

4/98

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

Ecole Nationale Polytechnique d'Alger
D.E.R de Génie Electrique et Informatique
Département d'Electronique



MEMOIRE
DE FIN D'ETUDES

En vue de l'obtention du Diplôme
d'Ingénieur d'Etat en Electronique

THEME

Application de la logique floue
à la commande de bras manipulateurs.

Conception et réalisation d'un logiciel
de simulation et de commandes : ROBSICOM
(ROBots Simulation and COMmands)

Réalisé par :

P.EL-H. FALL
N.LASSOUAOUI

Proposé et dirigé par :

C.LARBES
M.S. AIT CHEIKH

Promotion Juin 1998

80/A

General Register Office
Hertfordshire - ST. ALBANS
Registration Office

Dédicaces

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

Je dédie ce travail.

- *A mon très cher père, qu'il reçoive ma gratitude et mon cœur,*
- *A ma très chère mère,*
- *A la mémoire de ma grande- mère CHABHA,*
- *A mon frère MOHAND et à sa famille,*
- *A mes sœurs BAYA, FATIHA, LILA et leurs familles,*
- *Au personnel du centre médical de l'ENP,*
- *Au personnel de la bibliothèque et périodique,*
- *A ALDJIA, NASSIMA, MALIK, AZIZ, TOUFIK, MOUSTAFA, FERROUDIA, SAMIA, FALL et tous mes amis.*

Nadia

- *A mon père et ami,*
 - *A ma très chère mère,*
 - *A la mémoire de mon frère MAMADOU LAMME FALL,*
 - *A mon frère ami ABDOURAHMANE,*
 - *A ma nièce RAMAMBANE à BOUSSO et à toute la famille,*
 - *A mes oncles et tantes de Dakar, Mbour et Saint-louis,*
 - *A tous mes amis de Sénégal, d'Algérie et du Koweït,*
 - *A Dembané, Moussa, Pape Cheikh, etc,*
 - *A la mission diplomatique du Sénégal à Alger et particulièrement à Mr MBACKE QUEUE,*
 - *A tout ce peuple qui nous à accueilli et soutenu notre formation durant.*
 - *A tous ceux qui mérite notre gratitude et dont je ne peux citer.*
- Je leur dédie ce modeste travail.*

El-hadji

Handwritten text in a rectangular box, possibly a stamp or label, containing illegible characters.

Remerciements

Nous remercions très chaleureusement nos promoteurs M. C.LARBES et M.S.AIT CHEIKH, enseignants au Département d'Electronique de l'Ecole Polytechnique d'Alger de l'aide précieuse qu'ils nous ont prodiguée.

Nos remerciements vont aussi à tous nos honorables professeurs à qui nous devons en grande partie notre réussite.

Merci aussi à Dr ZERGUERRAS d'avoir bien voulu présider le jury et à Mr BOUSSEKSOU d'avoir accepté d'examiner ce travail.

Enfin toutes les personnes ayant contribué, de près ou de loin, à la réalisation de ce projet trouvent ici, l'expression de notre profonde gratitude.

SECRET
CONFIDENTIAL

[The remainder of the page contains extremely faint and illegible text, likely bleed-through from the reverse side of the document.]

هذا العمل يتمثل في تطبيق التحكم بواسطة المنطق الغامض على أنظمة معقدة (آليات) . تم استعمال أجهزة تحكم متعددة :

• جهاز تحكم ميداني

• جهاز تحكم سوجينو

مع دراسة عدد القواعد الغامضة فيها.

تم أيضا المقارنة بين التحكم الغامض و سابقها من نوع PID .

كل الجوانب الآلية (التمثيل الحركي و التحريكى ، التحكم بواسطة PID او بالمنطق الغامض ، التنشيط على

ثلاثة أبعاد و غيرها) تم جمعها في برنامج موحد ROBSICOM تحت نظام MS WINDOWS95 .

هذا العمل يصلح ان يكون منهجية مكملة للمقرر ، كما انه يستطيع ان يكون أداة للتصميم أذرع آلية.

Abstract:

This work concerns the application of fuzzy control to a complex systems (Robots) .

Many fuzzy controllers are used :

• Mamdani's Controller.

• Sugeno's Controller .

with variable number of fuzzy sets.

A comparison between these controllers and PID controllers is given.

A software :ROBSICOM (Robots Simulation and Commands) witch deals with all aspects of robotics (Cinematics, Dynamics, PID control, Fuzzy control, 3D animation,etc) is also implemented in MS WINDOWS 95 environnement.

This work could be used as a support for a course in robotics and eventually in the design stage of robot arms.

Resumé:

Cette étude concerne l'application de la commande floue à des systèmes complexes (Robots).

Différents controleurs flous sont utilisés:

• Controleur de Mamdani.

• Controleur de Sugeno.

avec la variation du nombre de règles.

La comparaison entre le controleur classique PID et le réglage par la logique floue est donnée.

Un logiciel :ROBSICOM (Robots Simulation et Commandes) qui englobe plusieurs aspects de la robotique (Cinématique, Dynamique, Commande floue, Animation 3D, etc) a été développé sous un environnement MS WINDOWS 95.

Cette étude peut servir comme un outil didactique ainsi qu'à l'étape de conception dans un projet robot.

1945
1946
1947

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

SOMMAIRE

11-1-1954
RECEIVED
SUPERVISOR
11-1-1954

Sommaire



Introduction Générale

Chapitre I Modélisation Cinématique

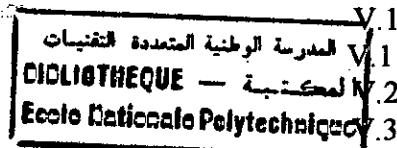
| | |
|---|------|
| Introduction | I.1 |
| I.1 Généralités sur les robots | I.1 |
| I.1.1 Définition des robots | I.1 |
| I.1.2 Description générale d'un robot | I.3 |
| I.1.2.1 Structure générale | I.3 |
| I.1.2.2 Classification morphologique | I.4 |
| I.2 Modélisation cinématique | I.5 |
| I.2.1 Modélisation cinématique directe | I.5 |
| I.2.1.1 Hypothèses de base pour le calcul du modèle | I.6 |
| I.2.1.2 Description de l'orientation | I.7 |
| I.2.1.2.1 Cosinus Directeurs | I.7 |
| I.2.1.2.2 Angles de Bryant | I.8 |
| I.2.1.3 détermination du modèle cinématique directe | I.9 |
| I.2.1.3.1 Méthode de Denavit-Hartenberg | I.9 |
| I.2.1.3.2 Application au bras articulé à 3DDL | I.11 |
| I.2.2 Modélisation cinématique inverse | I.14 |
| I.2.2.1 Algorithme de la méthode | I.14 |
| I.2.2.2 Application au bras articulé | I.15 |
| I.2.3 Cinématique des vitesses | I.16 |
| I.2.3.1 Translation et rotation différentielles | I.16 |
| I.2.3.2 Définition du vecteur différentiel | I.18 |
| I.2.3.3 Transformation de changement différentielle entre repères | I.19 |
| I.2.3.4 La jacobienne | I.20 |
| I.2.3.5 Méthode de calcul de la jacobienne | I.20 |
| I.3 Génération de trajectoires | I.22 |
| I.3.1 Interpolation avec des polynômes cubiques | I.23 |
| I.3.2 Génération de trajectoires cycloïdales | I.25 |
| I.3.3 Génération de trajectoires en fenêtre de Viviani | I.25 |
| I.3.4 Considérations pratiques | I.25 |
| Conclusion | I.26 |

Chapitre II Modélisation dynamique

| | |
|--|-------|
| Introduction | II.1 |
| II.1 Approche de Lagrange-Euler | II.1 |
| II.1.1 Energie cinétique du robot manipulateur | II.2 |
| II.1.2 Energie potentielle du robot manipulateur | II.4 |
| II.1.3 Equation dynamique du bras manipulateur | II.5 |
| II.1.4 Application | II.6 |
| II.2 Approche de Newton-Euler | II.6 |
| II.2.1 Rappel théoriques | II.7 |
| II.2.2 Application au structure mécanique du robot | II.9 |
| II.2.3 Equations récursives du mouvement du robot | II.10 |
| II.2.4 Equations récursives modifiées | II.12 |

| | |
|---|--------|
| II.2.5 Algorithme de Newton-Euler | II.10 |
| Conclusion | II.14 |
| Chapitre III Commande classique | |
| Introduction | III.1 |
| III.1 Simulation en boucle ouverte | III.2 |
| III.1.1 Applications | III.2 |
| III.1.2 Bras Scara | III.2 |
| III.1.3 Bras RRR | III.4 |
| III.1.4 Bras TRT | III.7 |
| III.2 Implémentation d'une loi de commande PID | III.7 |
| III.2.1 Introduction | III.7 |
| III.2.2 Algorithme des contrôleurs PID | III.8 |
| III.2.3 Application | III.8 |
| III.2.3.1 Algorithme de la commande PID | III.9 |
| III.2.3.2 Application PID | III.11 |
| III.2.3.2.1 Etude de l'influence des paramètres (Scara) | III.11 |
| III.2.3.2.2 Commande du bras RRR | III.16 |
| III.2.3.2.3 Commande du bras TRT | III.16 |
| Conclusion | III.18 |
| IV Méthodologies de commande floue | |
| Introduction | IV.1 |
| IV.1 Rappel des principes de la logique floue | IV.2 |
| IV.1.1 ensembles flous | IV.2 |
| IV.1.2 Opérateurs ensemblistes élémentaires | IV.3 |
| IV.1.3 Variables linguistiques et ensembles flous | IV.4 |
| IV.1.4 Fonction d'appartenance | IV.5 |
| IV.1.5 Logique floue et raisonnement approché | IV.6 |
| IV.2 Contrôleur flou | IV.6 |
| IV.2.1 Architecture d'un contrôleur flou | IV.6 |
| IV.2.1.1 L'interface de fuzzification | IV.7 |
| IV.2.1.2 La base de connaissances | IV.7 |
| IV.2.1.3 La logique et prise de décision | IV.8 |
| IV.2.1.4 L'interface de défuzzification | IV.8 |
| IV.2.2 Stratégie de fuzzification | IV.8 |
| IV.2.3 Stratégie d'inférence | IV.9 |
| IV.2.4 Agrégation des règles | IV.9 |
| IV.2.5 défuzzification | IV.10 |
| IV.2.5.1 Méthode du maximum | IV.10 |
| IV.2.5.2 Méthode de la moyenne des maxima | IV.10 |
| IV.2.5.3 Méthode du centre de gravité | IV.10 |
| IV.2.5.4 Méthode des hauteurs pondérées | IV.11 |
| IV.2.5.5 Méthode des hauteurs pondérées modifiée | IV.11 |
| IV.2.6 Contrôleur de Mamdani | IV.12 |
| IV.2.7 Contrôleur de Larsen | IV.13 |
| IV.2.8 Contrôleur de Sugeno | IV.13 |
| Conclusion | IV.14 |

| | |
|--|--------|
| V Commande floue de bras manipulateurs | |
| Introduction | V.1 |
| V.1 Description du schéma de contrôle typique | V.1 |
| V.2 Algorithme du contrôle flou | V.2 |
| V.2.1 Loi de la commande | V.3 |
| V.2.2 Implémentation | V.3 |
| V.3 Application à la commande du bras Scara | V.5 |
| V.3.1 fonction d'appartenance | V.5 |
| V.3.1.1 Distribution des fonctions d'appartenance | V.5 |
| V.3.1.2 Nombre de classes ou sous-ensembles flous | V.5 |
| V.3.2 Univers de discours | V.6 |
| V.3.3 Règles décisionnelles | V.6 |
| V.3.4 Méthodes d'inférence | V.6 |
| V.3.5 Période d'échantillonnage | V.6 |
| V.3.6 Simulation, résultats et interprétations | V.7 |
| V.4 Application à la commande du bras RRR | V.13 |
| Conclusion | V.13 |
| | |
| Chapitre VI Résultats et interprétations | |
| Introduction | VI.1 |
| VI.1 Comparaison à partir des résultats sur le Scara | VI.1 |
| VI.2 Caractéristiques d'un contrôleur PID | VI.2 |
| VI.3 Caractéristiques d'un contrôleur flou | VI.2 |
| VI.4 Etude comparative | VI.3 |
| Conclusion | VI.4 |
| | |
| VII Présentation du Logiciel | |
| Introduction | VII.1 |
| VII.1 Organisation générale | VII.1 |
| VII.2 Les différents articles liés au menu | VII.3 |
| VII.2.1 Application | VII.3 |
| VII.2.2 Cinématique | VII.4 |
| VII.2.3 Dynamique | VII.8 |
| VII.2.4 Trajectoire | VII.8 |
| VII.2.5 PID | VII.11 |
| VII.2.6 Flou | VII.11 |
| VII.2.7 Animation 3D | VII.13 |
| VII.2.8 Help | VII.13 |
| VII.3 Perspectives | VII.14 |
| Conclusion | VII.15 |
| | |
| Conclusion Générale | |
| | |
| Annexes | |
| | |
| Bibliographie | |



Amalgamated Telephone Company
Telephone Department
New York City

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

INTRODUCTION GENERALE

مجلس القضاء الاعلى
الجمهورية العراقية
القضاء الموحد
البيروت - لبنان

Handwritten text, likely a legal document or court record, covering the majority of the page. The text is extremely faint and illegible due to the quality of the scan.

INTRODUCTION GENERALE

Avec le développement fulgurant des microprocesseurs et de la circuiterie intelligente en général, l'utilisation des robots manipulateurs dans les chaînes de production automatisées est devenue de plus en plus fréquente. Cette robotisation qui est l'un des facteurs de l'augmentation de la productivité dans l'industrie du siècle finissant, n'a pu certainement se faire que grâce à des méthodes scientifiques de conception et de réalisation conférant à l'outil robot les performances désirées.

L'un des plus importants aspects dans l'étude d'un système asservi en général est la commande. Elle permet de trouver des solutions à donner aux paramètres des contrôleurs ou régulateurs utilisés dans le but de maîtriser le comportement du système à asservir dans les limites spatiales et temporelles considérées.

Le premier type de commande auquel on a fait appel fut la régulation PID classique. Ce type de régulateurs a l'avantage d'être efficace, fiable et simple à concevoir ; et de nombreuses méthodes ont été mises au point dans le but de faciliter le réglage de ces systèmes de commande [KET92].

Malgré l'existence de cette diversité de méthodes de calcul, les performances d'un PID restent bien souvent limitées en raison de la complexité réelle du système à commander : non linéarité, variation des paramètres, etc... alors que la plupart de ces systèmes doivent répondre à des exigences bien particulières. C'est le cas en robotique où malgré l'existence d'une chaîne mécanique articulée avec tout le couplage existant

entre les différents degrés de liberté, l'étude doit être faite selon un cahier des charges imposant dans la plupart des cas des déplacements précis selon des vitesses bien déterminées. De nombreuses stratégies ont été développées ces dernières décennies pour apporter des réponses au problème de la commande : Commande adaptative, Réseaux de neurones... Elles sont presque toutes basées sur les données numériques du système.

Or, des études faites [PED88] ont montré que, dans une boucle de régulation, l'homme réagit comme un contrôleur robuste non linéaire dont les paramètres varient au cours du temps. Cette stratégie de contrôle « humain » intègre la connaissance du processus et permet donc une action de contrôle même en présence de non linéarité, de distorsion, etc....

Ainsi les contrôleurs flous ont vu le jour dans l'idée simple de l'utilisation d'un mécanisme représentatif des jugements de l'être humain.

La conception de tels contrôleurs se base dans la théorie des ensembles flous qui permet de manipuler des données dont les caractéristiques s'accordent mal avec le calcul et la logique classique : données incertaines, mal qualifiées, et redondantes. Les bases de cette théorie généralisent la notion d'appartenance d'un objet à un ensemble donné. Elle permet, à la différence de la logique binaire, de donner à tout objet un degré d'appartenance qui peut prendre toute valeur comprise entre 0 et 1. Une telle assertion va permettre de construire des règles de logique qui s'adaptent bien à la modélisation des décisions humaines ; et donc permettront à la conception des contrôleurs flous.

Nous traiterons dans notre mémoire les différents aspects de la commande floue appliquée aux bras manipulateurs tout en faisant ressortir les avantages de cette nouvelle commande par rapport à ses précédentes les plus utilisées, mais aussi les limites et stratégies avancées de la part des chercheurs dans le but de palier à certaines insuffisances liées à la commande floue.

Ainsi notre mémoire est organisée en sept chapitres :

Le premier chapitre présente une vue d'ensemble sur la robotique. Il englobe des notions de base relatives aux robots ; on y traite notamment du vocabulaire utilisé. Il traite aussi l'étape primordiale de conception ou d'analyse d'un robot qu'est la modélisation. A ce niveau, nous introduirons la transformation de *Denavit-Hartenberg*

dans le cadre de l'étude géométrique et cinématique. Les bases de la cinématique des vitesses y sont aussi exposées. Ce chapitre se termine par l'étude de la génération de trajectoires.

Au deuxième chapitre, les principales approches de modélisation dynamique sont présentées ; à savoir *Euler-Lagrange*, *Newton-Lagrange*.

C'est au troisième chapitre qu'on aura à développer l'étude dynamique de quelques robots grâce à la commande classique PID. Nous mettrons l'accent sur les principes de la commande et les problèmes liés au contrôle d'un système aussi complexe qu'un robot par le biais de modèle linéarisé et découplé. Nous toucherons de plein fouet les limitations de la commande classique et la bienvenue d'un type de commande tel que la logique floue.

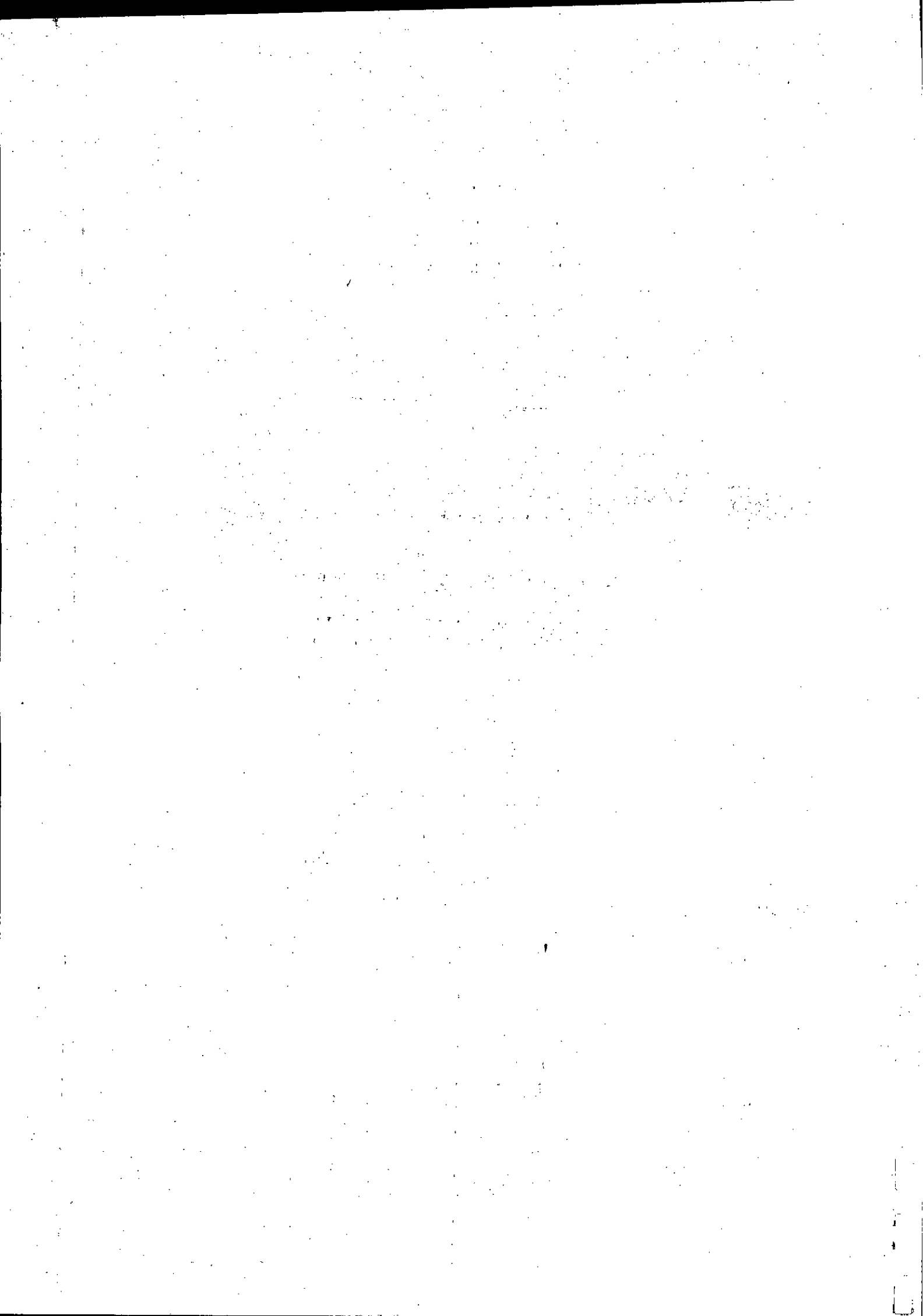
Au chapitre quatre nous aurons à survoler les bases théoriques de la logique floue. Nous y développerons la notion fertile d'ensemble flou et les principales arithmétiques qui se sont développées autour. L'étude des relations floues (de la génération des règles à la prise de décision) occupera naturellement dans ce chapitre sa place de choix. En effet, elle résume l'idée maîtresse de la commande floue.

L'objectif du chapitre cinq est de développer une commande floue sur quelques bras manipulateurs. On présentera en premier lieu l'architecture d'un contrôleur flou avec ses différents blocs que sont la fuzzification, l'inférence et la défuzzification. Ce chapitre fera ressortir les apports de la logique floue à la commande des robots mais aussi les limitations en termes d'inconvénients.

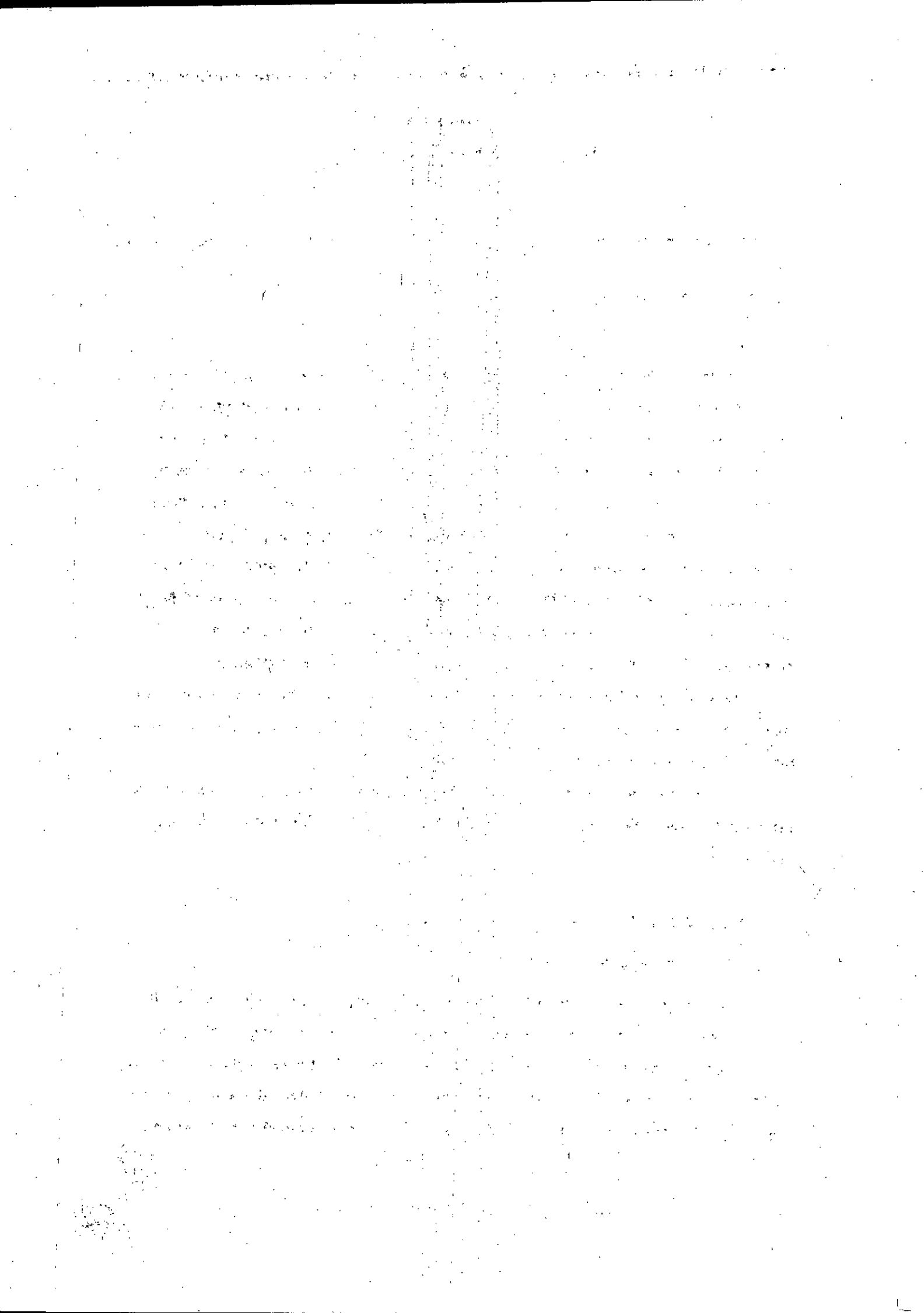
Dans le chapitre six nous présenterons le bilan de la commande. La comparaison floue PID nous permettra non seulement de mettre en exergue ce qu'offre une des deux méthodes par rapport à l'autre, mais aussi de tirer des équivalences entre les deux commandes moyennant certaines conditions à remplir.

Le dernier et septième chapitre est réservé à la présentation du logiciel autonome (ROBSICOM 3.0) que nous avons développé et qui englobe tous les aspects pratiques liés à notre projet..

Enfin, notre mémoire se clôture avec une conclusion générale qui nous permettra entre autres de développer des perspectives relatives à la commande des robots et des grands systèmes en général.



CHAPITRE I
MODELISATION
CINEMATIQUE



Chapitre I

MODELISATION CINEMATIQUE

INTRODUCTION

En général la synthèse d'une commande d'un processus donné passe toujours par une étape primordiale qui est la modélisation. Elle consiste à établir un ensemble d'équations mathématiques décrivant les relations d'entrées / sorties du système et obtenues grâce à l'application de lois physiques bien connues. Ces équations résument les relations entre les variables du système ; et entre elles et leurs dérivées respectives.

La cinématique des robots manipulateurs traite des caractéristiques géométriques du mouvement du robot comme une fonction du temps sans tenir compte des forces et couples qui en sont la cause. Elle permet de définir chaque configuration du robot par un ensemble de variables dites articulaires liées à la structure mécanique du bras et de projeter les points intéressants (outil terminal par exemple) dans un repère quelconque.

Le passage de l'espace articulaire à l'espace cartésien et vice versa en position aussi bien qu'en vitesse se fait grâce à des méthodes puissantes se reposant sur l'outil matriciel : la méthode de *Denavit-Hartenberg* par exemple.

Ce chapitre présente d'abord des généralités sur les robots, puis introduit les principes et les méthodes de base de la cinématique des robots. Il inclut aussi l'étude de la trajectoire.

I. 1 GENERALITES SUR LES ROBOTS

I. 1. 1 DEFINITION DES ROBOTS

La robotique est une science de l'ingénieur fondamentalement pluridisciplinaire. A elle seule elle utilise plus d'une technologie : la mécanique, l'hydraulique, l'électronique, l'informatique ; pour ne citer que cela. Elle est de ce fait un champ vaste et complexe dont une définition universelle fait défaut et semble même impossible. Le moins qu'on puisse dire est qu'elle est une discipline qui se charge de

regrouper l'ensemble des activités de construction et de mise en œuvre des **robots** [LAL94]. Ainsi la définition se ramène à celle du robot.

La **JIRA** (Japan Industrial Robot Association) propose :

“ Dispositifs versatiles et flexibles offrant des fonctions de placement similaires à celles des membres humains dont les fonctions de déplacement sont commandées par ses capteurs et ses moyens de reconnaissance.”

La **RIA** (Robotic Industries Association) des USA met plutôt l'accent sur l'aspect utilisation ; elle propose :

“Manipulateur multifonctions reprogrammable conçu pour déplacer des matériaux, des pièces, des outils ou des dispositifs spécialisés à travers des déplacements variables et programmables pour accomplir diverses tâches.”

L'**AFRI** (Association Française de Robotique Industrielle) donne aussi une définition relativement précise : “Mécanisme multifonctionnel à plusieurs degrés de liberté commandé directement par un opérateur humain ou par un système logique .”

Le robot industriel est présenté comme un manipulateur asservi en position, reprogrammable, polyvalent et capable de positionner et d'orienter des matériaux, pièces, outils ou dispositifs spécialisés. Il est présent dans des environnements hostiles, où il est destiné à accomplir des tâches dont l'homme est incapable de les effectuer (manipulation des matériaux radioactifs par exemple).

Quoique différentes, ces définitions présentent des ressemblances et une certaine complémentarité. Elles ressortent notamment certaines caractéristiques essentielles du robot qui le distinguent des machines automatisées (machines à commande numérique par exemple).

- La **versatilité** ou **flexibilité** qui est assurée par la démontabilité et l'adaptabilité de l'outil terminal ainsi que du fonctionnement de la commande en logique programmée (existence de la commande numérique).
- L'**automatisation complète** de la tâche qui suppose que celle-ci puisse se réaliser indépendamment de la connaissance “complète et à l'avance ” de l'environnement ; et que la machine puisse à elle seule appréhender la situation de cet environnement et établir les conditions à respecter en performance dans la mise en œuvre de son travail.

Passons maintenant à l'analyse de la structure du robot qui est une phase essentielle de design dans un projet robot.

I. 1. 2 DESCRIPTION GENERALE D'UN ROBOT

I. 1. 2.1 STRUCTURE GENERALE

Un robot manipulateur est formé par [COI95] :

- Une **structure mécanique** qui supporte l'organe terminal qui est la partie interactive à situer. C'est la partie la plus visible du robot.
- Les **capteurs** divers permettant le contrôle de l'état interne du robot (capteurs proprioceptifs) mais aussi de l'état de son environnement (capteurs extéroceptifs); et qui sont nécessaires à la commande.
- Un **système de commande** chargé de piloter les actionneurs du robot à partir de la définition du mouvement à exécuter transmise par le système décisionnel et les informations délivrées par les capteurs.
- Un **système décisionnel** qui assure la fonction de raisonnement et élabore le mouvement du robot manipulateur à partir de la définition de la tâche à exécuter. Ses fonctions de bases sont l'interprétation et la compréhension des messages émis par l'opérateur, la résolution des problèmes posés et la gestion de la base de données nécessaire au système.
- Un **système de communication** qui gère les messages transmis entre le système décisionnel et l'opérateur.

L'organigramme de la *figure I.1* résume cette structure.

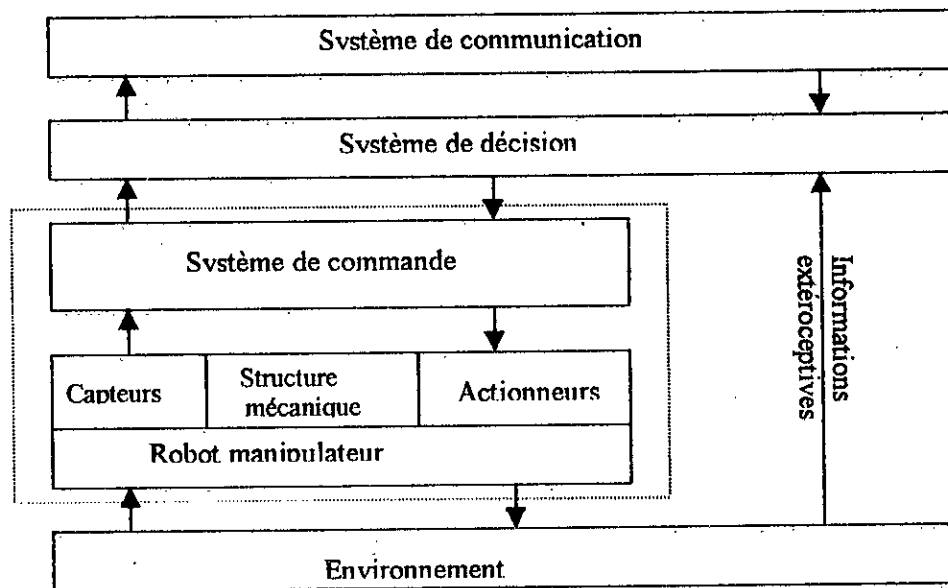


Figure I.1 Structure générale d'un robot manipulateur

I. 1. 2. 2 CLASSIFICATION MORPHOLOGIQUE

La classification des robots se fait selon plusieurs critères :

- Classification selon le niveau d'automatisation (point de vue commande),
- Classification selon le type d'asservissement (Boucle ouverte, Boucle fermée),
- Classification selon la nature de la tâche,
- Classification morphologique.

C'est surtout cette dernière qui est la plus souvent utilisée. Cette classification est basée sur l'architecture du robot ; elle donne à la fois la description de la chaîne mécanique que constitue les différents segments et les degrés de liberté concernant leur mouvement.

De manière grossière, cette classification fait distinguer 4 types de bras dans le cas de trois degrés de liberté:

a) **Bras cartésien (TTT) :**

Il est constitué de 3 articulations prismatiques. Son volume de travail est un parallélépipède.

b) **Bras cylindrique (TRT) :**

Il comprend 2 liaisons prismatiques et une liaison rotative dont les axes sont disposés selon un système de coordonnées cylindrique. Son volume de travail est une portion de cylindre.

c) **Bras sphérique (RRT) :**

Robot dont la structure présente 2 articulations pivots et une prismatique dont les axes sont disposés selon un système de coordonnées polaires. Le volume de travail est une calotte volumique de sphère.

d) **Bras articulé (RRR) :**

Il présente 3 liaisons toutes rotatives. Il présente une très grande flexibilité (mouvement dans différents plans).

La figure 1.2 présente la forme de ces différents robots.

Mécaniquement le manipulateur est constituée d'une chaîne de segments connectés entre eux par des articulations rotationnelles ou prismatiques. Une de ses deux extrémités est attachée à la base, tandis que l'autre est libre et reliée à l'outil terminal. Le mouvement relatif de chaque articulation entraîne celui des segments, ce qui permet de positionner l'extrémité libre et de donner à l'outil terminal la position et

l'orientation désirées.

La description du mouvement de chaque segment se fait par la cinématique directe.

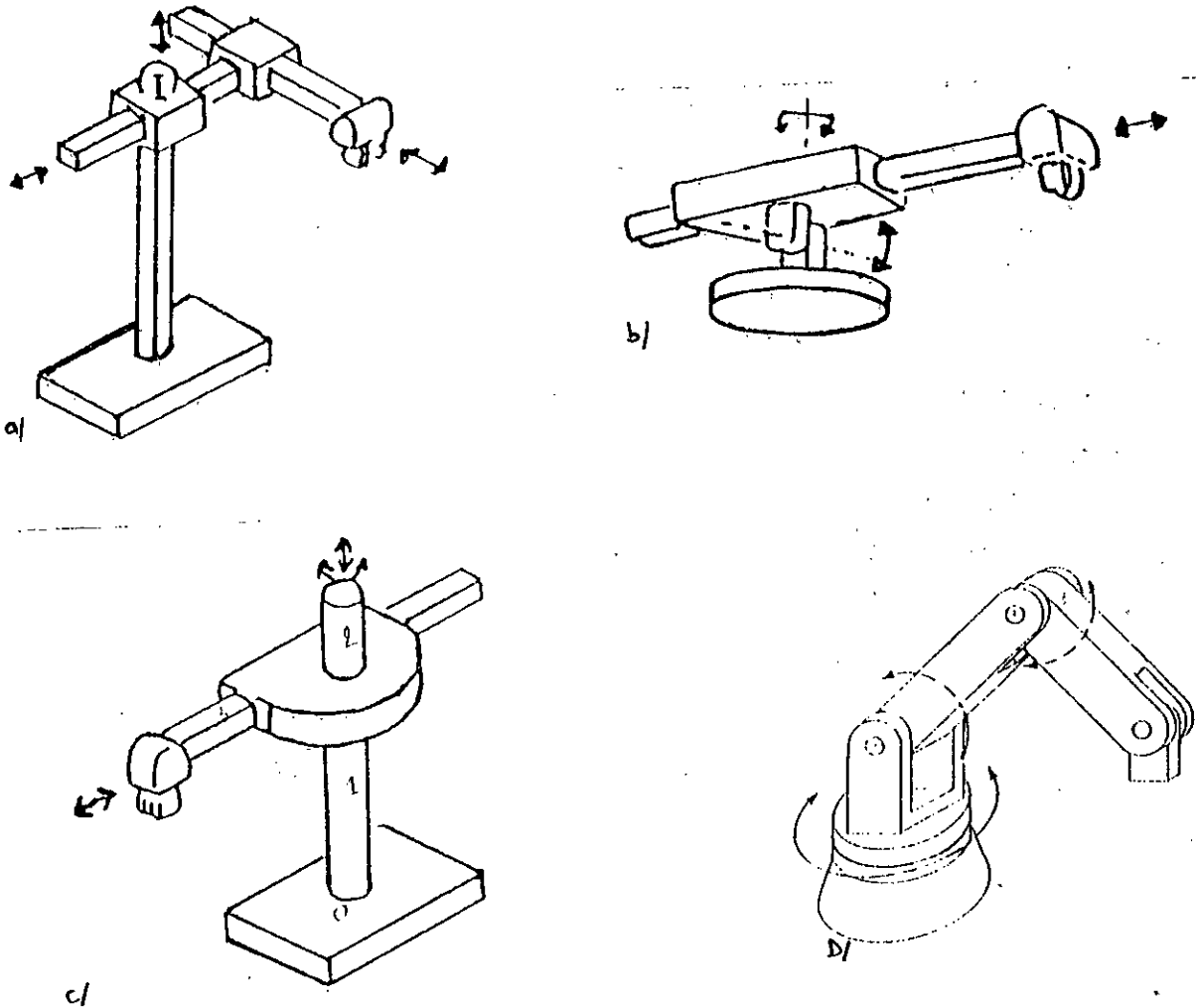


Figure 1.2 différents robots a) TTT b) RRT c) TRT d) RRR

I. 2 MODELISATION CINEMATIQUE

I. 2. 1 MODELISATION CINEMATIQUE DIRECTE

La description de chaque segment du bras est faite par le biais d'un repère qui lui est solidaire.

On appelle modèle cinématique direct la fonction f qui permet d'exprimer la situation de l'organe terminal du robot en fonction de la configuration du robot lui-même.

La situation de l'organe terminal est définie par les coordonnées dites **opérationnelles** (ou cartésiennes):

$$\mathbf{X}^T = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_m) \quad (\text{I.1})$$

La configuration du robot manipulateur est définie par n coordonnées **généralisées** (ou articulaires):

$$\mathbf{q}^T = (q_1, q_2, \dots, q_n) \quad (\text{I.2})$$

Le modèle est donc donné par la relation :

$$\mathbf{X} = f(\mathbf{q}) \quad (\text{I.3})$$

La modélisation cinématique inverse permet de faire le chemin inverse ; c'est-à-dire à partir de la connaissance de \mathbf{X} et des paramètres géométriques du robot trouver le vecteur \mathbf{q} de variables articulaires qui vérifie l'équation (1.3) au cas où il existe. C'est là un cas pratique très important sur lequel nous reviendrons dans la section suivante.

Dans ce qui suit les paramètres géométriques du bras manipulateur étant supposés connus, nous essayons de répondre à la question suivante :

Pour chaque vecteur des variables articulaires $(q_1, q_2, \dots, q_n)^T$, n étant le degré de liberté du manipulateur, quelle est la position et l'orientation de l'outil terminal correspondante, par rapport au référence fixe qui est la base ?

Répondre à cette question est l'objectif de la cinématique directe.

I. 2. 1. 1 HYPOTHESES DE BASE POUR LE CALCUL DU MODELE CINEMATIQUE DIRECT

Pour le calcul du modèle en cinématique direct on suppose les conditions suivantes :

- Les segments du robot sont parfaitement rigides (indéformables) et les articulations mécaniquement parfaites (pas d'élasticité, pas de frottements, etc ...)
- La base du robot est fixe.
- Le repère de référence est lié à la base du robot.
- Il n'existe pas d'obstacle dans le volume de travail du robot .
- Une tâche peut être exécutée en contrôlant la position et l'orientation de l'organe terminal.

Ces conditions admises, l'étude se simplifie énormément.

I. 2. 1. 2 DESCRIPTION DE L'ORIENTATION

L'orientation d'un repère R_n par rapport à un repère de base R_0 peut être définie de diverses manières : les *Angles d'Euler*, la Transformation *RPY* (*Roll*, *Pitch*, *Yaw*), les *Cosinus Directeurs* et Les *Angles de Bryant*.

Les deux premières méthodes ont été traitées [BEN97] ; nous allons uniquement présenter les deux dernières.

I. 2. 1. 2. 1 Les Cosinus Directeurs

Considérons (*figure I.3*) un repère de référence R_0 ayant E pour vecteur de base et e_1, e_2, e_3 pour vecteurs de base unités. On associe à un solide B un autre repère R_b de vecteur de base G et de vecteurs de base unités g_1, g_2, g_3 .

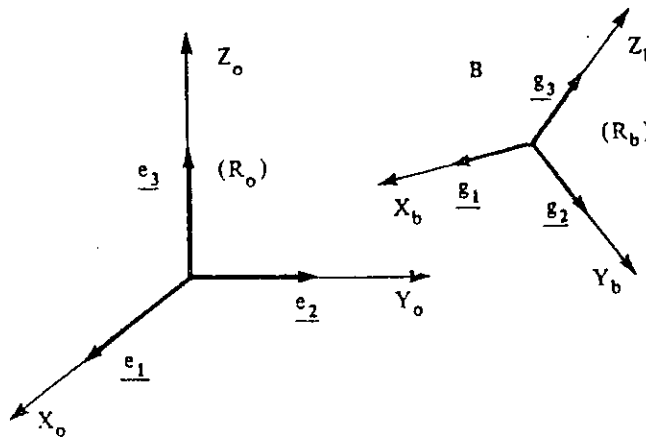


Figure I.3 : Description de l'orientation d'un repère à l'aide des cosinus directeurs

On peut écrire g_1, g_2, g_3 en fonction de e_1, e_2, e_3 , on a :

$$\begin{aligned} g_1 &= a_1 e_1 + a_2 e_2 + a_3 e_3 \\ g_2 &= b_1 e_1 + b_2 e_2 + b_3 e_3 \\ g_3 &= c_1 e_1 + c_2 e_2 + c_3 e_3 \end{aligned} \tag{I.4}$$

que l'on peut écrire sous forme matricielle :

$$\begin{pmatrix} g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} \quad (I.5)$$

c'est-à-dire :

$$G = A_g^E E \quad (I.6)$$

A_g^E est la matrice des cosinus directeurs qui donne l'orientation de G par rapport à E.

Cette matrice comporte 9 éléments cosinus directeurs, qui doivent vérifier :

$$\begin{aligned} |g_i| &= |e_i| = 1 & i=1, 2, 3. \\ g_i \cdot g_j &= e_i \cdot e_j = 0 & \text{si } i \neq j \end{aligned} \quad (I.7)$$

En définitive, seuls trois cosinus directeurs non choisis arbitrairement sont indépendants.

I. 2. 1. 2 Les Angles de Bryant

Dans ce cas, l'orientation de R_b est le résultat de trois rotations successives (figure I.4) :

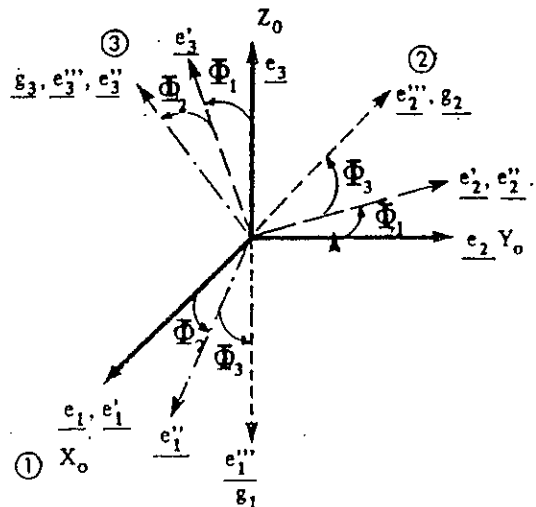


Figure I.4 : Description de l'orientation d'un repère à l'aide des angles de Bryant

- 1^{ère} rotation autour de e_1 d'un angle ϕ_1 , on obtient e_1', e_2', e_3' .
- 2^{ème} rotation autour de e_2' d'un angle ϕ_2 , on obtient e_1'', e_2'', e_3'' .
- 3^{ème} rotation autour de e_3'' d'un angle ϕ_3 , on obtient e_1''', e_2''', e_3''' identiques à g_1, g_2, g_3 .

on obtient :

$$G = A_i^g E$$

Avec

$$A_i^g = \begin{pmatrix} C_2 C_3 & C_1 S_3 + S_1 S_2 C_3 & S_1 S_3 - C_1 S_2 C_3 \\ -C_2 S_3 & C_1 C_3 - S_1 S_2 S_3 & S_1 C_3 + C_1 S_2 S_3 \\ S_2 & -S_1 C_2 & C_1 C_2 \end{pmatrix} \quad (I.8)$$

avec $\cos(\phi_i) = C_i$

$\sin(\phi_i) = S_i$

I. 2. 1. 3 DETERMINATION DU MODELE CINEMATIQUE DIRECTE

Définir le modèle géométrique direct revient à connaître :

1. L'orientation du repère R_i vis à vis de R_0 .
2. La position d'un point quelconque du corps (connu dans R_i) par rapport au repère R_0 .

Dans la littérature, il existe plusieurs méthodes pour l'évaluation de la position et de l'orientation d'un corps solide indéformable vis à vis d'un repère de référence.

On peut les classer en deux groupes principaux [COI95]:

- Méthodes utilisant les matrices 3×3 ,
- Méthodes utilisant les matrices 4×4 .

Parmi ces dernières la méthode de DENAVIT-HARTENBERG (matrices 4×4) est la plus utilisée.

I. 2. 1. 3. 1 Méthode de DENAVIT-HARTENBERG (DH)

Pour en décrire les relations de translations et de rotations entre les liaisons adjacentes d'un robot manipulateur, DENAVIT et HARTENBERG (1955) proposèrent une méthode matricielle pour en établir systématiquement les repères attachés à chaque liaison dans une chaîne articulée. La représentation de D-H résulte dans l'utilisation d'une transformation homogène 4×4 [PAU81] qui représente le système de coordonnées de chaque liaison par rapport au système de coordonnées de la liaison précédente. Et delà, à travers une séquence de transformations on peut ramener le système de coordonnées attaché à l'élément terminal au système de coordonnées attaché à la base (repère fixe).

Les repères liés aux segments sont déterminés selon les règles suivantes :

- L'axe z_{i-1} est le long de l'articulation i .
- L'axe x_i est normal à l'axe z_{i-1} : $x_i = z_{i-1} \wedge z_i$.
- L'axe y_i est déterminé de telle manière à constituer un trièdre direct.

Il est à noter que le choix du référentiel de base est arbitraire, pour autant que l'axe z_0 soit le long de l'axe de mouvement de l'articulation 1, de même la référence de l'outil terminal pourra être placée n'importe où, tant que l'axe x_n est normal à z_{n-1} . Il est maintenant possible de définir les paramètres cinématiques, de la manière suivante :

1. θ_i est l'angle de l'axe x_{i-1} vers l'axe x_i autour de l'axe z_{i-1} .
2. α_i est l'angle de l'axe z_{i-1} vers l'axe z_i autour de l'axe x_i .
3. a_i est la distance comprise entre l'intersection de l'axe z_{i-1} avec l'axe x_i et l'origine du référentiel i le long de l'axe x_i .
4. d_i est la distance comprise entre l'origine du référentiel $i-1$ et l'intersection de l'axe z_{i-1} avec l'axe x_i le long de l'axe z_{i-1} .

L'application de la méthode D-H avec un bras à n degrés de liberté se présente selon l'algorithme suivant [ROC 98] :

1. Etablir le système de coordonnées lié à la base, en s'assurant que l'axe z_0 est bien selon l'axe de rotation de l'articulation 1, et qu'il pointe vers l'« épaule » du bras manipulateur.
2. Pour chaque articulation $i=1, n-1$ faire les étapes 3 à 6 après numérotation des articulations et les segments.
3. Aligner l'axe z_i avec l'axe de l'articulation $i+1$.
4. Etablir l'origine de chaque système de coordonnées, celui-ci sera donné par l'intersection de l'axe z_{i-1} et z_i ou bien l'intersection de leurs normales communes avec l'axe x_i .
5. Etablir l'axe x_i selon la règle suivante : $x_i = \pm(z_{i-1} \wedge z_i) / \|z_{i-1} \wedge z_i\|$ ou bien le long de la normale commune aux axes z_i et z_{i-1} .
6. Etablir l'axe y_i , selon la règle suivante : $y_i = \pm(z_i \wedge x_i) / \|z_i \wedge x_i\|$ afin d'établir le trièdre direct (x_i, y_i, z_i) .
7. Etablir l'axe z_n selon la direction de l'axe z_{n-1} , et est sortant du corps du robot.

Déterminer les paramètres de chaque articulation : $i = 1, \dots, n-1$.

Une fois ces étapes effectuées, on détermine les matrices de transition homogènes relatives à chaque articulation, elles sont données en fonction des paramètres décrits ci-

dessus et en remarquant qu'un point r_i exprimé dans le référentiel i , peut être exprimé dans le référentiel $i+1$ en opérant les transformations successives suivantes :

1. Rotation autour de l'axe z_{i-1} d'un angle θ_i afin d'aligner l'axe x_{i-1} avec l'axe x_i .
2. Translation le long de l'axe z_{i-1} d'une distance d_i , afin de faire coïncider x_i et x_{i-1} .
3. Translation le long de l'axe x_i d'une distance a_i afin de faire coïncider les origines des axes x_i et x_{i-1} .
4. Rotation autour de l'axe x_{i-1} d'un angle α_i afin de faire coïncider les deux systèmes de coordonnées i et $i-1$.

Ce qui donne la matrice de transformation homogène suivante :

$$A_i^{i-1} = T_{z,d} \times R_{z,\theta} \times T_{x,a} \times R_{x,\alpha} \quad (\text{I.9})$$

$$A_i^{i-1} = \begin{pmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & +\cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{I.10})$$

Appliquons maintenant l'algorithme donné ci-dessus sur le bras articulé (RRR).

I. 2. 1. 3. 2 Application au bras articulé à 3 D.D.L

Le schéma descriptif de ce manipulateur est donné à la *figure I.5*.

L'implémentation des repères pour chaque liaison, en suivant les étapes de l'algorithme cité ci-dessus est illustré sur la *figure I.6* et les paramètres de la transformation D-H du robot sont regroupés dans le *tableau I.1* :

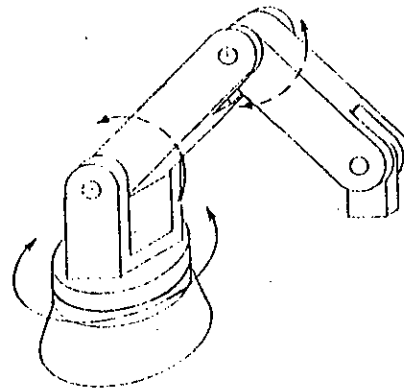


figure I.5 : Présentation du bras articulé

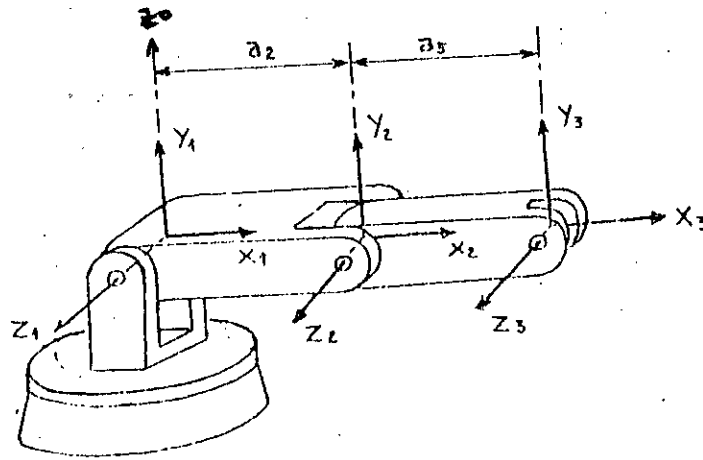


figure I.6 : Implémentation de la méthode D-H.

Tableau 1 : Paramètres de la transformation D-H pour le bras articulé .

| Segments | Variables | $\theta_{initial}$ | α_i | d_i | a_i |
|-----------|------------|--------------------|------------|-------|-------|
| Segment 1 | θ_1 | 0 | 90 | d_1 | 0 |
| Segment 2 | θ_2 | 0 | 0 | 0 | a_2 |
| Segment 3 | θ_3 | 0 | 0 | 0 | a_3 |

Ainsi en utilisant l'algorithme de la représentation de *Denavit-Hartenberg* nous aboutissons aux matrices de transformations homogènes suivantes :

$${}^0A_1 = \begin{pmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (I.11)$$

$${}^1A_2 = \begin{pmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (I.12)$$

$${}^2A_3 = \begin{pmatrix} C_3 & -S_3 & 0 & a_3 C_3 \\ S_3 & C_3 & 0 & a_3 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{I.13})$$

on a :

$${}^0A_3 = {}^0A_1 \times {}^1A_2 \times {}^2A_3 \quad (\text{I.14})$$

ce qui donne :

$${}^0A_3 = \begin{pmatrix} C_1 C_{23} & -C_1 S_{23} & S_1 & C_1 (a_3 C_{23} + a_2 C_2) \\ S_1 C_{23} & -S_1 S_{23} & -C_1 & S_1 (a_3 C_{23} + a_2 C_2) \\ S_{23} & C_{23} & 0 & a_3 S_{23} + a_2 S_2 + d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{I.15})$$

avec :

$$C_{23} = \cos(\theta_2 + \theta_3)$$

$$S_{23} = \sin(\theta_2 + \theta_3).$$

Le modèle géométrique direct s'obtient en égalisant la matrice de transformation homogène qui relie l'élément terminal à la base (dans l'exemple précédent 0A_3) et la matrice de transformation générale suivante :

$${}^0A_3 = \begin{pmatrix} X_x & Y_x & Z_x & P_x \\ X_y & Y_y & Z_y & P_y \\ X_z & Y_z & Z_z & P_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{I.16})$$

où :

$(X_x \ X_y \ X_z)^T$ est l'orientation de l'élément terminal suivant l'axe X de la base.

$(Y_x \ Y_y \ Y_z)^T$ est l'orientation de l'élément terminal suivant l'axe Y de la base.

$(Z_x \ Z_y \ Z_z)^T$ est l'orientation de l'élément terminal suivant l'axe Z de la base.

$(P_x \ P_y \ P_z)^T$ est la position de l'élément terminal.

On trouve donc :

$$\begin{aligned} x = P_x &= a_3 \cdot C_1 \cdot C_{23} + a_2 \cdot C_1 \cdot C_2 \\ y = P_y &= a_3 \cdot S_1 \cdot C_{23} + a_2 \cdot S_1 \cdot C_2 \\ z = P_z &= a_3 \cdot S_{23} + a_2 \cdot S_2 + d_1 \end{aligned} \quad (\text{I.17})$$

Ce système d'équations permet de faire le passage de l'espace articulaire à l'espace cartésien. Il s'agit d'un résultat important obtenu à travers la cinématique directe.

I. 2. 2 MODELISATION CINEMATIQUE INVERSE

La commande d'un bras manipulateur est généralement effectuée dans l'espace articulaire. Par contre la manipulation des objets et la réalisation d'une tâche sont effectuées dans l'espace cartésien. Ce qui veut dire que le contrôle de la position et de l'orientation de l'effecteur nécessite la connaissance du modèle géométrique inverse.

En cinématique inverse on essaye de répondre à la question suivante :
une position et une orientation étant donnée, quelle peut être la configuration que doit prendre le robot afin que son outil terminal atteigne le point désiré avec l'orientation fixée ?

Un tel problème n'a pas toujours de solution. En effet apporter une solution revient à trouver une fonction réciproque de f donnée à l'équation I.3. Or f étant non linéaire, les théorèmes d'existence et a fortiori d'unicité font défaut.

La résolution du problème cinématique inverse peut se faire à l'aide de plusieurs méthodes : analytiques, itératives, etc... Mais la plus utilisée repose sur l'identification et la résolution séquentielle à partir de la matrice de transformation obtenue [PAU81].

Dans ce qui suit nous décrirons brièvement la méthode, pour ensuite l'appliquer dans le cas du bras articulé.

I. 2. 2. 1 ALGORITHME DE LA METHODE

La méthode analytique que nous présentons est basée sur l'identification de la matrice de transformation homogène obtenue avec la cinématique directe du robot avec la matrice constante T donnant la position et l'orientation de l'effecteur.

$$T = \begin{pmatrix} X_x & Y_x & Z_x & P_x \\ X_y & Y_y & Z_y & P_y \\ X_z & Y_z & Z_z & P_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (I.18-a)$$

et

$${}^0A_3 = \begin{bmatrix} g_{11}(q,d) & g_{12}(q,d) & g_{13}(q,d) & g_{14}(q,d) \\ g_{21}(q,d) & g_{22}(q,d) & g_{23}(q,d) & g_{24}(q,d) \\ g_{31}(q,d) & g_{32}(q,d) & g_{33}(q,d) & g_{34}(q,d) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (I.18-b)$$

où

$q = (q_1, \dots, q_n)^T$ est un vecteur qui contient les variables articulaires,

$d = (d_1, \dots, d_m)^T$ contient les paramètres géométriques du bras (longueur des segments ...).

Les expressions des fonctions g_{ij} font ressortir généralement quatre formes permettant de dégager des solutions. La recherche des formes et la résolution d'une première permet de passer à une suivante en revenant à l'expression de T dans un repère lié à un segment inférieur [PAU81].

Les quatre formes rencontrées sont les suivantes :

- Première forme : $\alpha \cos(q_i) - \beta \sin(q_i) = 0$,
- Deuxième forme : $-\alpha \sin(q_i) + \beta \cos(q_i) = \gamma$.
- Troisième forme : $\begin{cases} d_j \cdot \sin(q_i) = \alpha \\ d_j \cdot \cos(q_i) = \beta \end{cases}$
- Quatrième forme : $\begin{cases} \alpha \cos(q_i + q_j) + \beta \cos(q_i) = x_1 \\ \alpha \sin(q_i + q_j) + \beta \sin(q_i) = x_2 \end{cases}$

Où :

$\alpha, \beta, \gamma, x_1, x_2$ sont des constantes ;

q_i, q_j sont des variables articulaires rotatives

et d_j variable articulaire prismatique.

La résolution de chacune des quatre formes utilise la fonction arctangente 2 (atan2) dans le cas de variables articulaires. La connaissance d'un paramètre permettra à la suite la détermination de la suivante. Nous allons appliquer cette méthode dans le cas du bras RRR. Pour plus de détails se référer à [PAU81] et [BEN97].

I. 2. 2. 2 APPLICATION AU BRAS ARTICULE

Le modèle géométrique inverse est obtenu en inversant l'équation (I.17). Il donne la configuration géométrique du robot en fonction de la position de l'élément terminal et ne conduit pas toujours à une solution unique.

En partant de (I.17), on obtient :

$$\theta_1 = \text{atan2}(p_y, p_x) \quad \text{ou} \quad \theta_1 = \theta_1 + 180 \quad (\text{I.19})$$

$$\theta_2 = \text{atan2}(x_2(a_3 C_3 + a_2) - a_3 x_1 S_3, x_1(a_3 C_3 + a_2) + a_3 x_2 S_3) \quad (\text{I.20})$$

$$\theta_3 = \text{atan2}(C_3, S_3) \quad (\text{I.21})$$

avec :

$$x_1 = C_1 P_x + S_1 P_y \quad (I.22)$$

$$x_2 = P_z - d_1 \quad (I.23)$$

$$C_3 = (x_1^2 + x_2^2 - a_2^2 - a_3^2) / (2 a_2 a_3) \quad (I.24)$$

$$S_3 = \pm \sqrt{1 - C_3^2} \quad (I.25)$$

On a donc quatre triplets de solutions $(\theta_1, \theta_2, \theta_3)$.

I. 2. 3 CINEMATIQUE DES VITESSES

Dans cette section on considère les transformations cinématiques par rapport à la vitesse . La cinématique de vitesse définit à tout instant les vitesses d'axes du robot à partir de la position et de l'orientation de son outil . On introduit alors , la dérivée par rapport au temps pour les matrices de position et d'orientation . Dans ce qui suit , nous présentons :

- La transformation d'un mouvement différentiel d'un repère à un autre ;
- La transformation d'un mouvement différentiel de l'espace cartésien à l'espace articulaire , ou vice versa .

RAPPEL

Soit la matrice $M = (m_{ij})$, $i=1 \text{ à } 4 ; j=1 \text{ à } 4$.

La dérivée des éléments de cette matrice par rapport à la variable x est la matrice dT , tel que :

$$dM = \left(\frac{\partial m_{ij}}{\partial x} \right) \quad i=1 \text{ à } 4 ; j=1 \text{ à } 4 . \quad (I.26)$$

I.2.3.1 TRANSLATION ET ROTATION DIFFERENTIELLES

Dans ce qui suit , on considère seulement le mouvement de translation et de rotation .

Soit un système de coordonnées T , on a :

$$T+dT = \text{trans}(dx, dy, dz) \text{ rot}(k, d\theta) T \quad (I.27)$$

où :

$\text{trans}(dx, dy, dz)$ est une translation de dx, dy, dz par rapport à la base .

et $\text{rot}(k, d\theta)$ est une rotation de $d\theta$ autour du vecteur k par rapport à la base .

qui donne

$$dT = (\text{trans}(dx, dy, dz) \text{rot}(k, d\theta) - 1) T \quad (\text{I.28})$$

dT est le changement différentiel de translation et de rotation par rapport au repère de la base .

On peut aussi exprimer le changement différentiel par rapport au repère T , tel que :

$$T+dT = T \cdot {}^T\text{trans}(dx, dy, dz) \cdot {}^T\text{rot}(k; d\theta) \quad (\text{I.29})$$

${}^T\text{trans}(dx, dy, dz)$ est une translation de dx, dy, dz par rapport au système de coordonnées T .

${}^T\text{rot}(k, d\theta)$ est une rotation de $d\theta$ autour du vecteur k par rapport au système de coordonnées T .

Dans ce cas dT est donné par :

$$dT = T ({}^T\text{trans}(dx, dy, dz) \cdot {}^T\text{rot}(k, d\theta) - 1) \quad (\text{I.30})$$

On définit alors l'opérateur différentiel Δ , par

$$\Delta = (\text{trans}(dx, dy, dz) \text{rot}(k, d\theta) - 1) \quad (\text{I.31})$$

d'après (I.28), on a donc :

$$dT = \Delta T \quad (\text{I.32})$$

qui est le changement différentiel de translation et de rotation par rapport au repère de la base

$$(I.30) \Rightarrow dT = T \Delta \quad (\text{I.33})$$

qui est le changement différentiel de translation et de rotation par rapport au repère ~~au repère~~ T .

Expression de Δ :

$$\text{On a : } \text{trans}(dx, dy, dz) = \begin{pmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{I.34})$$

$$\text{et } \text{rot}(k, d\theta) = \begin{pmatrix} 1 & -k_x d\theta & k_y d\theta & 0 \\ k_x d\theta & 1 & -k_x d\theta & 0 \\ -k_y d\theta & k_x d\theta & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{I.35})$$

avec $k = k_x i + k_y j + k_z k$

donc :

$$\Delta = \begin{pmatrix} 0 & -k_x d\theta & k_y d\theta & d_x \\ k_x d\theta & 0 & -k_x d\theta & d_y \\ -k_y d\theta & k_x d\theta & 0 & d_z \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{I.36})$$

Remarque :

La rotation différentielle $\text{rot}(k, d\theta)$ est équivalente à trois rotations successives par rapport aux axes x, y, z :

$$\text{rot}(k, d\theta) = \text{rot}(x, \delta_x) \cdot \text{rot}(y, \delta_y) \cdot \text{rot}(z, \delta_z) \quad (\text{I.37})$$

avec

$\text{rot}(x, \delta_x)$ est la rotation différentielle de δ_x par rapport à l'axe x .

$\text{rot}(y, \delta_y)$ est la rotation différentielle de δ_y par rapport à l'axe y .

$\text{rot}(z, \delta_z)$ est la rotation différentielle de δ_z par rapport à l'axe z .

$$\delta_x = k_x d\theta$$

$$\delta_y = k_y d\theta$$

$$\delta_z = k_z d\theta$$

I.2.3.2 DEFINITION DU VECTEUR DIFFERENTIEL

Soit $d = dx i + dy j + dz k$ le vecteur de translation différentielle.

$$\delta = \delta_x i + \delta_y j + \delta_z k$$

Le vecteur différentiel D exprime les composantes de translation différentielle par rapport à x, y, z , et des rotations différentielles par rapport à $\delta_x, \delta_y, \delta_z$.

$$D = \begin{pmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{pmatrix}$$

I.2.3.3 LA TRANSFORMATION DE CHANGEMENT DIFFERENTIEL ENTRE REPERE

Dans cette section : on veut calculer l'opérateur différentiel ${}^T\Delta$ par rapport au système de coordonnées T , sachant que l'opérateur différentiel Δ par rapport à la base existe.

Si on fait égalité entre (I.32) et (I.33), on trouve :

$${}^T\Delta = T^{-1} \Delta T \quad (I.38)$$

qui donne le graphe de transformation représenté par la figure(I.7):

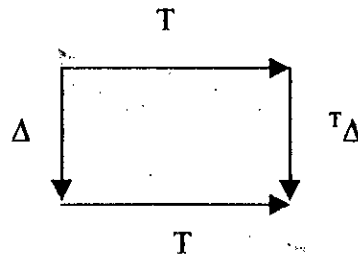


Figure I.7: Graphe de transformation du changement différentiel.

$$\text{si } T = \begin{pmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (I.39)$$

alors :

$${}^T\Delta = \begin{pmatrix} 0 & -{}^T\delta_z & {}^T\delta_y & {}^T d_x \\ {}^T\delta_z & 0 & -{}^T\delta_x & {}^T d_y \\ -{}^T\delta_y & {}^T\delta_x & 0 & {}^T d_z \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (I.40)$$

avec

$$\begin{aligned}
 {}^T d_x &= n \cdot ((\delta \times p) + d) \\
 {}^T d_y &= o \cdot ((\delta \times p) + d) \\
 {}^T d_z &= a \cdot ((\delta \times p) + d) \\
 & \\
 {}^T \delta_x &= n \cdot \delta \\
 {}^T \delta_y &= o \cdot \delta \\
 {}^T \delta_z &= a \cdot \delta
 \end{aligned}
 \tag{I.41}$$

$$\begin{pmatrix} {}^T d_x \\ {}^T d_y \\ {}^T d_z \\ {}^T \delta_x \\ {}^T \delta_y \\ {}^T \delta_z \end{pmatrix} = \begin{pmatrix} n_x & n_y & n_z & (p \times n)_x & (p \times n)_y & (p \times n)_z \\ o_x & o_y & o_z & (p \times o)_x & (p \times o)_y & (p \times o)_z \\ a_x & a_y & a_z & (p \times a)_x & (p \times a)_y & (p \times a)_z \\ 0 & 0 & 0 & n_x & n_y & n_z \\ 0 & 0 & 0 & o_x & o_y & o_z \\ 0 & 0 & 0 & a_x & a_y & a_z \end{pmatrix} \begin{pmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{pmatrix}
 \tag{I.42}$$

I.2.3.4 LA JACOBIEENNE

Dans la section précédente , on a décrit le déplacement de l'outil terminal par le vecteur différentiel D , donné dans l'espace cartésien . On définit de la même manière le vecteur

$D_q = (dq_1 \quad dq_2 \quad dq_3 \quad dq_4 \quad dq_5 \quad dq_6)$ pour un bras à 6ddl , celui ci est donné dans l'espace articulaire .

La matrice Jacobienne J qui définit les vitesses dans l'espace cartésien, relie les vecteurs D et D_q par :

$$D = J \cdot D_q
 \tag{I.43}$$

Chaque vecteur colonne de la matrice J consiste en une rotation différentielle et une translation différentielle correspondant à un changement différentiel dq_i pour chaque articulation .

I.2.3.5 METHODE POUR LE CALCUL DE LA JACOBIEENNE

Ils existent plusieurs méthodes pour le calcul de la matrice jacobienne , nous présentons une méthode qui est basée sur le calcul des matrices de transformation homogène :

Soit la matrice de transformation homogène 0T_6 , d'après la section (I.2.1.3) , on peut écrire :

$${}^0T_6 = {}^0A_1 \times {}^1A_2 \times {}^2A_3 \times {}^3A_4 \times {}^4A_5 \times {}^5A_6$$

$$\frac{\partial T_6}{\partial q_i} = {}^0T_6 {}^{T_6} \Delta_i$$

(I.44)

$${}^{T_6} \Delta_i = (A_i A_{i+1} \dots A_6)^{-1} {}^{i-1} \Delta_i (A_i A_{i+1} \dots A_6)$$

et $T = A_i A_{i+1} \dots A_6$ appliqué au système d'équation (I.32) ;

Deux cas se présentent :

1^{er} cas :

l'articulation est pivot, où les d_i sont nuls :

Donc (I.41) avec $d_i=0$ donne

$${}^{T_6} d_{ix} = n \cdot (\delta \times p)$$

$${}^{T_6} d_{iy} = o \cdot (\delta \times p)$$

$${}^{T_6} d_{iz} = a \cdot (\delta \times p)$$

(I.45)

$${}^{T_6} \delta_{ix} = n \cdot \delta$$

$${}^{T_6} \delta_{iy} = o \cdot \delta$$

$${}^{T_6} \delta_{iz} = a \cdot \delta$$

2^{ème} cas :

l'articulation est glissière, où les δ_i sont nuls :

Donc (I.41) avec $\delta_i = 0$ donne :

$${}^{T_6} d_{ix} = n_z$$

$${}^{T_6} d_{iy} = o_z$$

$${}^{T_6} d_{iz} = a_z$$

(I.46)

$${}^{T_6} \delta_{ix} = 0$$

$${}^{T_6} \delta_{iy} = 0$$

$${}^{T_6} \delta_{iz} = 0$$

ALGORITHME

Les mouvements différentiels sont connus dans l'espace local articulaire, dans cette section nous présentons un algorithme de calcul des mouvements différentiels dans l'espace cartésien lié au référentiel de base.

On a :

$${}^0T_6 = {}^0A_1 \times {}^1A_2 \times {}^2A_3 \times {}^3A_4 \times {}^4A_5 \times {}^5A_6$$

On définit :

$${}^i T_6 = {}^i A_{i+1} \times {}^{i+1} A_{i+2} \times \dots \times {}^5 A_6 \quad (1.47)$$

Deux cas se présentent :

- 1^{er} cas :

Une articulation pivot :

$$\frac{\partial X}{\partial \theta_{i+1}} = {}^i n_y {}^i P_x - {}^i n_x {}^i P_y; \quad \frac{\partial Y}{\partial \theta_{i+1}} = {}^i o_y {}^i P_x - {}^i o_x {}^i P_y; \quad \frac{\partial Z}{\partial \theta_{i+1}} = {}^i a_y {}^i P_x - {}^i a_x {}^i P_y \quad (1.48a)$$

$$\frac{\partial \phi_x}{\partial \theta_{i+1}} = {}^i n_z; \quad \frac{\partial \phi_y}{\partial \theta_{i+1}} = {}^i o_z; \quad \frac{\partial \phi_z}{\partial \theta_{i+1}} = {}^i a_z \quad (1.48b)$$

avec $\frac{\partial X}{\partial \theta_{i+1}}, \frac{\partial Y}{\partial \theta_{i+1}}, \frac{\partial Z}{\partial \theta_{i+1}}, \frac{\partial \phi_x}{\partial \theta_{i+1}}, \frac{\partial \phi_y}{\partial \theta_{i+1}}, \frac{\partial \phi_z}{\partial \theta_{i+1}}$ les translations différentielles suivant les axes

x, y, z et les rotations différentielles suivant les axes x, y et z respectivement.

- 2^{ème} cas :

L'articulation est glissière :

$$\frac{\partial X}{\partial \theta_{i+1}} = {}^i n_z; \quad \frac{\partial Y}{\partial \theta_{i+1}} = {}^i o_z; \quad \frac{\partial Z}{\partial \theta_{i+1}} = {}^i a_z \quad (1.49a)$$

$$\frac{\partial \phi_x}{\partial \theta_{i+1}} = \frac{\partial \phi_y}{\partial \theta_{i+1}} = \frac{\partial \phi_z}{\partial \theta_{i+1}} = 0 \quad (1.49b)$$

avec $\frac{\partial X}{\partial \theta_{i+1}}, \frac{\partial Y}{\partial \theta_{i+1}}, \frac{\partial Z}{\partial \theta_{i+1}}, \frac{\partial \phi_x}{\partial \theta_{i+1}}, \frac{\partial \phi_y}{\partial \theta_{i+1}}, \frac{\partial \phi_z}{\partial \theta_{i+1}}$ translation différentielle suivant l'axe x, l'axe y, l'axe z, rotation différentielle suivant x, y, z respectivement.

1.3 GENERATION DE TRAJECTOIRES

Dans ce paragraphe, on va présenter des méthodes de génération de trajectoires dans le cas où la forme de ces dernières (dans le temps et dans l'espace) est décrite en terme de fonctions de variables articulaires. Chaque point de la trajectoire est spécifié

par une position et une orientation désirée du repère de l'outil terminale OUTIL par rapport à celui de la base BASE.

Chacun de ces points est converti en un ensemble de variables articulaires désirées en appliquant la cinématique inverse, ensuite une fonction lisse est trouvée pour chaque articulation du bras manipulateur. Cette dernière doit passer par les points intermédiaires préspecifiés.

La durée requise pour chaque segment de temps doit être la même pour chaque articulation.

On notera que la forme de la fonction variable articulaire n'est pas forcément similaire à celle des autres variables.

1.3.1 INTERPOLATION AVEC POLYNOMES CUBIQUES

Considérons le problème de déplacement de l'outil terminal d'une position initiale à une position finale. En utilisant la cinématique inverse, l'ensemble des valeurs des variables articulaires qui correspondent à la position et à l'orientation finale peut être calculé.

La position initiale étant connue dans l'espace des variables électriques, nous devons trouver pour chaque variable articulaire qui a pour valeur à t_0 et à t_f la position finale de l'articulation. Il existe plus qu'une fonction lisse $q(t)$ satisfaisant la condition précédente.

Cependant avec les conditions aux limites en position et en vitesse :

$$\begin{cases} q(t_0) = q_0 \\ q(t_f) = q_f \end{cases} \quad (1.50)$$

et

$$\begin{cases} \dot{q}(t_0) = 0 \\ \dot{q}(t_f) = 0 \end{cases} \quad (1.51)$$

Ces conditions peuvent être satisfaites par un polynôme d'ordre au moins égal à trois.

La résolution [BEN97] donne :

$$q(t) = q_0 + 3 \frac{q_f - q_0}{t_f^2} t^2 - 2 \frac{q_f - q_0}{t_f^3} t^3 \quad (1.52)$$

En général il existe des points intermédiaires dans la trajectoire du robot par lesquels le bras doit passer avec une vitesse donnée. Dans ce cas les conditions initiales de l'équation (I. 51) deviennent :

$$\begin{cases} \dot{q}(t_0) = \dot{q}_0 \\ \dot{q}(t_f) = \dot{q}_f \end{cases} \quad (I.53)$$

La solution polynomiale d'ordre trois est :

$$q(t) = q_0 + \dot{q}_0 t + \left(\frac{3(q_f - q_0)}{t_f^2} - \frac{2\dot{q}_0 + \dot{q}_f}{t_f} \right) t^2 - \left(\frac{2(q_f - q_0)}{t_f^3} - \frac{\dot{q}_f + \dot{q}_0}{t_f^2} \right) t^3 \quad (I.54)$$

on peut calculer le polynôme cubique liant n'importe quelles positions initiales et finales avec des vitesses initiales et finales quelconques.

Si on a les vitesses articulaires désirées en chaque point, on pourra tout simplement utiliser (I.54) pour chaque segment afin de trouver les polynômes requis.

Il existe diverses méthodes pour spécifier les vitesses aux points intermédiaires :

- 1- L'utilisateur spécifie la vitesse en chaque point intermédiaire en terme de vitesses linéaires (cartésiennes) et angulaires de l'outil terminal à l'instant voulu.
- 2- Le système choisit automatiquement les vitesses aux points intermédiaires en appliquant une heuristique convenable et ce dans l'espace cartésien ou électrique.
- 3- Le système choisit automatiquement les vitesses aux points intermédiaires de telle manière à ce que l'accélération soit continue en ces points.

Dans la première option, les vitesses cartésiennes aux points intermédiaires sont liées aux vitesses angulaires par la jacobienne inverse évaluée en ces points. Si la jacobienne est singulière en un point intermédiaire, l'utilisateur n'a pas le choix d'une vitesse arbitraire en ce point.

Quoique très utile qu'un algorithme de génération de trajectoires soit capable de faire passer le manipulateur par un point avec une vitesse imposée par l'utilisateur, ceci risque de constituer un lourd fardeau pour l'utilisateur car il est contraint de donner toujours ces spécifications. C'est ainsi qu'un système convenable doit inclure la deuxième et/ou la troisième option.

Parfois on veut donner des conditions aux limites pour les accélérations, dans ce cas un polynôme d'ordre au moins égal à cinq est nécessaire. La résolution se fait de manière analogue [CRA86].

I.3.2 GENERATION DE TRAJECTOIRES CYCLOÏDALES

La trajectoire cycloïdale présente des fois un intérêt surtout quand une continuité d'ordre deux est exigée. La fonction cycloïdale est donnée par :

$$q(t) = A (\omega t - \sin (\omega t)) + B \quad (1.55)$$

ce qui donne pour vitesse et accélération :

$$\begin{cases} \dot{q}(t) = A\omega(1 - \cos(\omega t)) \\ \ddot{q}(t) = A\omega^2 \sin(\omega t) \end{cases} \quad (1.56)$$

La condition aux limites (équation 1.50) donne la solution :

$$q(t) = \frac{q_f - q_0}{2\pi} \left(2\pi \frac{t}{t_f} - \sin\left(2\pi \frac{t}{t_f}\right) \right) + q_0 \quad (1.57)$$

Cette fonction présente un jerk (dérivée de l'accélération) continu. Cette propriété est très importante dans certains cas de robots. En effet une continuité du jerk se traduit par une continuité dans la variation des efforts appliqués au système (couples ou forces). La relation fondamentale de la dynamique le démontre.

I.3.3 GÉNÉRATION DE TRAJECTOIRES EN FENETRE DE VIVIANI

La fenêtre de Viviani est donnée par l'intersection entre le cylindre d'équation :

$$x^2 + \left(y - \frac{R}{2}\right)^2 = \left(\frac{R}{2}\right)^2 \quad (1.58)$$

et la sphère

$$x^2 + y^2 + z^2 = R^2 \quad (1.59)$$

Il s'agit d'une trajectoire dans l'espace qui est souvent donnée en version paramétrée.

I.3.4 CONSIDERATIONS PRATIQUES

En pratique la génération de la trajectoire fait intervenir la dynamique du système. En effet la trajectoire doit prendre en compte une valeur d'accélération maximum que peut supporter le manipulateur à tout instant ; et qui dépend de sa dynamique [CRA86]. En effet la plupart des actionneurs ne sont pas caractérisés par une valeur maximale de couple ou d'accélération mais par une courbe couple-vitesse.

CONCLUSION

Dans ce chapitre nous avons étudié les aspects essentiels de la cinématique des robots. L'application de la méthode DH permet le passage de l'espace articulaire à l'espace cartésien de manière systématique. Cependant la non-bijectivité du modèle cinématique directe (fonction f de l'équation 1.3) fait que la résolution de la cinématique inverse est plus délicate.

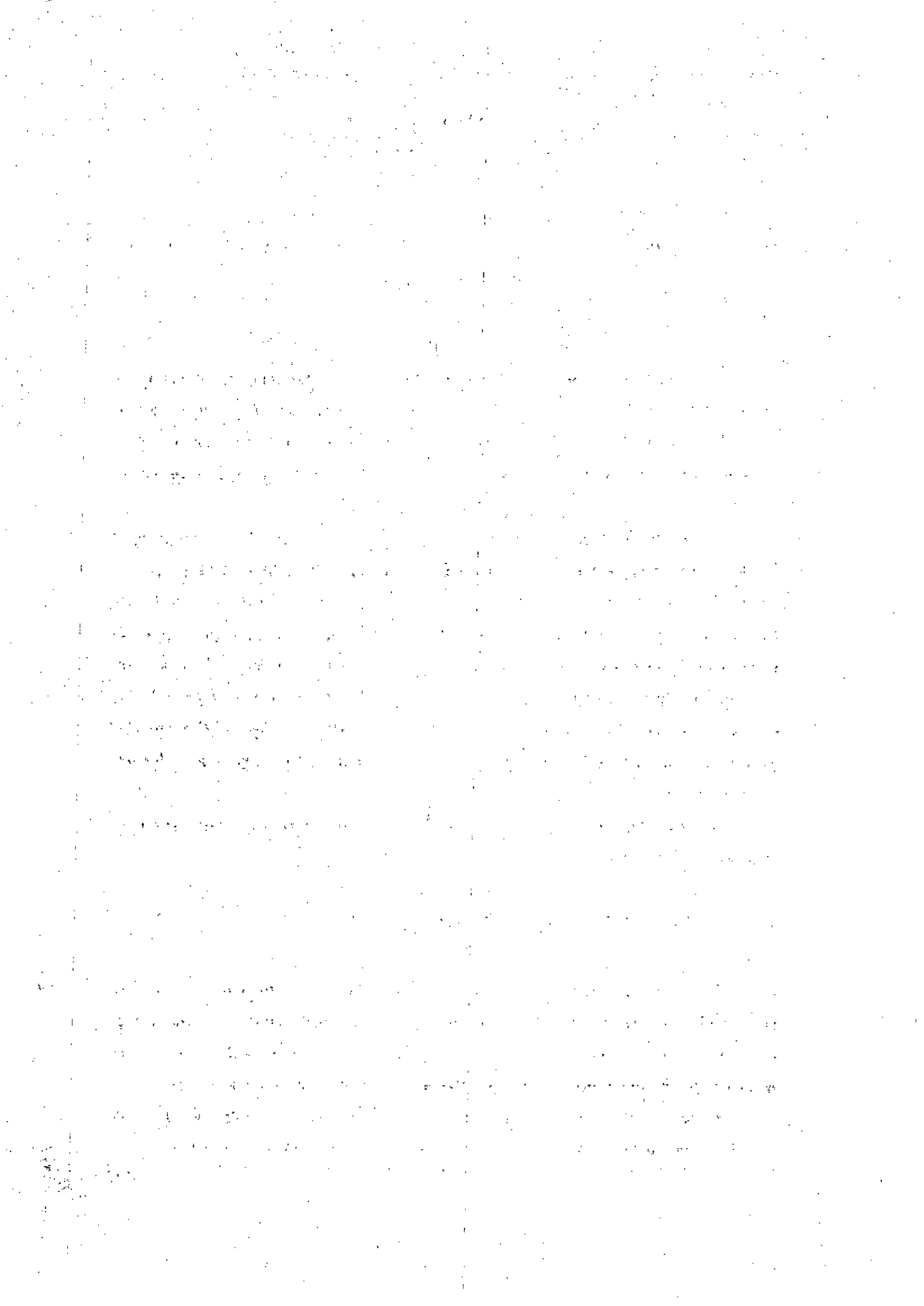
La méthode analytique est une méthode trigonométrique qui fait appel aux transformations homogènes et est applicable sur la plupart des manipulateurs industriels. Cependant dès que le degré de couplage entre les différentes variables articulaires est très accentué, la résolution qui est d'une certaine mesure intuitive devient difficile voire impossible. En plus cette méthode ne donne aucune indication sur le choix d'une solution parmi toutes celles obtenues par la résolution.

La cinématique inverse intervient dans la pratique comme on l'a vue dans la génération de trajectoire. Ce qui lui donne beaucoup d'importance malgré l'inexistence d'algorithme unifié.

Avec la cinématique de vitesse, on peut avoir :

- La vitesse d'un repère par rapport à un autre.
- La connaissance de la matrice Jacobienne J permet de définir les vitesses dans l'espace cartésien en connaissant les vitesses dans l'espace articulaire, le chemin inverse est possible en utilisant la jacobienne inverse.

CHAPITRE II
MODELISATION
DYNAMIQUE



Chapitre II

MODELISATION DYNAMIQUE

INTRODUCTION

Le modèle dynamique d'un bras manipulateur est l'ensemble des équations mathématiques décrivant le comportement dynamique du système. De telles équations sont utilisées pour la simulation par ordinateur, du mouvement du bras manipulateur ainsi pour la conception d'une commande en vue de satisfaire les performances imposées.

Le modèle du bras manipulateur peut être déterminé à partir des lois connues de la physique ; telles que celles de *NEWTON* et de *LAGRANGE*. Des approches conventionnelles comme l'approche de *LAGRANGE-EULER* [L-E] ou celle de *NEWTON -EULER* [N-E] peuvent aussi être systématiquement appliquées pour la détermination du modèle dynamique du bras manipulateur. Cependant, il est à noter qu'il existe plusieurs variantes de ces deux approches telles que « *Uicker's-L.E* », « *Hollerbach's recursive Lagrange* », « *Lee's generalized d'Alembert G.D.* » etc. Ces approches peuvent aboutir à des équations dont la structure est différente, car elles ont été conçues pour différentes tâches.

Dans ce qui suit, nous allons uniquement présenter les approches de Lagrange-Euler et Newton-Euler.

II. 1 APPROCHE DE LAGRANGE-EULER

Les équations générales du mouvement du bras manipulateur peuvent être convenablement exprimées par l'application directe du formalisme de Lagrange-Euler, celui-ci, associé à la représentation de D-H, peut constituer un algorithme efficace et compact afin d'aboutir aux équations décrivant le mouvement du bras manipulateur.

L'algorithme est exprimé sous forme matricielle, ce qui facilite l'analyse et l'implémentation sur ordinateur. La détermination des équations dynamiques d'un bras

manipulateur à n ddl (degrés de liberté) est basée sur le fait que la matrice homogène de transformation relie le référentiel de la liaison i est celui de la liaison i-1.

L'équation de L-E est donnée par :

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \left(\frac{dq_i}{dt} \right)} \right] - \frac{\partial \mathcal{L}}{\partial q_i} = T_i \tag{II. 1}$$

où :

L : Fonction de Lagrange, $L = E_c - E_p$.

E_c : Energie cinétique totale du bras manipulateur.

E_p : Energie potentielle totale du bras manipulateur.

q_i : $i^{\text{ème}}$ coordonnée généralisée.

dq_i/dt : dérivée première par rapport au temps de q_i

T_i : force généralisée appliquée à la $i^{\text{ème}}$ articulation.

II. 1. 1 ENERGIE CINETIQUE DU ROBOT MANIPULATEUR

Le formalisme de Lagrange-Euler exige la connaissance de l'énergie cinétique du système physique, qui à son tour exige la connaissance de la vitesse de chaque liaison.

Soient ${}^i r_i$ les coordonnées homogènes d'un point fixe appartenant à la liaison i, par rapport au référentiel de la liaison i, ${}^0 r_i$ ses coordonnées par rapport au référentiel fixe de la base, on a donc :

$${}^0 r_i = A_i^0 \times {}^i r_i \tag{II. 2}$$

avec : $A_i^0 = A_1^0 \times A_2^1 \times \dots \times A_i^{i-1}$.

Puisqu'il s'agit d'une liaison rigide, il s'ensuit que : $\frac{d^i r_i}{dt} = 0$,

d'où :

$${}^0 v_i = \frac{d^0 r_i}{dt} = {}^i r_i \left(\frac{d^0 A_i}{dt} \right) \tag{II. 3}$$

D'une manière générale, on trouve :

$${}^0v_i = \sum_{j=1}^i \frac{\partial A_i^0}{\partial q_j} \frac{dq_j}{dt} \cdot {}^i r_i \quad (\text{II. 4})$$

En remarquant que la dérivée précédente de la matrice A_i^0 peut être obtenue aisément à l'aide de la matrice Q_i définie pour une structure à articulations rotationnelles, comme suit :

$$Q_j = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{II. 5})$$

et pour une structure à articulations prismatiques, comme suit :

$$Q_j = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{II. 6})$$

on obtient :

$$U_{ij} = \frac{\partial A_i^0}{\partial q_j} = \begin{cases} 0 & \text{si } j \geq i. \\ A_{j-1}^0 \cdot Q_j \cdot A_i^{j-1} & \text{si } j \leq i. \end{cases}$$

D'où en utilisant les notations précédentes, on aura :

$${}^0v_i = \sum U_{ij} \frac{dq_j}{dt} \cdot {}^i r_i \quad (\text{II. 7})$$

Si on désigne par E_c l'énergie cinétique de la liaison i , et par dE_c l'énergie cinétique d'une particule de masse dm de l'articulation i , on a :

$$dE_c = \frac{1}{2} \left((dx_i / dt)^2 + (dy_i / dt)^2 + (dz_i / dt)^2 \right) \cdot dm \quad (\text{II. 8})$$

$$dE_c = \frac{1}{2} \text{trace} ({}^0v_i \cdot {}^0v_i^T) \cdot dm \quad (\text{II. 9})$$

En substituant, l'équation (II.7) dans (II.9), on trouve :

$$dE_c = \frac{1}{2} \text{trace} \left(\sum_{p=1}^i \sum_{r=1}^i U_{ip} ({}^i r_i \cdot dm \cdot {}^i r_i^T) U_{ir}^T \cdot \dot{q}_p \cdot \dot{q}_r \right) \quad (\text{II. 10})$$

L'énergie cinétique totale de l'articulation i est donc donnée par :

$$E_c = \frac{1}{2} \text{trace} \left[\sum_{p=1}^i \sum_{r=1}^i U_{ip} \left(\int {}^i r_i \cdot {}^i r_i^T \cdot dm \right) \cdot U_{ir}^T \cdot q_p \cdot q_r \right] \quad (\text{II. 11})$$

Le terme entre parenthèses représente l'inertie de l'articulation et peut s'écrire sous forme matricielle comme suit :

$$\begin{pmatrix} \int x_i^2 . dm & \int x_i . y_i . dm & \int x_i . z_i . dm & \int x_i . dm \\ \int y_i . x_i . dm & \int y_i^2 . dm & \int y_i . z_i . dm & \int y_i . dm \\ \int x_i . z_i . dm & \int z_i . y_i . dm & \int z_i^2 . dm & \int z_i . dm \\ \int x_i . dm & \int y_i . dm & \int z_i . dm & \int dm \end{pmatrix} \quad (\text{II. 12})$$

Cette matrice d'inertie peut être exprimée à l'aide des moments d'inertie de la liaison i par rapport aux principales axes du référentiel (x_i, y_i, z_i).

$$\begin{pmatrix} \frac{-I_{xx} + I_{yy} + I_{zz}}{2} & I_{xy} & I_{xz} & m_i \bar{x}_i \\ I_{xy} & \frac{I_{xx} - I_{yy} + I_{zz}}{2} & I_{yz} & m_i \bar{y}_i \\ I_{xz} & I_{yz} & \frac{I_{xx} + I_{yy} - I_{zz}}{2} & m_i \bar{z}_i \\ m_i \bar{x}_i & m_i \bar{y}_i & m_i \bar{z}_i & m_i \end{pmatrix} \quad (\text{II. 13})$$

avec (x_i, y_i, z_i) coordonnées du centre de gravité du solide, exprimés dans le repère auquel il est lié.

Il s'ensuit que l'énergie totale du bras manipulateur sera la suivante :

$$E_{c \text{ totale}} = \frac{1}{2} \sum_{i=1}^n \left(\sum_{p=1}^i \sum_{r=1}^i \text{trace}(U_{ip} . J_i . U_{ir}^T) . \dot{q}_p . \dot{q}_r \right) \quad (\text{II. 14})$$

II. 1. 2 ENERGIE POTENTIELLE DU BRAS MANIPULATEUR

L'énergie potentielle de l'articulation i, E_{pi} est donné par :

$$E_{pi} = -m_i . g . (A_i^0 . {}^i r_i) \quad (\text{II.15})$$

L'énergie potentielle totale du bras manipulateur sera donc :

$$E_p = \sum_{i=1}^n -m_i . g . (A_i^0 . {}^i r_i) \quad (\text{II. 16})$$

Tel que g soit le vecteur gravitation : $g = (0 \ 0 \ -|g| \ 0)^T$.

II. 1. 3 EQUATION DYNAMIQUE DU BRAS MANIPULATEUR

La fonction de Lagrange sera donnée par :

$$L = \frac{1}{2} \sum_{i=1}^n \left(\sum_{p=1}^i \sum_{r=1}^i \text{trace}(U_{ip} \cdot J_p \cdot U_{ir}^T) \cdot \dot{q}_p \cdot \dot{q}_r \right) + \sum_{i=1}^n m_i \cdot g(A_i^0 \cdot r_i^i) \quad (\text{II. 17})$$

En appliquant l'équation (II. 1) à la fonction Lagrangienne donnée ci-dessus, on trouve :

$$T_i = \sum_{j=i}^n \sum_{k=1}^j \text{tr}(U_{jk} \cdot J_j \cdot U_{jk}^T) \cdot \frac{d^2 q_k}{dt^2} + \sum_{j=i}^n \sum_{k=1}^j \sum_{m=1}^j \text{tr}(U_{jkm} \cdot J_j \cdot U_{jm}^T) \cdot \dot{q}_k \cdot \dot{q}_m - \sum_{j=i}^n m_j g \cdot U_{ji} \cdot r_j^j \quad (\text{II. 18})$$

avec :

$$U_{ijk} = \begin{cases} A_0^{K-1} Q_K A_{K-1}^{j-1} Q_j A_{j-1}^i & k \leq j \leq i \\ A_0^{j-1} Q_j A_{j-1}^{k-1} Q_k A_{k-1}^i & j \leq k \leq i \\ 0 & i < j \quad \text{ou} \quad i < k \end{cases} \quad (\text{II.19})$$

T_i est le couple généralisé appliqué à la $i^{\text{ème}}$ articulation.

L'écriture matricielle de l'équation dynamique du mouvement du bras manipulateur est la suivante :

$$T = D(q) \cdot \frac{d^2 q}{dt^2} + h(q, \frac{dq}{dt}) + c(q) \quad (\text{II. 20})$$

Avec :

$$\bullet \quad D_{ik} = \sum_{j=\max(i,k)}^n \text{tr}(U_{jk} \cdot J_j \cdot U_{ji}^T) \quad i, k = 1, n.$$

$$\bullet \quad h_{ikm} = \sum_{j=\max(i,k,m)}^n \text{tr}(U_{jkm} \cdot J_j \cdot U_{ji}^T) \quad i, k, m = 1, n.$$

$$\bullet \quad h_i = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \cdot \dot{q}_k \cdot \dot{q}_m \quad i = 1, n.$$

$$\bullet \quad c_i = \sum_{j=i}^n -m_j \cdot g \cdot U_{ji} \cdot r_j^j \quad i = 1, n.$$

Remarques :

1. Les coefficients c_i , $D_{i k}$, $h_{i k m}$ sont fonction des variables d'articulation θ_i et des paramètres d'inertie du robot et sont appelés les coefficients dynamiques du robot manipulateur. A partir des équations données ci-dessus il est aisé de donner à chacun d'eux le sens physique correspondant.
2. Les coefficients $D_{i k}$ représentent l'accélération des variables d'articulation. Pour $i=k$, D_{ii} est relié à l'accélération de l'articulation i quand le couple T_i est appliqué. Pour $i \neq k$, $D_{i k}$ est relié au couple de réaction induit par l'accélération k et appliqué à l'articulation i ou vice versa.
3. Les coefficients c_i représentent l'effet de la charge dû à la gravité de la liaison i .
4. $h_{i k m}$ est relié aux vitesses angulaires des articulations k et m , qui en se combinant induisent un couple de réaction exercé sur l'articulation i . En particulier, pour $k=m$, il s'agira de la force centrifuge générée par la vitesse angulaire $d\theta_k / dt$ et exercée sur l'articulation i ; autrement dit il s'agira de la force de Coriolis générée par les vitesses angulaires des articulations k et m . Enfin, la physique nous montre que :

$$h_{i k m} = h_{i m k}.$$

II. 1. 4 APPLICATION

Les modèles dynamiques de certains bras tels que le RRR, Puma560 à 3 ddl, le Rhino, Scara sont donnés en annexe (cf. Annexe A2).

II. 2 APPROCHE DE NEWTON - EULER

Dans la partie précédente, on a élaboré les équations différentielles non linéaires du second ordre obtenues par le formalisme d'*Euler-Lagrange*. L'utilisation de ces équations pour déterminer les forces généralisées à partir des positions, vitesses, accélérations généralisées pour une trajectoire donnée devient une tâche ardue et fastidieuse dès que le nombre de degrés de liberté est supérieur à deux, en plus elle nécessite un temps de calcul énorme. Dans le but d'améliorer la commande en temps réel du robot, un modèle dynamique simplifié a été proposé. Ce modèle néglige les forces de Coriolis et centrifuges, et réduit remarquablement le temps de calcul de la commande. Néanmoins, l'effet Coriolis et centrifuge est très important quand le robot

est amené à suivre des trajectoires rapides. Cela dit, le modèle dynamique simplifié restreint le mouvement du robot aux plages des faibles vitesses qui ne sont pas désirables dans un environnement industriel.

La méthode de *Newton-Euler* est actuellement considérée comme l'approche la plus favorable et la plus exploitable. En effet, sa structure récursive la rend facilement implémentable sur des calculateurs numériques. Le formalisme de N-E peut être considéré comme l'inverse du modèle dynamique obtenu par l'approche d'E-L, et donc utilisé dans la commande inverse des robots manipulateurs. La méthode de N-E utilise deux étapes pour la dérivation des forces généralisées à partir des positions, vitesses et accélérations désirées. Ces deux étapes vont être déterminées dans les sections suivantes.

II. 2. 1 RAPPELS THEORIQUES

Soit le repère fixe (O, X, Y, Z) et le repère mobile (O, X^*, Y^*, Z^*) tournant autour du repère fixe à la vitesse angulaire w (figure II.1).

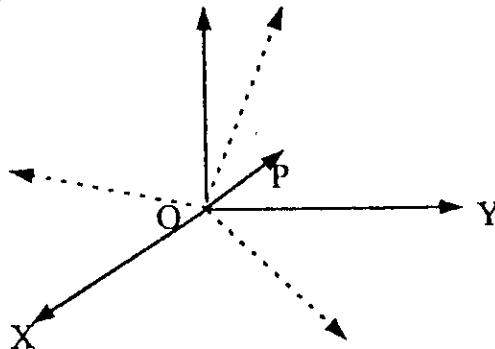


figure II.1 : Rotation d'un repère par rapport à un autre.

La vitesse et l'accélération du point mobile P par rapport au repère fixe (O, X, Y, Z) sont données par :

$$\frac{dr}{dt} = \frac{d^*r}{dt} + w \times r \quad (\text{II. 21})$$

$$\frac{d^2r}{dt^2} = \frac{d^{**}r}{dt^2} + 2w \times \frac{d^*r}{dt} + w \times (w \times r) + \frac{dw}{dt} \times r \quad (\text{II. 22})$$

tel que $\frac{d^*r}{dt}$ et $\frac{d^{*2}r}{dt^2}$ sont respectivement les vitesses et accélérations relatives du point P par rapport au repère mobile (O,X*,Y*,Z*).

Le terme $2w \times \frac{d^*r}{dt}$ représente l'accélération complémentaire ou de Coriolis, et le terme $w \times (w \times r)$ représente l'accélération centrifuge.

Dans le cas où le repère (O,X*,Y*,Z*) possède un mouvement de translation par rapport au repère fixe (figure II.2).

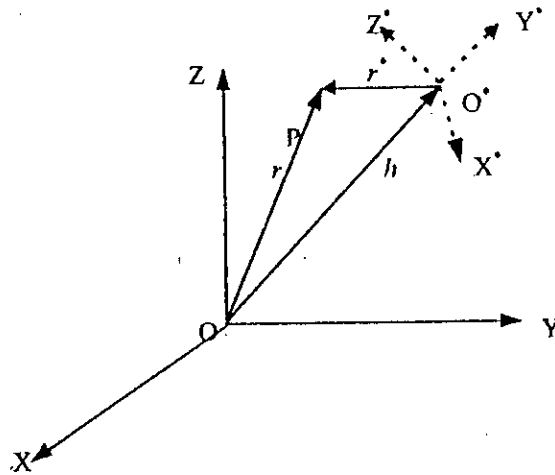


figure II.2 : Mouvement de translation et de rotation d'un repère.

La vitesse et accélération absolues du point P s'expriment par :

$$v(t) = \frac{dr^*}{dt} + \frac{dh}{dt} = \frac{d^*r^*}{dt} + w \times r^* + \frac{dh}{dt} \quad (\text{II. 23})$$

$$a(t) = \frac{d^{*2}r^*}{dt^2} + 2w \times \frac{d^*r^*}{dt} + w \times (w \times r^*) + \frac{dw}{dt} \times r^* + \frac{d^2h}{dt^2} \quad (\text{II. 24})$$

Les équations (II.22) à (II.24) vont être utilisées pour déterminer les vitesses et accélérations linéaires et angulaires de chaque liaison par rapport à la liaison adjacente.

**II. 2. 2 APPLICATION AUX REPERES LIES A LA STRUCTURE
MECANIQUE DU ROBOT**

Soit w_i et w_i^* les vitesses linéaires et angulaires du repère (x_i, y_i, z_i) par rapport à la base

(x_0, y_0, z_0) , w_i^* est la vitesse angulaire du point O_i par rapport à la base (figure II.3).

on a : $w_i = w_{i-1} + w_i^*$

avec

$$w_i^* = \begin{cases} z_{i-1} \dot{q}_i & \text{si } R \\ 0 & \text{si } T \end{cases}$$

où R désigne une liaison rotationnelle et T une liaison translationnelle.

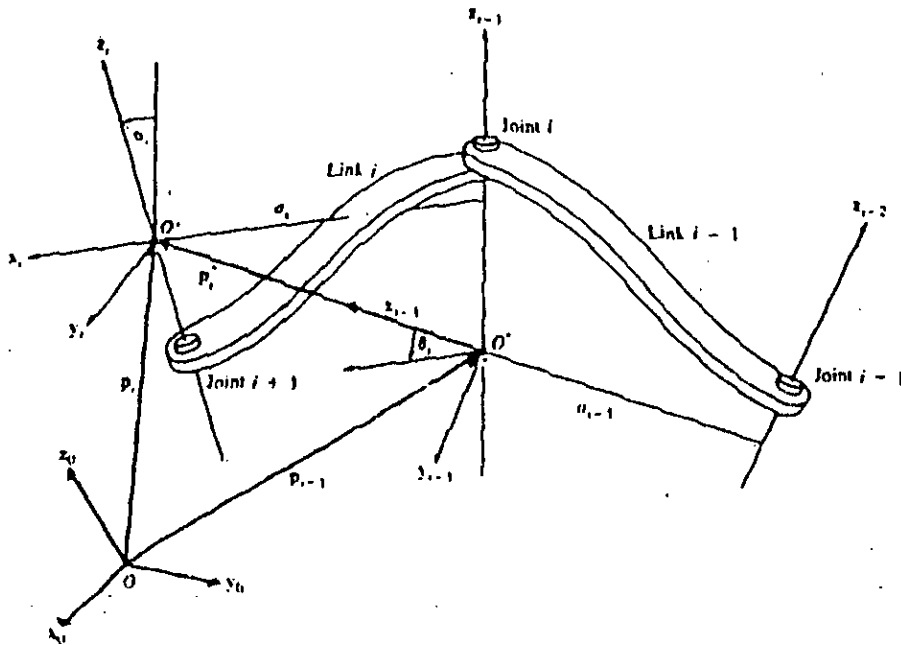


figure II.3 : Relation entre divers repères d'une chaîne articulée.

on obtient donc :

$$w_i = \begin{cases} w_{i-1} + z_{i-1} \dot{q}_i & \text{si } R \\ w_{i-1} & \text{si } T \end{cases} \tag{II. 25}$$

et

$$\dot{w}_i = \begin{cases} \dot{w}_{i-1} + z_{i-1} \ddot{q}_i + w_{i-1} \times (z_{i-1} \dot{q}_i) & \text{si } R \\ \dot{w}_{i-1} & \text{si } T \end{cases} \quad (\text{II. 26})$$

Les vitesses et accélérations linéaires de la liaison i sont obtenues en utilisant les équations (II.22) à (II.25) et peuvent être exprimées par :

$$v_i = \begin{cases} w_i \times P_i^* + v_{i-1} & \text{si } T \\ z_{i-1} \dot{q}_i + w_i \times P_i^* + v_{i-1} & \text{si } R \end{cases} \quad (\text{II. 27})$$

$$a_i = \begin{cases} \dot{w}_i \times P_i^* + w_i \times (w_i \times P_i^*) + \dot{v}_{i-1} & \text{si } T \\ z_{i-1} \ddot{q}_i + \dot{w}_i \times P_i^* + 2w_i \times (z_{i-1} \dot{q}_i) + w_i \times (w_i \times P_i^*) + \dot{v}_{i-1} & \text{si } R \end{cases} \quad (\text{II. 28})$$

II. 2. 3 EQUATIONS RECURSIVES DU MOUVEMENT DU ROBOT

Après qu'on a obtenu les informations cinématiques de chaque liaison (v_i , a_i , w_i , \dot{w}_i), l'étape suivante sera de déterminer les équations de mouvement du robot en utilisant le principe de d'Alembert. Le principe met en évidence les conditions d'équilibre statique, et s'annonce comme suit : « Pour chaque système mécanique, la somme algébrique des forces extérieures et des forces de résistance dans n'importe quelle direction et appliquées sur un point est nulle ».

Considérant une chaîne de liaisons articulées (figure II.4)

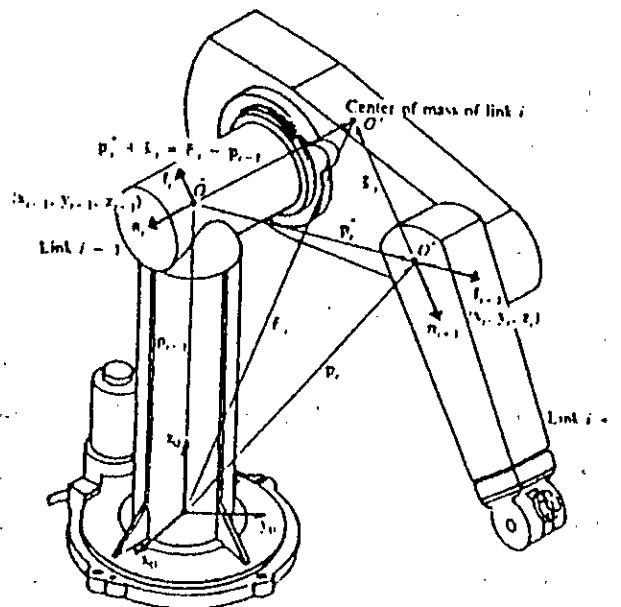


Figure II.4 : Forces et moments pour diverses liaisons.

On définit les paramètres suivants :

O_i : centre de masse de la liaison i .

\bar{s}_i : position du centre de masse par rapport au repère $R_i (x_i, y_i, z_i)$

\bar{v}_i et \bar{a}_i : vitesses et accélérations linéaires absolues du centre de masse (par rapport à la base).

F_i : la force extérieure totale exercée sur la liaison i en son centre de masse.

N_i : le moment extérieur total exercé sur la liaison i en son centre de masse.

I_i : matrice d'inertie de la liaison i calculée autour de son centre de masse et par rapport à la base.

f_i : force appliquée sur la liaison i par la liaison $i-1$ par rapport au repère R_{i-1} .

n_i : moment appliqué sur la liaison i par la liaison $i-1$ par rapport au repère R_{i-1} .

on a :

$$F_i = \frac{d(m_i \bar{v}_i)}{dt} = m_i \bar{a}_i \quad (\text{II. 29})$$

$$N_i = \frac{d(I_i \omega_i)}{dt} = I_i \dot{\omega}_i + \omega_i \times (I_i \omega_i) \quad (\text{II. 30})$$

La vitesse et l'accélération absolue du centre de masse sont données par :

$$\bar{v}_i = v_i + \omega_i \times \bar{s}_i \quad (\text{II. 31})$$

$$\bar{a}_i = \dot{v}_i + \dot{\omega}_i \times \bar{s}_i + \omega_i \times (\omega_i \times \bar{s}_i) \quad (\text{II. 32})$$

Les équations reliant les forces et moments exercées sur la liaison i à celles exercées sur la liaison $i+1$ sont données par :

$$f_i = m_i \bar{a}_i + f_{i+1} \quad (\text{II. 33})$$

$$n_i = n_{i+1} + P_i^* \times f_{i+1} + (P_i^* + \bar{s}_i) \times F_i + N_i \quad (\text{II. 34})$$

Ces équations récursives peuvent être utilisées pour obtenir les forces et moments pour chaque liaison i . f_{n+1} , n_{n+1} sont respectivement les forces et moments appliqués par l'élément terminal sur un objet extérieur du système.

Dans la modélisation cinématique, on a vu que si la liaison i est rotationnelle, elle va tourner d'un angle q_i par rapport au R_{i-1} suivant l'axe z_{i-1} . Donc le couple appliqué sur l'articulation i est la somme de la projection de n_i sur l'axe z_{i-1} et du moment visqueux. Dans le cas d'une articulation prismatique (translationnelle), cette dernière va se déplacer de q_i le long de l'axe z_{i-1} . La force appliquée sur l'articulation i est donc la somme de la projection de f_i sur l'axe z_{i-1} et une force due au déplacement visqueux

On obtient donc :

$$\tau_i = \begin{cases} n_i^T z_{i-1} + b_i \dot{q}_i & siR \\ f_i^T z_{i-1} + b_i \dot{q}_i & siT \end{cases} \quad (\text{II. 35})$$

Les équations de N-E se composent de deux ensembles d'équations récursives : les équations directes « *forward equations* » données par (II.9), (II.10), (II.12) et (II.16), et les équations de retour « *backward equations* » données par (II.13), (II. 14), (II.17) et (II.18).

Les équations directes propagent les informations cinétiques de la base jusqu'à l'élément terminal. Pour les équations de retour, les couples et force de chaque liaison sont calculés récursivement de l'élément terminal jusqu'à la base.

II. 2. 4 EQUATIONS RECURSIVES MODIFIEES

L'inconvénient majeur des équations récursives précédentes est le fait que les matrices d'inertie I_i et les paramètres géométriques (s_i, P_j^*) sont établies par rapport à la base, donc ils changent quand le robot est en mouvement. Il existe une version améliorée de la méthode de N-E qui consiste à référencer toutes les grandeurs : les vitesses, accélérations, matrices d'inertie, centres de masse, forces et moments par rapport à ces repères. Par cette nouvelle méthode le temps de calcul diminue remarquablement et devient proportionnel au nombre de dll.

Soit la matrice de rotation du repère R_i vers le repère R_{i-1} .

$${}^{i-1}R_i = \begin{pmatrix} c\theta_i & -c\alpha_i s\theta_i & -s\alpha_i s\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i \\ 0 & s\alpha_i & c\alpha_i \end{pmatrix} \quad (\text{II.36})$$

Les équations de N-E par rapport au repère lié à chaque liaison deviennent :

Forward equations : $i=1,2,\dots,n$

$${}^iR_0 w_i = \begin{cases} {}^iR_{i-1} ({}^{i-1}R_0 w_{i-1} + z_{i-1} \dot{q}_i) & siR \\ {}^iR_{i-1} ({}^{i-1}R_0 w_{i-1}) & siT \end{cases} \quad (\text{II. 37})$$

$${}^i R_0 w_i = \begin{cases} {}^i R_{i-1} [{}^{i-1} R_0 \dot{w}_{i-1} + z_0 g_i + ({}^{i-1} R_0 w_{i-1}) \times z_0 \dot{q}_i] & \text{si } R \\ {}^0 R_{i-1} ({}^{i-1} R_0 \dot{w}_i) & \text{si } T \end{cases}$$

$${}^i R_0 a_i = \begin{cases} ({}^i R_0 \dot{w}_i) \times ({}^i R_0 P_i^*) + ({}^i R_0 \dot{w}_i) \times [({}^i R_0 w_i) \times ({}^i R_0 P_i^*)] + {}^i R_{i-1} ({}^{i-1} R_0 a_{i-1}) & \text{si } T \\ {}^i R_{i-1} [z_0 \ddot{q}_i + {}^{i-1} R_0 a_{i-1}] + ({}^i R_0 \dot{w}_i) \times ({}^i R_0 P_i^*) + 2({}^i R_0 w_i) \times ({}^i R_{i-1} z_0 \dot{q}_i) + ({}^i R_0 w_i) \times [({}^i R_0 \dot{w}_i) \times ({}^i R_0 P_i^*)] & \text{si } R \end{cases}$$

$${}^i R_0 \bar{a}_i = ({}^i R_0 \dot{w}_i) \times ({}^i R_0 \bar{s}_i) + ({}^i R_0 w_i) \times [({}^i R_0 w_i) \times ({}^i R_0 \bar{s}_i)] + {}^i R_0 a_i$$

Backward equations : $i=n, n-1, \dots, 1$

$${}^i R_0 F_i = m_i {}^i R_0 \bar{a}_i$$

$${}^i R_0 N_i = ({}^i R_0 I_i \quad {}^0 R_i) ({}^i R_0 \dot{w}_i) + ({}^i R_0 w_i) \times [({}^i R_0 I_i \quad {}^0 R_i) ({}^i R_0 w_i)]$$

$${}^i R_0 f_i = {}^i R_0 F_i + {}^i R_{i+1} ({}^{i+1} R_0 f_{i+1})$$

$$\begin{aligned} {}^i R_0 n_i &= {}^i R_{i+1} [{}^{i+1} R_0 n_{i+1} + ({}^{i+1} R_0 P_i^*) \times ({}^{i+1} R_0 f_{i+1})] + \\ &+ ({}^i R_0 P_i^* + {}^i R_0 \bar{s}_i) \times ({}^i R_0 F_i) + {}^i R_0 N_i \end{aligned}$$

et

$$\tau_i = \begin{cases} ({}^{i-1} R_0 n_i)^T ({}^{i-1} R_{-1} z_0) + b_i \dot{q}_i & \text{si } R \\ ({}^i R_0 f_i)^T ({}^{i-1} R_{-1} z_0) + b_i \dot{q}_i & \text{si } T \end{cases}$$

II. 2. 5 ALGORITHME DE NEWTON-EULER

Conditions initiales : n nombre de degré de liberté :

$$w_0 = \dot{w}_0 = v_0 = 0 \quad a_0 = [g_x \quad g_y \quad g_z]$$

Etape1 : attribuer au compteur la valeur $i=1$

Etape2 : calcul de ${}^iR_0 w_i, {}^iR_0 \dot{w}_i, {}^iR_0 a_i, {}^iR_0 \bar{a}_i$

Etape3 : test si $i = n$ goto Etape4 sinon $i=i+1$ goto Etape2

Etape4 : calcul de f_{n+1}, n_{n+1} en cas de charge , s'il n'y a pas de charge , ils sont nuls.

Etape5 : calcul de ${}^iR_0 F_i, {}^iR_0 N_i, {}^0R_i f_i, {}^iR_0 n_i$ et τ_i .

Etape6 : si $i = 1$ goto stop sinon $i = i - 1$ goto Etape5.

CONCLUSION

L'approche d'Euler-Lagrange de nature analytique n'est pas adéquat dans le cas où le nombre de degré de liberté dépasse 2, cela est dû aux calculs ardues et fastidieux que nécessite la méthode.

L'approche de Newton-Euler de nature récursive, nous permet de rendre la dérivation des équations dynamiques du robot plus aisée. L'utilisation des vecteurs positions-vitesses-accélérations relatives, ainsi que les matrices de rotation d'un repère à un autre, donne l'aspect récursif à l'algorithme et le rend facilement implémentable sur calculateurs numériques. L'inconvénient de ce formalisme est sa forme non structurée, et donc, ne peut pas être utilisé pour la synthèse de la commande.

CHAPITRE III
COMMANDE
CLASSIQUE

Faint, illegible text, possibly bleed-through from the reverse side of the page. The text is arranged in several paragraphs and appears to be a formal document or report.

Chapitre III

COMMANDE CLASSIQUE

INTRODUCTION

Pour réaliser une tâche désirée, le robot doit suivre une trajectoire bien définie dans le temps comme dans l'espace (la commande). Ce qui veut dire qu'une action de contrôle est nécessaire afin que la configuration prise par le système soit asservie à cette commande et ne dépende que de celle-ci.

La commande engage en premier lieu l'appareillage utilisé aussi bien pour la mesure (capteurs) que pour la génération des efforts agissant sur les différents degrés de liberté du bras manipulateur (actionneurs). Cependant, les énormes progrès dans le domaine de la métrologie et de l'instrumentation font que la disponibilité des organes de commande et de mesure est une condition toujours réalisée [SCH94][ASC92].

Ainsi la commande se réduit à la définition de l'Algorithme de régulation : relation définissant l'écart entre la consigne (commande) et la mesure.

Les premiers types d'algorithmes ont été basés sur une analyse mathématique du processus à réguler. Le système est donc remplacé par sa fonction de transfert. Ce modèle est associé en boucle fermée avec une fonction de transfert linéaire choisie de façon à réaliser les conditions de compromis adéquat entre la stabilité et la précision du système global ainsi formé.

Les régulateurs PID (Proportionnelle-Intégrale-Dérivée) sont les plus anciens et les plus répandus à l'échelle industrielle en raison de leur simplicité de mise en œuvre et de leurs performances acceptables [KET92].

Dans ce chapitre nous allons exposer brièvement le principe de tels régulateurs et leur application à la commande de bras manipulateurs. Mais avant cela, ayons une idée sur la fonction de transfert de quelques robots en observant leur comportement en boucle ouverte.

III. 1 SIMULATION EN BOUCLE OUVERTE

La simulation en boucle ouverte du modèle dynamique du bras manipulateur avec des entrées de type échelon, permet d'estimer globalement leur fonction de transfert. Elle permet aussi de voir leur degré de stabilité et les interactions entre les différents segments.

III. 1. 1 APPLICATIONS

Les équations d'état obtenues constituent après l'étude dynamique un système différentiel dont la résolution numérique est possible. Nous utiliserons la méthode de Runge-Kutta d'ordre 4 pour la bonne précision qu'elle donne.

Les modèles dynamiques des bras utilisés sont donnés en Annexe 2.

III. 1. 2 BRAS SCARA

Les réponses du robot Scara à 2ddl en position et en vitesse pour une entrée échelon (u_1 , u_2) sur un intervalle de temps de 10 secondes sont présentées par les figures suivantes.

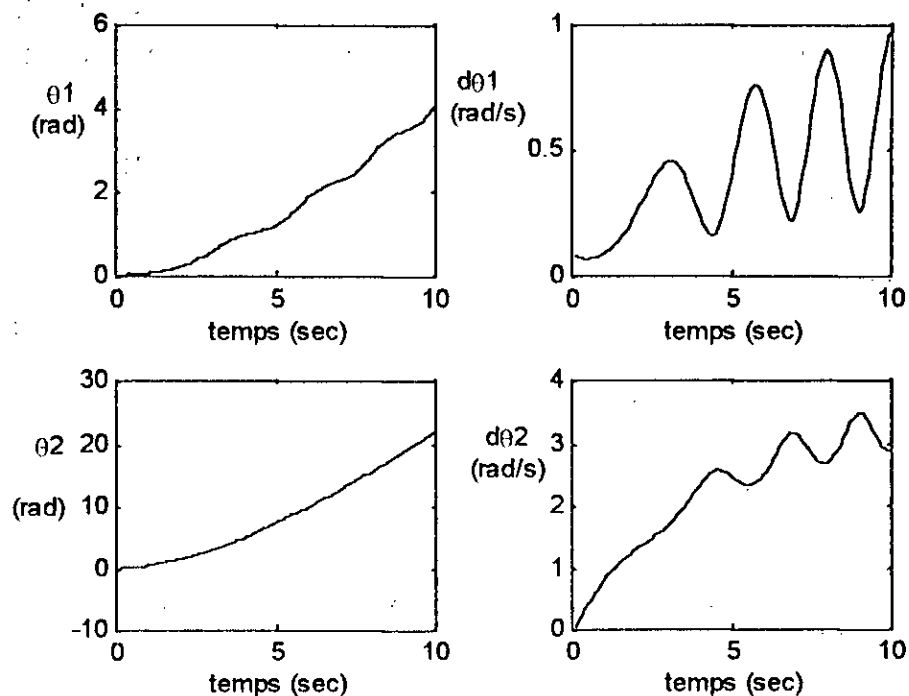


Figure III.1 : Réponse indicielle du bras Scara $u_1=u_2=1$.

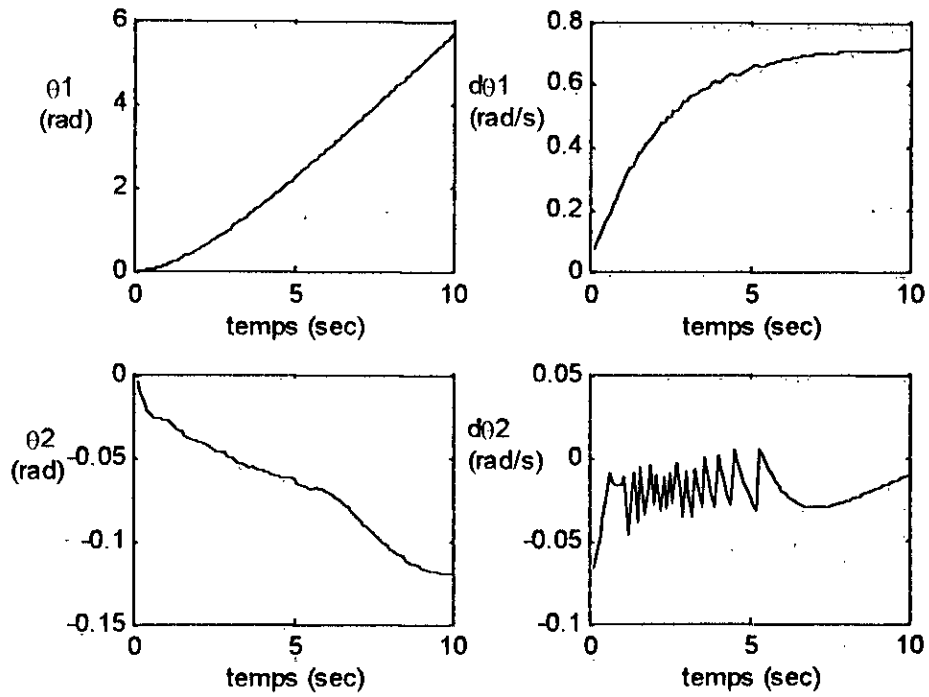


Figure III.2. Réponse indicielle du bras Scara $u_1 = 1, u_2 = 0$.

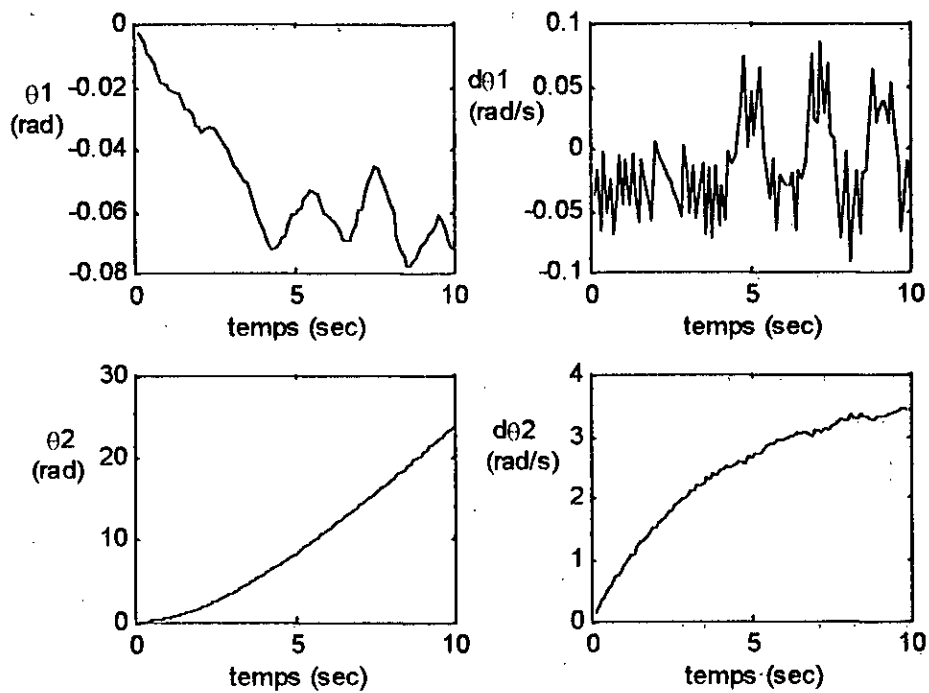


Figure III.3 : Réponse indicielle du bras Scara $u_1=0, u_2=1$.

ANALYSE ET INTERPRETATION

D'après la figure III.1, les variables articulaires θ_1 et θ_2 décrivent une parabole, donc le comportement du robot est semblable à celui d'un double intégrateur. Afin de mesurer le degré de couplage entre les articulations, on attaque une articulation par un échelon alors que l'autre est libre

La figure III.2 montre le couplage entre la première articulation et la deuxième. Il s'agit d'un couplage faible vu la dynamique de θ_2 .

La figure III.3 montre l'effet du deuxième segment sur le premier ; ce dernier est presque immobile.

Des deux dernières figures, Il apparaît clairement que ce robot est un système faiblement couplé.

III. 1. 3 BRAS RRR

Les réponses du bras articulé en position et en vitesse pour une entrée échelon sur un intervalle de temps de 10 secondes sont présentées par les figures suivantes :

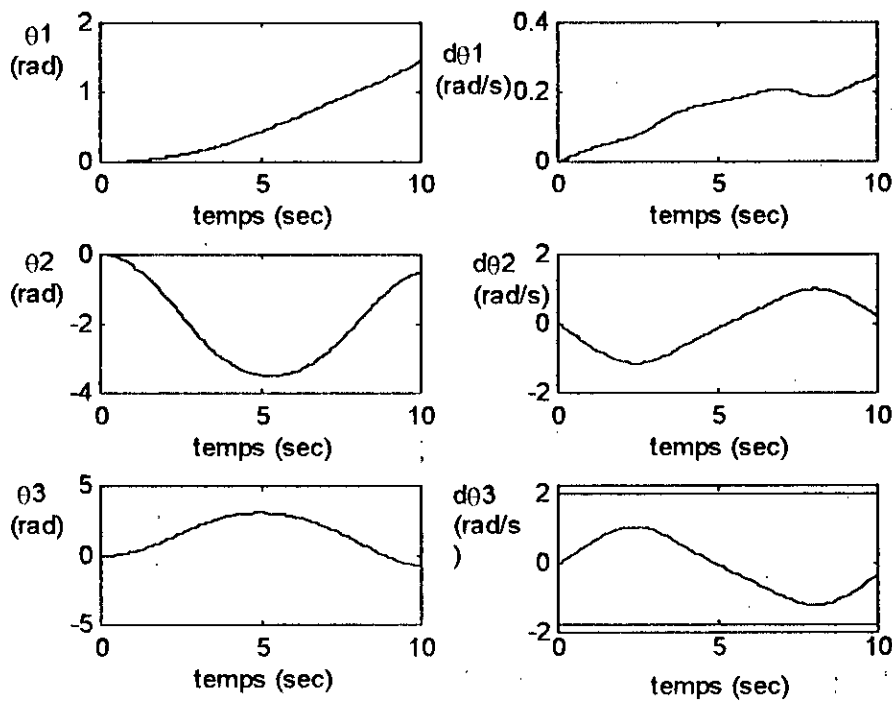


Figure III.4. Réponse indicielle du bras articulé pour $u_1 = u_2 = u_3 = 1$.

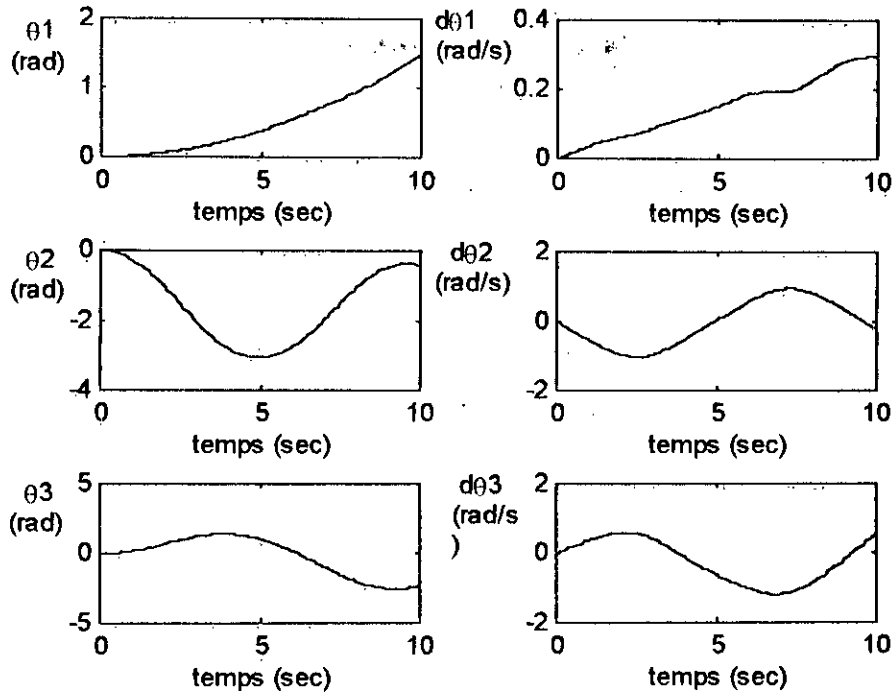


Figure III.5. Réponse indicielle du bras articulé pour $u_1 = 1, u_2 = u_3 = 0$.

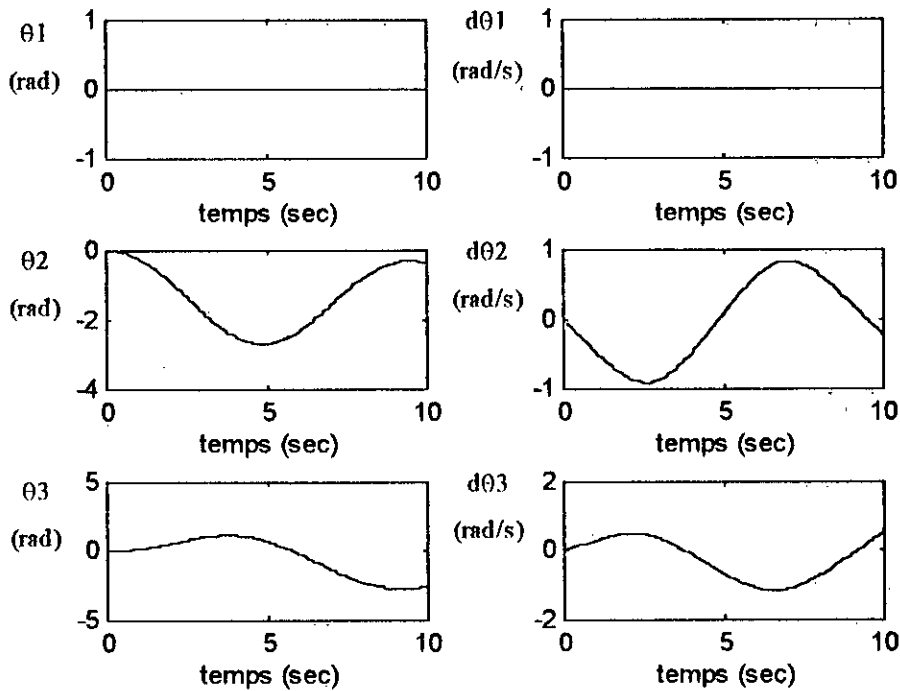


Figure III.6. Réponse indicielle du bras articulé pour $u_1 = u_3 = 0, u_2 = 1$.

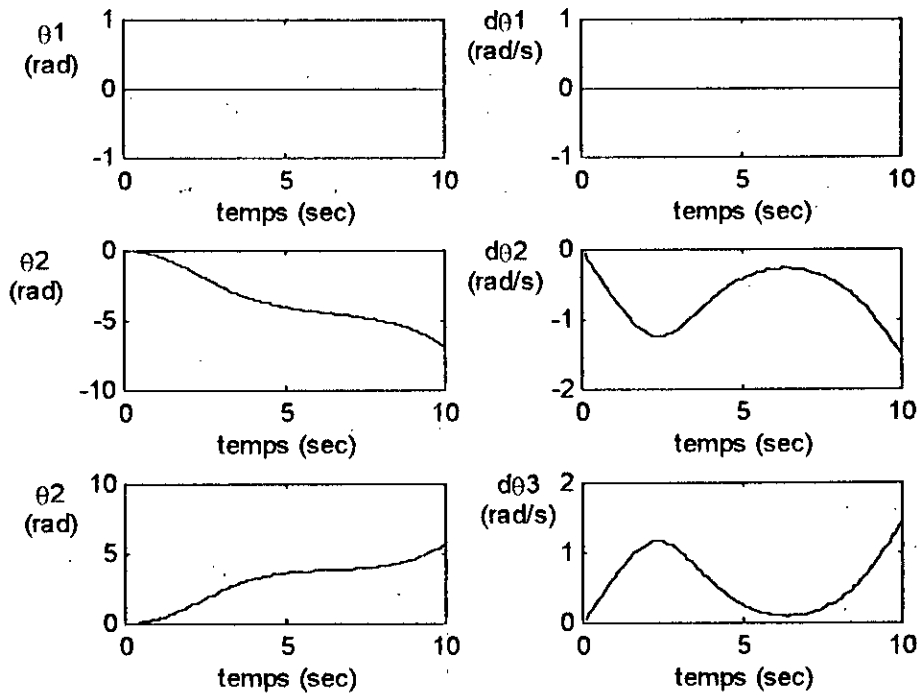


Figure III.7. Réponse indicielle du bras articulé pour $u_1 = u_2 = 0, u_3 = 1$.

ANALYSE ET INTERPRETATION

Il est clair d'après la figure III.4 que le comportement du bras articulé est semblable à celui d'un double intégrateur (les trois variables articulaires décrivent des paraboles).

Dans le but de mesurer le degré de couplage entre articulations, on a attaqué une articulation

Par un échelon alors que les autres sont libres. La figure III.5 montre le couplage entre la première articulation et les deux autres, il s'agit d'un couplage important vu la dynamique de θ_2 et θ_3 . La figure III.6 montre l'effet du deuxième segment sur les deux autres, on constate que le premier segment est immobile, le troisième segment décrit presque le même mouvement que le deuxième mais dans le sens contraire. On peut interpréter cela par le fait que le manipulateur tend à revenir à sa position de départ (principe des moindres actions).

La figure III.7 montre l'effet du troisième segment sur les autres. Alors que le premier reste immobile, le deuxième segment présente le même mouvement que le troisième mais avec un sens opposé.

On conclut que :

- Le premier segment influe sur les deux autres mais le contraire n'est pas vrai.
- Ce robot est un système fortement couplé.

Le système est instable en boucle ouverte, donc nécessite une commande en boucle fermée.

III. 1. 4 Bras TRT

Voir Annexe 3.

III. 2 IMPLEMENTATION D'UNE LOI DE COMMANDE PID

III. 2. 1 INTRODUCTION

Pour améliorer les performances d'un système asservi on doit trouver un compromis entre la stabilité et la précision qui sont deux grandeurs antagonistes. Le point de vue de l'automatique classique consiste à introduire dans la chaîne du système des fonctions de transfert convenablement choisies afin que leur existence puisse changer de manière bénéfique la transmittance du système [figure III-8]. On parle de correcteur.

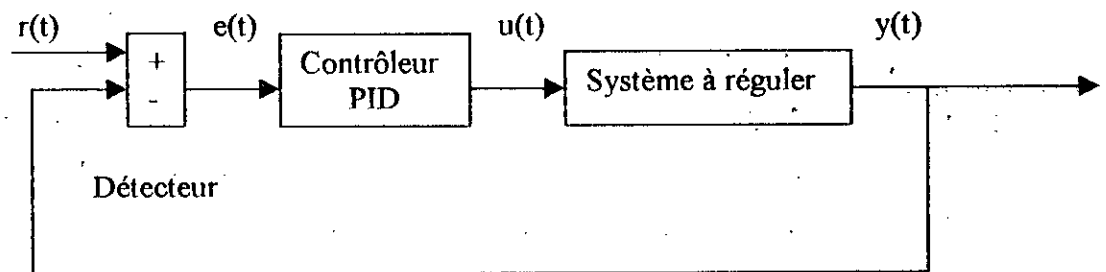


Figure III.8 : schéma bloc d'un contrôleur PID.

Le correcteur peut être de type :

- PD (avance de phase),
- PI (retard de phase),
- PID.(avance et retard de phase).

Dans ce qui suit, nous présentons une étude générale du contrôleur PID, nous allons ensuite appliquer la commande PID sur le bras Scara et Articulé RRR à 3ddl, nous discutons les avantages et les inconvénients de cette commande, et nous terminons ce chapitre par une conclusion générale.

III. 2. 2 ALGORITHME DES CONTROLEURS PID :

L'algorithme de base des contrôleurs PID se présente comme suit :

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (\text{III.1})$$

où

- $u(t)$ est la variable de contrôle.
- $e(t)$ est l'erreur ou variable à contrôler, c'est la différence entre la consigne $r(t)$ et la sortie du système $y(t)$.

La variable de contrôle est ainsi la somme de trois termes :

- Un terme P : proportionnel à l'erreur
- Un terme I : proportionnel à l'intégrale de l'erreur
- Un terme d : proportionnel à la dérivée de l'erreur.

Les paramètres du contrôleur sont K_p : la constante proportionnelle, T_i : la constante d'intégration et T_d : la constante de dérivation.

III. 2. 3 APPLICATION

Pour assurer un bon suivi de la trajectoire, il est nécessaire d'envisager la prise en compte du modèle complet du manipulateur.

Le schéma de commande doit considérer l'ensemble des actionneurs. Le système est considéré comme un système multivariable.

Le couple de commande est :

$$\Gamma = K_v \dot{e}(t) + K_p e(t) + K_i \int_0^t e(\xi) d\xi \quad (\text{III.2})$$

Où K_p , K_v , K_i sont des matrices diagonales constantes, des gains du correcteur PID.

$e(t)$ et $\dot{e}(t)$ sont respectivement le vecteur d'erreur en variables articulaires et en vitesses articulaires.

$$e(t) = q_{id}(t) - q_i(t). \quad (\text{III.3})$$

$$\dot{e}(t) = \dot{q}_{id}(t) - \dot{q}_i(t). \quad (\text{III.4})$$

avec : q_{id} , \dot{q}_{id} sont la position et la vitesse articulaires désirées.

q_i , \dot{q}_i sont la position et la vitesse articulaires prises par le bras manipulateur.

$t = k \Delta t$; Δt est la période d'échantillonnage.

on aura :

$$\Gamma_i(k \Delta t) = K_{p_i} e_i(k \Delta t) + K_{v_i} \dot{e}_i(k \Delta t) + K_{i_i} \Delta t \sum_{j=0}^k e_i(j \Delta t). \quad (\text{III.5})$$

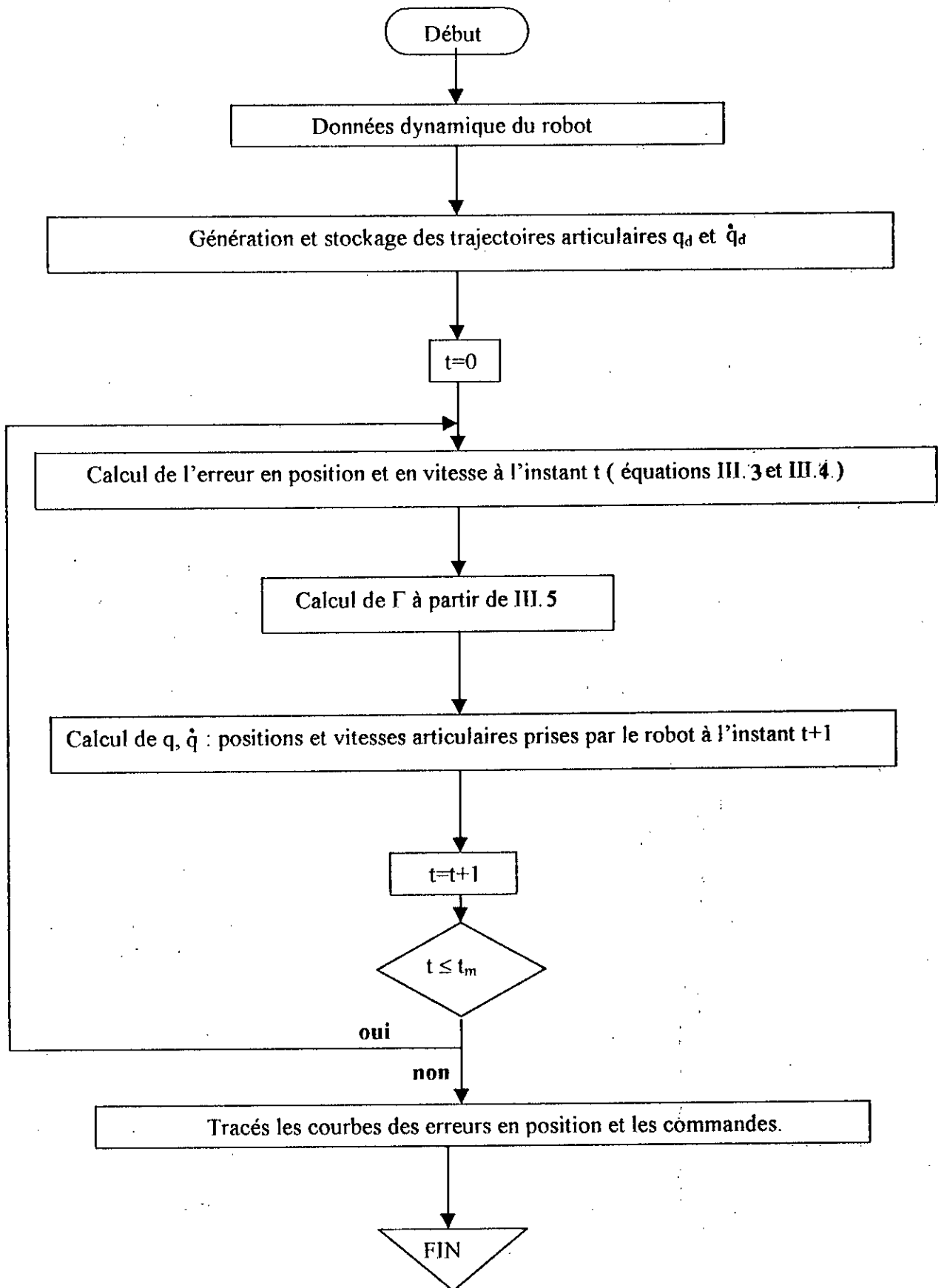
L'indice i représente le $i^{\text{ème}}$ segment du bras.

Ce type de contrôle est simple, car chaque articulation du robot est commandée séparément des autres.

Les performances d'une telle loi de commande appliquée à un système non linéaire couplé, sont très difficile à décrire. On outre, le choix convenable des matrices K_p , K_v , K_i est impossible à faire pour toutes les configurations du bras et pour des trajectoires variées. Donc le choix des gains reste arbitraire.

III. 2. 3. 1 ALGORITHME DE LA COMMANDE PID

L'organigramme ci-dessous présente l'algorithme de la commande PID.



III. 2. 3. 2 APPLICATION

III. 2. 3. 2. 1 Étude de l'influence des paramètres (Bras Scara)

Commande PID avec une consigne échelon pour

$$K_{p1}=100, K_{p2}=120, K_{v1}=120, K_{v2}=100, K_{i1}=5, K_{i2}=5$$

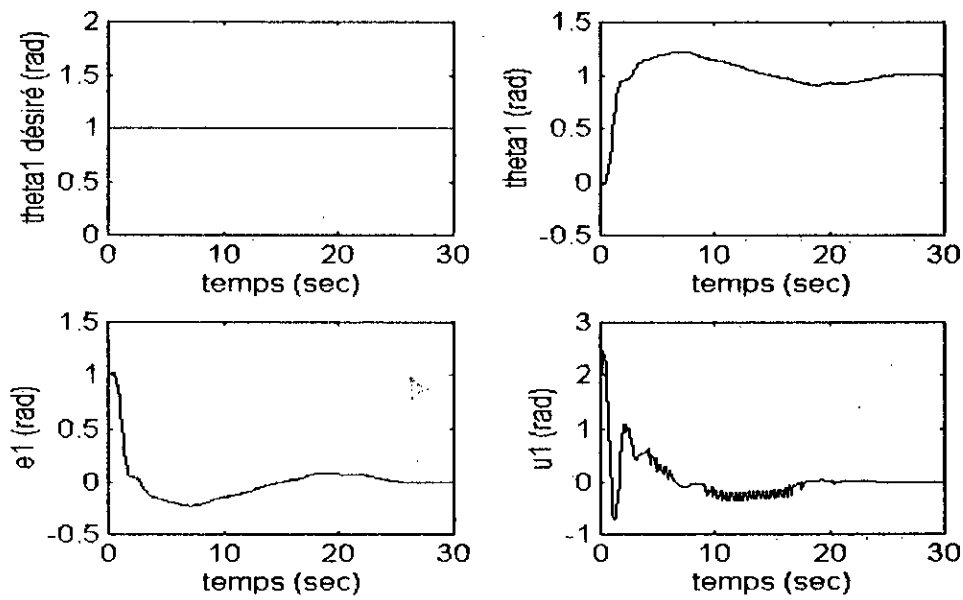


Figure III.9. Réponse du premier segment du bras Scara

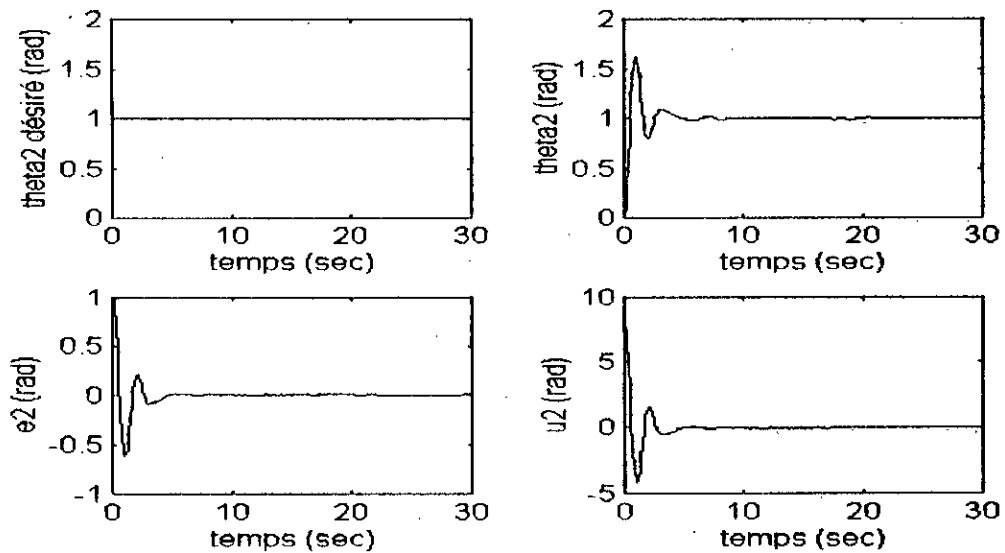


Figure III.10. Réponse du deuxième segment du bras

Commande PID avec une consigne échelon pour
 $K_{p1}=70$, $K_{p2}=120$, $K_{v1}=120$, $K_{v2}=100$, $K_{i1}=5$, $K_{i2}=5$.

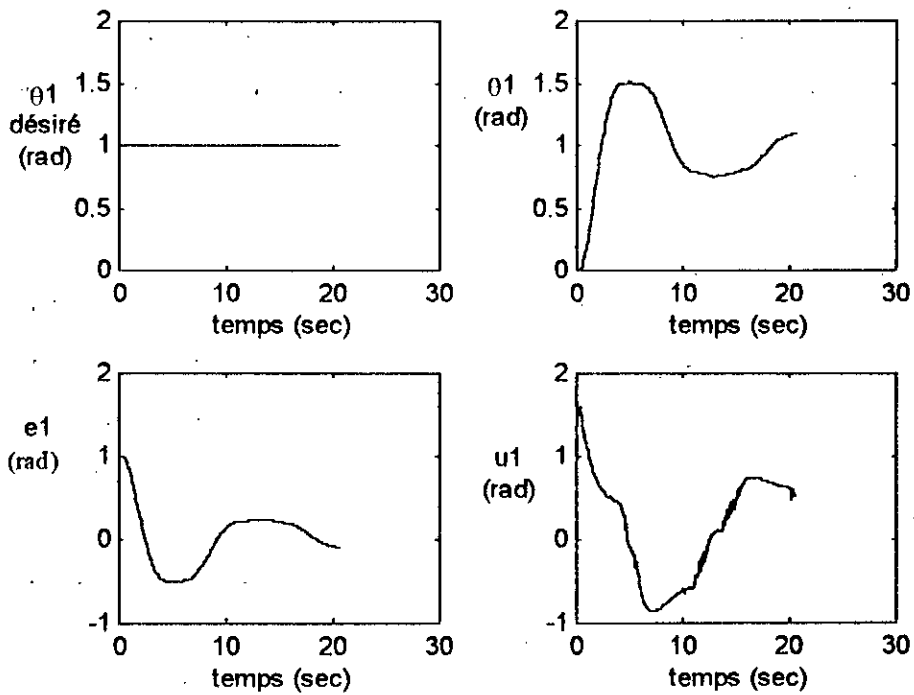


Figure III.11. Réponse du premier segment du bras Scara.

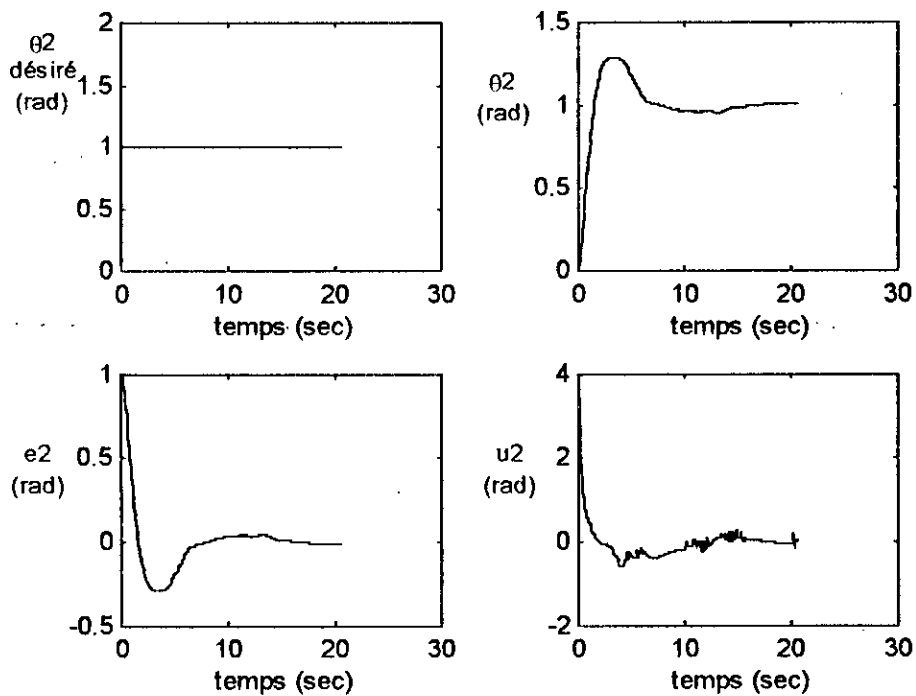


Figure III.12. Réponse du deuxième segment du bras Scara.

ANALYSE ET INTERPRETATION

La figure III.9 et la figure III.10 présentent les réponses du segment 1 et segment 2 du bras Scara pour une loi de commande de type PID, pour une poursuite de trajectoire de type échelon.

Les coefficients utilisés sont :

$$K_{p1}=100, K_{p2}=120, K_{v1}=120, K_{v2}=100, K_{i1}=5, K_{i2}=5.$$

On constate d'après ces figures que le système converge vers la position désirée, avec une erreur d'environ de 0.3 rad sur chaque segment pendant le régime transitoire.

- La figure III.11 et la figure III.12 présentent les réponses du segment 1 et segment 2 du bras Scara pour les mêmes commande et consigne que le cas précédent, pour un coefficient proportionnel pour le premier segment $K_{p1}=70$.

Dans ce cas, on voit bien que l'erreur sur le premier segment est plus importante, elle est de l'ordre de 0.5rad pendant le régime transitoire.

- D'après cette étude, on conclut que le choix des coefficients des matrices K_p, K_v, K_i influe sur la réponse du système, et un choix convenable de ces paramètres est important pour une bonne poursuite de trajectoire, donc de meilleures performances.

Remarque :

Il faut noter qu'une trajectoire de type échelon n'existe pas en pratique (un système mécanique (donc à inertie) ne peut pas se déplacer d'un point à un autre en une durée de temps nulle).

Le but du choix de cette trajectoire est de montrer l'influence des paramètres du régulateur PID.

Commande PID pour une consigne cycloïdale pour
 $K_{p1}=100$, $K_{p2}=80$, $K_{v1}=50$, $K_{v2}=50$, $K_{i1}=5$, $K_{i2}=15$.

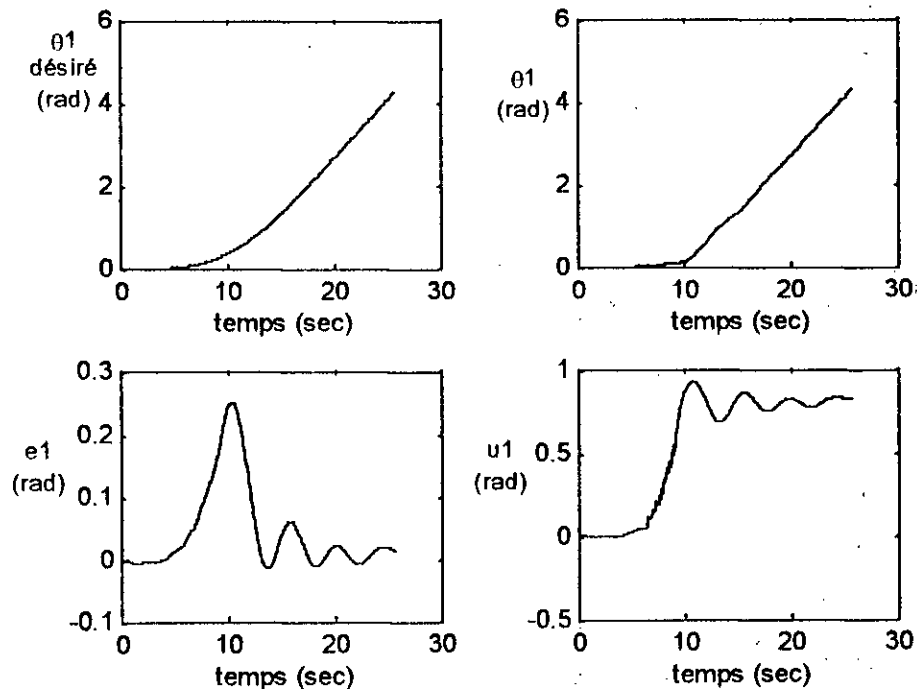


Figure III.13. Réponse du premier segment du bras Scara.

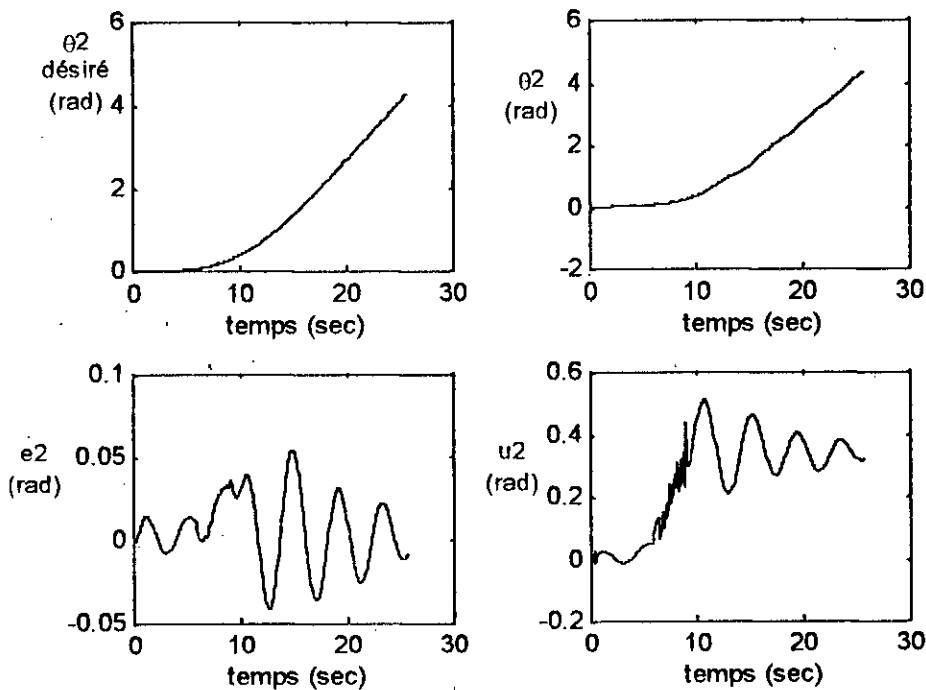


Figure III.14. Réponse du deuxième segment du bras Scara.

Commande PID pour une consigne cycloïdale pour

$$K_{p1}=50, K_{p2}=80, K_{v1}=80, K_{v2}=80, K_{i1}=15, K_{i2}=20$$

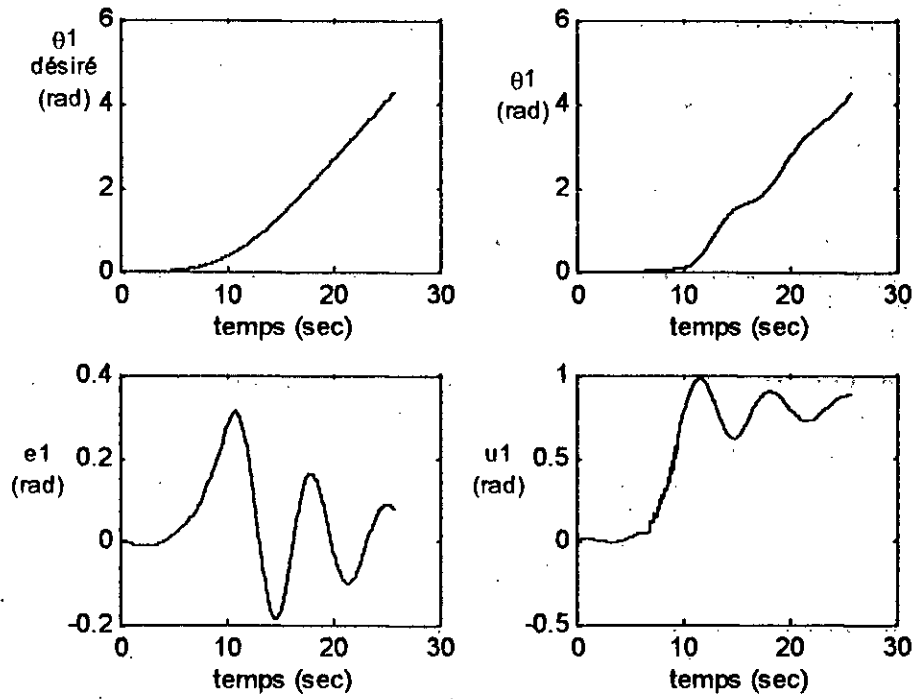


Figure III.15. Réponse du premier segment du bras Scara.

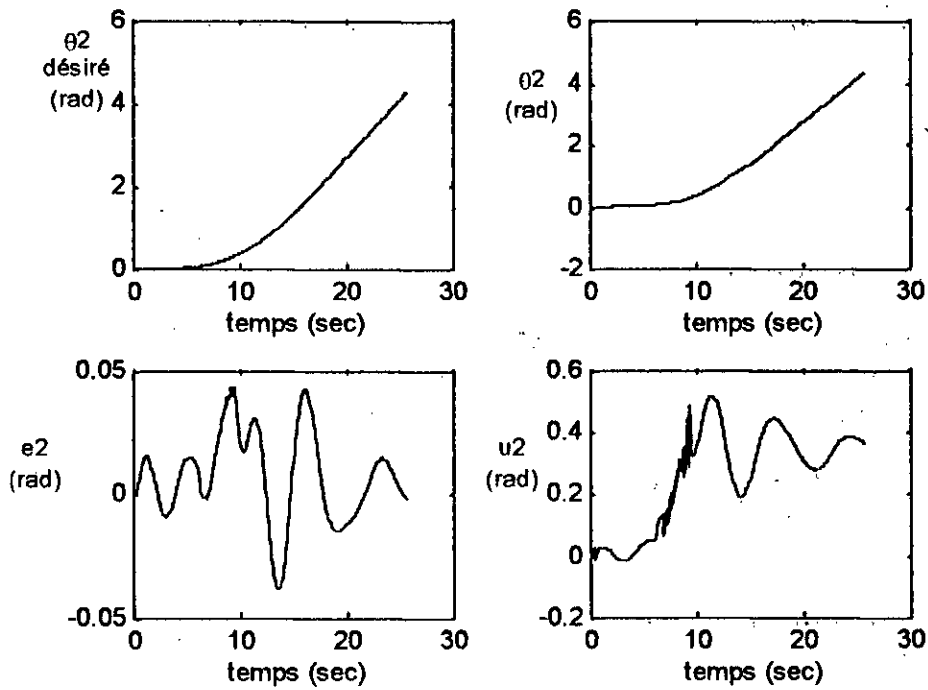


Figure III.16. Réponse du deuxième segment du bras Scara.

ANALYSE ET INTERPRETATION

Les figure III.13 et III.14 présentent les réponses du segment 1 et segment 2 du bras Scara respectivement pour une loi de commande de type PID, pour une poursuite de trajectoire de type cycloïdale.

Les coefficients utilisés sont :

$$K_{p1}=100, K_{p2}=80, K_{v1}=50, K_{v2}=50, K_{i1}=5, K_{i2}=15.$$

On constate d'après ces figures que le système converge vers la position désirée, avec une erreur importante de 0.24 rad pendant le régime transitoire pour le premier segment.

Par contre, avec le deuxième segment le système converge avec une erreur maximale de 0.05rad.

Les figures III.15 et III.16 présentent les réponses du segment 1 et segment 2 du bras Scara respectivement pour les mêmes commande et consigne, pour un coefficient proportionnel pour le segment 1 égal à 50.

Dans ce cas, nous constatons que l'erreur sur le premier segment est plus importante (0.3 rad) par rapport au cas de la figure13, le système converge lentement vers la position désirée.

De même, on conclut que le choix des coefficients des matrices K_p , K_v , K_i influe sur la réponse du système, et un choix convenable de ces paramètres est important pour une bonne poursuite de trajectoire donc pour une meilleure commande

III. 2 .3 .2. 2 Bras Articulé [BEN97]

L'application de l'algorithme de la commande PID sur le bras Articulé RRR, pour une poursuite de trajectoire de type cycloïdale, pour les coefficients du contrôleur PID suivants :

$$K_{p1}=2670, K_{p2}=2670, K_{p3}=333;$$

$$K_{v1}=30, K_{v2}=120, K_{v3}=80;$$

$$K_{i1}=K_{i2}=K_{i3}=5;$$

Les résultats de simulation sont résumés par la figure (III.17)

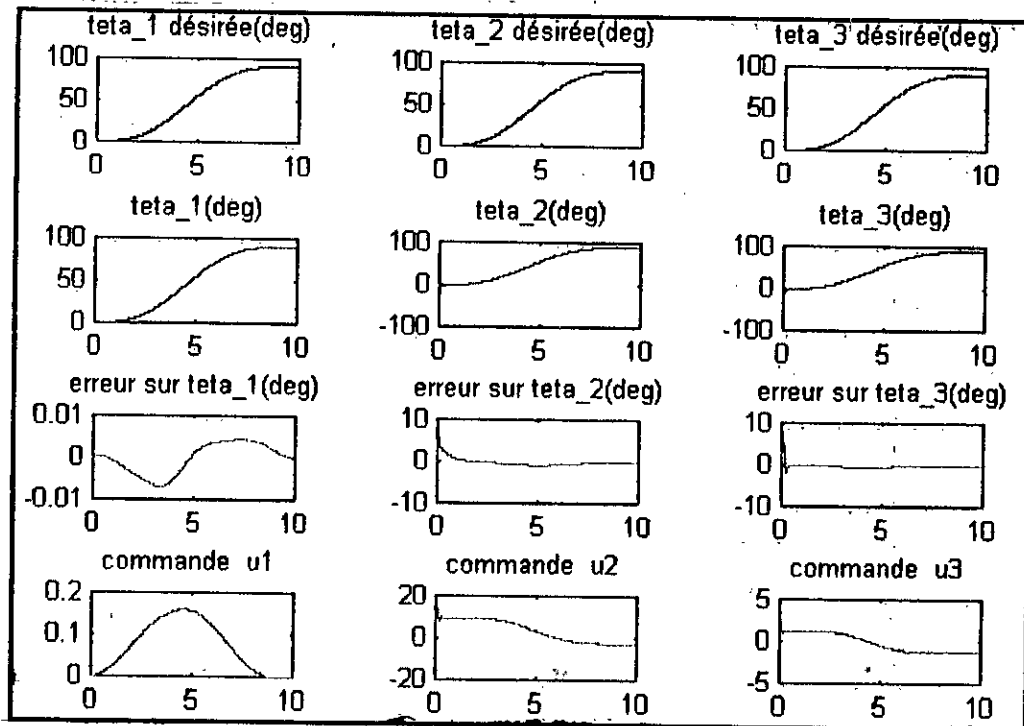


Figure III.17. Réponse du bras RRR pour une commande classique PID

ANALYSE ET INTERPRETATION

On risque d'avoir des erreurs statiques finales de position pour les variables articulaires. Le choix d'une valeur assez faible de K_i permet d'éliminer ces erreurs finales sans pour autant s'éloigner du second ordre.

III. 2. 3. 2. 3 Bras TRT

Voir Annexe3.

CONCLUSION

Après avoir présenté les problèmes de commande d'un système robotique à travers la commande classique PID, on peut conclure que:

- Le choix des gains K_p , K_v , K_i est empirique (Le correcteur ne prend pas en charge les évolutions temporelles du système à commander), en plus ce choix n'est valable que pour une consigne particulière.
- La forme de l'erreur de suivi de la trajectoire n'est pas prévisible, d'où le risque de perdre le contrôle et la stabilité du robot.
- Les réponses obtenues avec la commande PID présentent des caractéristiques médiocres en régime transitoire (temps de dépassement et temps de réponse importants).

L'application de cette commande nécessite la connaissance du modèle complet du bras manipulateur, qui généralement, est un système non linéaire, couplé, multivariable.

L'application d'une commande qui va se soucier moins de la difficulté mathématique liée à la considération du modèle du robot est donc la bienvenue. Et c'est en la commande floue qu'il va falloir chercher une solution. La commande se fera par une méthodologie tout à fait nouvelle qui permet de modéliser la connaissance de l'opérateur, et son contrôle conséquent à cette connaissance plutôt que les paramètres numériques du système.

CHAPITRE IV
METHODOLOGIES
DE
COMMANDE FLOUE



Chapitre IV
METHODOLOGIES
DE COMMANDE
FLOUE

INTRODUCTION

Bien que des tentatives aient été menées depuis très longtemps pour la formalisation des connaissances subjectives [ARA94], le véritable mérite de la réussite de cette nouvelle stratégie revient au Professeur Iranien Lotfi Zadeh qui a exposé pour la première fois à l'Université de Berkeley (Californie) en 1965 ce qui est connu de nos jours sous le nom de logique floue. Une telle méthodologie devrait faire le rapprochement tant attendu entre le contrôle mathématique précis basé sur le modèle du système, et la prise de décision humaine qui repose sur la connaissance « floue » de l'opérateur sur le système.

Des expériences ont montré que, dans de nombreux cas, les résultats obtenus avec un contrôleur flou sont meilleurs que ceux obtenus avec des algorithmes de contrôle conventionnel [KET92]. En particulier, la méthodologie du contrôleur flou apparaît utile quand les processus sont très complexes à analyser par les techniques quantitatives conventionnelles comme c'est le cas des robots manipulateurs, ou quand les sources d'information sont jugées non précises ou incertaines.

Cet avantage le contrôleur flou le doit à la base mathématique très pratique qu'est la logique floue.

Dans ce chapitre, nous allons présenter la méthodologie de conception d'un contrôleur flou. Ce tour d'horizon inclut :

- Le concept d'ensemble flou et les opérations sur ce nouvel objet mathématique.
- Le développement (dérivation) des règles de contrôle flou,
- Les stratégies de fuzzification et de défuzzification,
- La définition d'implication floue.

IV. 1 RAPPELS DES PRINCIPES DE LA LOGIQUE FLOUE

Dans ce qui suit, nous allons résumer les concepts de la théorie des ensembles flous nécessaires pour la conception des contrôleurs flous.

IV. 1. 1. ENSEMBLES FLOUS

Soit U une collection continue ou discrète d'objets dénotés $\{u\}$. U est appelé univers de discours et u représente l'élément générique de U .

Définition 1 : Ensemble flou

Un ensemble flou peut être vu comme une généralisation du concept d'ensemble ordinaire dont la fonction d'appartenance prend seulement deux valeurs $\{0,1\}$ (ensemble booléen).

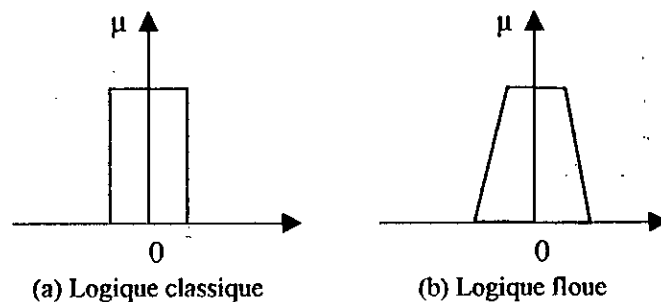


Figure IV.1 : Fonctions caractéristiques : logique classique- logique floue

Ainsi, sur l'univers de discours U , un ensemble flou F est caractérisé par une fonction d'appartenance μ_F prenant des valeurs dans l'intervalle $[0,1]$.

$$\mu_F : U \rightarrow [0,1]$$

Cet ensemble flou F peut être représenté comme un ensemble de paires (élément générique, degré d'appartenance) ordonnées :

$$F = \{ (u, \mu_F(u)) / u \in U \}$$

Et on le notera :

- U continu : $F = \int_U \mu_F(u) / u$

$$\bullet \text{ U discret : } F = \sum_{i=1} \mu_F(u_i) / u_i \quad (\text{IV.1})$$

Définition 2 : Support, point de commutation, singleton flou

Le support d'un ensemble flou F est l'ensemble de tous les points u dans U tels que $\mu_F(u) > 0$. En particulier, l'élément u de U tel que $\mu_F(u) = 0.5$ est appelé point de commutation. Un ensemble flou dont le support est un singleton dans U avec $\mu_F(u) = 1$ est appelé singleton flou.

IV. 1. 2 OPERATIONS ENSEMBLISTES ELEMENTAIRES [ZAD88]

Soit A et B deux ensembles flous dans U ayant respectivement μ_A et μ_B comme fonctions d'appartenances. L'union, l'intersection, et la complémentation des ensembles flous sont définies à l'aide de leur fonction d'appartenance [DUB88].

Définition 3 : Union \cup

La fonction d'appartenance $\mu_{A \cup B}$ de la réunion $A \cup B$ est définie, pour tout $u \in U$, par :

$$\mu_{A \cup B}(u) = \max \{ \mu_A(u), \mu_B(u) \}. \quad (\text{IV.2})$$

Définition 4 : Intersection \cap

La fonction d'appartenance $\mu_{A \cap B}$ de l'intersection $A \cap B$ est définie, pour tout $u \in U$, par :

$$\mu_{A \cap B}(u) = \min \{ \mu_A(u), \mu_B(u) \}. \quad (\text{IV.3})$$

Définition 5 : Complémentation

La fonction d'appartenance $\mu_{A'}$ du complément A' d'un ensemble flou A est définie, pour tout $u \in U$, par :

$$\mu_{A'}(u) = 1 - \mu_A(u). \quad (\text{IV.4})$$

Définition 6: Produit cartésien

Si A_1, A_2, \dots, A_n sont des ensembles flous dans respectivement U_1, U_2, \dots, U_n , le produit cartésien de A_1, A_2, \dots, A_n est un ensemble flou dans l'espace produit

$U_1 * U_2 * \dots * U_n$ ayant pour fonction d'appartenance :

$$\mu_{A_1 * A_2 * \dots * A_n}(u_1, u_2, \dots, u_n) = \min \{ \mu_{A_1}(u_1), \mu_{A_2}(u_2), \dots, \mu_{A_n}(u_n) \}$$

ou

$$\mu_{A_1 * A_2 * \dots * A_n}(u_1, u_2, \dots, u_n) = \mu_{A_1}(u_1) \bullet \mu_{A_2}(u_2) \bullet \dots \bullet \mu_{A_n}(u_n) \quad (\text{IV.5})$$

Définition 7: Relation floue

Une relation floue est un ensemble dans $U_1 * U_2 * \dots * U_n$ exprimée comme

$$R_{U_1 * U_2 * \dots * U_n}(u_1, u_2, \dots, u_n) = ((u_1, u_2, \dots, u_n), \mu_R(u_1, u_2, \dots, u_n)) / (u_1, u_2, \dots, u_n) \in U_1 * U_2 * \dots * U_n$$

(IV.6)

Définition 8: Composition "Sup-Star"

Si R et S sont deux relations floues dans respectivement $U * V$ et $V * W$, la composition de ces deux relations est une relation floue dénotée par $R \circ S$ et elle est définie par :

$$R \circ S = \{ [(u, v), \sup(\mu_R(u, v) \oplus \mu_S(v, w))] , u \in U, v \in V, w \in W \} \quad (\text{IV.7})$$

où \oplus est l'un des opérateurs de la classe des normes triangulaires (cf. Annexe A 4).

IV. 1. 3 VARIABLES LINGUISTIQUES ET ENSEMBLES FLOUS

Définition 9: Nombre flou

Dans un univers continu U, un nombre flou F est un ensemble flou (dans U) normal et convexe :

$$\begin{aligned} \max \{ \mu_F(u), u \in U \} &= 1 && \text{(normal)} \\ \mu_F(\lambda u_1 + (1 - \lambda) u_2) &= \min \{ \mu_F(u_1), \mu_F(u_2) \} && \text{(convexe)} \quad (\text{IV.8}) \end{aligned}$$

pour tout $(u_1, u_2) \in U * U$ et $\lambda \in [0, 1]$.

Définition 10: Variables linguistiques

L'utilisation des ensembles flous constitue une voie privilégiée pour la manipulation de concepts vagues et imprécis : petit, grand, chaud, ... En particulier, on peut employer les ensembles flous pour représenter des variables linguistiques. Une variable linguistique peut être vue comme une variable dont la valeur est un nombre flou ; autrement dit que ses valeurs sont définies en termes linguistiques (flous).

Une variable linguistique est caractérisée par un 5-uple $(x, T(x), U, G, M)$ dans lequel :

- x est le nom de la variable (vitesse par exemple),
- $T(x)$ est l'ensemble des valeurs linguistiques de x dont chacune est un nombre flou défini sur U ($T(\text{vitesse}) = \{\text{lente, moyenne, rapide, ...}\}$),
- G est une règle syntaxique pour la génération des noms des valeurs de x ,
- M est une règle sémantique qui associe à chaque valeur un sens.

Dans l'exemple de la variable " vitesse ", si chaque terme est défini dans l'univers de discours $U = [0, 100]$ par exemple selon la figure IV.2, M interprétera " lente " comme une vitesse inférieure ou égale à 40 km/h, et " rapide " comme une vitesse au delà de 70 km/h.

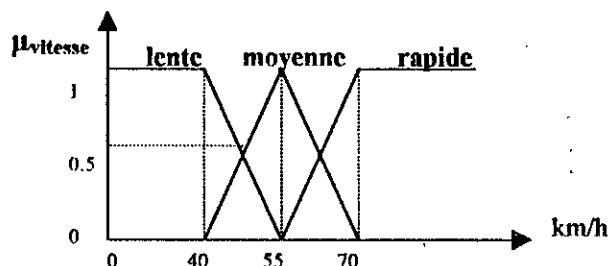


Figure IV.2 : Représentation graphique des vitesses floues

IV. 1. 4 FONCTION D'APPARTENANCE

La définition des éléments d'un ensemble flou comme des relations entre ensembles flous repose totalement sur la caractérisation par fonction d'appartenance. Ceci rend primordial le choix de la forme d'une telle fonction surtout pour les traitements numériques sur calculateurs.

Le plus souvent, on utilise les formes suivantes [BUH94]:

- Fonction Triangulaire : Elle est définie par trois paramètres définissant les coordonnées des coins du triangle donnant la forme de la fonction.
- Fonction Trapézoïdale : Elle est définie par quatre paramètres,
- Fonction Gaussienne : Définie par son centre et sa variance.
- Fonction Sigmoidale.

La forme et les expressions de telles fonctions sont données en Annexe (cf. Annexe A4)

IV. 1. 5 LOGIQUE FLOUE ET RAISONNEMENT APPROCHEE

En raisonnement approché, il y a deux règles d'inférence floue importantes proposées par Zadeh [ZAD88] nommés le " modus ponens généralisé " (MPG) et le " modus tollens généralisé " (MTG) qui sont une généralisation de la logique formelle : règles du tiers exclus.

Prémisse 1 : x est A'

Prémisse 2 : si x est A alors y est B (MPG) (IV.9)

Conséquence : y est B'

Prémisse 1 : y est B'

Prémisse 2 : si x est A alors y est B (MTG) (IV.10)

Conséquence : x est A'

où A, A', B et B' sont des ensembles flous représentant des variables linguistiques.

En pratique, on définit un opérateur flou dénommé implication floue permettant d'évaluer le degré de vérité d'une règle R donnée par la forme " Si x est A et y est B " ; et cela à partir des valeurs de la prémisse d'une part, et celle de la conclusion d'autre part.

$$\mu_R(x, y) = \text{Imp.}(\mu_A(x), \mu_B(y)) \quad (\text{IV.11})$$

Les opérateurs les plus utilisés en commande floue sont les implications dites de Mamdani et Larsen.

- Implication de Mamdani : $\mu_R(x, y) = \min(\mu_A(x), \mu_B(y))$ (IV.12)

- Implication de Larsen : $\mu_R(x, y) = \mu_A(x) \cdot \mu_B(y)$ (IV.13)

IV. 2 CONTROLEUR FLOU

IV. 2. 1 ARCHITECTURE D'UN CONTROLEUR FLOU

Muni des différents concepts de la théorie des ensembles flous, on peut désormais développer les notions importantes des contrôleurs flous.

La *figure IV.3* montre le schéma synoptique général d'un contrôleur flou.

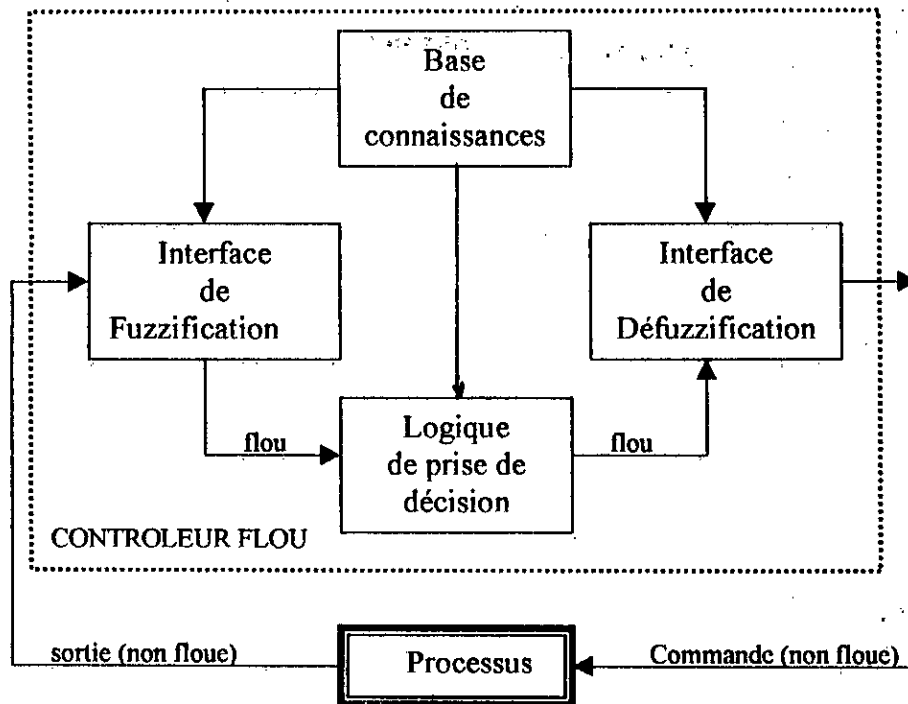


Figure IV.3 Schéma synoptique général d'un contrôleur flou

Ce schéma est composé :

- d'une interface de fuzzification à l'entrée,
- d'une base de connaissances,
- d'une logique de prise de décision,
- d'une interface de défuzzification à la sortie,
- du système à contrôler.

IV. 2. 1. 1 L'interface de fuzzification

Elle inclut les fonctions suivantes :

- mesure des variables d'entrée,
- représentation d'une cartographie d'échelle transférant la plage des variables d'entrée aux univers de discours correspondants,
- représentation de la fonction de fuzzification convertissant les données d'entrée en valeurs linguistiques convenables pouvant être vues comme des étiquettes d'ensembles flous.

IV. 2. 1. 2 La base de connaissances

Elle comprend une connaissance du domaine d'application et des buts du contrôle prévu. Elle est donc composée d'une base de données fournissant les règles de contrôle linguistique qui traduisent les astuces de contrôle des experts du domaine.

IV. 2. 1. 3. La logique de prise de décision

C'est le noyau du contrôleur flou ; elle est capable de simuler la prise de décision de l'être humain en se basant sur les concepts flous et les règles d'inférence en logique floue (voir plus haut).

IV. 2. 1. 3. L'interface de défuzzification

Elle réalise les fonctions suivantes :

- une cartographie d'échelle convertissant la plage des variables de sortie aux univers de discours appropriés,
- une défuzzification fournissant une action de contrôle (physique) à partir d'une action de contrôle floue.

IV. 2. 2. STRATEGIE DE FUZZIFICATION

La fuzzification est une opération qui transforme une mesure en une évaluation subjective conformément à la théorie de la logique floue. Ainsi une valeur x_0 de l'univers du discours U est relié à l'ensemble flou A_x par l'opération de fuzzification qui donne à x_0 son degré d'appartenance à A_x . Il existe deux méthodes de fuzzification suivant la définition de A_x [SAL98]:

- A_x est un singleton flou défini par :

$$\mu_{A_x} = \begin{cases} 1 & \text{si } x = x_0 \\ 0 & \text{si } x \neq x_0 \end{cases} \quad (\text{IV.14})$$

- $A_x(x_0) = 1$ et $\mu_{A_x}(x)$ décroît lorsque x s'éloigne de x_0 . Dans ce cas, on ne considère plus que la valeur de x caractérise la valeur réelle de la variable.

Les deux stratégies sont illustrées à la figure III.4 cinématique inverse dessous.

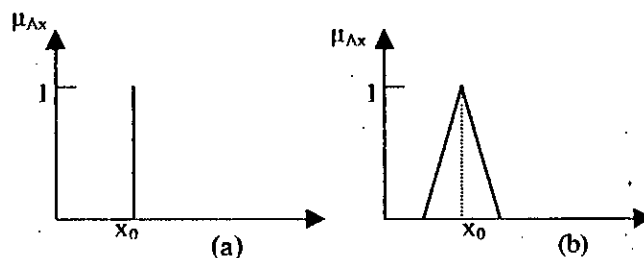


Figure IV. 4 Deux stratégies de la fuzzification

IV. 2. 3. STRATEGIE D'INFERENCE

Le module d'inférence est le bloc dénommé par logique de prise de décision de la figure IV.3. Comme son nom l'indique il permet de passer à travers des transformations utilisant les règles d'inférence des ensembles flous de l'espace d'entrée vers ceux de l'espace de sortie et ce, en se basant sur les opérations de la logique floue.

Les règles sont données sous la forme suivante :

R_1 : Si x est A_1 et y est B_1 , alors z est C_1 .

R_2 : Si x est A_2 et y est B_2 , alors z est C_2 .

etc...

où x et y étant les variables de l'espace d'entrée réparti en sous-ensembles flous $A_1, A_2, \dots, A_m, B_1, B_2, \dots, B_n$; et z la variable de sortie élément d'ensembles flous C_1, C_2, \dots

A partir de la règle « Si A alors B » et du fait A' , on déduit un nouveau fait B' qui est caractérisé par un ensemble flou dont la fonction d'appartenance est :

$$\mu_{B'}(y) = \sup_{x \in X} \min(\mu_{A'}(x), \mu_R(x, y)) \quad (\text{IV.15})$$

IV. 2. 4. AGREGATION DES REGLES

Avec les implications usuellement utilisées en contrôle flou, l'agrégation des règles s'effectue à l'aide d'une conorme triangulaire. Les méthodes utilisées pour déterminer la fonction d'appartenance de la sortie peuvent être classées dans trois types [SAL98]:

- **Méthode de Tsukamoto :**

La sortie totale du système flou est une somme pondérée des degrés de confiance de chaque règle floue et des fonctions d'appartenance de la sortie.

- **Méthode de Mamdani :**

C'est la plus utilisée. La sortie totale est calculée par application de l'opérateur max aux sorties des différentes règles. Ceci revient à considérer que l'ensemble des N règles constituant la base de règles floues sont liées par l'opérateur de disjonction "ou". On obtient alors :

$$\mu_{B'}(y) = \max_{i=1 \text{ à } n} \mu_{B_i'}(y) \quad (\text{IV.16})$$

où n est le nombre de règles activées dans le jeu des règles du système flou.

- **Méthode de Sugeno :**

La conclusion de chaque règle est une combinaison linéaire des variables d'entrée plus une constante. La sortie finale est égale à la moyenne pondérée de la sortie de chaque règle floue.

IV.2.5 DEFUZZIFICATION

Le rôle de la défuzzification est de transformer la partie floue issue de l'inférence en une grandeur numérique de commande. Malheureusement, il n'existe pas une procédure systématique pour choisir la stratégie de défuzzification. Mais comme il s'agit de la logique floue vue comme alternative pour la commande de processus complexes, un critère de choix s'impose d'elle-même ; à savoir la simplicité de calcul. Ce critère a conduit aux défuzzificateurs suivants [BUH94],[LEE90] :

IV.2.5.1 Méthode du maximum

La méthode consiste à examiner l'ensemble flou B issu de l'inférence pour en choisir comme sortie (commande) la valeur y pour laquelle $\mu_B (y)$ est un maximum. L'inconvénient de cette méthode est l'indétermination qui se présente lorsque la sortie présente un palier (plusieurs maxima).

IV.2.5.2 Méthode de la moyenne des maxima

Avec cette méthode on détermine d'abord dans l'ensemble flou B issu de l'inférence les valeurs de y pour lesquelles $\mu_B (y)$ est un maximum ; ensuite, on détermine la commande en faisant la moyenne de ces valeurs.

IV.2.5.3 Méthode du centre de gravité

Le centre de gravité de la surface délimitée par la fonction d'appartenance de sortie et l'axe des abscisses est calculé. Sa valeur détermine la commande. Soit $\mu_B (y)$ la forme de la fonction d'appartenance de l'ensemble B obtenu en sortie d'inférence. La formule du centre de gravité est donnée par :

$$\bar{y} = \frac{\int_S y \mu_B(y) dy}{\int_S \mu_B(y) dy} \quad (IV.17)$$

où S est le support de $\mu_B(y)$: c'est-à-dire l'ensemble définissant l'univers du discours.

Remarque :

En pratique l'intégration est approchée par la méthode des rectangles ou mieux celle des trapèzes bien connues en analyse numérique. Le résultat devient exact quand l'univers de discours est discret. On obtient la version discrète de l'intégrale :

$$\bar{y} = \frac{\sum_{i=1}^l y_i \mu_B(y_i)}{\sum_{i=1}^l \mu_B(y_i)} \quad (IV.18)$$

IV.2.5.4 Méthode des hauteurs pondérées

La procédure avec cette méthode consiste à prendre comme valeur de commande la moyenne pondérée des centres de gravité \bar{y}_k de l'ensemble flou A_k issu de chaque règle R_k , chacun de ces centres de masse étant pondéré par le degré de confiance de la règle μ_k .

On obtient la formule suivante dans le cas de N règles:

$$\bar{y} = \frac{\sum_{k=1}^N \mu_k \bar{y}_k}{\sum_{k=1}^N \mu_k} \quad (IV.19)$$

L'avantage de cette méthode est qu'elle est simple à implémenter car le centre de gravité des fonctions d'appartenance habituellement utilisées sont connus. Cependant, cette formule ne prend aucune considération de la forme de la fonction d'appartenance de sortie.

IV.2.5.5 Méthode des hauteurs pondérées modifiée

Il s'agit de la version améliorée de la méthode précédente. Ici le calcul de la fonction d'appartenance de sortie est effectué ; et un coefficient de correction δ est introduit. La formule IV.19 devient :

$$\bar{y} = \frac{\sum_{k=1}^N (\mu_k \bar{y}_k / \delta_k^2)}{\sum_{k=1}^N (\mu_k / \delta_k^2)} \quad (IV.20)$$

Le coefficient δ_k mesure l'étendue de la fonction d'appartenance de la variable de sortie associée à la règle R_k . Pour une fonction d'appartenance triangulaire ou trapézoïdale, δ peut être le support du triangle ou du trapèze, et pour une fonction gaussienne elle peut être sa variance.

IV.2.6 CONTROLEUR DE MAMDANI

Il s'agit du premier contrôleur flou réalisé en 1974 par le professeur E. Mamdani à Queen Mary College pour la régulation d'une machine à vapeur [MAM 77].

La méthode de Mamdani repose sur l'utilisation de l'opérateur min pour la combinaison des prémisses et pour l'implication. L'agrégation des règles est réalisée par l'opérateur max, et la défuzzification est effectuée par la méthode du centre de gravité.

La figure IV.5 illustre le principe de cette méthode pour les deux règles suivantes :

R_1 : Si (e est Positive, et Δe est Zéro) alors u est Zéro

R_2 : Si (e est Positive_Petite, et Δe est Positive_Petite) alors u est Positive_Petite

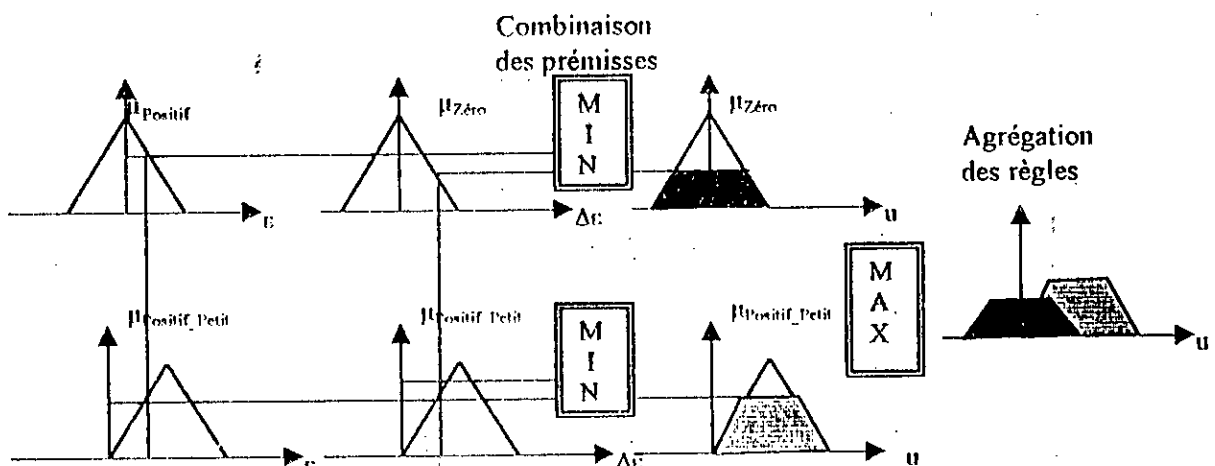


Figure IV. 5 Principe de la méthode de Mamdani

IV.2.7 CONTROLEUR DE LARSEN

La méthode de Larsen utilise le produit pour la combinaison des prémisses et pour l'implication. L'agrégation des règles est réalisée par l'opérateur max, et la défuzzification est effectuée par la méthode du centre de gravité.

IV.2.8 CONTROLEUR DE SUGENO

Ce modèle de régulateur flou a été développé initialement par Sugeno et Takagi pour la modélisation des systèmes à partir de données numériques [SUG85].

Une règle typique dans le système de Sugeno est de la forme :

$$R_k : \text{Si } (x_i \text{ est } F_i^k, \text{ et } \dots, \text{ et } x_n \text{ est } F_n^k) \text{ alors } y = f_k(x)$$

où les F_i^k sont des ensembles flous de l'espace d'entrée, tandis que $f(x)$ une fonction numérique prenant ses valeurs dans l'espace de sortie. En général, $f(x)$ est une fonction polynomiale des variables d'entrée. La conception d'un contrôleur flou de Sugeno consiste d'abord à définir les ensembles flous de ces dites fonctions.

Soient par exemple x_1, \dots, x_n les entrées du système flou et y la sortie (commande). On définit d'abord les m_i ensembles flous F_i^k dans U_i pour chaque x_i , tel que pour tout x_i de U_i , il existe au moins un $\mu_{F_i^k}(x_i) \neq 0$; où $i = 1, 2, \dots, n$ et $j = 1, 2, \dots, m_i$.

Ensuite, on construit la base de règles du système flou qui comporte $M = \prod_{i=1}^n m_i$ règles

telles que :

$$R_k : \text{Si } (x_1 \text{ est } F_1^{k1}, \text{ et } \dots, \text{ et } x_n \text{ est } F_n^{kn}) \text{ alors } y_k(x) = a_0^k + a_1^k x_1 + \dots + a_n^k x_n$$

où les paramètres a_i^k sont des nombres réels.

La sortie est obtenue par la méthode du centre de gravité.

CONCLUSION

Basés sur les éléments méthodologiques propres à la logique floue, plusieurs algorithmes de régulation floue ont été développés et appliqués sur plusieurs systèmes dans des domaines différents. Par exemple P. J. King et E. H. Mamdani [KET92] ont décrit l'application de tels algorithmes pour le contrôle de processus dynamiques.

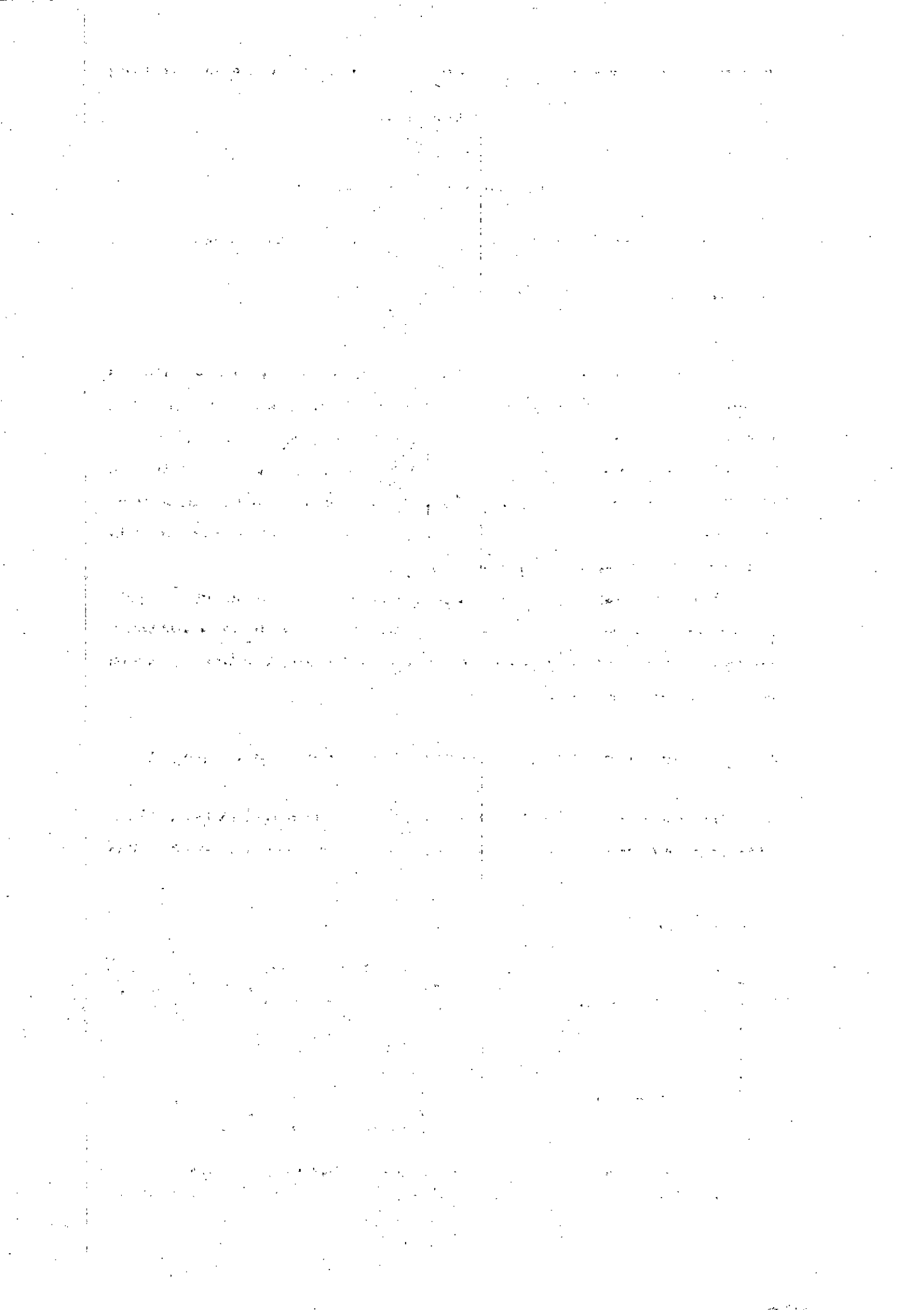
Les résultats ainsi obtenus par ces brillants chercheurs montrent :

- d'une part, les intérêts de tels algorithmes surtout lorsqu'il s'agit de la régulation de systèmes complexes ou difficiles à modéliser,
- d'autre part, la performance non négligeable des contrôleurs flous vis-à-vis des contrôleurs conventionnels PID.

Cependant, la plupart des expériences faites ont concerné les processus lents (en général chimiques). L'adaptation voire l'amélioration et l'utilisation de l'un de ces algorithmes dans un contrôle plus rapide et plus précis ont attiré par la suite l'attention des chercheurs [PED88].

L'application de tels algorithmes pourrait certainement être une bonne alternative pour la résolution des problèmes liés à la commande des bras manipulateurs.

CHAPITRE V
COMMANDE FLOUE
DE BRAS
MANIPULATEURS



Chapitre V
COMMANDE FLOUE
DE BRAS MANIPULATEURS

INTRODUCTION

La mise en œuvre d'un contrôleur flou présente quelques aspects pratiques importants : le choix de la forme de la fonction d'appartenance, de la méthode d'inférence et de défuzzification, c'est-à-dire de l'algorithme. En commande, la plupart des algorithmes utilisés sont basés sur le schéma simple proposé par Mamdani [MAM77]. La majorité a été appliquée avec succès, mais il s'agit en général de processus industriels lents. Ils ont aussi montré des limites de performances sur des systèmes rapides et fortement non linéaires.

Dans cette partie, nous nous sommes inspiré du même algorithme pour commander en position les articulations du bras SCARA à 2 degrés de liberté étudié précédemment et dont la simulation en boucle ouverte et le contrôle à l'aide d'un PID a été fait. Notre étude se fera aussi sur le bras RRR.

V. 1. DESCRIPTION DU SCHEMA DE CONTROLE TYPIQUE

La majorité des contrôleurs développés utilisent le schéma simple proposé par Mamdani pour les systèmes mono-entrée / mono-sortie. Ce schéma est représenté par la *figure V.1*

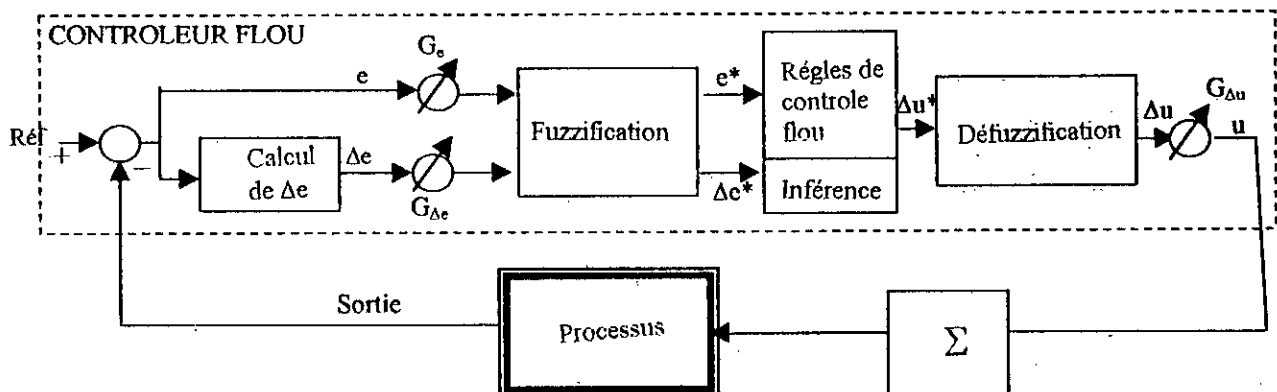


Figure V.1 Schéma bloc d'une boucle de régulation à contrôleur flou de type Mamdani

Le système est composé :

- du contrôleur flou comprenant :
 - un bloc de calcul de la variation de l'erreur (Δe) au cours du temps,
 - les facteurs de l'échelle ou gains associés à l'erreur (e), à sa variation (Δe) et à la variation de la commande (Δu),
 - un bloc de fuzzification de l'erreur et sa variation,
 - une base des règles de contrôle flou ,
 - une logique floue utilisée pour l'évaluation des règles de contrôle flou (inférence),
 - un bloc de défuzzification servant à convertir la variation de la commande floue en une valeur numérique,
- du processus à contrôler.

V. 2. ALGORITHME DU CONTROLE FLOU

Le succès des algorithmes de commande floue des processus complexes est dû au choix de méthodes relativement pratiques, permettant avec une simplicité notable, la mise au point de tels algorithmes.

Ces méthodes consistent à formuler un ensemble de règles de décision en termes linguistiques, utilisant des ensembles flous pour décrire les amplitudes de l'erreur, de sa variation et de la variation de la commande appropriée. En combinant ces règles, on peut dresser des tables de décision permettant de donner les valeurs de la sortie du contrôleur correspondant aux situations d'intérêt.

Pour ce faire, on est appelé à quantifier ces différentes variables. Les niveaux de quantification ainsi obtenus forment les univers de discours (pour l'erreur, sa variation, la variation de la commande et éventuellement pour les gains correspondants) et les valeurs obtenues sont assignées à un ensemble de sous-ensembles flous linguistiquement étiquetés par des termes ou symboles.

Les valeurs des degrés d'appartenance sont subjectivement attribuées pour définir le sens des étiquettes des sous-ensembles flous selon certaines formes de fonction d'appartenance.

Ces sous-ensembles flous pourront être combinés grâce aux opérateurs fondamentaux suivants qu'il est important de rappeler :

- réunion : $A \cup B$ correspondant à la fonction « ou » définie par :

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (\text{V.1})$$

- intersection : $A \cap B$ correspondant à la fonction « et » donnée par :

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (\text{V.2})$$

- complémentation : A' correspondant à la fonction « non » définie par :

$$\mu_{A'}(x) = 1 - \mu_A(x) \quad (\text{V.3})$$

V. 2. 1. LOI DE COMMANDE

La loi adoptée est fonction des entrées choisies par notre contrôleur, en l'occurrence, l'erreur et sa variation sur une période d'échantillonnage T_e :

$$u(k T_e) = f(e(k T_e), \Delta e(k T_e)) \quad (\text{V.4})$$

Par conséquent, l'activation de l'ensemble des règles de décision associées donne la variation de la commande Δu nécessaire et qui constitue donc la sortie de notre contrôleur.

Cette variation de la commande est obtenue par une simple lecture d'une table de décision prédéfinie hors ligne. La loi de commande est donnée de manière classique sous la forme suivante :

$$u(k T_e) = G_{\Delta u} * \Delta u(k T_e) + u(k T_e - T_e) \quad (\text{V.5})$$

où $G_{\Delta u}$ est le gain associé à la commande, en général choisi faible pour assurer la stabilité du système et $\Delta u(k T_e)$ est la variation de la commande à l'instant T_e .

V. 2. 2. IMPLEMENTATION

La valeur de la commande à l'instant $k T_e$ est obtenue selon l'algorithme suivant :

1. calcul de l'erreur présente de la commande et de sa variation :

$$e(k T_e) = \text{référence} - \text{sortie du processus} \quad (\text{V.6})$$

$$\Delta e(k T_e) = \frac{e(k T_e) - e(k T_e - k T_e)}{T_e} \quad (\text{V.7})$$

\downarrow (erreur)

2. normalisation de l'erreur et de sa variation à l'aide des facteurs d'échelle ou gains correspondants :

$$e^*(k T_e) = G_e * e(k T_e) \quad (\text{V.8})$$

$$\Delta e^* (k T_e) = G_{\Delta e} * \Delta e (k T_e) \quad (V.9)$$

3. fuzzification ou conversion des valeurs obtenues en variables flous,
4. lecture du niveau de quantification du sous-ensemble flou résultant de la variation de la commande dans la table de décision par application de la règle d'inférence,
5. détermination de la commande à appliquer à l'entrée du processus à réguler à l'aide du facteur d'échelle ou gain correspondant (Equation V.5).

L'organigramme de la figure V.2 regroupe toutes ces étapes.

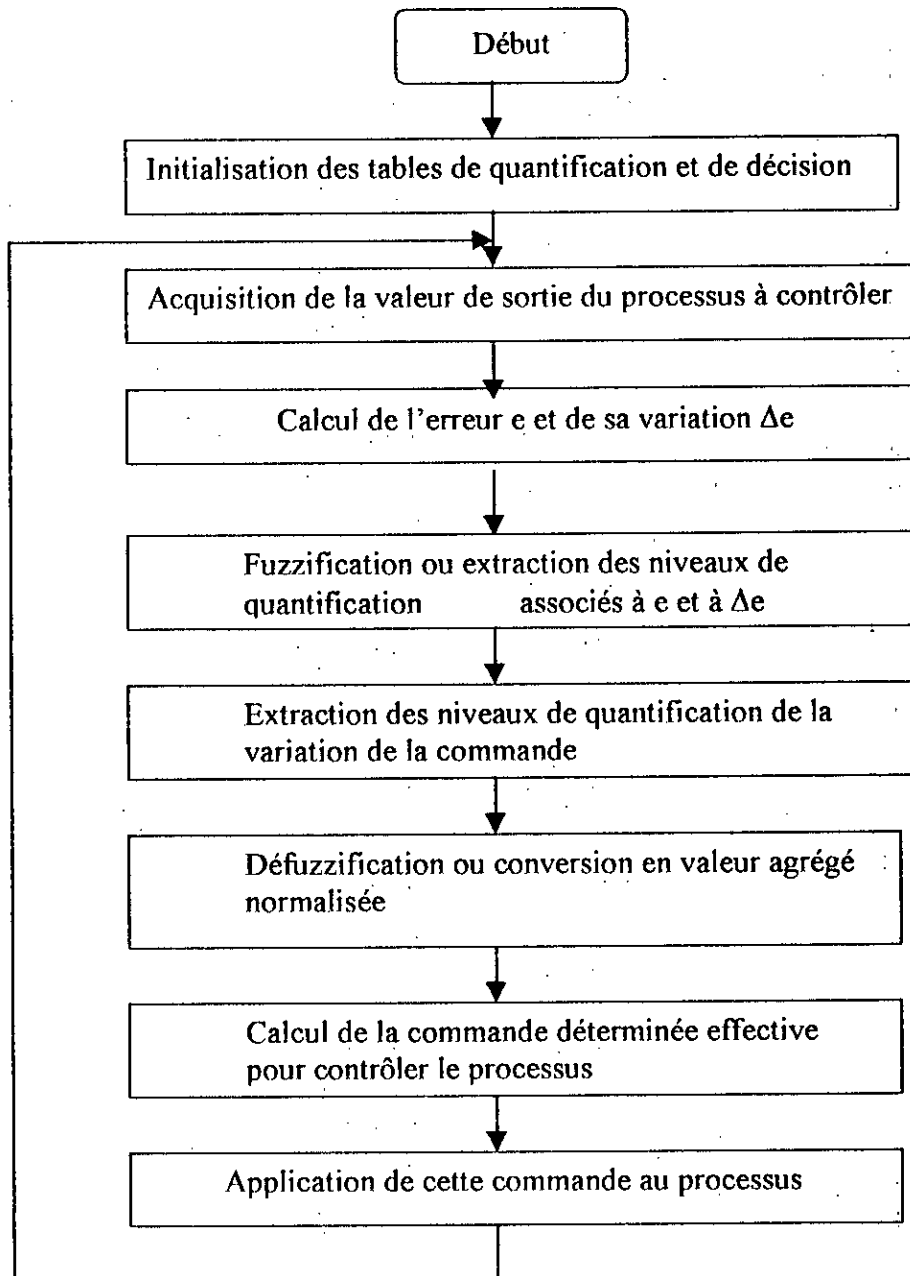


figure V. 2 : Organigramme de l'algorithme de contrôle flou

V. 3. APPLICATION A LA COMMANDE DU BRAS SCARA

Le principe de la commande reste le même que celui de la PID ; chaque articulation est commandée séparément dans l'espace articulaire.

V. 3. 1. FONCTION D'APPARTENANCE

Pour le choix des formes des fonction d'appartenance, nous avons opté pour la triangulaire. Cela s'explique avant tout par la simplicité de mise en œuvre. Mais aussi pour les performances obtenues avec cette forme par des études déjà faites [PED88].

V. 3. 1. 1. DISTRIBUTION DES FONCTIONS D'APPARTENANCE

Eu égard à la nature complexe de notre modèle et en raison de certaines investigations [BUH94], nous avons adopté une distribution uniforme (équidistant) avec des formes symétriques. Ce qui donne un chevauchement de 50% entre les sous-ensemble (voir figure V.3). Une telle distribution facilite énormément la programmation ; et offre une performance optimale [BOV91].

V. 3. 1. 2. NOMBRE DE CLASSES OU SOUS-ENSEMBLE FLOUS

Afin d'étudier l'influence du nombre de classes dans les performances de la commande, nous avons fait l'étude avec un nombre de 3, 5 et 7.

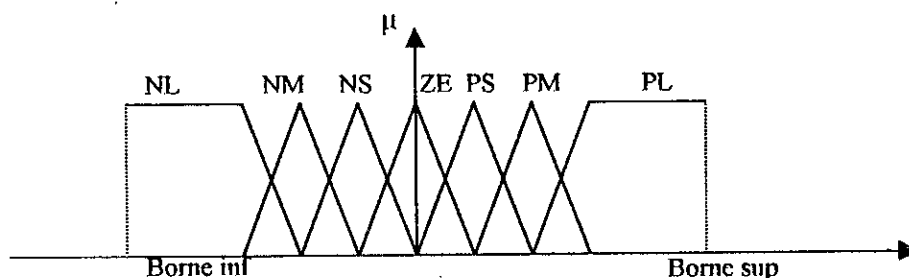


figure V.3 Distribution des ensembles flous

V. 3. 2. UNIVERS DE DISCOURS

Dans la commande du SCARA on a affaire à des angles (variables articulaires). L'erreur et sa variation ont été naturellement bornée entre -3.15 et 3.15 ($\pm\pi$). La valeur de la commande est normalisée à 1 en valeur absolue. C'est le gain à la sortie du contrôleur flou qui sert de cartographie d'échelle.

V. 3. 3. REGLES DECISIONNELLES

Les règles décisionnelles se composent de paires situation/action de la forme :

Si e est A et Δe est B, alors Δu est C.

Cet ensemble de règles doit regrouper toutes les situations possibles que peut rencontrer le contrôleur flou lors de son fonctionnement. Dans notre application, nous avons opté pour la base de règles de *Mac Vicar-Whelan* [LEE90] qui constitue une base de données très pratique pour l'arithmétique des règles. Ce tableau est donné à la *figure V.4* dans l'exemple de 7×7 classes.

V. 3. 4. METHODES D'INFERENCE

Nous avons essayé les méthodes max-min et som-prod pour des besoins de comparaison.

V. 3. 5. PERIODE D'ECHANTILLONNAGE

Vu la dynamique de notre système, la simulation sur 10 secondes est dans beaucoup des cas presque optimal. Nous avons subdivisé l'intervalle de temps en 256 niveaux ; ce qui donne une période d'échantillonnage de 39 ms. Ce paramètre caractérisant la bande passante de notre contrôleur est d'une grande importance. Au cas où le comportement du robot doit être observé sur une période plus grande, nous avons tout de même conservé les 256 points. Ce choix ($256=2^8$) est lié à la programmation sous C++ ; l'intervalle de temps est échantillonné à 256 niveaux et la référence à l'un de ces derniers est faite à partir d'une indice codée sur un octet : il en résulte une meilleure gestion de la mémoire.

Toutefois la dynamique du système a toujours donné des résultats dans les limites de 60 secondes.

| | | | | | | | | |
|------------------------|----------------|----|----|----|----|----|----|----|
| $\Delta e \rightarrow$ | $\downarrow e$ | NL | NM | NS | ZE | PS | PM | PL |
| NL | NL | NL | NL | NL | NL | NM | NS | ZE |
| NM | NL | NL | NL | NM | NS | ZE | PS | PM |
| NS | NL | NL | NM | NS | ZE | PS | PM | PL |
| ZE | NL | NM | NS | ZE | PS | PM | PL | PL |
| PS | NM | NS | ZE | PS | PM | PL | PL | PL |
| PM | NS | ZE | PS | PM | PL | PL | PL | PL |
| PL | ZE | PS | PM | PL | PL | PL | PL | PL |

figure V.4 Matrice de Mac Vicar- Whelan dans le cas 7x7 classes

V. 3. 6. SIMULATION, RESULTATS ET INTERPRETATION

Les paramètres du bras utilisés sont donnés en Annexe (voir Annexe A5).

Les courbes de simulations sont données à partir de la figures V.5.

Avec un contrôleur de Sugeno à 3 classes (figure V.5) nous avons obtenu une réponse assez rapide du système mais avec un grand dépassement pour le premier segment.

Avec l'augmentation du nombre de règles à 5 (figure V.6) il y a toujours un dépassement et le système devient plus lent. Cela s'explique d'abord par la forme des fonctions d'appartenance choisie : palier aux extrémités ; ce qui correspond dans le tableau de Mac-Vicar à une sortie maximale. Dans le deuxième cas (5 règles) le nombre d'ensembles devenant plus important, le temps de calcul augmente.

Il est à noter que les meilleurs résultats sont obtenus pour le deuxième segment.

A 7 règles des résultats plus optimaux ont été obtenus. Le problème du dépassement est résolu avec un temps de réponse pratiquement le même qu'au premier cas (ordre de 10 s).

Avec le contrôleur de Mamdani on a un dépassement trop important avec 3 classes (figure V.8); ce dépassement est réduit quand on augmente le nombre de classes à 7 (figure V.9) avec des meilleurs performances.

Toutefois il est à noter que le deuxième segment a un comportement plus optimal que le premier ; à cause du couplage, il est plus sensible à l'erreur et à sa

variation, donc plus sujet à un contrôle optimal.

Toutefois, même si ces résultats peuvent être améliorés avec un contrôle plus optimal des gains pour éventuellement réduire le temps de réponse, il est important de noter dès maintenant l'absence de divergence comme c'était des fois le cas avec les contrôleurs PID.

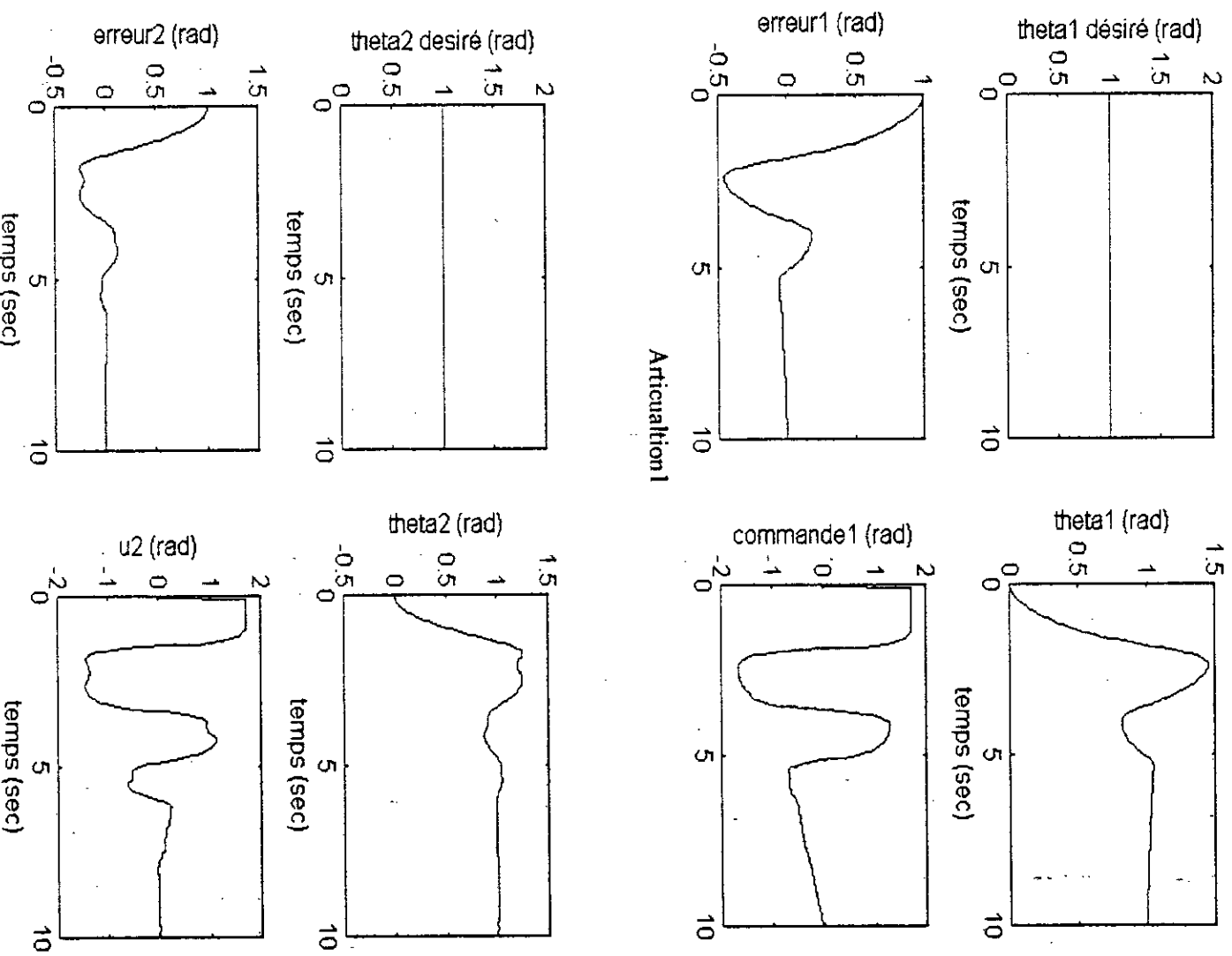
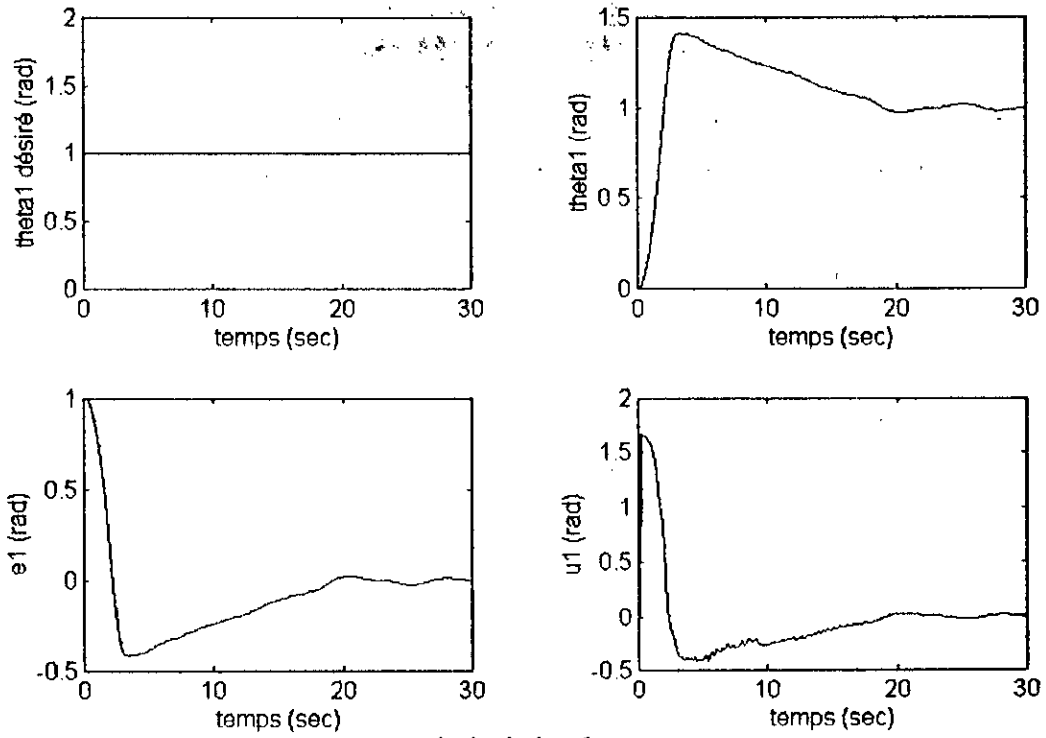
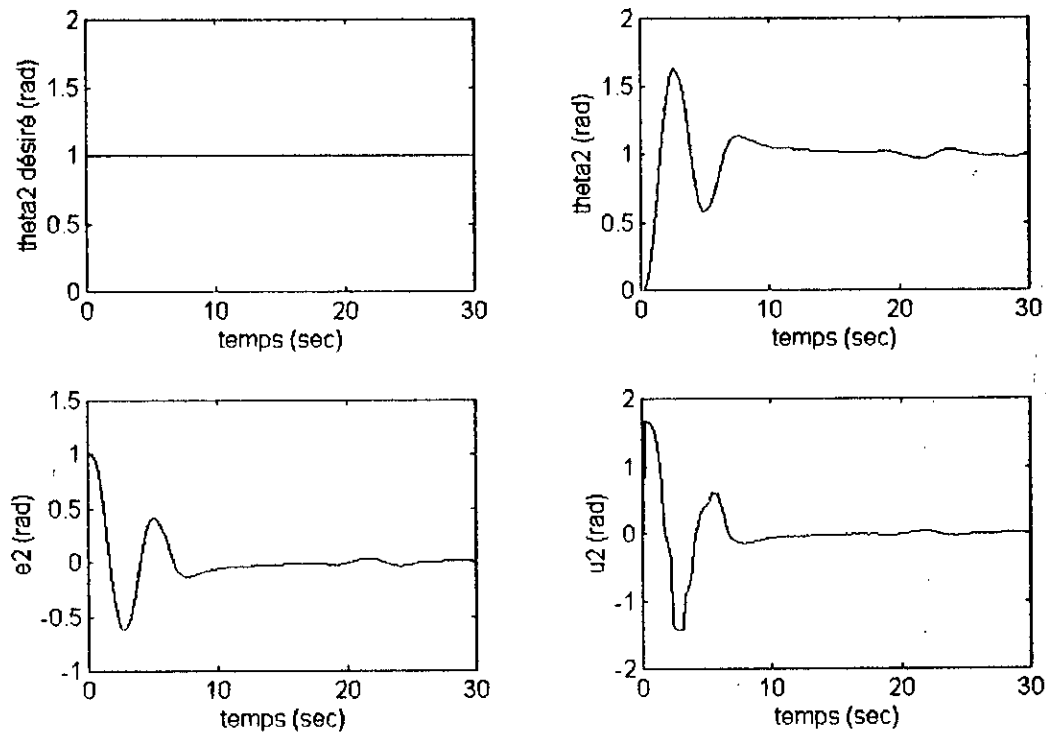


Figure V.5 Contrôleur Som-Prod à 3 règles



Articulation 1



Articulation 2

Figure V. 6 Contrôleur Som-Prod à 5 règles

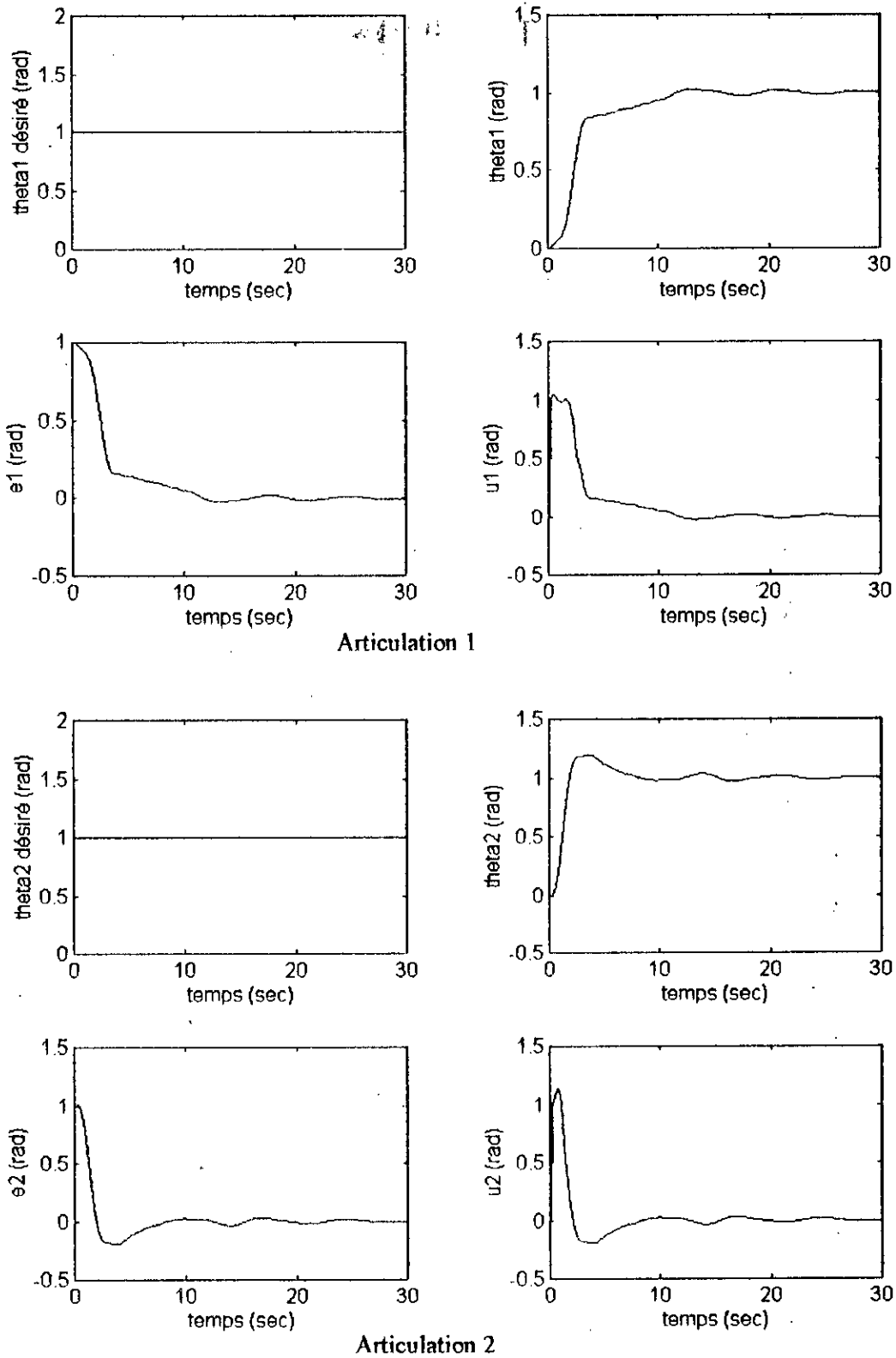
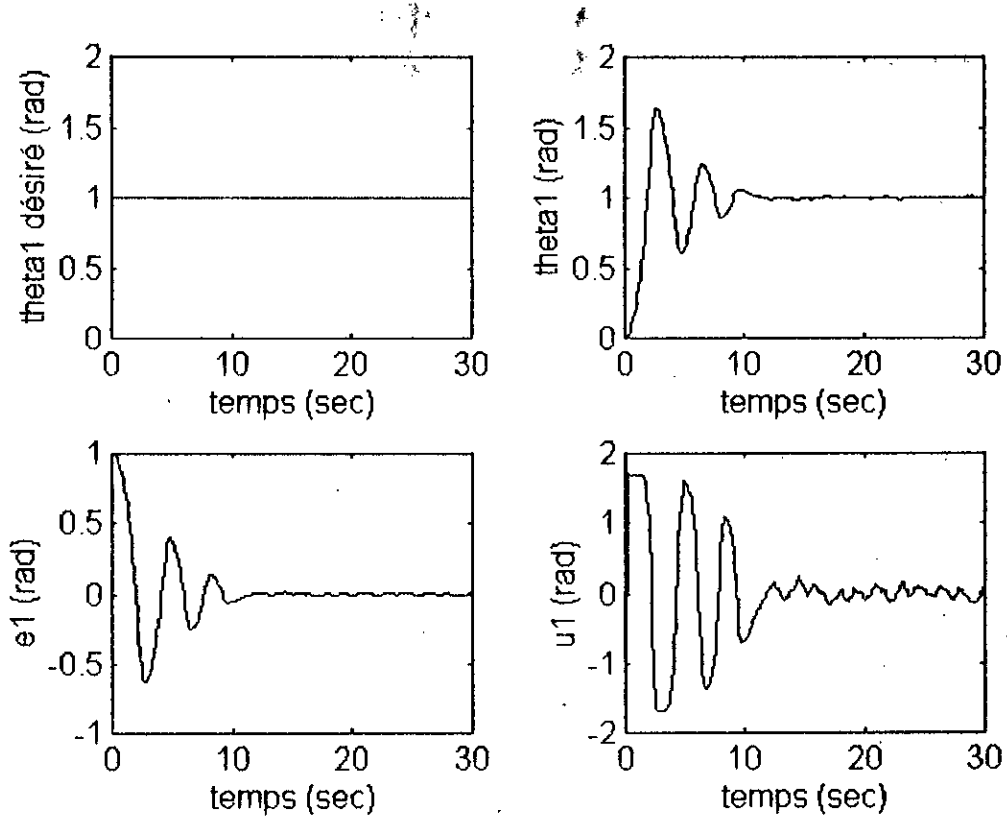
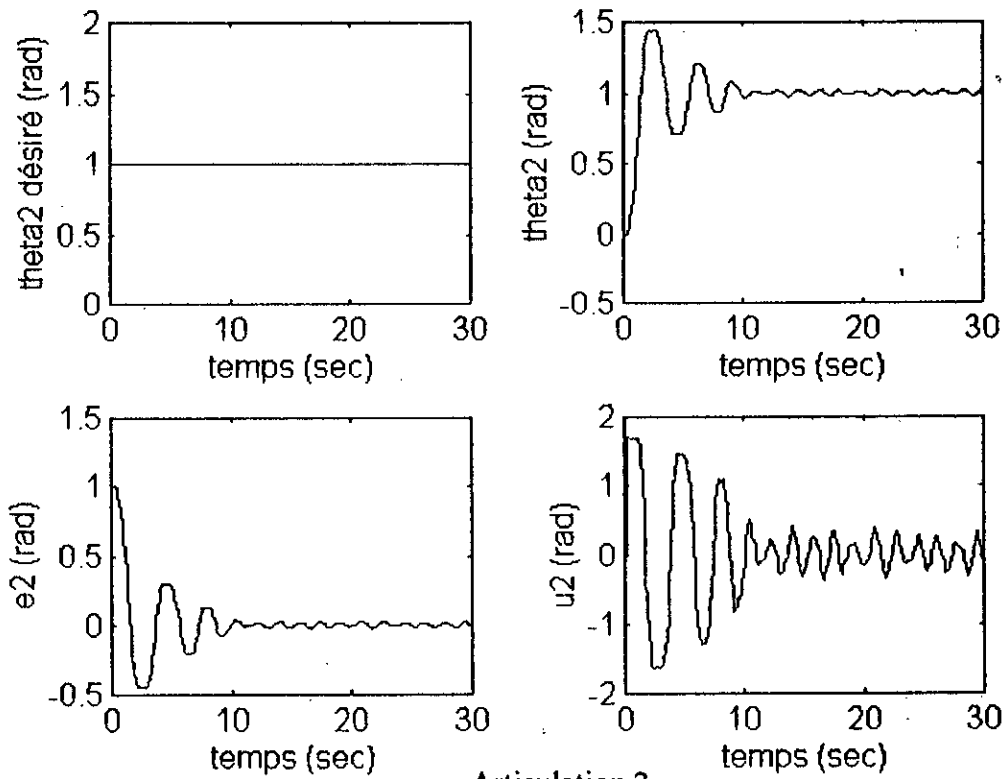


Figure V.7 contrôleur flou Som-Prod à 7 classes



Articulation 1



Articulation 2

Figure V.8 Contrôleur flou de Mamdani à 3 règles

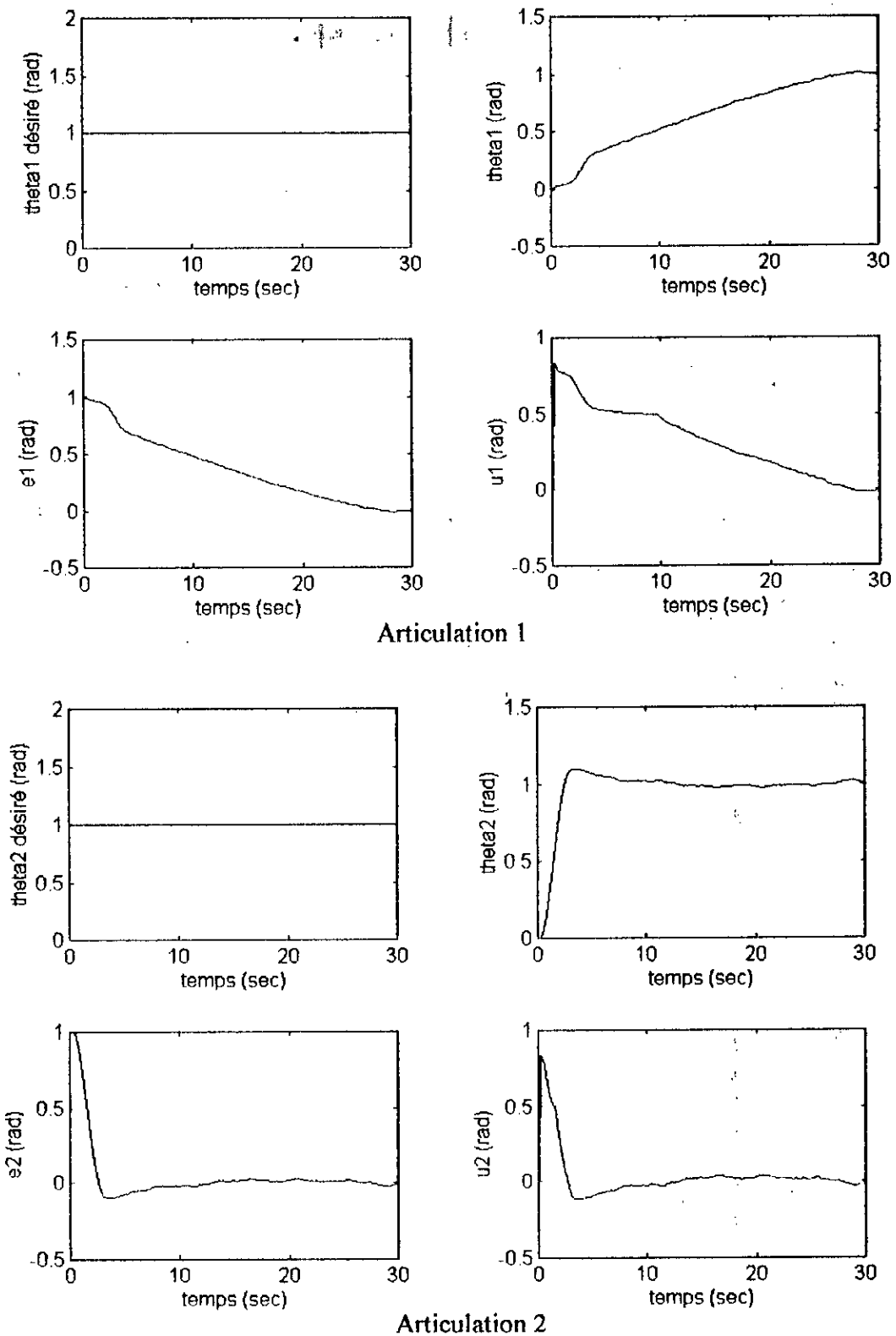


Figure V.9 contrôleur flou de Mamdani à 7 règles

V. 4. APPLICATION A LA COMMANDE DU BRAS RRR

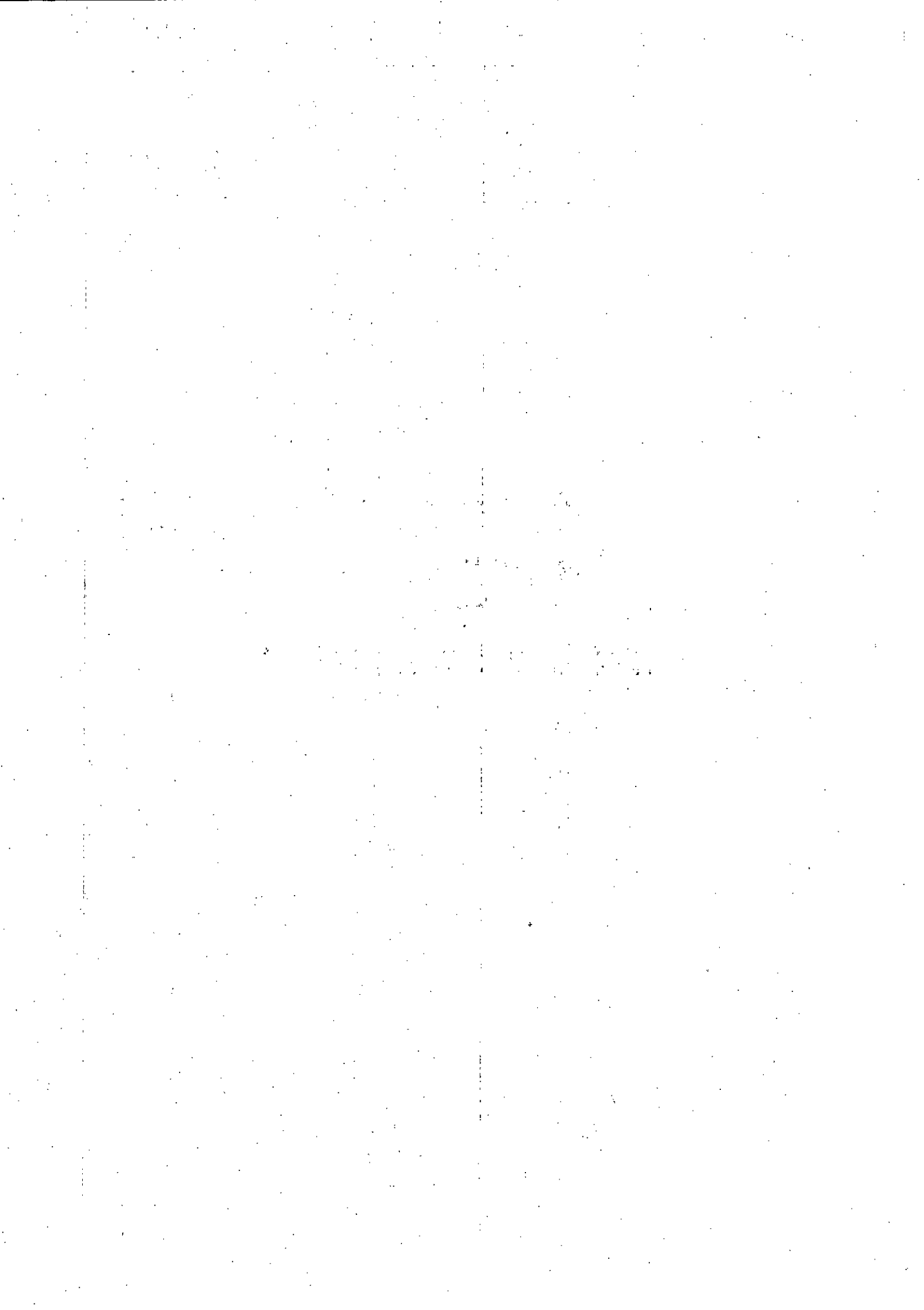
Notre application au bras RRR a suivi le même démarche. Les résultats de simulation sont en annexe (voir Annexe A5).

CONCLUSION

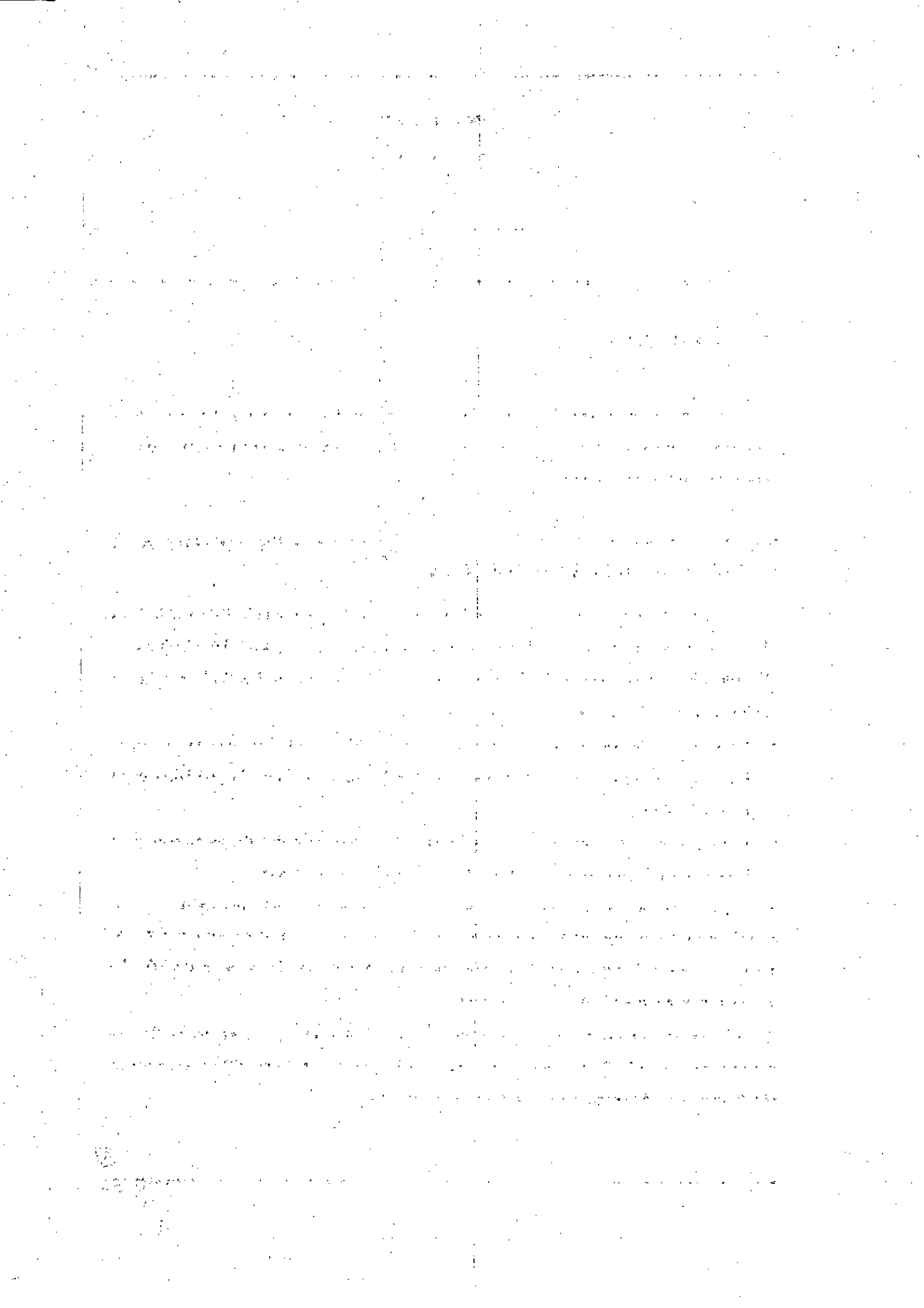
Dans ce chapitre, nous avons appliqué la logique floue à la commande en position de bras manipulateurs.

Les résultats obtenus ont été meilleur que ceux du contrôle conventionnel avec une régulation PID : la stabilité est mieux assurée (dépassement réduit), un gain en temps de réponse, et une meilleur prise en compte des perturbation.

Cependant un surcroît de performances est envisageable toujours dans les limites du contrôle flou : Adaptation des gains (de l'erreur et de sa variation) à la dynamique du système, supervision floue du contrôle PID, etc...



CHAPITRE VI
RÉSULTATS
ET
INTERPRÉTATIONS



Chapitre VI
RESULTATS
ET
INTERPRETATIONS

INTRODUCTION

Ce chapitre a pour but de faire une étude comparative entre ces deux commandes, et par la même occasion, faire la synthèse et l'analyse des résultats trouvés dans les chapitres précédents.

VI. 1 ETUDE COMPARATIVE A PARTIR DES RESULTATS OBTENUS AVEC LE BRAS SCARA

En comparant les résultats obtenus de l'application de la commande PID sur le bras Scara, pour une consigne échelon (figure III.9 et figure III.10), avec les résultats de l'application de la commande floue avec un contrôleur som-prod à 7 classes sur le même bras (fig V.7), nous constatons :

- Pour trouver de bons résultats avec la commande PID, il fallait trouver les matrices K_p , K_v , K_i qui permettait d'atteindre un bon compromis entre la précision et le temps de réponse.
- Les réponses des deux segments en boucle PID, présentent des dépassements plus importants par rapport à ceux observés avec les contrôleurs flous.
- L'erreur de position du premier segment avec la boucle PID pendant le régime transitoire est plus importante que celle avec le contrôleur flou, donc ce dernier cas nécessite moins de temps que le premier pour converger vers la position désirée. La même remarque pour le deuxième segment.

Pour le temps de réponse et de dépassement, le contrôleur flou présente de meilleures performances que le PID. Donc en régime transitoire, le contrôleur PID présente des résultats assez médiocres par rapport au contrôleur flou.

VI. 2 CARACTERISTIQUES D'UN CONTROLEUR PID

Un système PID se caractérise comme suit :

1/ Il ne peut réguler en principe que des processus linéaires. Souvent une partie seulement de la courbe de régulation peut être décrite suivant un mode linéaire, ce qui est une limitation importante au début de la régulation PID.

2/ Un bon réglage suppose qu'une description mathématique du processus est possible, et il doit encore être parfait dans bien des cas par une approche expérimentale. Le réglage est le gros problème des régulations PID.

3/ Une boucle de régulation PID n'est réglable de façon optimale dans la plupart des cas que pour une plage réduite.

4/ Une boucle de régulation PID est en général passablement sensible aux parasites.

Les parasites sur le signal de mesure ont toujours une grande influence. De plus, les modifications du processus dues au vieillissement et à la dérive des capteurs et des actionneurs, ont une influence notable sur le réglage d'une régulation PID.

5/ Les exigences imposées à l'appareillage de mesure, notamment au capteur de la valeur réelle, sont très grandes. Chaque variable du processus à réguler doit être mesurée précisément par un capteur. La fiabilité de la boucle de régulation PID dépend donc fortement de la qualité des composants utilisés.

VI. 3 CARACTERISTIQUES D'UN CONTROLEUR FLOU

Un système en logique floue se caractérise par :

1/ La logique floue s'adapte à la régulation de processus aussi bien linéaires que non-linéaires.

C'est une différence importante entre la commande floue et régulation PID, car la plupart des processus sont non-linéaires .

2/ La simplicité des réglages est la différence la plus frappante entre une régulation PID et une régulation floue.

La régulation en logique floue utilise les connaissances exprimées en mots et en phrases ordinaires par un utilisateur humain. Le système communique avec l'utilisateur dans son langage, adapté exactement à l'application . Dans beaucoup de cas, la facilité de maîtriser le système est une raison suffisante pour appliquer la régulation en logique

floue.

4/ La régulation en logique floue peut surpasser la régulation en PID pour ce qui est de la qualité de la réponse dynamique du système. Les dépassements et les temps de réponse peuvent être nettement inférieurs à ceux d'une boucle PID si le réglage est fait soigneusement [BOV 91] [SCH95].

5/ Une autre raison d'utiliser la logique floue est la tolérance des erreurs, plus grande que celle de la régulation PID. On peut traiter la réaction aux perturbations du processus par des règles spécifiques complémentaires.

6/ La fiabilité d'une régulation en logique floue peut être très grande, parce qu'on peut tenir compte dès l'énoncé des règles de la défaillance des capteurs et des actionneurs.

V. 4 ETUDE COMPARATIVE

D'après les caractéristiques citées ci dessus , on peut affirmer :

- * La simplicité de la maîtrise d'un dispositif au moyen de la logique floue est l'une de ses propriétés remarquables qui sautent aux yeux des utilisateurs.
- * Une boucle de régulation traditionnelle en PID suppose une analyse chiffrée du processus. Au contraire la logique floue se contente d'une analyse logique de la connaissance du processus. Ce dernier mode de description est souvent le seul possible, surtout pour les processus non linéaires.
- * La régulation en logique floue peut piloter des processus multivariables, complexes et grands systèmes.
- * L'application de la logique floue donne une plus grande liberté que la régulation PID dans le choix des capteurs à utiliser .
- * Une régulation en logique floue réagira souvent mieux aux perturbations que la boucle PID (i.e. plus robuste).

CONCLUSION

Les régulateurs électroniques *Proportionnel-Intégral-Dérivé* sont basés sur une analyse mathématique du processus à réguler. Les limitations de cette commande tenaient surtout à l'existence et à la connaissance d'une analyse mathématique exécutable. L'analyse mathématique de processus complexe est souvent très coûteuse, quand elle n'est pas totalement impossible.

Cette limitation est une des raisons importantes qu'on fait que la recherche s'est orientée vers d'autres techniques pour rendre possible malgré tout le pilotage de certains processus.

On peut penser que la logique floue constitue un complément précieux dans le cas de systèmes difficilement identifiables ou dont les paramètres subissent des variations brutales.

CHAPITRE VII
PRESENTATION
DU
LOGICIEL

Faint, illegible text, possibly bleed-through from the reverse side of the page. The text is arranged in several paragraphs and appears to be a formal document or report. Due to the low contrast and high noise, the specific words and sentences cannot be transcribed accurately.

Chapitre VII
PRESENTATION
DU LOGICIEL

INTRODUCTION

Robsicom est un outil de travail dont l'objectif est de regrouper tous les aspects pratiques de la technologie des robots. Il s'adresse avant tout aux étudiants en Génie Electrique intéressés par la Robotique ; et est particulièrement destiné à être un support du cours de Robotique enseigné en cinquième année Electronique à l'E.N.P.

Ce travail que nous présentons est l'aboutissement d'un effort continu développé avec l'encadrement du Labo d'Automatique et dont les jalons ont été posés grâce au travail d'élèves ingénieurs des deux dernières promotions (1996 et 1997).

Nous l'avons appelé ROBSICOM 3.0 (ROBots : SIMulation et COMmande).

Il a été développé exclusivement à l'aide de l'EDI (Environment Development Interface) Borland C++ 4.0 et est comme la plupart des logiciels de commerce d'aujourd'hui une application qui fonctionne sous Microsoft Windows 95 et même sous Windows 3.11.

Ce chapitre sert de documentation à ce produit.

V II. 1. ORGANISATION GENERALE

Le logiciel s'est voulu un environnement convivial pour l'utilisateur avec qui il doit interagir en temps réel. Ainsi il a utilisé pas mal de fonctionnalités offertes par la POO (Programmation Orientée Objet). Il utilise les fenêtres, menus, boîtes de dialogues, boîtes d'édition, etc...

Pour rester autonome, l'exécutable a été construit en statique n'utilisant ainsi aucune bibliothèque (fichiers DLL).

La figure VII.1 donne l'architecture de son fichier « makefile ».

Les fichiers *.cpp (C++) sont les fichiers sources ; ils contiennent en détails :

- Final.cpp : l'Application et ses fonctions virtuelles (Initialisation de la fenêtre principale, la fonction Run() qui s'exécute dès le lancement, etc...)
- Fen0.cpp : la fenêtre des simulations. Elle est le support des différents tracés de courbes.
- Fen1.cpp : la fenêtre de visualisation. Elle gère la visualisation animée.

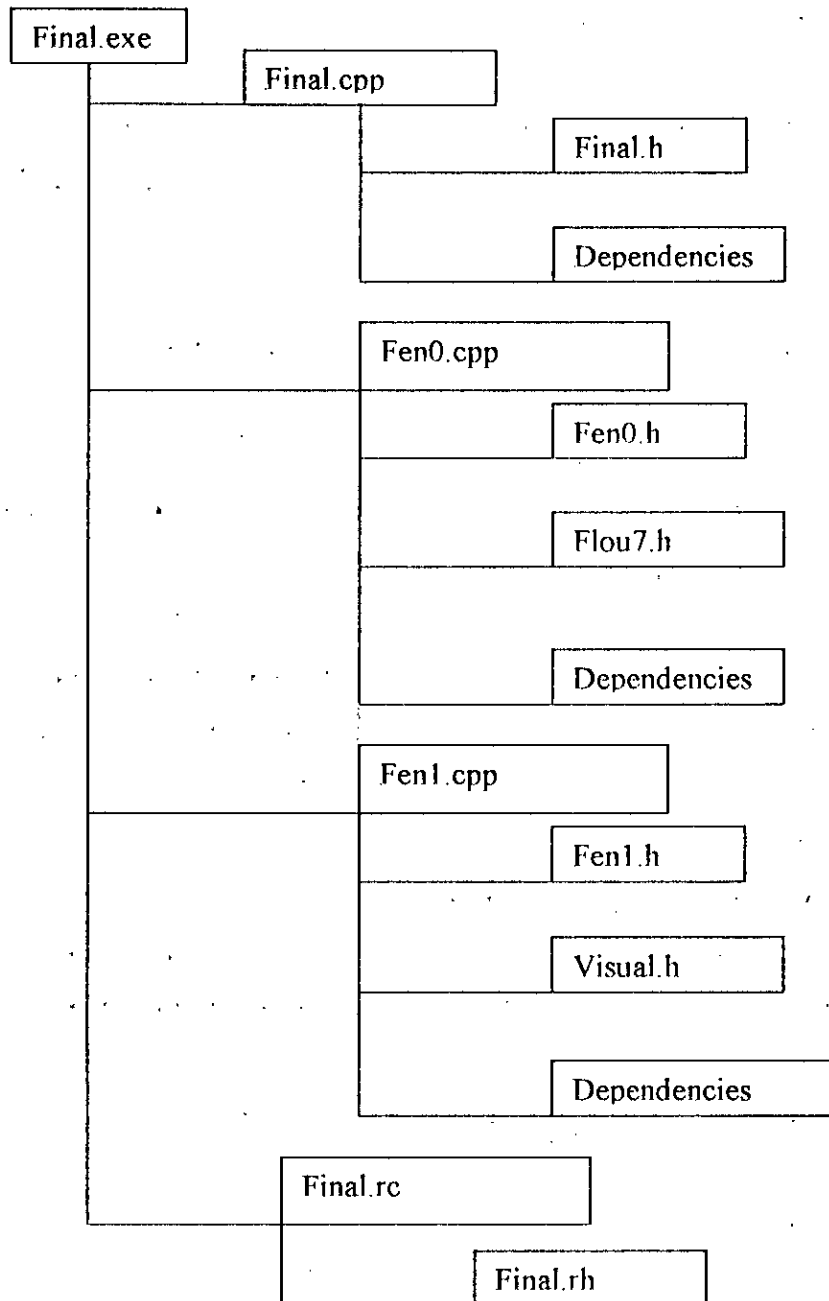


figure VII.1 Architecture de Robsicom 3.0

Remarque : Toujours dans l'esprit du C++, ces fichiers incluent des unités (extensions *.h « header »)

Les fichiers « Dependencies » : Ce sont les fichiers bibliothèques de la Société Borland dérivant de l'OWL (Objects Windows Library) et permettant l'accès aux différentes fonctions (mathématiques, graphiques, etc ...)

Le fichier Final.rc est la version non compilée des ressources (menus, boîtes de dialogues,...). Il inclut, comme d'ailleurs tous les fichiers *.cpp la liste des identificateurs de ressources regroupés dans Final.rh.

Une telle organisation a permis au logiciel de supporter des articles tous liés à un seul menu qui se présente dès son lancement. Examinons les fonctionnalités qu'ils nous offrent.

VII. 2. LES DIFFERENTS ARTICLES LIES AU MENU

Le menu comporte neuf sous-menus surgissants dans lesquels un séparateur délimite les articles permettant ainsi un regroupement selon l'orientation ou le domaine d'intérêt.

Passons en revue les huit « pop-up » (menus surgissants) que sont :

- Application : Fonctions globales liées au logiciel (Bibliothèque, Quitter, etc...).
- Cinématique : Modélisation cinématique directe et inverse des bras manipulateurs.
- Dynamique : Simulation en boucle ouverte de quelques bras.
- Trajectoire : Présentation des trajectoires utilisées.
- PID : Simulation de la commande PID de quelques bras.
- Flou : fonctions d'appartenance des ensembles flous et commande flou de robots.
- Animation 3D : Visualisation animée de bras selon une trajectoire définie.
- Help : L'aide en ligne du logiciel.

La *figure VII.2* indique le menu principal de ROBSICOM 3.0.

VII. 2. 1. APPLICATION

Dans ce « pop-up » on trouve les trois articles que sont :

- Biblio : qui est une rubrique permettant à l'utilisateur d'accéder aux paramètres

géométriques et dynamiques des robots utilisés dans le logiciel et qui sont :

- Bras Scara : 2 degrés de liberté,
- Bras Puma : 2 degrés de liberté,
- Bras Rhino : 2 degrés de liberté,
- Bras TTT : 3 degrés de liberté,
- Bras TRT : 3 degrés de liberté,
- Bras RRT : 3 degrés de liberté,
- Bras RRR : 3 degrés de liberté,
- Puma 560 : 3 degrés de liberté,

La *figure VII.3* montre la boîte de dialogue présentatrice de la bibliothèque de ROBSICOM 3.0.

- Effacer : fonction permettant à l'utilisateur d'initialiser la zone cliente de la fenêtre principale. En réalité c'est une fonction exécutée lors du passage d'un article à un autre utilisant une routine de tracé sur l'écran.
- Quitter : fermeture de l'application ; cette fonction ne diffère en rien de la fermeture par menu système (trois cases entête à droite d'une fenêtre Microsoft Windows 95).

VII. 2. 2. CINEMATIQUE

Ce menu regroupe la cinématique de tous les bras étudiés dans ce projet comme celle des robots figurant dans le programme du Cours de Robotique de cinquième année.

La séparation entre bras de 2 degrés de liberté (ddl) et 3 ddl est nette. La cinématique directe précède pour chaque bras la cinématique inverse ; et on dispose d'une visualisation en 3D de la position initiale et finale pour certains robots. Cette visualisation ne fait intervenir aucune notion de dynamique ni de trajectoire ; elle est faite dans le grand souci de présenter un outil pédagogique et d'accès facile même pour un néophyte en robotique.

La *figure VII.4* montre les deux étapes de la cinématique directe :

- a) Entrée des variables articulaires
- b) Lecture de la configuration résultante dans l'espace cartésien.

La *figure VII.5* montre les deux étapes opposées de la cinématique inverse.

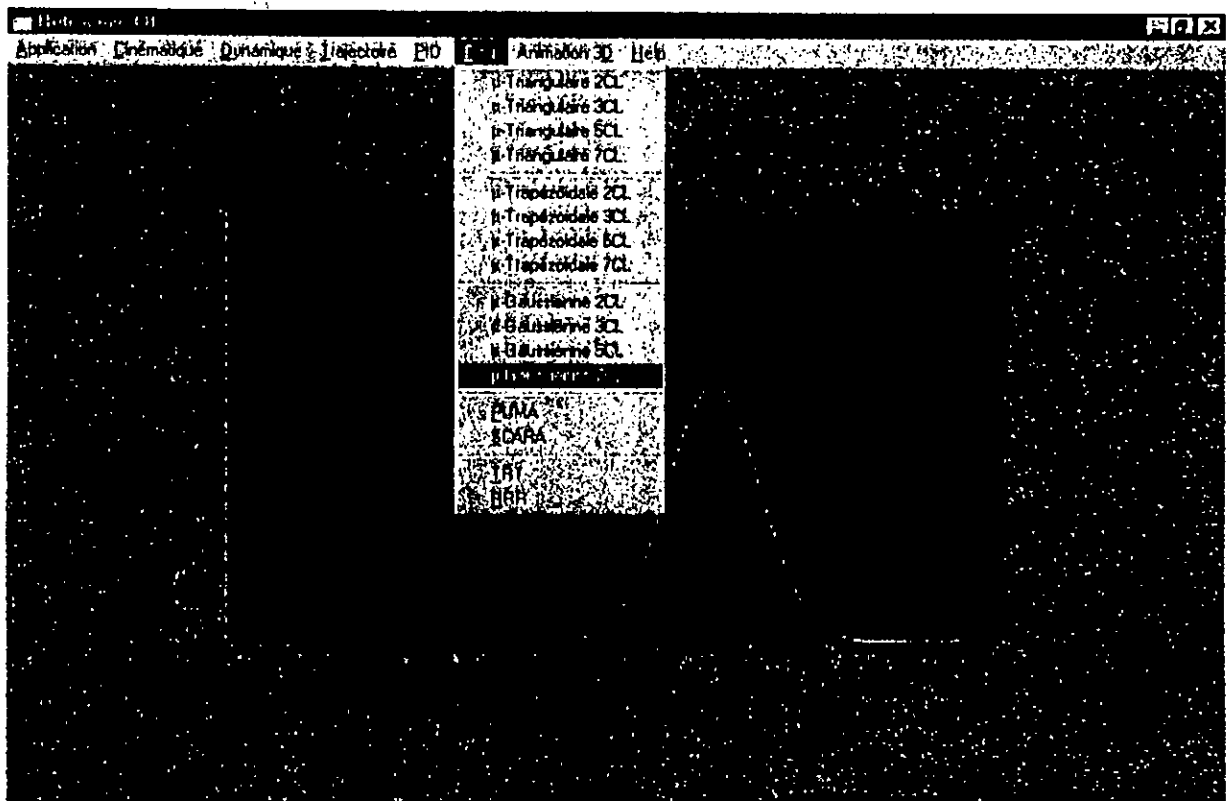


Figure VII.2 Menu principal de ROBSICOM 3.0

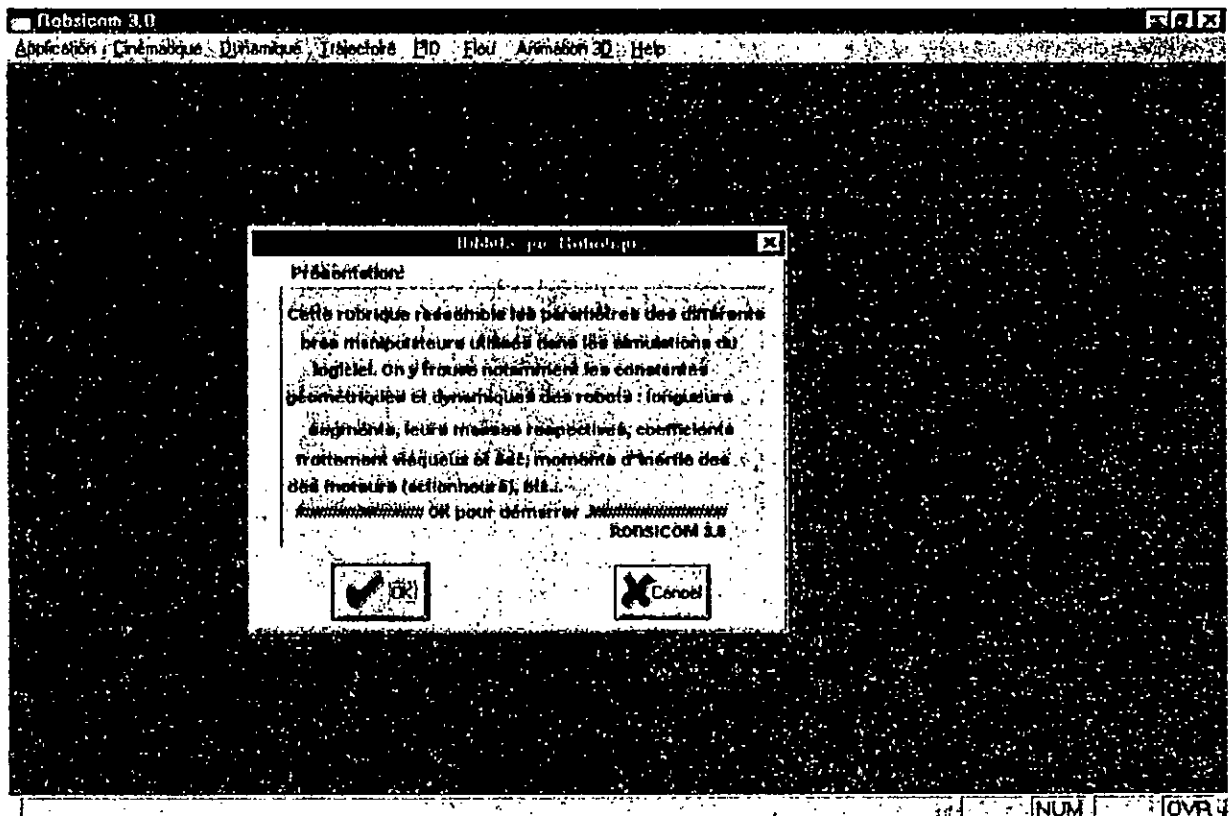


Figure VII.3 Présentation de la Bibliothèque de ROBSICOM 3.0

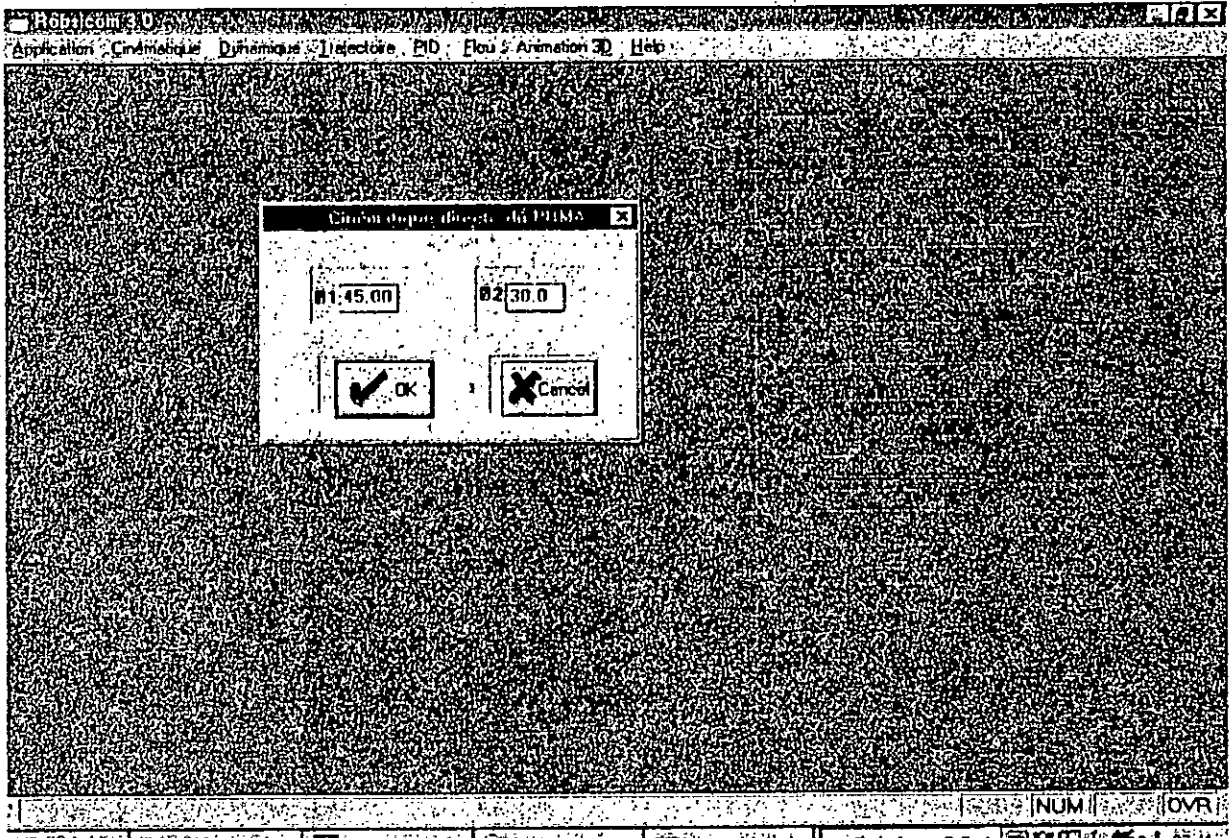


Figure VII.4 a Première étape de la cinématique directe

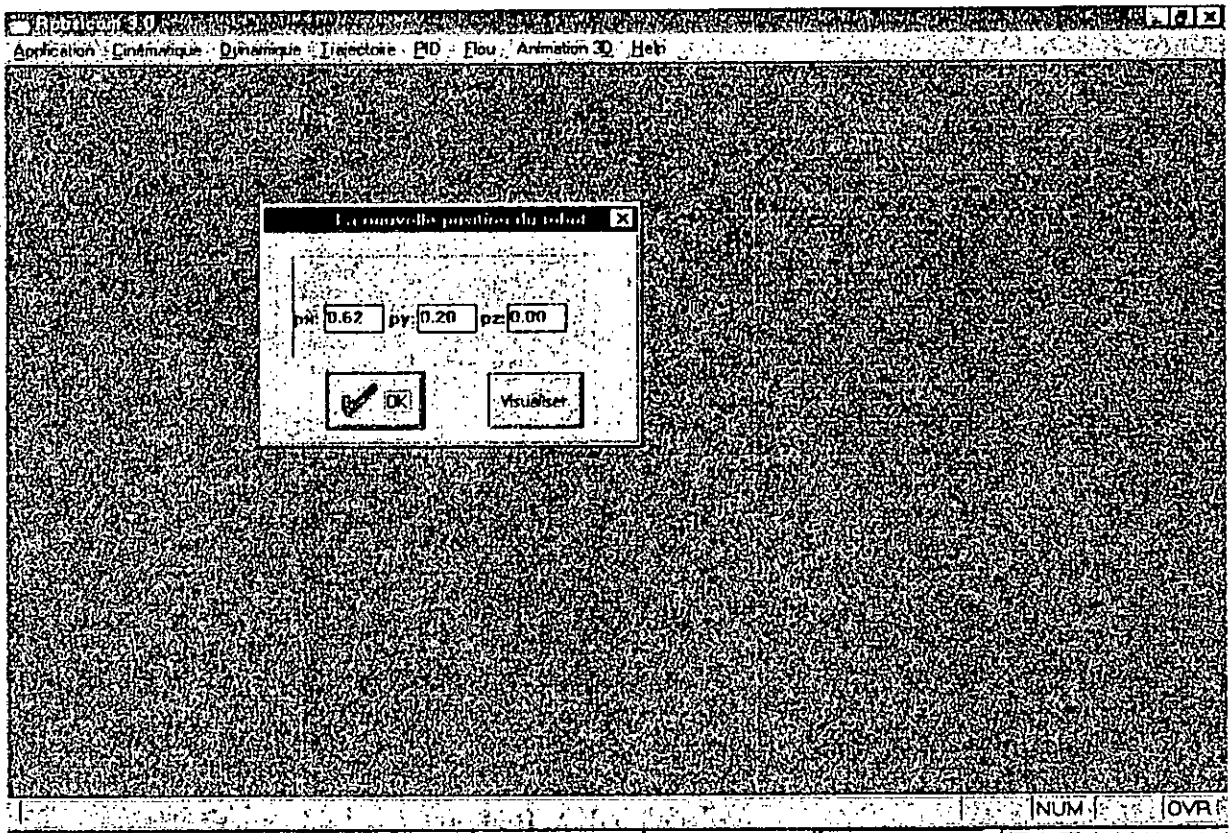


Figure VII.4 b Deuxième étape de la cinématique directe

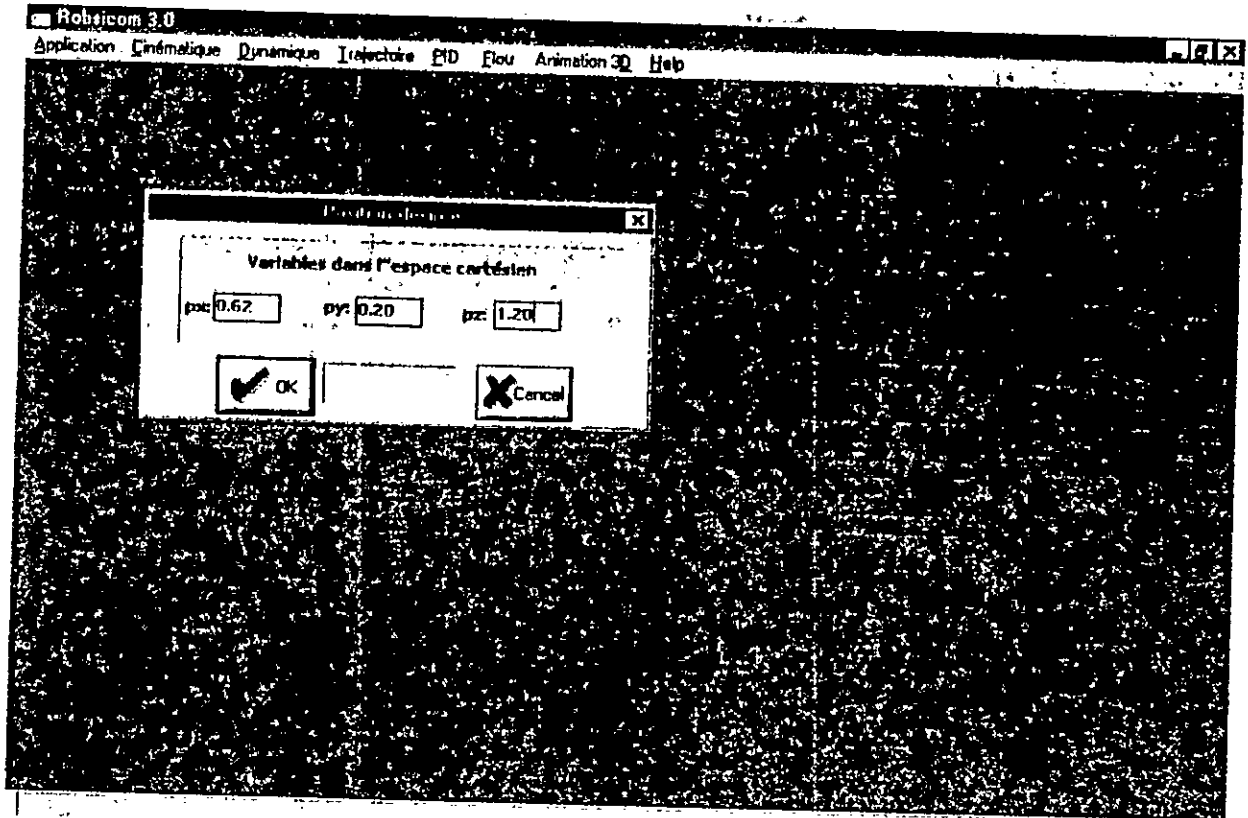


Figure VII.5a Première étape de la cinématique inverse

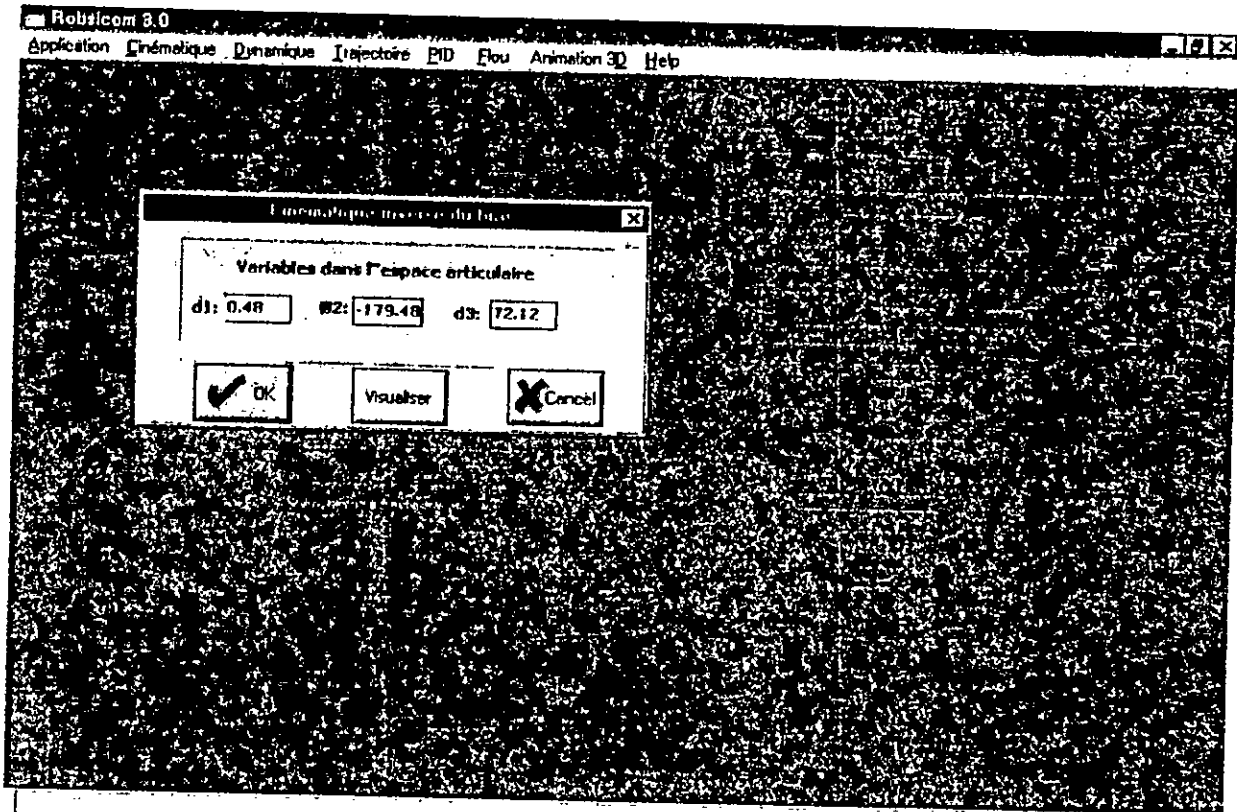


Figure VII.5b Deuxième étape de la cinématique inverse

Remarque :

Le déroulement normal des opérations de modélisation cinématique est assuré avec le bouton OK ; tandis que CANCEL permet à l'utilisateur d'interrompre le cours de l'opération liée à la boîte de dialogue.

VII. 2. 3. DYNAMIQUE

Dans ce menu nous proposons à l'utilisateur un outil lui permettant de simuler le comportement en boucle ouverte des bras suivant :

- Scara
- Puma
- RRR
- Puma 560

En sélectionnant un article de ce menu apparaît à l'écran quatre ou six courbes présentées sur un fond noir donnant ainsi les variables articulaires et leurs dérivées (position et vitesse). La graduation des courbes est faite de manière à focaliser au mieux les zones de courbes qui présentent un intérêt. La durée de simulation est fixée à 10 secondes.

La *figure VII.6* montre les résultats obtenus avec le Scara et le Puma 560.

VII. 2. 4. TRAJECTOIRE

Cette partie permet de visualiser le mouvement en poursuite de trajectoire. Elle propose les trajectoires classiques :

- Polynomiale,
- Cycloïdale
- Fenêtre de Viviani

A l'entrée d'un article on peut visualiser l'évolution d'une variable articulaire et ses deux premières dérivées par rapport au temps (position, vitesse, accélération). Cet article présente un intérêt plutôt pédagogique ; il recèle une fonction de description de trajectoire qui sera utilisée lors de la visualisation animée. Toutefois l'utilisateur peut jouer à partir d'ici sur la durée de poursuite (temps initial, temps final).

La *figure VII.7* montre un exemple de trajectoire.

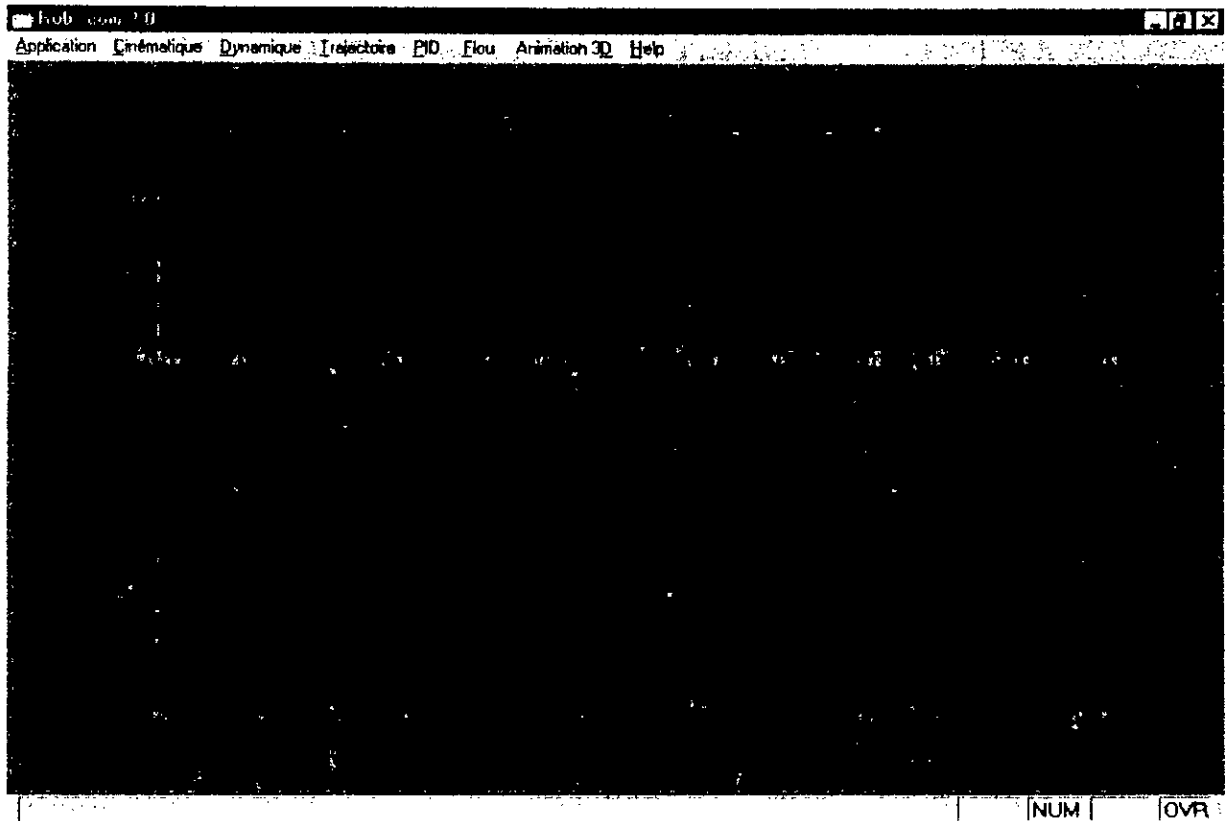


Figure VII. 6a Dynamique du bras Scara en boucle ouverte $u_1=u_2=1$

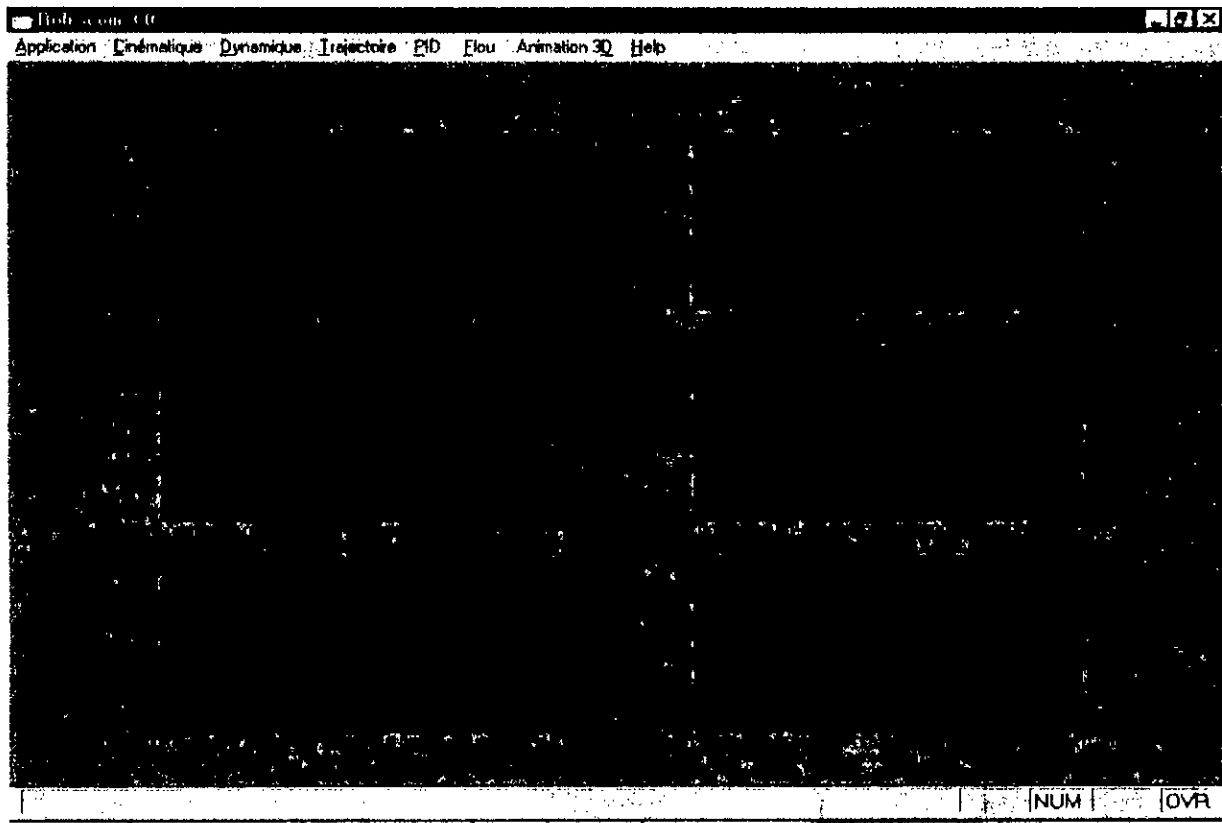


Figure VII. 6b Dynamique du bras Puma 560 en boucle ouverte $u_1=1, u_3=u_2=0$.

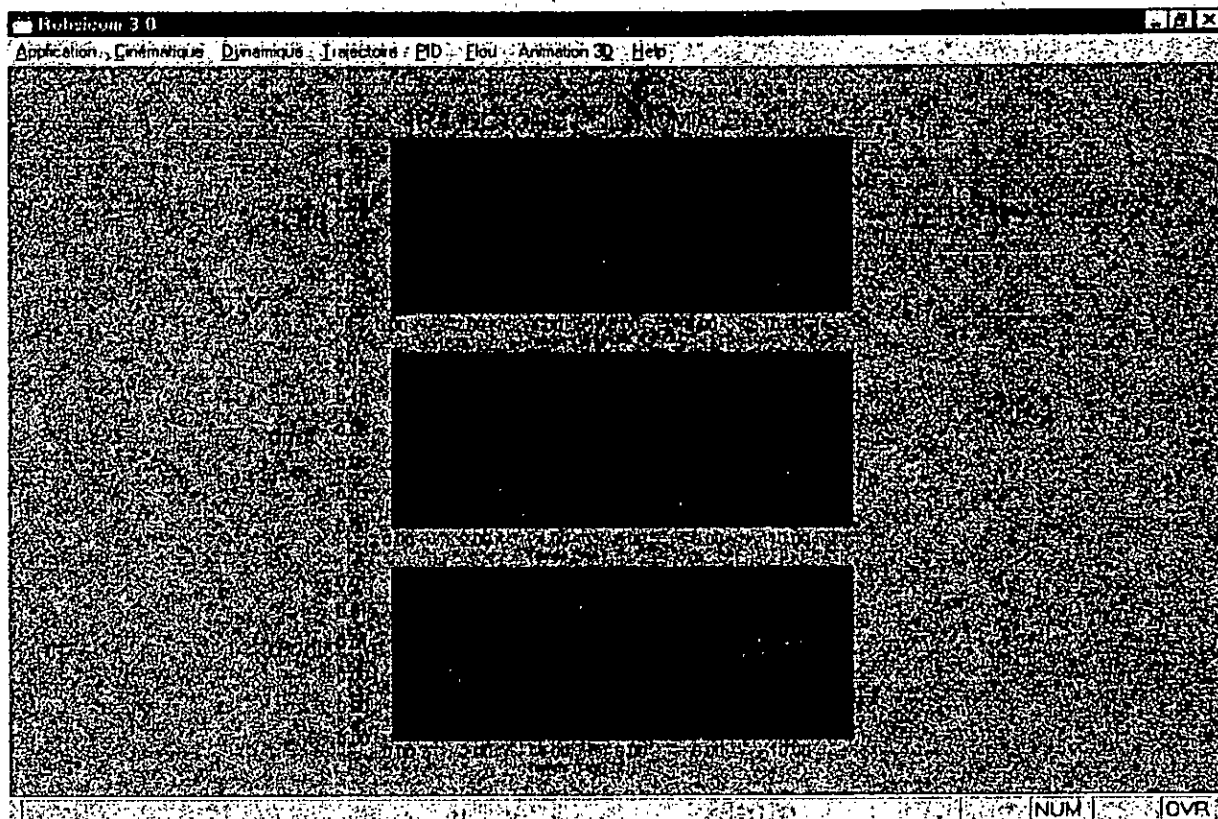


Figure VII.7 Trajectoire polynomiale générée de $t_0=0$ à $t_f=10s$

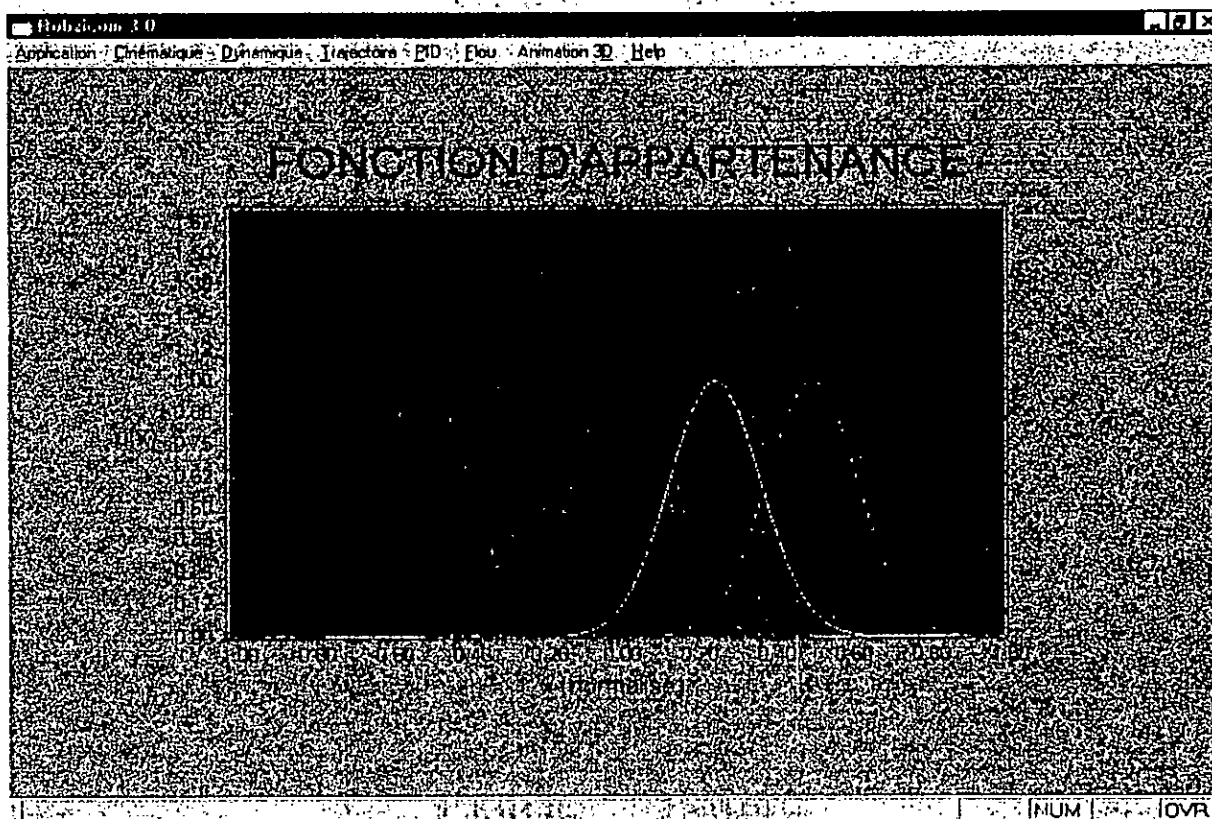


Figure VII.9 fonction d'appartenance gaussienne à 3 classes

VII. 2. 5. PID

On a voulu donner ici à l'utilisateur un outil de simulation pour la commande conventionnelle PID sur les bras suivants :

- Le Scara
- Le Puma
- Le RRR
- Le TRT
- Le Puma 560

Des boites de dialogue apparaissent pour donner à l'utilisateur un moyen de jouer sur les valeurs des constantes k_p , k_v , k_i du correcteur PID. Les courbes donnant :

- La référence ou configuration désirée : consigne,
- L'évolution de la variable articulaire au cours du temps : mesure,
- L'erreur de position (consigne – mesure),
- La commande délivrée par le correcteur.

La *figure VII.8* donne les résultats de la simulation avec le bras Scara dans le cas de valeurs de constantes assurant une bonne convergence.

VII - 2 -6 - FLOU

Ce menu contient l'aspect pratique de la commande floue dans notre projet. Il présente d'abord les formes de fonctions d'appartenance avec des classes variables sur lesquelles des tests ont été faits pour la conception de contrôleurs flous. La fuzzification des variables de commande a été faite avec de telles distributions :

- Triangulaire : de 2, 3, 5, 7 classes,
- Trapézoïdale : de 2, 3, 5, 7 classes,
- Gaussienne : de 2, 3, 5, 7 classes.

La *figure VII.9* montre des exemples de fonctions d'appartenance.

La simulation par commande floue occupe également sa place. La démarche est similaire à celle avec la PID. Ici l'utilisateur choisit sa fonction d'appartenance et le nombre de classes avec lesquelles il veut travailler.

La *figure VII.10* montre les résultats de simulation obtenus avec le contrôleur de Sugeno à 7 classes dans la commande en position du bras Scara.

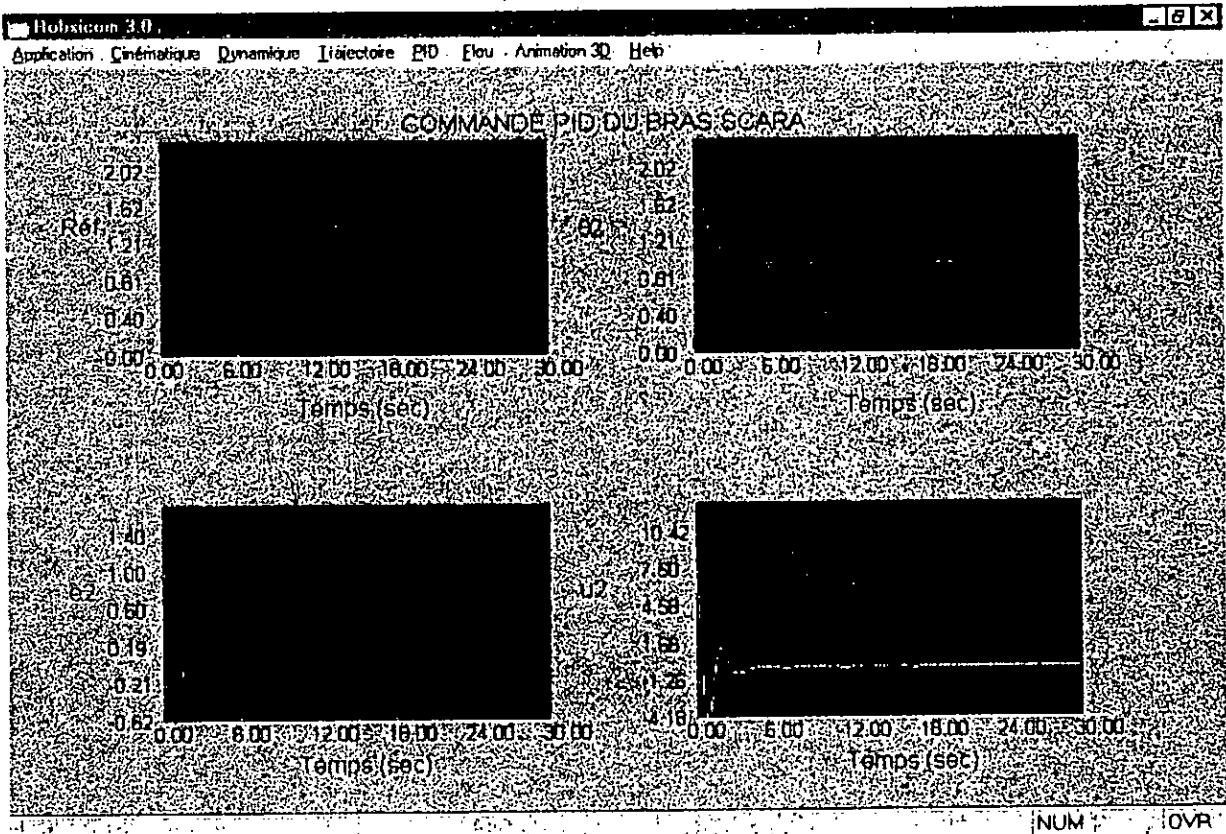


Figure VII.8 Comportement de la deuxième articulation du Scara en commande PID

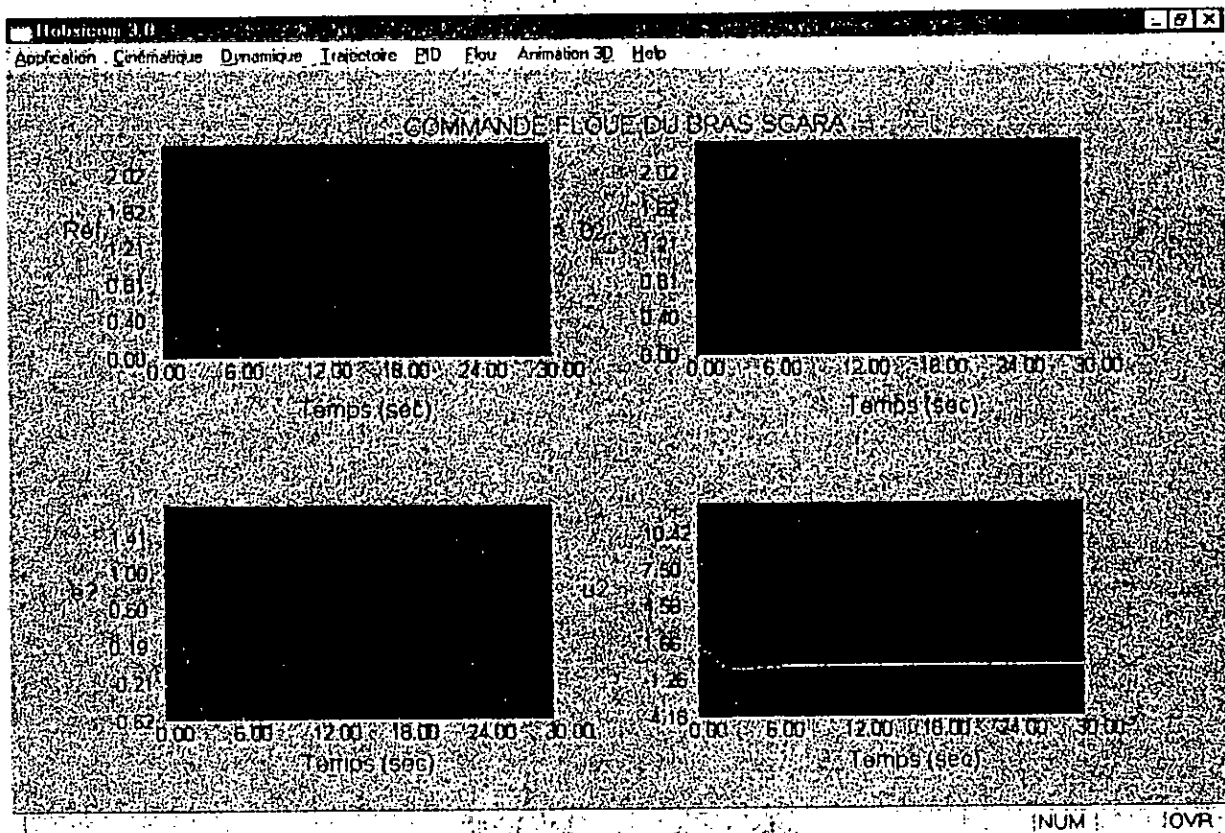


Figure VII.10 Comportement de la deuxième articulation du Scara en commande floue

VII. 2.7. ANIMATION

On se propose dans cette partie la visualisation graphique en mouvement des bras manipulateurs. Les dimensions du robot sont toujours données dans la biblio et l'utilisateur peut les modifier.

A ce menu une boîte de dialogue permet à l'utilisateur d'entrer le point cible. La position de départ correspond à celle où toutes les variables articulaires sont nulles.

La figure VII.11 montre l'animation du Puma à 2 ddl.

VII. 2. 8. HELP

Cette partie présente le logiciel :

- Boite à propos...
- Utilisation de l'aide
- Menu de ROBSICOM 3.0

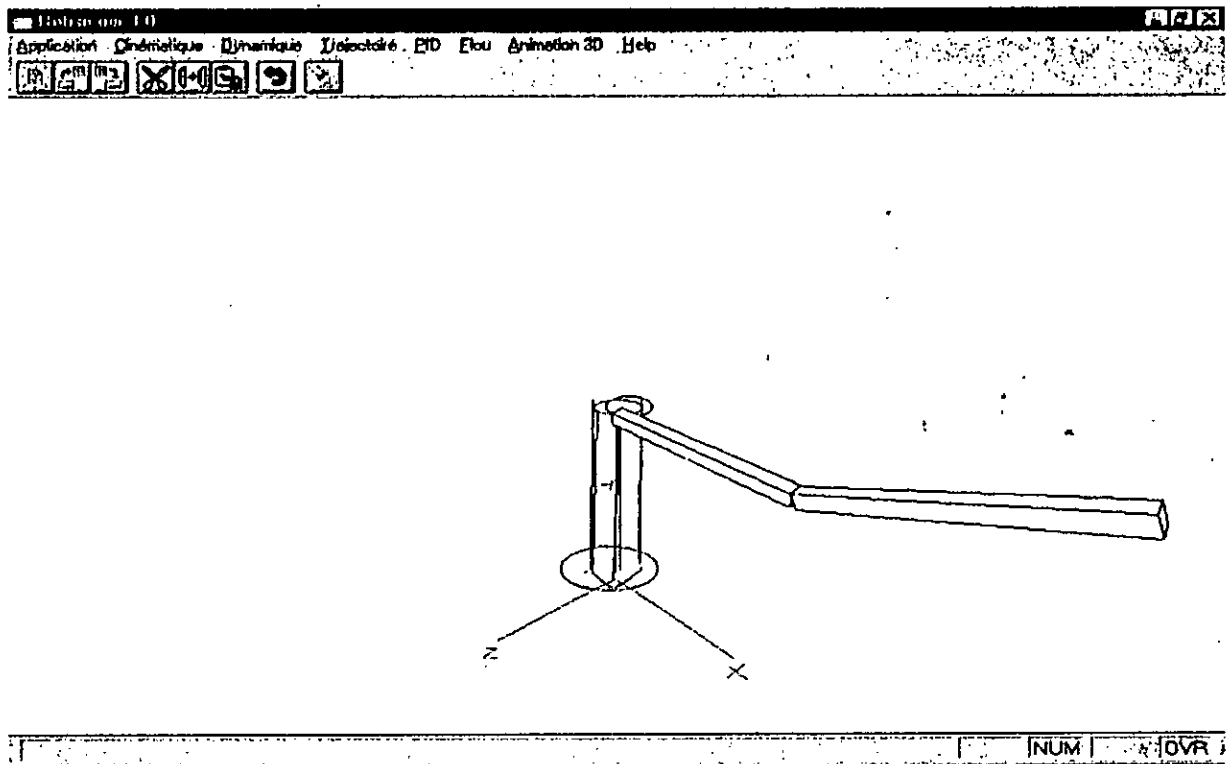


Figure VII. 11 Visualisation animée du bras Puma

VII. 3. PERSPECTIVES

Robsicom emploie des programmes externes tels que le logiciel Matlab. En utilisant les nouveaux standards de Microsoft Windows API (Application Programming Interface), le logiciel de Robsicom ne sera que plus performant.

Nous proposons une étude permettant l'implémentation de la méthode DH (Denavit-Hartenberg) qui sera un outil permettant de trouver la cinématique directe de n'importe quel bras à la seule définition de sa morphologie géométrique.

Dans la visualisation animée la visualisation des repères permettrait de mieux suivre le mouvement des articulations.

Les courbes de simulation présentées dans ce logiciel ont été obtenues grâce à des fonctions de tracé sur écran gérant l'échelle et la graduation de manière automatique. On pourrait présenter en Outils un article utilisant cette fonction pour la visualisation de n'importe quelle courbe (graphisme).

Robsicom reste ouvert à des développements pouvant le performer et l'étendre à des applications nouvelles et cela grâce à la puissance de la programmation orientée objet.

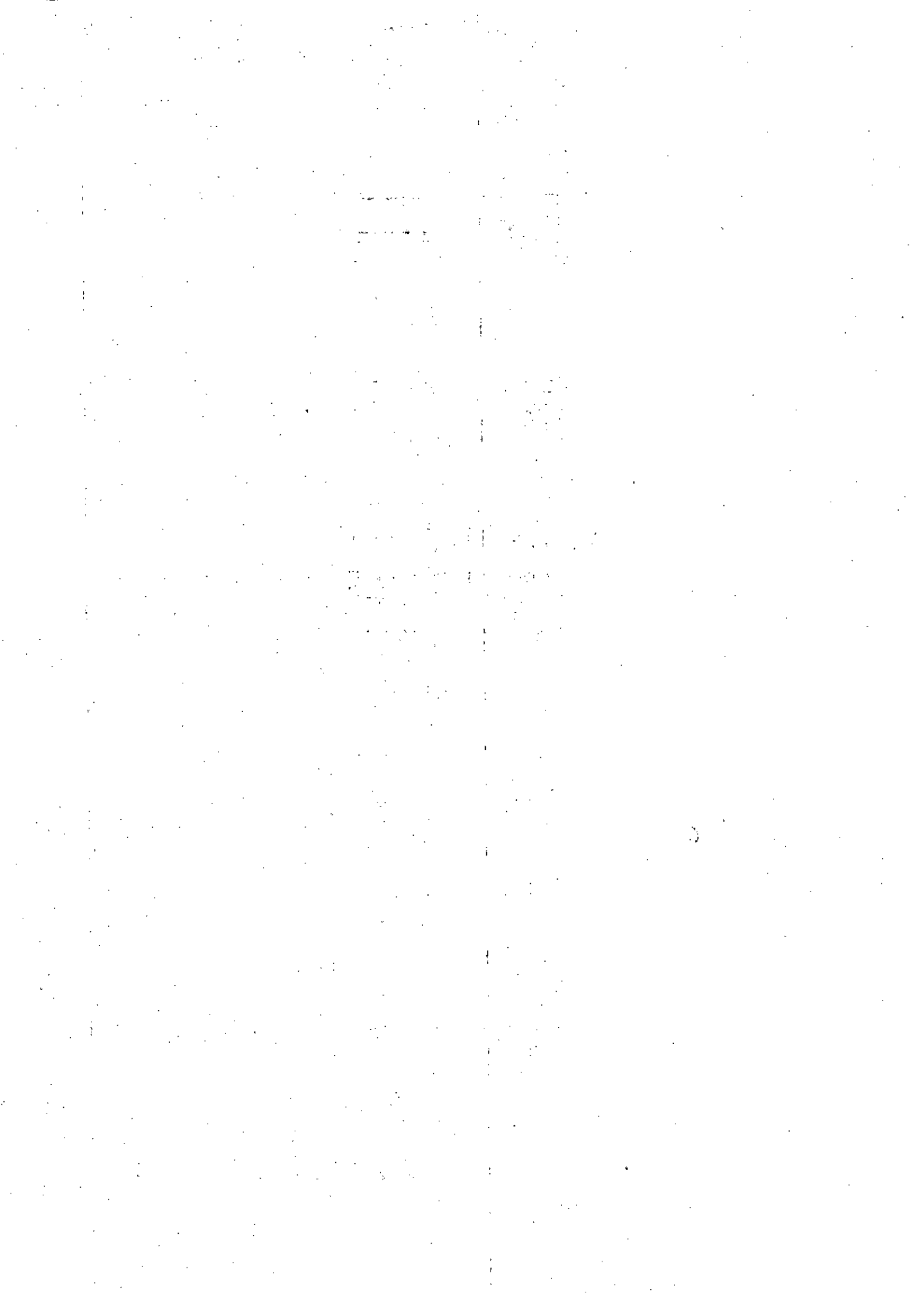
CONCLUSION

Ce logiciel nous a permis de donner à notre projet son aspect pratique. De ce fait, il épouse le plan de notre travail. Il englobe ainsi les outils qui nous ont permis de faire l'étude des bras dont les modèles sont disponibles dans la littérature ; et de nous confronter aux problèmes pratiques liés à la dynamique des robots : instabilité, non linéarité, couplage, etc...

Au delà des courbes de simulation dont l'interprétation est une démarche scientifique importante pour la maîtrise des systèmes, et qui sont présentées dans ce logiciel ; la visualisation animée a apporté un surcroît de clarté dans l'étude et une approche plus poussée vers la réalité.

Toutefois même si la réalisation pratique reste la finalité dans tout travail de conception, il reste vrai qu'un projet bien étudié permet la maîtrise de la majorité des paramètres dès l'étape de simulation.

Ce logiciel se veut d'abord un outil didactique mais aussi un moyen de simulation et de conception dans un projet robot.



CONCLUSION GENERALE

1. The first part of the document discusses the importance of maintaining accurate records of all transactions and activities. It emphasizes that this is essential for ensuring transparency and accountability in the organization's operations.

2. The second part outlines the various methods and tools used to collect and analyze data. This includes the use of surveys, interviews, and focus groups to gather qualitative information, as well as the application of statistical software for quantitative analysis.

3. The third part describes the process of identifying and measuring key performance indicators (KPIs). It highlights the need to select metrics that are relevant to the organization's strategic goals and to establish a baseline for comparison.

4. The fourth part details the implementation of a data management system. This involves setting up a secure database to store all collected information and ensuring that access is restricted to authorized personnel only.

5. The fifth part discusses the importance of regular reporting and communication of findings. It stresses that management should be kept informed of progress and any emerging trends or issues in a timely manner.

6. The sixth part addresses the challenges often encountered during the data collection and analysis process. These include issues such as data quality, incomplete responses, and the time and cost associated with large-scale data gathering.

7. The seventh part provides recommendations for improving the overall effectiveness of the data collection process. This includes suggestions for streamlining procedures, providing training to staff, and using technology to automate repetitive tasks.

8. The eighth part concludes by summarizing the key points discussed throughout the document and reiterating the commitment to data-driven decision-making and continuous improvement.

CONCLUSION GENERALE

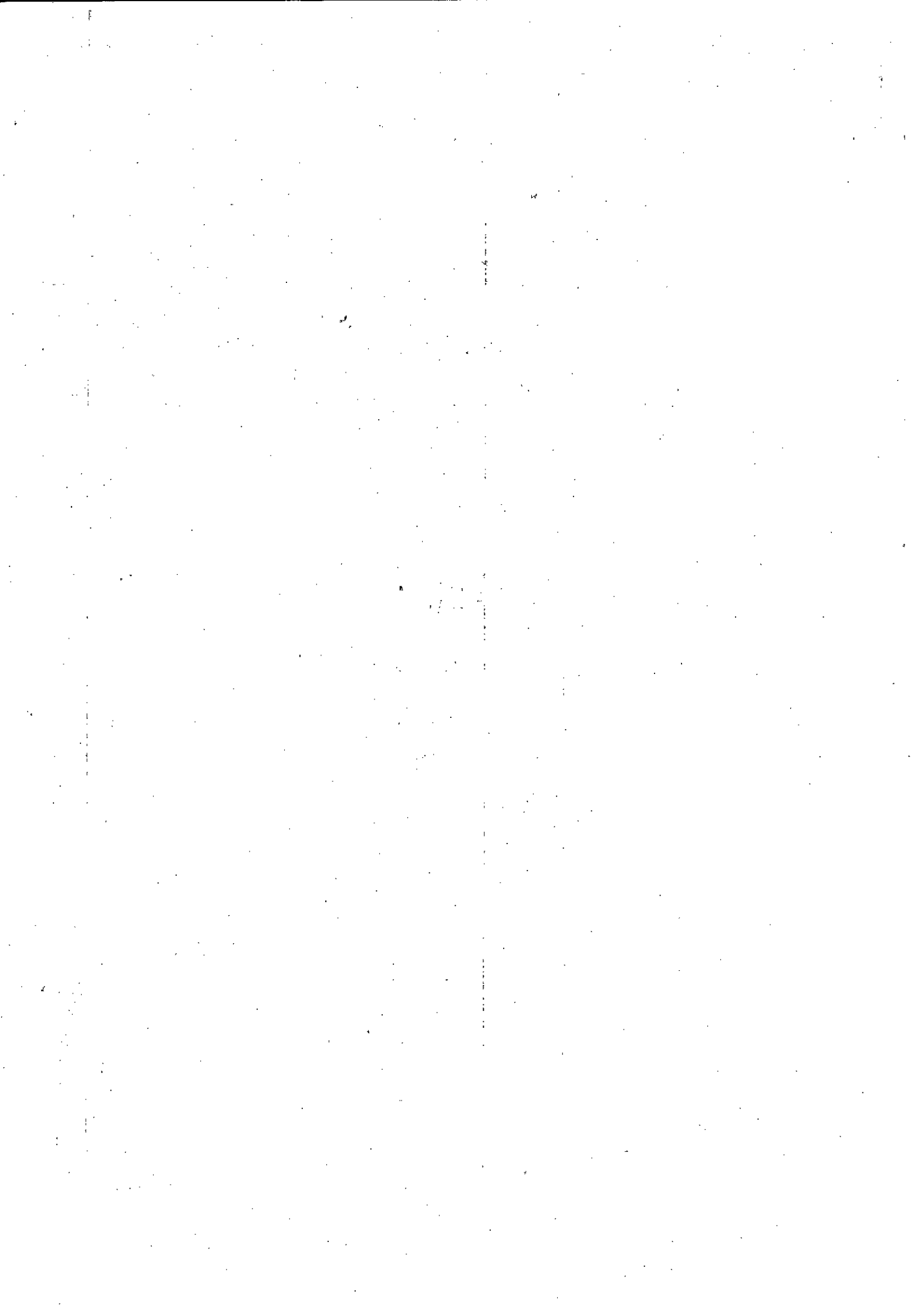
L'objectif principal de ce travail était d'essayer d'ouvrir la voie vers la maîtrise des techniques de commande des robots manipulateurs. Nous avons fait ressortir les limites du contrôle PID à la commande des systèmes interconnectés en général et des robots en particulier.

Notre étude a concerné principalement le robot Scara mais l'extension a été faite dans la mesure du possible à d'autres types de bras classiques (RRR, TRT) ; et notre démarche reste valable pour n'importe quelle structure.

Nous avons concrétisé notre travail à travers un logiciel (ROBSICOM 3.0) qui est un outil de simulation (graphique et dynamique) et de commande en robotique : une application fonctionnant sous Microsoft Windows 95.

En perspective, ce travail reste ouvert sur les aspects suivants :

- L'élaboration de lois de commande plus performantes, en position et en force, pour le contrôle du bras. On peut citer par exemple la commande floue adaptative avec l'utilisation des SOC « Self Organization Controllers ».
- La poursuite de trajectoires quelconques avec les types contrôles introduits,
- L'amélioration de la visualisation animée en 3D du logiciel et la création d'une carte interface d'utilisation pour un robot (hardware).



ANNEXES

THE UNIVERSITY OF CHICAGO
DEPARTMENT OF CHEMISTRY
RESEARCH REPORT NO. 100
BY
J. H. GOLDSTEIN AND
R. F. W. WILSON
PUBLISHED BY THE UNIVERSITY OF CHICAGO PRESS
CHICAGO, ILLINOIS
1951

NOTATION

Dans ce qui suit, nous utilisons cette notation :

$$S_i = \sin(\theta_i), \quad S_{ij} = \sin(\theta_i + \theta_j).$$

$$C_i = \cos(\theta_i), \quad C_{ij} = \cos(\theta_i + \theta_j).$$

1.1 MODELISATION CINEMATIQUE DU BRAS SCARA A 2DDL**MODELISATION GEOMETRIQUE DIRECT :**

Les paramètres D-H du bras Scara sont donnés dans le tableau suivant :

| Segments | Variables | $\theta_{initial}$ | α_i | d_i | a_i |
|-----------|------------|--------------------|------------|-------|-------|
| Segment 1 | θ_1 | 0 | 0 | 0 | a_1 |
| Segment 2 | θ_2 | 0 | 0 | 0 | a_2 |

La matrice de transformation globale est :

$${}^0A_2 = \begin{pmatrix} C_{12} & -S_{12} & 0 & a_1 C_{12} + a_2 C_2 \\ S_{12} & C_{12} & 0 & a_1 S_{12} + a_2 S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$a_1 = l_1, \quad a_2 = l_2.$$

l_1, l_2 longueurs des segments.

MODELISATION GEOMETRIQUE INVERSE :

$$\text{Nous avons : } P_x = a_1 C_{12} + a_2 C_1$$

$$P_y = a_1 S_{12} + a_2 S_1$$

$$\theta_1 = \text{atan2}(P_y(a_1 C_2 + a_2) - a_1 P_x S_2, P_x(a_1 C_2 + a_2) + a_1 P_y S_2)$$

$$\theta_2 = \text{atan2}(S_2, C_2)$$

avec :

$$C_2 = \frac{P_x^2 + P_y^2 - a_1^2 - a_2^2}{2a_1 a_2}$$

$$S_2 = \pm \sqrt{1 - C_2^2}$$

1.2 MODELISATION CINEMATIQUE DU BRAS RHINO A 2DLL

Ce bras possède mêmes modèles géométriques direct et indirect que le bras Scara à 2dll.

1.3 MODELISATION CINEMATIQUE DU BRAS PUMA560 A 3DLL [NED96]

MODELISATION GEOMETRIQUE DIRECT :

Les paramètres D-H de ce bras sont résumés dans le tableau suivant :

| Segments | Variables | α_i | a_i | d_i | θ_i |
|----------|------------|------------|-------|-------|------------|
| Segment1 | θ_1 | -90 | 0 | 0 | θ_1 |
| Segment2 | θ_2 | 0 | l_2 | d_2 | θ_2 |
| Segment3 | θ_3 | 0 | l_3 | 0 | θ_3 |

La matrice de transformation globale est :

$${}^0A_3 = \begin{pmatrix} C_1 C_{23} & -C_1 S_{23} & -S_1 & C_1(a_2 C_2 + a_3 C_{23}) - d_2 S_1 \\ S_1 C_{23} & -S_1 S_{23} & C_1 & S_1(a_2 C_2 + a_3 C_{23}) + d_2 C_1 \\ -S_{23} & -C_{23} & 0 & -(a_2 S_2 + a_3 S_{23}) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

MODELISATION GEOMETRIQUE INVERSE :

$$\theta_1 = \text{Tan}^{-1} \left[\frac{-\text{arm} \cdot P_y \cdot \sqrt{P_x^2 + P_y^2 - d_2^2} - P_x d_2}{-\text{arm} \cdot P_x \cdot \sqrt{P_x^2 + P_y^2 - d_2^2} - P_y d_2} \right] \quad -\pi \leq \theta_1 \leq \pi$$

$$\theta_2 = \text{Tan}^{-1} \left[\frac{\sin \alpha \cdot \cos \beta + \text{arm} \cdot \text{elbow} \cdot \cos \alpha}{\cos \alpha \cdot \sin \beta - \text{arm} \cdot \text{elbow} \cdot \sin \alpha \cdot \cos \beta} \right] \quad -\pi \leq \theta_2 \leq \pi$$

$$\theta_3 = \phi - \pi / 2.$$

$$\cos \phi = \frac{a_2^2 + l_3^2 - P_x^2 + P_y^2 + P_z^2 + d_2^1}{2 \cdot l_2 \cdot l_3}$$

$$\sin \phi = \text{arm. elbow} \sqrt{1 - \cos^2 \phi}$$

$$\cos \alpha = \frac{-\text{arm.} \sqrt{P_x^2 + P_y^2 - d_2^2}}{\sqrt{P_x^2 + P_y^2 + P_z^2 - d_2^1}}$$

$$\sin \alpha = \frac{-P_z}{\sqrt{P_x^2 + P_y^2 + P_z^2 - d_2^2}}$$

$$\cos \beta = \frac{P_x^2 + P_y^2 + P_z^2 - d_2^2 + l_2^2 - l_3^2}{2 \cdot a_2 \sqrt{P_x^2 + P_y^2 + P_z^2 - d_2^2}}$$

arm et *elbow* sont des indicateurs de configuration associés aux trois premières articulations du bras Puma 560 [NED].

CALCUL DU MODELE DYNAMIQUE DU ROBOT PUMA 560

La forme du modèle est

$$(M(q))\ddot{q} + D(q)[\dot{q} \quad \dot{q}] + C(q)[\dot{q}^2] + G(q) = T$$

les matrices U_{ij} sont :

$$U_{11} = \begin{pmatrix} -s_1 & 0 & -c_1 & 0 \\ c_1 & 0 & -s_1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad U_{21} = \begin{pmatrix} -s_1 c_2 & s_1 s_2 & -c_1 & -(l_2 c_2 s_1 + d_2 c_1) \\ c_1 c_2 & -c_1 s_2 & -s_1 & l_2 c_2 c_1 - d_2 s_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{22} = \begin{pmatrix} -c_1 s_2 & -c_1 c_2 & 0 & -l_2 c_1 s_2 \\ -s_1 s_2 & -s_1 c_2 & 0 & -l_2 s_1 s_2 \\ -c_2 & s_2 & 0 & -l_2 c_2 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{31} = \begin{pmatrix} -s_1 c_{23} & s_1 s_{23} & c_1 & -s_1(l_2 c_2 + l_3 c_{23}) - d_2 c_1 \\ -s_1 s_{23} & -s_1 c_{23} & -s_1 & c_1(l_2 c_2 + l_3 c_{23}) - d_2 s_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{32} = \begin{pmatrix} -c_1 s_{23} & -c_1 c_{23} & 0 & -c_1(l_2 s_2 + l_3 s_{23}) \\ -s_1 s_{23} & -s_1 c_{23} & 0 & -s_1(l_2 s_2 + l_3 s_{23}) \\ -c_{23} & s_{23} & 0 & -(l_2 c_2 + l_3 c_{23}) \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{33} = \begin{pmatrix} -c_1 s_{23} & -c_1 c_{23} & 0 & -l_3 c_1 s_{23} \\ -s_1 s_{23} & -s_1 c_{23} & 0 & -l_3 s_1 s_{23} \\ -c_{23} & s_{23} & 0 & -l_3 c_{23} \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{12} = U_{13} = U_{23} = 0$$

avec

$$c_i = \cos(q_i) \quad c_{ij} = \cos(q_i + q_j)$$

$$s_i = \sin(q_i) \quad s_{ij} = \sin(q_i + q_j)$$

La matrice J_j est :

$$J_j = \begin{pmatrix} \alpha_i & 0 & 0 & x_i \\ 0 & \beta_i & 0 & 0 \\ 0 & 0 & \gamma_i & z_i \\ x_i & 0 & z_i & m_i \end{pmatrix}$$

avec

$$\begin{aligned} \alpha_i &= \frac{-I_{xxi} + I_{yyi} + I_{zzi}}{2} \\ \beta_i &= \frac{I_{xxi} - I_{yyi} + I_{zzi}}{2} \\ \gamma_i &= \frac{I_{xxi} + I_{yyi} - I_{zzi}}{2} \end{aligned} \quad i=1,2,3$$

et

$$\begin{aligned} x_2 &= -\frac{l_2}{2} m_2 \\ x_3 &= -\frac{l_3}{2} m_3 \\ z_1 &= -\frac{l_1}{2} m_1 \end{aligned}$$

Calcul de la matrice d'inertie :

$$\begin{aligned} M_{11}(q) &= \text{trace}(U_{11} J_1 U_{11}^T) + \text{trace}(U_{21} J_2 U_{21}^T) + \text{trace}(U_{31} J_3 U_{31}^T) \\ &= \alpha_1 + \gamma_1 + \alpha_2 c_2^2 + \beta_2 s_2^2 + \gamma_2 + 2 x_2 (l_2 c_2^2) + m_2 (l_2^2 c_2^2 + d_2 (d_2 + e)) + \alpha_3 c_{23}^2 + \beta_3 s_{23}^2 \\ &+ \gamma_3 + 2 x_3 (c_{23} (l_2 c_2 + l_3 c_{23})) + m_3 ((l_2 c_2 + l_3 c_{23})^2 + d_2^2) \\ &= I_1 + I_2 c_{23}^2 + I_3 c_{22} + I_4 c_2 c_{23} \end{aligned}$$

$$\begin{aligned} M_{22}(q) &= \text{trace}(U_{22} J_2 U_{22}^T) + \text{trace}(U_{32} J_3 U_{32}^T) \\ &= \alpha_2 + \beta_2 + \alpha_3 + \beta_3 + m_3 l_2 l_3 c_3 + m_3 l_2^2 \\ &= I_7 + I_4 c_3 \end{aligned}$$

$$\begin{aligned} M_{33}(q) &= \text{trace}(U_{33} J_3 U_{33}^T) \\ &= \alpha_3 + \beta_3 + 2 x_3 l_3 + m_3 l_3^2 \\ &= I_8 \end{aligned}$$

$$\begin{aligned} M_{12}(q) &= \text{trace}(U_{22} J_2 U_{21}^T) + \text{trace}(U_{32} J_3 U_{31}^T) \\ &= x_2 (d_2 s_2) + m_2 (l_2 d_2 s_2) + x_3 (d_2 s_{23}) + m_3 (l_2 s_2 + l_3 s_{23}) d_2 \\ &= I_5 s_{23} + I_6 s_2 \end{aligned}$$

$$\begin{aligned} M_{13}(q) &= \text{trace}(U_{33} J_3 U_{31}^T) \\ &= x_3 (d_2 s_{23}) + m_3 (l_3 d_2 s_{23}) = I_5 s_{23} \end{aligned}$$

$$\begin{aligned}
 M_{23}(q) &= \text{trace}(U_{33} J_3 U_{32}^T) \\
 &= \alpha_3 + \beta_3 + x_3 (2 l_3 + l_2 c_3) + m_3 (l_3^2 + l_2 l_3 c_3) \\
 &= I_8 + 0.5 I_4 c_3
 \end{aligned}$$

$$M_{21}(q) = M_{12}(q) \quad ; \quad M_{31}(q) = M_{13}(q) \quad ; \quad M_{32}(q) = M_{23}(q).$$

avec

$$I_1 = \alpha_1 + \gamma_1 + \beta_2 + \gamma_2 + \beta_3 + \gamma_3 + m_3 d_2^2 + m_2 d_2 (d_2 + e)$$

$$= I_{yy1} + I_{xx2} + m_2 d_2 (d_2 + e) + m_3 d_2^2 + I_{xx3}$$

$$I_2 = \alpha_3 - \beta_3 = I_{yy3} - I_{xx3}$$

$$I_3 = \alpha_3 - \beta_3 + m_3 l_2^2 = I_{yy3} - I_{xx3} + m_3 l_2^2$$

$$I_4 = m_3 l_2 l_3$$

$$I_5 = 0.5 m_3 l_3 d_2$$

$$I_6 = 0.5 m_2 l_2 (d_2 + e) + m_3 d_2 l_2$$

$$I_7 = \alpha_2 + \beta_2 + \alpha_3 + \beta_3 + m_3 l_2^2 = I_{zz2} + I_{zz3} + m_3 l_2^2$$

$$I_8 = I_{zz3}$$

Calcul de la matrice des couples de coriolis :

$$D_{11}(q) = \left(\frac{\partial M_{11}}{\partial q_2} + \frac{\partial M_{12}}{\partial q_1} - \frac{\partial M_{12}}{\partial q_1} \right)$$

$$= -2 I_2 c_{23} s_{23} - 2 I_3 c_2 s_2 - I_2 (s_2 c_{23} + c_2 s_{23})$$

$$D_{12}(q) = \left(\frac{\partial M_{11}}{\partial q_3} + \frac{\partial M_{13}}{\partial q_1} - \frac{\partial M_{23}}{\partial q_1} \right)$$

$$= -2 I_2 c_{23} s_{23} - I_4 c_2 s_{23}$$

$$D_{13}(q) = \left(\frac{\partial M_{12}}{\partial q_3} + \frac{\partial M_{13}}{\partial q_2} - \frac{\partial M_{23}}{\partial q_1} \right)$$

$$= 2 I_5 c_{23}$$

$$D_{21}(q) = \left(\frac{\partial M_{12}}{\partial q_2} + \frac{\partial M_{22}}{\partial q_1} - \frac{\partial M_{12}}{\partial q_2} \right) = 0$$

$$D_{23}(q) = \left(\frac{\partial M_{22}}{\partial q_3} + \frac{\partial M_{23}}{\partial q_2} - \frac{\partial M_{23}}{\partial q_2} \right) = -I_4 s_3$$

$$D_{31}(q) = \left(\frac{\partial M_{31}}{\partial q_3} + \frac{\partial M_{32}}{\partial q_1} - \frac{\partial M_{12}}{\partial q_3} \right) = 0$$

$$D_{32}(q) = \left(\frac{\partial M_{31}}{\partial q_3} + \frac{\partial M_{33}}{\partial q_1} - \frac{\partial M_{13}}{\partial q_3} \right) = 0$$

$$D_{33}(q) = \left(\frac{\partial M_{32}}{\partial q_3} + \frac{\partial M_{33}}{\partial q_2} - \frac{\partial M_{23}}{\partial q_3} \right) = 0$$

Calcul des éléments de la matrice des couples de centrifuges :

$$C_{11}(q) = \frac{1}{2} \left(\frac{\partial M_{11}}{\partial q_1} + \frac{\partial M_{11}}{\partial q_1} - \frac{\partial M_{11}}{\partial q_1} \right) = 0$$

$$C_{12}(q) = \frac{1}{2} \left(\frac{\partial M_{12}}{\partial q_2} + \frac{\partial M_{12}}{\partial q_2} - \frac{\partial M_{22}}{\partial q_1} \right)$$

$$= I_5 c_{23} + I_6 c_2$$

$$C_{13}(q) = \frac{1}{2} \left(\frac{\partial M_{13}}{\partial q_3} + \frac{\partial M_{13}}{\partial q_3} - \frac{\partial M_{33}}{\partial q_1} \right) = I_5 c_{23}$$

$$C_{21}(q) = \frac{1}{2} \left(\frac{\partial M_{21}}{\partial q_1} + \frac{\partial M_{21}}{\partial q_1} - \frac{\partial M_{11}}{\partial q_2} \right) = I_2 c_{23} s_{23} + I_3 c_2 s_2 + 0.5 I_4 (s_{23} c_2 + s_2 c_{23})$$

$$C_{22}(q) = \frac{1}{2} \left(\frac{\partial M_{22}}{\partial q_2} + \frac{\partial M_{22}}{\partial q_2} - \frac{\partial M_{22}}{\partial q_2} \right) = 0$$

$$C_{23}(q) = \frac{1}{2} \left(\frac{\partial M_{23}}{\partial q_3} + \frac{\partial M_{23}}{\partial q_3} - \frac{\partial M_{33}}{\partial q_2} \right) = -0.5 I_4 s_3$$

$$C_{31}(q) = \frac{1}{2} \left(\frac{\partial M_{31}}{\partial q_1} + \frac{\partial M_{31}}{\partial q_1} - \frac{\partial M_{11}}{\partial q_3} \right) = I_2 s_{23} c_{23} + 0.5 I_4 s_{23} c_2$$

$$C_{32}(q) = \frac{1}{2} \left(\frac{\partial M_{32}}{\partial q_2} + \frac{\partial M_{32}}{\partial q_2} - \frac{\partial M_{22}}{\partial q_3} \right) = 0.5 I_4 s_3$$

$$C_{33}(q) = \frac{1}{2} \left(\frac{\partial M_{33}}{\partial q_3} + \frac{\partial M_{33}}{\partial q_3} - \frac{\partial M_{33}}{\partial q_3} \right) = 0$$

Calcul des couples de gravitations :

$$r_1^T = \begin{bmatrix} 0 & 0 & -\frac{l_1}{2} & 1 \end{bmatrix}^T$$

$$r_2^2 = \begin{bmatrix} -\frac{l_2}{2} & 0 & d_2 & 1 \end{bmatrix}^T$$

$$r_3^3 = \begin{bmatrix} -\frac{l_3}{2} & 0 & 0 & 1 \end{bmatrix}^T$$

$$g = \begin{bmatrix} 0 & 0 & -|g| & 1 \end{bmatrix}^T$$

donc :

$$G_1(q) = -m_1 g^T U_{11} r_1^1 - m_2 g^T U_{21} r_2^2 - m_3 g^T U_{31} r_3^3 = 0$$

$$G_2(q) = -m_2 g^T U_{22} r_2^2 - m_3 g^T U_{32} r_3^3 = -0.5 m_2 g l_2 c_2 - 0.5 m_3 g l_3 c_{23} - m_3 g l_2 c_2$$

$$G_3(q) = -m_3 g^T U_{33} r_3^3 = -0.5 m_3 g l_3 c_{23}$$

CALCUL DU MODELE DYNAMIQUE DU ROBOT SCARA

La forme du modèle est :

$$(M(q) + I_M)q + D(q)[\dot{q} \quad \dot{q}] + C[\dot{q}^2] + H(q) = T$$

avec

$$I_M = \begin{pmatrix} n_1^2 I_{M1} & 0 \\ 0 & n_2^2 I_{M2} \end{pmatrix} \quad H(q) = \begin{pmatrix} n_1^2 (f_{v1} \dot{q}_1 + f_{s1} \operatorname{sgn}(\dot{q}_1)) \\ n_2^2 (f_{v2} \dot{q}_2 + f_{s2} \operatorname{sgn}(\dot{q}_2)) \end{pmatrix}$$

Les matrices U_{ij} sont :

$$U_{11} = \begin{pmatrix} -s_1 & -c_1 & 0 & -l_1 s_1 \\ c_1 & -s_1 & 0 & l_1 c_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{21} = \begin{pmatrix} -s_{12} & -c_{12} & 0 & -l_2 s_{12} - l_1 s_1 \\ c_{12} & -s_{12} & 0 & l_2 c_{12} + l_1 c_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{22} = \begin{pmatrix} -s_{12} & -c_{12} & 0 & -l_2 s_{12} \\ c_{12} & -s_{12} & 0 & l_2 c_{12} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$U_{12} = 0$$

avec :

$$\begin{aligned} c_i &= \cos(q_i) & c_{ij} &= \cos(q_i + q_j) \\ s_i &= \sin(q_i) & s_{ij} &= \sin(q_i + q_j) \end{aligned}$$

La matrice J_j est :

$$J_j = \begin{pmatrix} \frac{1}{3} m_1 l_1^2 & 0 & 0 & -\frac{1}{2} m_1 l_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{2} m_1 l_1 & 0 & 0 & m_1 \end{pmatrix}$$

Calcul de la matrice d'inertie :

$$\begin{aligned} M_{11}(q) &= \text{trace}(U_{11} J_1 U_{11}^T) + \text{trace}(U_{21} J_2 U_{21}^T) \\ &= \frac{1}{3} m_1 l_1^2 + \frac{1}{3} m_2 l_2^2 + m_2 l_1^2 + m_2 l_1 l_2 c_2 \end{aligned}$$

$$\begin{aligned} M_{12}(q) &= \text{trace}(U_{22} J_2 U_{21}^T) \\ &= \frac{1}{3} m_2 l_2^2 + \frac{1}{2} m_2 l_1 l_2 c_2 \end{aligned}$$

$$\begin{aligned} M_{22}(q) &= \text{trace}(U_{22} J_2 U_{22}^T) \\ &= \frac{1}{3} m_2 l_2^2 \end{aligned}$$

$$M_{12}(q) = M_{21}(q)$$

Calcul du vecteur des couples de centrifuges :

$$D_1(q) = \left(\frac{\partial M_{11}}{\partial q_2} + \frac{\partial M_{12}}{\partial q_1} - \frac{\partial M_{12}}{\partial q_1} \right) = -m_2 l_1 l_2 s_2$$

$$D_2(q) = \left(\frac{\partial M_{11}}{\partial q_1} + \frac{\partial M_{11}}{\partial q_1} - \frac{\partial M_{11}}{\partial q_1} \right) = 0$$

Calcul de la matrice des couples de centrifuges :

$$C_{11}(q) = \frac{1}{2} \left(\frac{\partial M_{11}}{\partial q_1} + \frac{\partial M_{11}}{\partial q_1} - \frac{\partial M_{11}}{\partial q_1} \right) = 0$$

$$C_{12}(q) = \frac{1}{2} \left(\frac{\partial M_{12}}{\partial q_2} + \frac{\partial M_{12}}{\partial q_2} - \frac{\partial M_{22}}{\partial q_1} \right) = -\frac{1}{2} m_2 l_1 l_2 s_2$$

$$C_{21}(q) = \frac{1}{2} \left(\frac{\partial M_{21}}{\partial q_1} + \frac{\partial M_{21}}{\partial q_1} - \frac{\partial M_{11}}{\partial q_2} \right) = \frac{1}{2} m_2 l_1 l_2 s_2$$

$$C_{22}(q) = \frac{1}{2} \left(\frac{\partial M_{22}}{\partial q_2} + \frac{\partial M_{22}}{\partial q_2} - \frac{\partial M_{22}}{\partial q_2} \right) = 0$$

Valeurs utilisées pour les simulations :

f_{v1}, f_{v2} = coefficients de frottement visqueux,

f_{s1}, f_{s2} = coefficients de frottement secs,

Im_1, Im_2 = moments d'inertie des moteurs,

m_1, m_2 = masses des segments,

km_1, km_2 = constantes de proportionnalité des moteurs.

l_1, l_2 = longueurs des segments

I_1, I_2 = moment d'inertie des segments,

n_1, n_2 = réducteurs.

$$m_1 = 15.91, m_2 = 11.36, l_1 = .432, l_2 = .432;$$

$$f_{v1} = .0007, f_{v2} = .0007, f_{s1} = .001, f_{s2} = .001;$$

$$Im_1 = .0001, Im_2 = .0001, km_1 = .18, km_2 = .18;$$

$$n_1 = 120, n_2 = 50;$$

LE MODELE DU BRAS RHINO

L'équation du modèle dynamique est :

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) = \tau$$

La matrice d'inertie :

$$M(\theta) = \begin{pmatrix} l_1 + l_2 c_2 & l_3 + \frac{1}{2} l_2 c_2 \\ l_3 + \frac{1}{2} l_2 c_2 & l_3 \end{pmatrix}$$

Matrice des couples de coriolis et centrifuges :

$$C(\theta, \dot{\theta}) = \begin{pmatrix} (l_2 s_2) \left(\dot{\theta}_1 \dot{\theta}_2 + \frac{1}{2} \dot{\theta}_2^2 \right) \\ \frac{1}{2} l_2 \dot{\theta}_1^2 s_2 \end{pmatrix}$$

matrice de gravitation :

$$G(\theta) = \begin{pmatrix} l_4 c_1 + l_3 c_{12} \\ l_3 c_{12} \end{pmatrix}$$

Les caractéristiques du bras sont :

$$m_1 = 15.91 \text{Kg}$$

$$m_2 = 11.36 \text{Kg}$$

$$l_1 = l_2 = l = 0.432 \text{m}$$

MODELISATION DU BRAS ARTICULE RRR

Le modèle est donné par cette équation :

$$\begin{pmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \end{pmatrix} = (M(q) + I_M)q + D(q)(\dot{q} - \ddot{q}) + C(q)(\dot{q}^2) + F(q) + G(q)$$

On a :

$$I_{ixx} = I_{izz} = m \left(\frac{L^2}{3} + \frac{R^2}{4} \right)$$

$$I_{iyy} = \frac{mR^2}{2}$$

$$I_{ixy} = I_{iyz} = I_{ixz} = 0$$

$$\alpha_i = \frac{-I_{ixx} + I_{iyy} + I_{izz}}{2}$$

$$\beta_i = \frac{I_{ixx} - I_{iyy} + I_{izz}}{2}$$

$$\gamma_i = \frac{I_{ixx} + I_{iyy} - I_{izz}}{2}$$

Calcul des éléments M_{ij} :

$$M_{11} = I_{1yy} + s_2^2 I_{2xx} + c_2^2 I_{2yy} + m_3 (a_2 c_2)^2 + s_{23}^2 \cdot I_{3xx} + c_{23}^2 I_{3yy} + m_3 a_2 a_3 c_{23} c_2$$

$$M_{22} = I_{2zz} + I_{3zz} + (a_2^2 + a_2 a_3 s_3) m_3$$

$$M_{33} = I_{3zz}$$

$$M_{23} = M_{32} = \frac{a_2 a_3 m_3 c_3}{2} + I_{3zz}$$

$$M_{12} = M_{13} = M_{21} = M_{31} = 0 ;$$

Calcul les éléments de la matrice C :

$$C_{12} = (I_{3xx} - I_{3yy}) c_{23} + (I_{2xx} - I_{2yy}) c_2 s_2 + (m_2 - m_3) a_2^2 c_2 s_2 - 0.5 m_3 a_2 a_3 s_{223}$$

$$C_{21} = (I_{2yy} - I_{2xx}) c_2 s_2 + (\alpha_3 - \beta_3) c_{23} s_{23} + 0.5 m_3 a_2 a_3 s_{223} + m_3 a_2^2 c_2 s_2$$

$$C_{23} = -m_3 (a_3^2 c_{23} s_{23} + 0.5 a_3 s_{223})$$

$$C_{31} = (\alpha_3 - \beta_3) c_{23} s_{23} + 0.5 m_3 a_2 a_3 s_{23} c_2$$

$$C_{32} = 0.5 a_2 a_3 m_3 s_2$$

$$C_{11} = C_{13} = C_{22} = C_{33} = 0;$$

$$D_{11} = -a_2^2 c_2^2 m_2 c_1 s_1$$

$$D_{12} = 0$$

$$D_{23} = -0.5 a_2 a_3 m_3 s_{223}$$

$$D_{13} = D_{21} = D_{22} = D_{31} = D_{32} = D_{33} = 0$$

Calcul les termes de gravitations :

$$G_1 = 0$$

$$G_2 = 0.5 m_2 g c_2 + m_3 g (0.5 a_3 c_{23} + a_2 c_2)$$

$$G_3 = 0.5 a_3 m_3 g c_{23}$$

Les éléments de la matrice F :

F_{ii} coefficient de frottement visqueux ramené à l'articulation .

$$F_{ij} = 0, \text{ pour } i \neq j.$$

Les éléments du vecteur I_M :

I_{Mi} : représente le moment d'inertie de l'actionneur i .

Les caractéristiques de ce bras sont :

$$m_1 = m_2 = m_3 = 10 \text{Kg.}$$

Les longueurs :

$$d_1 = a_2 = a_3 = 1;$$

Le rayon des segments :

$$r = 10 \text{m}$$

Coefficients de frottements visqueux :

$$f_1 = 75; f_2 = 10; f_3 = 1.$$

Coefficients proportionnels au signal de commande :

$$k_1 = 40; k_2 = 20; k_3 = 40.$$

MODELE DYNAMIQUE DU BRAS TRT

Le modèle dynamique sous forme d'équations d'état :

$$\dot{X} = A X + B U + D$$

$$Y = C X$$

$$X = [X_1 \ X_2 \ X_3 \ X_4 \ X_5 \ X_6] = [q_1 \ \dot{q}_1 \ q_2 \ \dot{q}_2 \ q_3 \ \dot{q}_3]$$

avec :

$$Y^T = [y_1 \ y_2 \ y_3] = [q_1 \ q_2 \ q_3] \text{ vecteur de sortie .}$$

$$U^T = [U_1 \ U_2 \ U_3] \text{ vecteur de commande}$$

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & \frac{-f_1}{m_1 + m_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\frac{f_2}{m_3 j^*} & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \frac{-f_3}{m_3} \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 0 & 0 \\ \frac{K_1}{m_1 + m_3} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{K_3}{m_3 j^*} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{K_3}{m_3} \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$d^i = (0 \quad -g \quad 0 \quad -2.(X_5 - l_2).X_4.X_6 / j^* \quad 0 \quad (X_5 - l_2/2).X_4^2)$$

$$j^* = (l_2)^2/3 - l_2.X_5 + X_5^2$$

Les caractéristiques du bras utilisés sont :

Les masses :

$$m_1=20\text{Kg} ; m_2=0\text{kg} ; m_3=10\text{Kg} ;$$

La longueur du deuxième segment :

$$l_2=0.75\text{m} ;$$

Les coefficients de frottement visqueux :

$$f_1=30 \text{Ns /m} ; f_2=7.825\text{Nms/rad} ; f_3=20\text{Ns/m}.$$

Les coefficients proportionnels au signal de commande :

$$k_1=100\text{N/V} ; k_2=10 ; k_3=10\text{N/V}.$$

ETUDE DU BRAS TRT :

I. EN BOUCLE OUVERTE

Les caractéristiques du bras TRT sont :

Les masses : $m_1=20$ Kg ; $m_2=0$; $m_3=10$ Kg ;

La longueur du deuxième segment : $l_2= 0.75$ m ;

Les coefficients de frottement visqueux : $f_1=30$ N.Sec⁻¹ ; $f_2=7.825$ N.Sec⁻¹ ; $f_3=20$ N.Sec⁻¹

Les coefficients de proportionnalité des actionneurs : $K_1=100$; $K_2=K_3=10$;

Les résultats de simulation en boucles ouverte, du bras TRT, sont illustrés par la figure 1 :

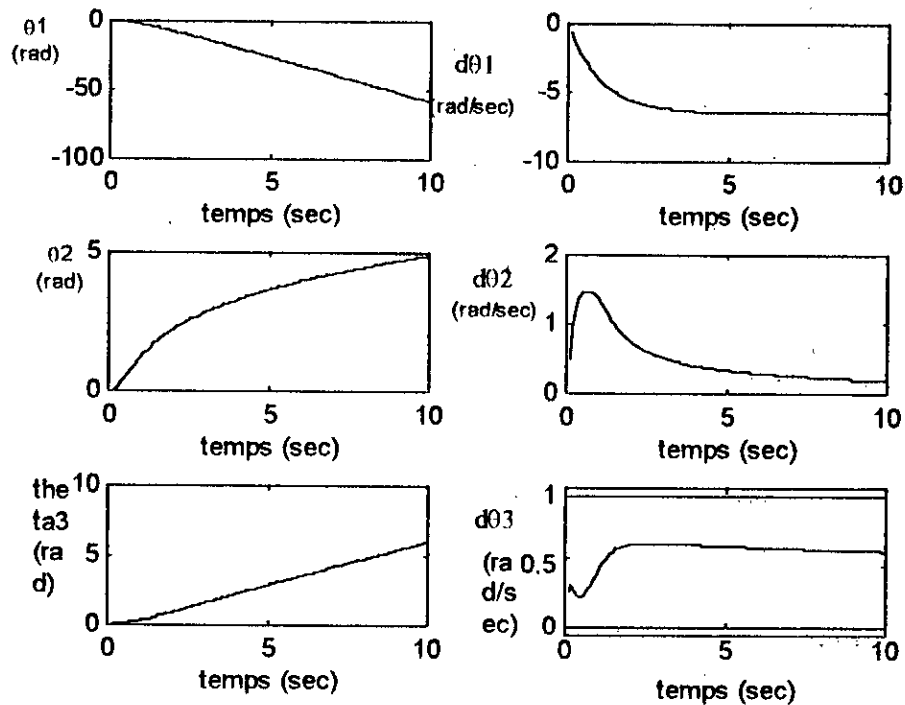
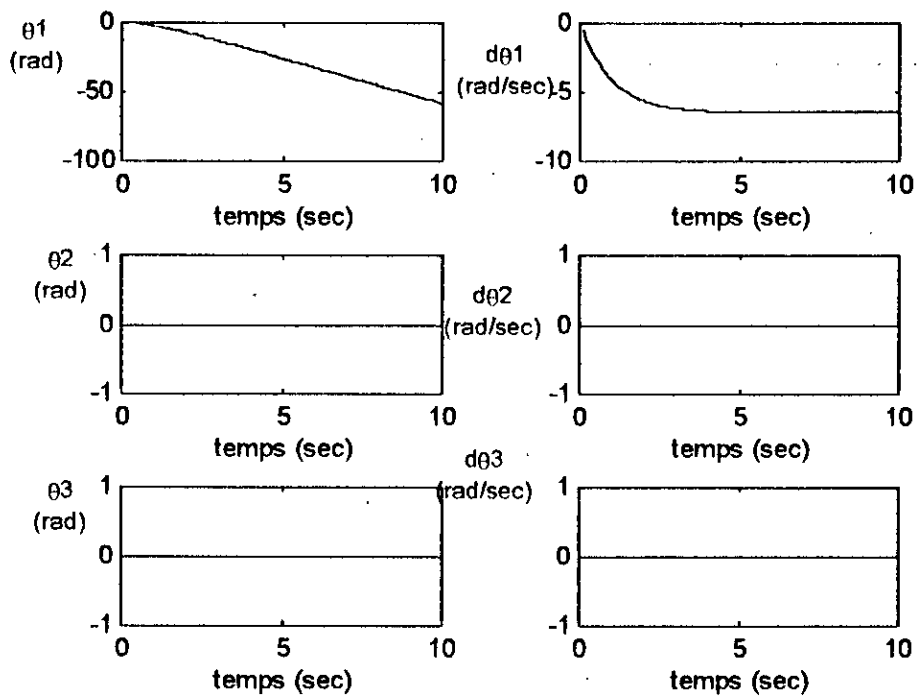
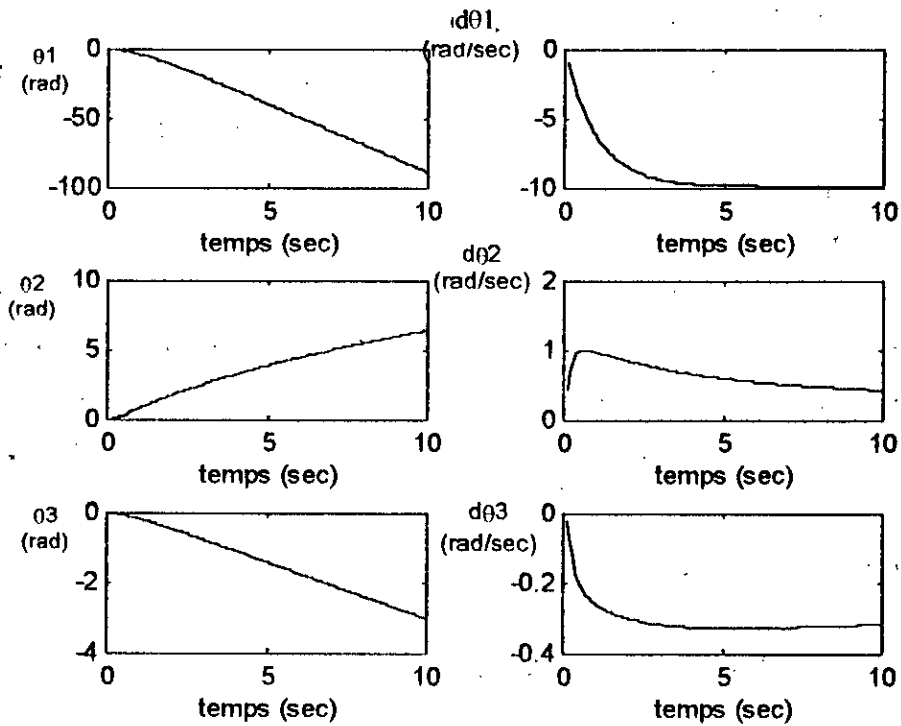


Figure1 : Réponse indicielle du bras TRT pour $u_1=u_2=u_3=1$

Figure2 : Réponse indicielle du bras TRT pour $u_1=1, u_2=u_3=0$.Figure3 : Réponse indicielle du bras TRT pour $u_2=1, u_1=u_3=0$.

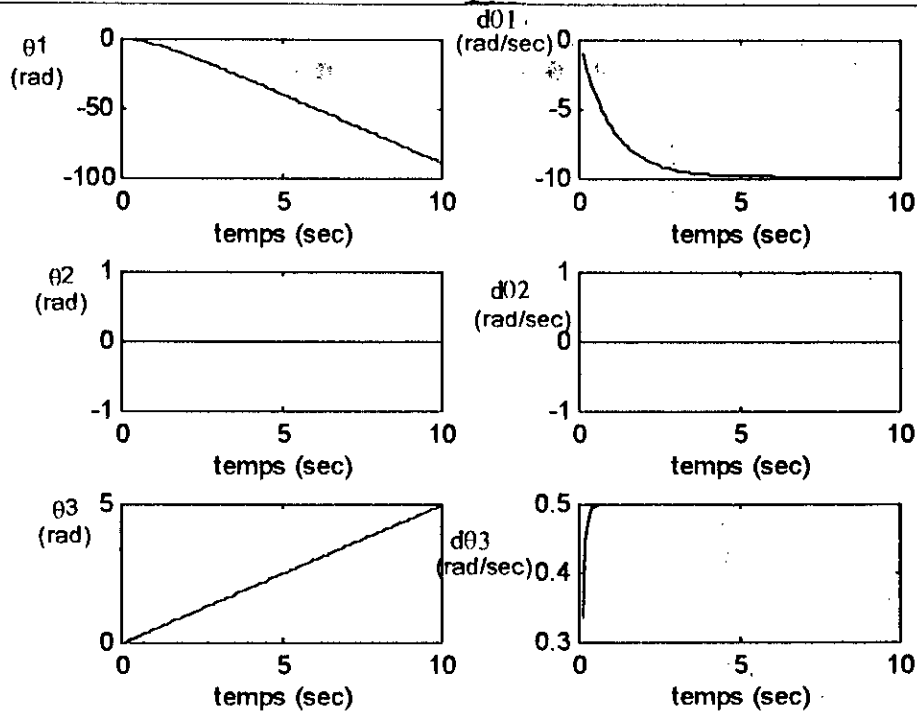


Figure4 : Réponse indicielle du bras TRT pour $u_3=1$, $u_1=u_2=0$.

ANALYSE ET INTERPRETATION

D'après la figure 1, pendant le régime permanent, les segments 1 et 3 décrivent des trajectoires linéaires, des droites, (la pente est une constante) ; par contre le segment 2 décrit une parabole, il se comporte comme un double intégrateur.

A partir de la figure 2, nous remarquons que les segments 2 et 3 sont immobiles, donc le segment 1 n'influe pas sur les deux autres. Par contre, la figure 3 montre que le segment 3 tend à avoir le même mouvement que celui du segment 2, mais dans le sens inverse (principe des moindres actions), on conclut que le segment 2 influe sur le troisième, mais ce dernier n'influe pas sur l'autre, du fait lorsqu'on applique un échelon sur le troisième, le deuxième reste immobile (figure 4).

En plus, le premier segment poursuit la même trajectoire, quelle que soit l'entrée appliquée aux deux autres segments.

On conclut que le bras TRT est un système couplé et instable, d'où nécessité d'une commande pour le stabiliser.

2. ETUDE EN BOUCLE FERMEE

Commande PID pour une consigne cycloïde pour

$$K_{p1}=2500, K_{v1}=750, K_{i1}=2, K_{p2}=50, K_{v2}=20, K_{i2}=10, K_{p3}=30, K_{v3}=10, K_{i3}=15.$$

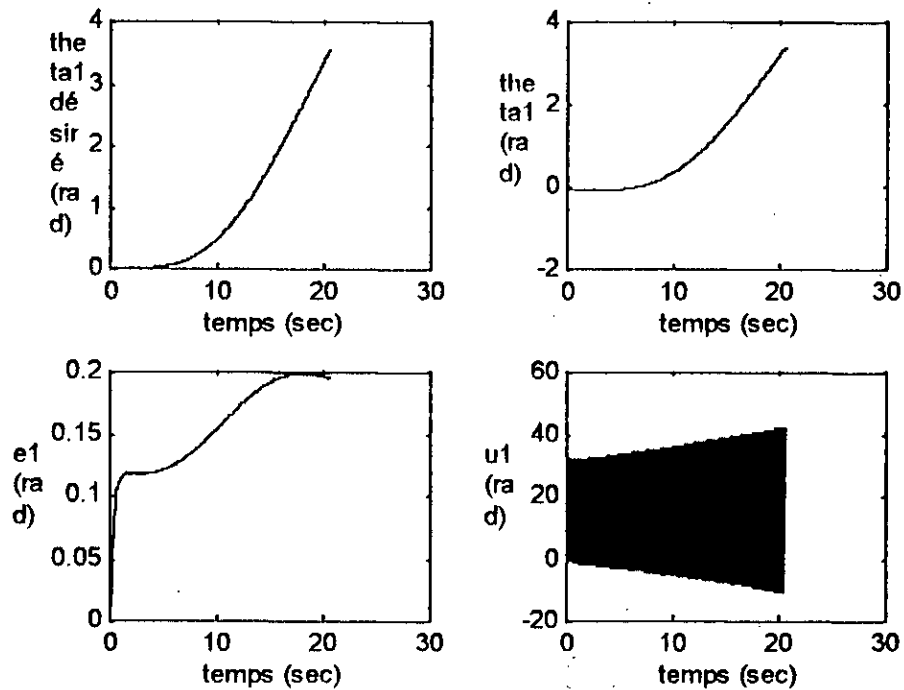


Figure 5 : Réponse du premier segment du bas TRT

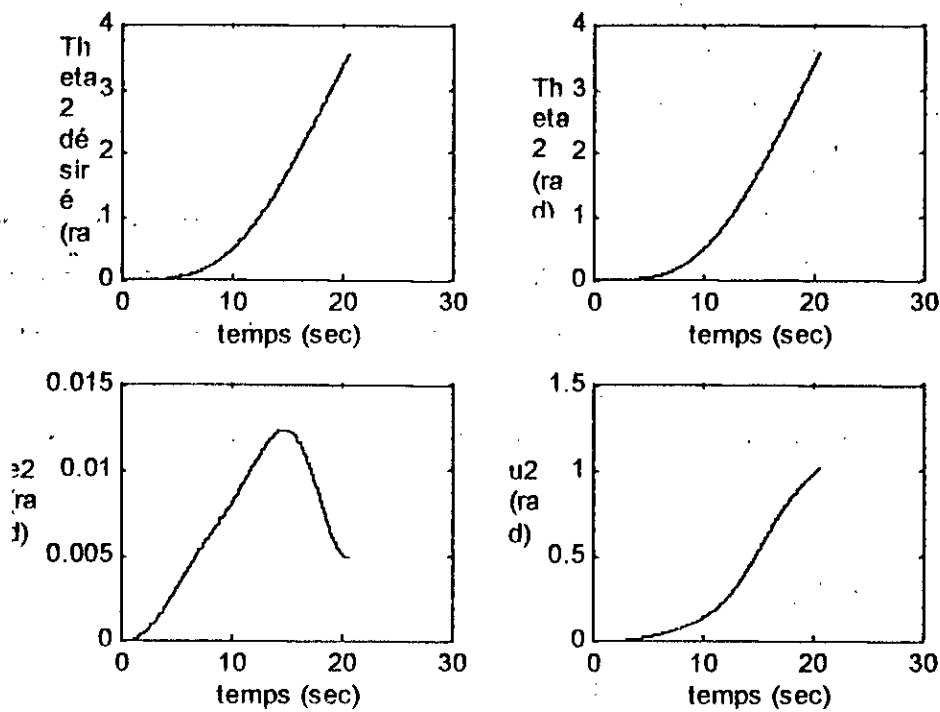


Figure 6 : Réponse du deuxième segment du bras TRT.

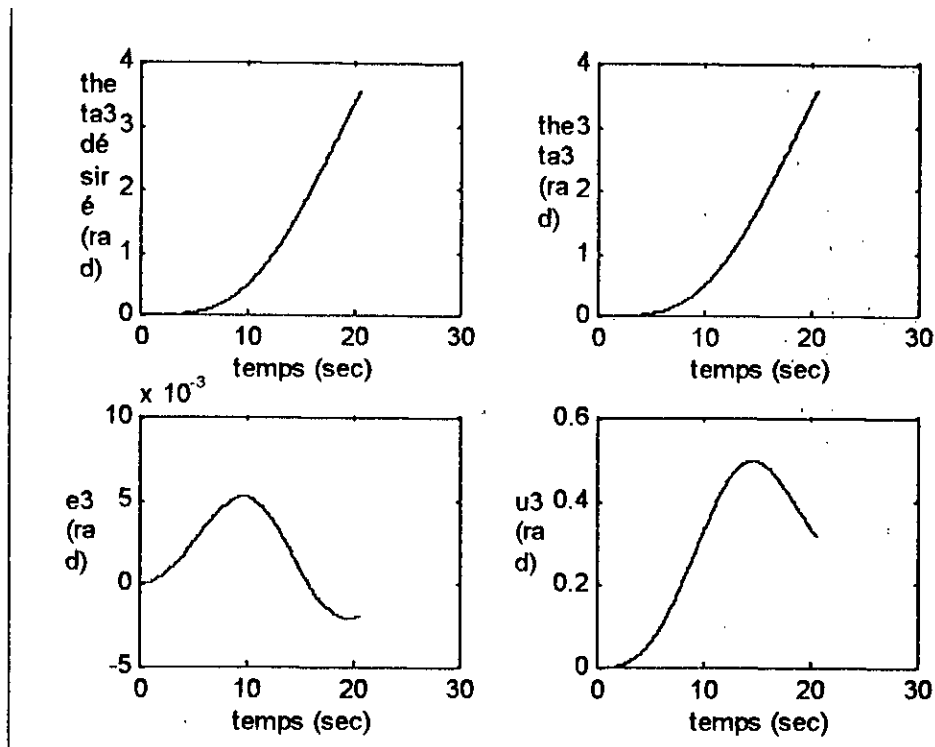


Figure 7 : Réponse du troisième segment du bras TRT.

ANALYSE ET INTERPRETATION

D'après la figure 5, on constate qu'il existe une erreur permanente de 0.2 rad pour le premier segment, ce qui présente une erreur assez importante. Par contre, le deuxième segment et le troisième poursuivent la trajectoire cycloïdale avec des erreurs inférieures à 0.012 rad et 0.005 rad respectivement, ceux qui présentent une bonne précision.

On peut conclure, qu'avec le contrôleur PID, cité ci-dessus, le bras TRT possède de bonnes performances.

ANNEXE 4

I. NORME TRIANGULAIRE :

Produit borné : $x \oplus y = \max(0, x+y-1)$

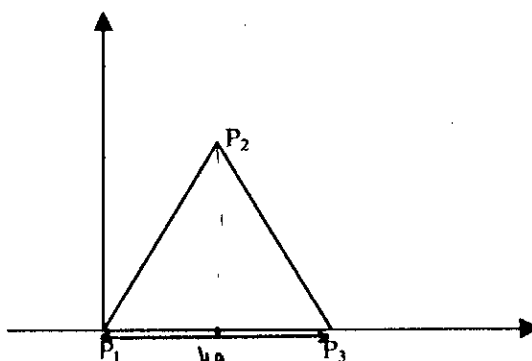
II. CONORME TRIANGULAIRE :

Somme bornée : $x \oplus y = \min(1, x+y)$

III. FORMES DES FONCTIONS D'APPARTENANCE :**1. La forme Triangulaire**

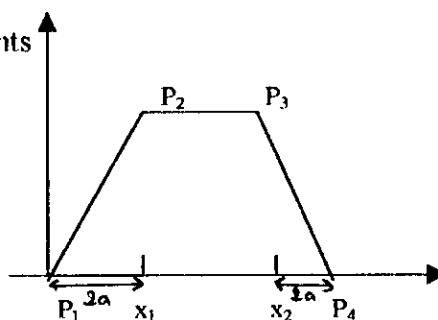
L'allure triangulaire est complètement définie par 3 points P_1, P_2, P_3 .

$$\mu_A(x) = \begin{cases} 1 - \frac{|x - x_1|}{2a} & x_1 - 2a \leq x \leq x_1 + 2a \\ 0 & \text{ailleurs} \end{cases}$$

**2. La forme Trapézoïdale**

Cette forme est définie par quatre points

P_1, P_2, P_3 et P_4 .

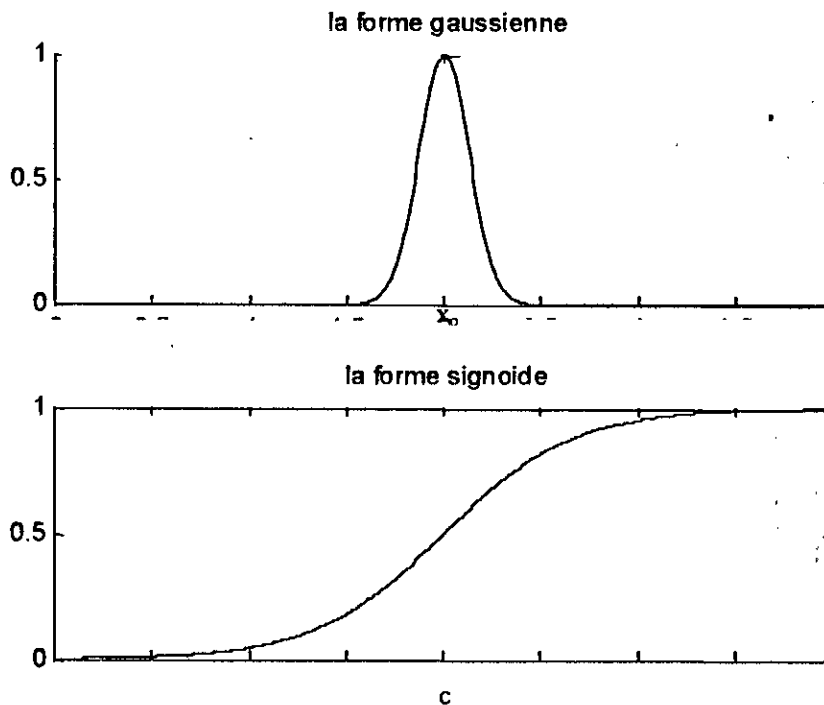


$$\mu_A(x) = \begin{cases} 1 - \frac{|x - x_1|}{2a} & x_1 - 2a \leq x \leq x_1 \\ 1 & x_1 \leq x \leq x_2 \\ 1 - \frac{|x - x_2|}{2a} & x_2 \leq x \leq x_2 + 2a \end{cases}$$

a: est un paramètre .

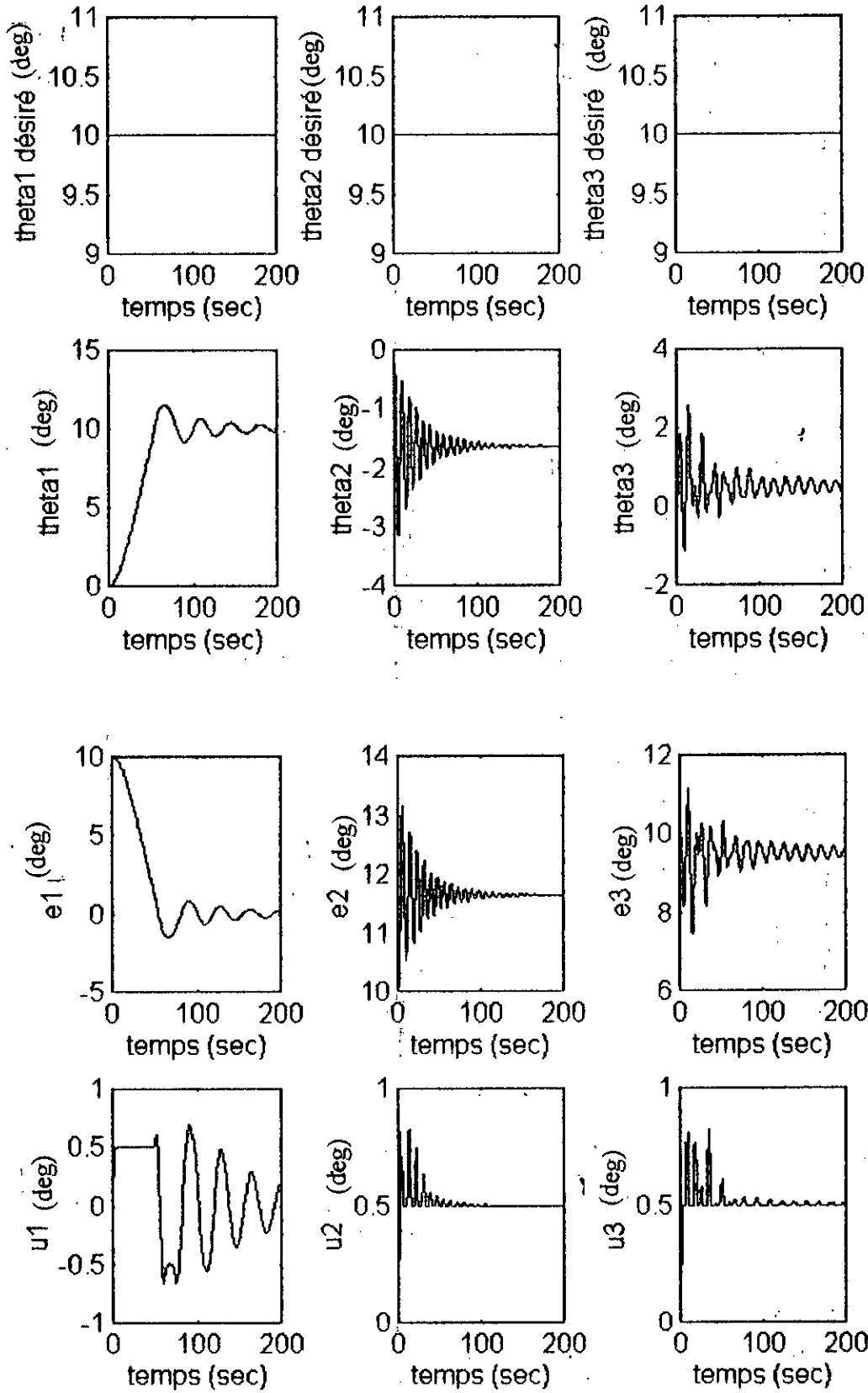
FORME GAUSSIENNE :

$$\mu_A(x) = \exp(- (x - x_0) / a)^2); \quad -\infty \leq x \leq +\infty$$



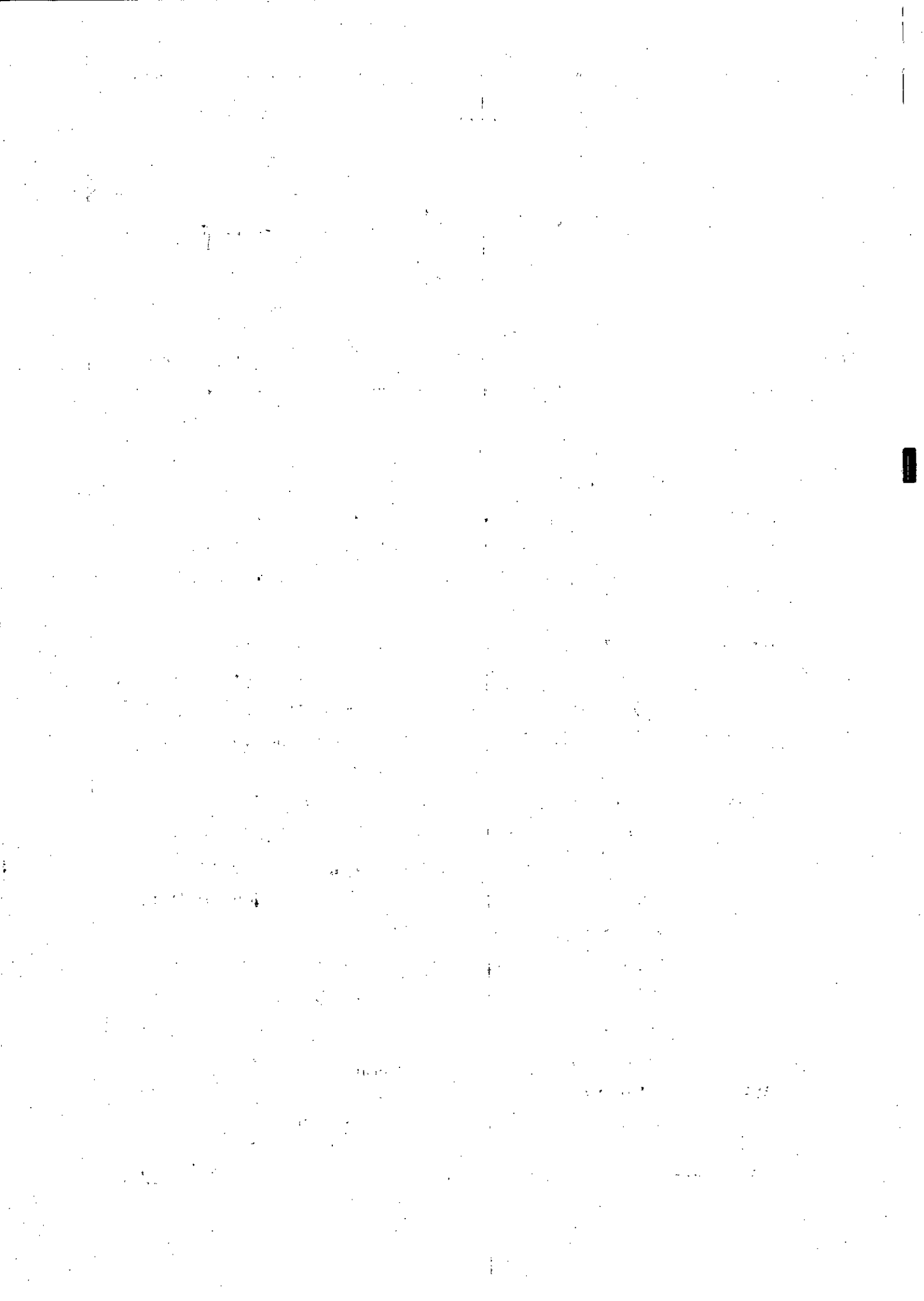
FORME SIGMOÏDE :

$$\mu_A(x) = \frac{1}{1 + e^{-a(x-c)}}$$



Commande floue du bras RRR avec Som-Prod 7 règles $u_1=u_2=10, u_3=10$

**REFERENCES
BIBLIOGRAPHIQUES**



BIBLIOGRAPHIE

- [ASC92] ASCH G., "Capteurs utilisés en instrumentation industrielle", Dunod, 1992.
- [ARA94] ARAGO 14, "logique floue", Observatoire Français Des Techniques Avancées, Masson, Paris, 1994.
- [BAL95] Bali.N., "Etude des performances de la commande prédictive généralisée : Application aux robots manipulateurs PUMA et SCARA", Thèse de magister , ENP, 1995.
- [BEN97] M.Benkrid, M.Mama, "Modélisation et Commandes De Bras Manipulateurs", Thèse d'ingénieur, ENP, 1997.
- [BOU93] Bouchon. Meunier, Logique floue PUF 1993.
- [BOU95] S. Boursas, « Modélisation et commande floue d'un bras de robot à une liaison flexible » Thèse d'ingénieur, institut d'électronique, U.S.T.H.B., Déc. 1995.
- [BOU97] S. Boulkhrachef S. et KADRI L. " Méthodologies de commande de la machine asynchrone incluant la logique floue", Thèse d'ingénieur, Département d'Electrotechnique, ENP., Juillet 1997.
- [BOV91] S.Boverie, B.Demaya, A.Titli, "Fuzzy logic control compared with other automatic control approaches", IEEE, 1991.
- [BUH94] H. Bühler, réglage par logique floue. PPR, 1994.
- [COI 95] COIFFET.C "Robots : définitions et classements", Technique de l'ingénieur (Traité Mesure et Contrôle), Doc R7700, Paris, 1995
- [CRA86] CRAIG.J, " Introduction to robotics : Mechanics and control", Edition Addison Wesley, 1986.
- [DJE96] N.Djeddi et D.Dribine, "Modélisation Cinématique Et Dynamique Des Bras Manipulateurs" , Thèse d'ingénieur, ENP, 1996.
- [DUB88] D.Dubois ,H.Prade et C.Testemale "Fuzzy Sets and Systems", Weightened fuzzy pattern matching , 28, 1988.
- [FU87]] FU.SK., GONZALES R.C, LEE C.S.G "Robots : Control, sensing, Vision and Intelligence" , Mac Graw Hill Company, 1987.

- [GUE95] L. Guenfaf, "Etude comparative de différentes stratégies de commande adaptative : Application à un robot manipulateur", Thèse de Magister, ENP, 1995. +
- [HEL93] D.H. Hellendorn et M. Reinfrank, An introduction to Fuzzy control. Springer verlag 1993.
- [HER96] R. Herrero et al., "A Highly nonlinear fuzzy control algorithm for servo systems positioning," in Proc. IFAC intelligent components and instruments for control Application, vol. 32, n° 1, pp. 57-65, Jan/Feb. 1996.
- [JAM93] M. Jamshidi, N. Vadiie, T.J. Rov, "Fuzzy Logic and Control"
[JAN.95] J.-S.R. Jang et C.T. Sun, "Neuro-fuzzy modeling and control," in Proc. IEEE, Vol. 83, n°3, mars, 1995, pp. 378-405.
- [KAU87] A. Kauffman, « Nouvelle logique pour l'intelligence artificielle. » Hermès, 1987.
- [KET92] R. Ketata. Méthodologie de régulation numérique incluant la logique floue. Thèse de doctorat, N° 199, INSA, Toulouse, 1992.
- [LALL 94] LALLEMAND J.P et ZEGHLOULS.S, " Robotique : Aspects fondamentaux, Modélisation mécanique, C.A.O robotique et commande", Edition MASSON, Paris 1994
- [LEE190] C.C. Lee, "Fuzzy logic in control system : Fuzzy logic controller-part I" IEEE Trans. Syst., Man. And cybern., vol. 20, pp. 404-418, Feb. 1990.
- [LEE290] C.C. Lee, "Fuzzy logic in control system : Fuzzy logic controller-part II" IEEE Trans. Syst., Man. And cybern., vol. 20, pp. 419-435, Feb. 1990.
- [MAM77] E.H Mamdani, " Application of fuzzy logic to approximate reasoning using linguistic synthesis", IEEE transactions on computers, vol. C.26, n° 12, pp. 1182-1191. 1977.
- [MIZ95] M. Mizumuto, "Realization of PID controls by fuzzy control methods ", ELSEVIER, 1995.
- [NED96] Nedjari et Boukari "Commande Adaptative décentralisée, Application en Robotique", Thèse d'ingénieur, Département d'Automatique, ENP, 1996.
- [PAU81] R.P. PAUL "Robots Manipulators : Mathematics, Programming and

- Control'', The MIT Press, 1981.
- [PED88] Pedtycz W. 'Fuzzy Control and Fuzzy Systems'', Department of Electrical Engineering, University of Manitoba, Winnipeg, Canada, R. S.P.Tauton, Sonerset, England, 1988.
- [ROC98] B.B.Rochdi 'Commande Adaptative Décentralisée des robots manipulateurs'', Thèse de Magistère, Département d'Automatique, ENP, Avril 1998.
- [SAL98] B.Salim 'Commande Adaptative par systèmes flous, Application à la commande des robots manipulateurs'', Thèse de Magistère, Département d'Automatique, ENP, Avril 1998.
- [SCH094] SCHOLTEN H., "logique floue et régulation PID'', Publitrionic, Paris, 1994.
- [SUG85] M.Sugeno et T.Takagi, , " Fuzzy identification of systems and its Applications to modeling and control, " IEEE Trans. Syst. Man and Cybernetics, vol. SMC-15, n°. 1 pp. 116-132, 1985.
- [TAK83] T.Takagi et M.Sugeno, "Derivation of fuzzy control rules from human operators control actions," in IFAC Fuzzy Information, Marseille, France, 1983.
- [TZA85] S.Tzafestas, "Incremental Fuzzy PID Control, » IEEE Trans. Industrial Electronics, vol 12, n°1 pp 116-132, 1985.
- [ZAD88] L.A Zadeh. " Fuzzy logic'', IEEE Computer, pp. 83-93, April 1988.

