

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

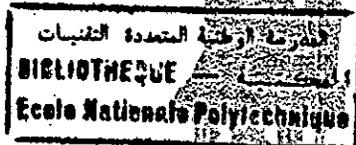
Handwritten signature



ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT DE GENIE ELECTRIQUE ET INFORMATIQUE

Filière : AUTOMATIQUE



Projet de Fin d'Etudes

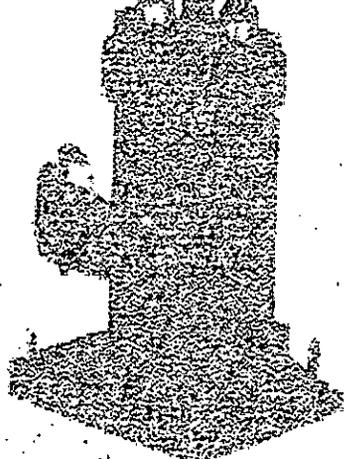
En vue de l'obtention du diplôme d'ingénieur d'état en Automatique

THEME

COMMANDES FLOUES ADAPTATIVES DIRECTES
APPLIQUÉES AU ROBOT À STRUCTURE CYLINDRIQUE

Proposé et dirigé par :
H.CHEKIREB

Etudié par :
Fatma-Zohra DOUDAH
Asmaâ-Wided LONGOU



Promotion : **JUIN 2001**

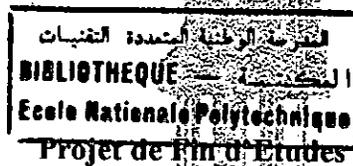
E.N.P 10 Avenue Hassen Badi El-Harrach ALGER

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



ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT DE GENIE ELECTRIQUE ET INFORMATIQUE
Filière : AUTOMATIQUE



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

THEME

**COMMANDES FLOUES ADAPTATIVES DIRECTES
APPLIQUÉES AU ROBOT À STRUCTURE CYLINDRIQUE**

Proposé et dirigé par :
H.CHEKIREB

Etudié par :
**Fatma-Zohra DOUDAH
Asmaâ-Wided LONGOU**

Promotion : **JUIN 2001**

E.N.P 10 Avenue Hassen Badi El-Harrach ALGER

Remerciements

*Il nous est particulièrement agréable de témoigner notre reconnaissance à Monsieur **H.CHEKIREB** chargé de cours à l'ENP, pour la disponibilité qu'il nous a accordé pour le suivi de ce travail et pour nous avoir fait profiter de ses précieux conseils.*

*Nos vifs remerciements s'adressent à Monsieur **B.HEMICI**, chargé de cours à l'ENP pour avoir accepté de présider ce jury.*

*Nous tenons à remercier également Monsieur **R.ILOUL**, chargé de cours à l'ENP pour nous avoir fait l'honneur de juger ce travail.*

Nos remerciements s'adressent également à tous les enseignants de la filière Automatique .

Dédicaces

المدرسة الوطنية المتعددة الفتيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

F-Zohra Leila

Je dédie ce travail à :

la mémoire de ma très chère grand mère

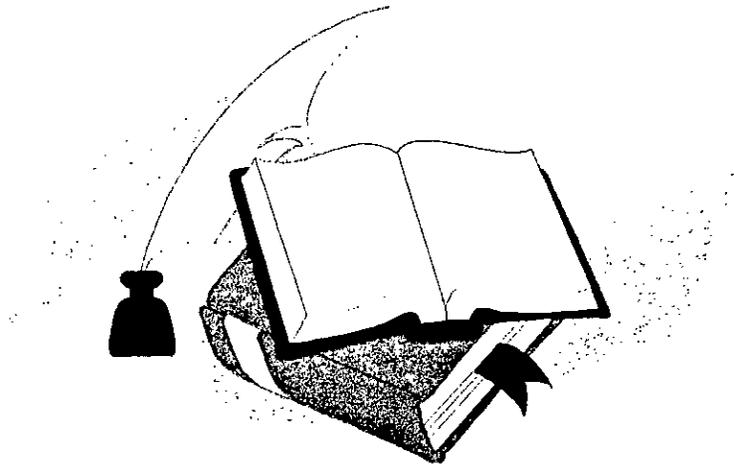
ma très chère mère

mes tantes

mes oncles

mes amis

et à tous ceux qui aiment la science



Asmaâ-Wided

*Je dédie ce travail à mes parents, les êtres qui me
sont les plus chers*

Chapitre I : SYSTEMES FLOUS	
I.1 Introduction	1
I.2 Description des systèmes flous	1
I.3 Modélisation d'un système par logique flou	4
I.3.1 Modélisation d'un système de Sugeno d'ordre 1	4
I.4 Remarques sur la logique flou	6
I.4.1 La logique flou et les appareils «grand public»	6
I.4.1.1 Application dans les produits bruns et les appareils photographiques	7
I.4.1.2 La logique flou dans les produits blancs	7
 Chapitre II : MODELISATION DU ROBOT	
II.1 Introduction	9
II.2 Modélisation cinématique	10
II.2.1 Introduction	10
II.2.2 Transformations élémentaires	10
II.2.3 Présentation du robot à structure cylindrique	12
II.2.3.1 Application de la méthode de DH sur le robot à structure cylindrique	13
II.3 Modélisation dynamique	15
II.3.1 Formalisme de Lagrange	15
II.3.2 Calcul du modèle dynamique du robot à structure cylindrique	16
 Chapitre III : COMMANDE FLOUE ADAPTATIVE DIRECTE DÉCENTRALISÉE BASÉE SUR LA METHODE DU COUPLE CALCULÉ	
III.1 Introduction générale	18
III.2 Commande floue adaptative directe décentralisée basée sur la méthode du couple Calculé	19
III.2.1 Introduction	19
III.2.2 Formulation du problème	20
III.2.3 Dynamique de l'erreur filtrée	22
III.2.4 Analyse de la stabilité	22
III.2.5 Résultats de la simulation	24
 Chapitre IV : COMMANDE FLOUE ADAPTATIVE DIRECTE LINÉARISANTE (TYPE 1)	
IV.1 Introduction	30
IV.2 Formulation du problème	30
IV.3 Dynamique de l'erreur	31
IV.4 Analyse de la stabilité	33
IV.5 Résultats de la simulation	35
 Chapitre V : COMMANDE FLOUE ADAPTATIVE DIRECTE LINÉARISANTE (TYPE 2)	
V.1 COMMANDE FLOUE ADAPTATIVE DIRECTE DÉCENTRALISÉE LINÉARISANTE	42
V.1.1 Introduction	42
V.1.2 Formulation du problème	42
V.1.3 Adaptation par logique floue	44
V.1.4 Dynamique de l'erreur	44
V.1.5 Analyse de la stabilité	46

V.1.6 Résultats de la simulation	47
V.2 COMMANDE FLOUE ADAPTATIVE DIRECTE DÉCENTRALISÉE LINÉARISANTE AVEC ADAPTATION DES FONCTIONS D'APPARTENANCES	
V.2.1 Introduction	54
V.2.2 Position du problème	54
V.2.3 Approximation par un système flou	55
V.2.4 Dynamique de l'erreur	56
V.2.5 Analyse de la stabilité	58
V.2.6 Résultats de la simulation	59
CONCLUSION GÉNÉRALE	68
ANNEXE 1	70
ANNEXE 2	76
ANNEXE 3	88
ANNEXE 4	91
BIBLIOGRAPHIE	

INTRODUCTION GÉNÉRALE

المدرسة الوطنية المتعددة التخصصات
 المكتبة — BIBLIOTHEQUE
 Ecole Nationale Polytechnique

La robotique est un domaine très vaste dans ses techniques et l'évolution de ses technologies, beaucoup utilisée dans l'industrie car les robots assurent une bonne productivité ; une bonne qualité du produit et une quantité en un temps record.

Les robots sont très utilisés dans les milieux hostiles où l'homme ne peut pas intervenir pour cause de danger ou l'incapacité. Pour ces raisons, nous nous sommes intéressés à la commande des robots car pour pouvoir les manipuler (leur faire suivre des trajectoires pour des tâches précises), il faut les commander.

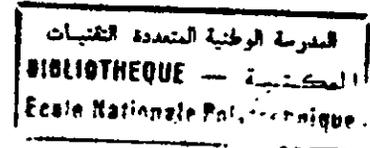
Le robot pour lequel on a synthétisé nos commandes est un robot à structure cylindrique, il présente trois articulations : une rotation autour de l'axe vertical, une translation suivant le même axe et une translation suivant l'axe horizontal.

Nous nous sommes intéressés à la commande floue adaptative directe, en effet l'intérêt de telles commandes est de conserver les mêmes performances de commande, malgré les variations qui peuvent subvenir sur le système à commander, de plus ces commandes sont réalisables dans des domaines admissibles.

Les commandes que nous présentons sont : en premier lieu, la commande floue adaptative directe décentralisée basée sur la méthode du couple calculé, elle exploite le modèle directe du robot, dans l'expression de la commande, la partie non-linéaire de l'écriture décentralisée de l'équation du mouvement est approximée par un système flou de Sugeno d'ordre zéro. Les deux commandes suivantes sont linéarisantes : la première est de type1, se base sur le modèle directe, les fonctions floues de son expression de commande sont de Sugeno d'ordre un, la deuxième est de type2 et se base sur le modèle inverse et les fonctions floues sont de Sugeno d'ordre zéro. Ces commandes reposent sur la table des règles floues qui contient la combinaison des ensembles flous (de type Gaussienne), donc pour trois ensembles flous pour chacune des deux variables d'entrée du système flou (position et vitesse), on aboutit à neuf règles. Dans le but de réduire la taille de la base des règles floues, on a été amené à utiliser une adaptation des centres et des variances des ensembles flous et cela au niveau de la dernière commande qui est la commande floue adaptative directe décentralisée linéarisante avec adaptation des fonctions d'appartenance, nous sommes arrivés au résultat important qui est de réduire énormément la taille de l'algorithme, on a pris une règle pour la première articulation, une règle pour la deuxième articulation et trois règles pour la troisième articulation.

Chapitre I

SYSTEMES FLOUS



I.1 Introduction :

La logique floue est un concept d'appréhension des systèmes complexes jusqu'alors mal perçus par les méthodes d'analyse traditionnelles. En effet, le principe de fonctionnement de tout opérateur logique, consiste à interpréter une information, en lui affectant les valeurs 0 ou 1 selon qu'elle appartienne ou non à un ensemble de considérations données. Cet ensemble est supposé connu, défini, et invariable. En réalité, les choses sont loin d'être aussi strictes, et le cerveau humain est là pour nous rappeler qu'une information peut être interprétée différemment selon le contexte dans lequel elle survient. L'objectif de la logique floue sera donc de se rapprocher le plus possible de l'analyse qu'aurait effectuée le cerveau dans des conditions similaires. Par exemple, la logique floue doit être capable d'augmenter automatiquement le volume d'un autoradio, en fonction de bruits parasites de toutes natures (moteur, vitesse, état de la route, vent), et de ne pas agir dans le cas d'une conversation entre passagers du véhicule [1].

I.2 Description des systèmes flous :

La figure 1 montre la configuration de base des systèmes logiques flous. Le système logique flou représente une application de $U \subset \mathfrak{R}^n$ vers \mathfrak{R} où U l'univers de discours est tel que $U = U_1 \times \dots \times U_n$ avec $U_i \subset \mathfrak{R}, i = 1 \dots n$. Un système flou est en général constitué par les quatre éléments de base suivants (figure 1.1) [12] :

- Base des règles floues
- Moteur d'inférence flou
- Fuzzificateur
- Défuzzificateur

Dans ce qui suit, nous donnons la fonction descriptive de chacun de ces éléments. Avant cela, nous rappelons la définition d'un ensemble flou A défini sur l'univers de discours U . L'ensemble flou A de l'univers de discours U est caractérisé par une fonction d'appartenance $\mu_A : U \rightarrow [0, 1]$ qui associe à chaque élément x de U une valeur $\mu_A(x)$ dans l'intervalle $[0, 1]$ et représentant le degré d'appartenance de x dans A .

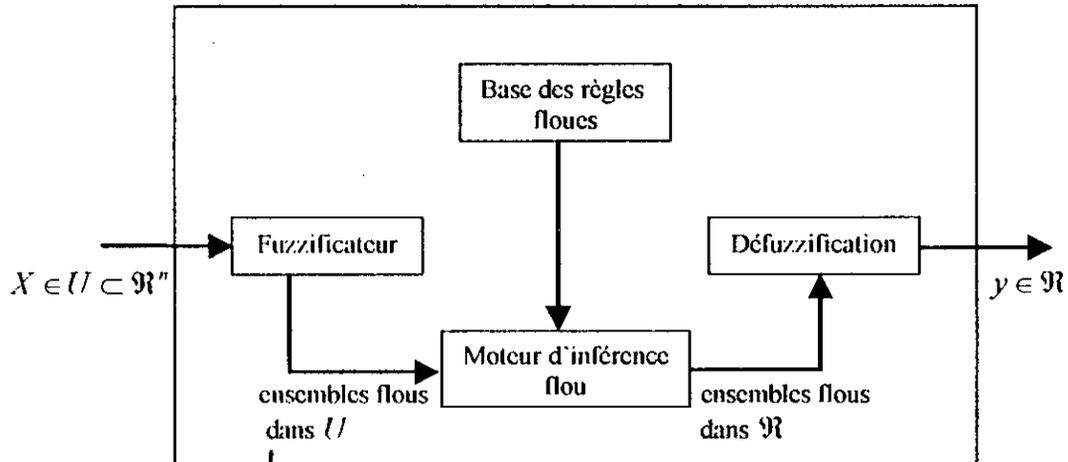


Figure 1.1 : Configuration de base des systèmes logiques flous

La base des règles flous rassemble l'ensemble des règles flous de la forme *Si- Alors* décrivant en termes linguistiques le comportement du système.

$$\begin{array}{l}
 R_l : \text{Si } x_1 \text{ est } A_1^l \text{ et } \dots \text{ et } x_n \text{ est } A_n^l ; \\
 \quad \quad \quad \text{Alors } y_l \text{ est } B^l .
 \end{array} \tag{1.1}$$

où $X = [x_1 \dots x_n]^T \in U$ et $y \in \mathfrak{R}$ sont respectivement les entrées et les sorties du système flou, A_i^l et B^l sont respectivement les ensemble flous de U et \mathfrak{R} et $l = 1 \dots m$.

Chaque règle floue *Si- Alors* de (1.1) définit une implication floue $A_1^l \times \dots \times A_n^l \rightarrow B^l$, qui est un ensemble défini dans l'espace produit $U \times \mathfrak{R}$.

La règle d'implication floue est définie par l'opération produit telle que :

$$\mu_{A_1^l \dots A_n^l \rightarrow B^l}(X, y) = \mu_{A_1^l \dots A_n^l}(X) \mu_{B^l}(y) \tag{1.2}$$

où $\mu_{A_1^l \dots A_n^l}(X)$ est donné par :

$$\mu_{A_1^l \dots A_n^l}(X) = \mu_{A_1^l}(x_1) * \dots * \mu_{A_n^l}(x_n) \tag{1.3}$$

Le symbole $*$ dénote la t-norme, qui traduit la conjonction « et » dans (1.1), laquelle est définie par l'opération suivante :

$$u * v = u \cdot v \quad \text{produit algébrique} \tag{1.4}$$

Le *moteur d'inférence flou* permet à partir d'un fait observé de la base des règles floues de fournir une décision en exploitant le raisonnement approximatif.

Soit A_x un ensemble flou arbitraire dans U constituant le fait observé, alors chaque R_l de la base de connaissance détermine un ensemble flou $A_x \circ R_l$ dans \mathfrak{R} basé sur la règle compositionnelle *sup-star* d'inférence suivante :

$$\mu_{A_x \circ R_l}(y) = \sup_{X \in U} [\mu_{A_x}(X) * \mu_{A'_1, \dots, A'_n \rightarrow B'}(X, y)] \quad (1.5)$$

où $*$ est la t-norme (1.4) et $\mu_{A'_1, \dots, A'_n \rightarrow B'}(X, y)$ est déterminée par la règle d'implication floue de (1.2). L'ensemble flou final $A_x \circ (R_1, \dots, R_m)$ déterminé par les m règles est obtenue par disjonction de $\mu_{A_x \circ R_l}(y)$ pour $l = 1 \dots m$:

$$\mu_{A_x \circ (R_1, \dots, R_m)}(y) = \mu_{A_x \circ R_1}(y) \dot{+} \dots \dot{+} \mu_{A_x \circ R_m}(y) \quad (1.6)$$

où $\dot{+}$ dénote la t-conorme définie par :

$$u \dot{+} v = \max(u, v) \quad \text{union floue} \quad (1.7)$$

En utilisant l'opération produit (1.2) et prenant (1.3), alors l'inférence est appelée inférence produit. L'équation (1.5) devient ainsi :

$$\mu_{A_x \circ R_l}(y) = \sup_{X \in U} [\mu_{A_x}(X) \mu_{A'_1}(x_1) \dots \mu_{A'_n}(x_n) \mu_{B'}(y)] \quad (1.8)$$

Le *fuzzificateur* schématise le vecteur $X = [x_1 \dots x_n]^T \in U$ en un ensemble flou A_x dans U . A_x est un singleton flou c'est à dire :

$$\mu_{A_x}(X') = \begin{cases} 1 & \text{si } X' = X \\ 0 & \text{si } X' \neq X \end{cases} \quad X' \in U \quad (1.9)$$

Alors nous appelons ce singleton fuzzificateur.

Le *défuzzificateur* schématise les ensembles flous dans \mathfrak{R} en un point dans \mathfrak{R} . La sortie défuzzifiée y est définie par :

$$y(X) = \frac{\sum_{l=1}^m \bar{y}_l \left(\prod_{i=1}^n \mu_{A'_i}(x_i) \right)}{\sum_{l=1}^m \left(\prod_{i=1}^n \mu_{A'_i}(x_i) \right)} \quad (1.10)$$

où \bar{y}_i est le point auquel μ_{A_i} atteint sa valeur maximale ou encore l'abscisse du centre de gravité.

I.3 Modélisation d'un système par logique floue :

On s'intéresse à la manière avec laquelle on peut représenter n'importe quel système en utilisant le raisonnement flou.

I.3.1 Modélisation d'un système de Sugeno d'ordre 1 :

Dans cette modélisation, la sortie s'écrit sous forme d'un polynôme de degrés un en fonction des entrées, ce qui donne une valeur bien précise de la sortie .

En prenant le vecteur d'état $X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ et les m règles :

$$R_1 : \text{Si } x_1 \text{ est } A_1^1 \text{ et } x_2 \text{ est } A_2^1 \dots \text{ et } x_n \text{ est } A_n^1 \text{ alors } y_1 = \sum_{i=1}^n a_i^1 x_i$$

$$R_2 : \text{Si } x_1 \text{ est } A_1^2 \text{ et } x_2 \text{ est } A_2^2 \dots \text{ et } x_n \text{ est } A_n^2 \text{ alors } y_2 = \sum_{i=1}^n a_i^2 x_i$$

⋮

$$R_m : \text{Si } x_1 \text{ est } A_1^m \text{ et } x_2 \text{ est } A_2^m \dots \text{ et } x_n \text{ est } A_n^m \text{ alors } y_m = \sum_{i=1}^n a_i^m x_i$$

On définit :

$$\alpha_j = \prod_{i=1}^n \mu_{A_i^j}(x_i)$$

$$y = \frac{\alpha_1 y_1 + \alpha_2 y_2 + \dots + \alpha_m y_m}{\alpha_1 + \alpha_2 + \dots + \alpha_m}$$

$$y = \frac{\sum_{j=1}^m \alpha_j y_j}{\sum_{j=1}^m \alpha_j} = \frac{\sum_{j=1}^m \alpha_j \left(\sum_{i=1}^n a_i^j x_i \right)}{\sum_{j=1}^m \alpha_j}$$

$$y = \left(\begin{array}{l} \frac{\alpha_1 x_1}{\sum_{j=1}^m \alpha_j} a_1^1 + \frac{\alpha_2 x_1}{\sum_{j=1}^m \alpha_j} a_1^2 + \dots + \frac{\alpha_m x_1}{\sum_{j=1}^m \alpha_j} a_1^m + \frac{\alpha_1 x_2}{\sum_{j=1}^m \alpha_j} a_2^1 + \frac{\alpha_2 x_2}{\sum_{j=1}^m \alpha_j} a_2^2 + \dots + \frac{\alpha_m x_2}{\sum_{j=1}^m \alpha_j} a_2^m \\ \dots \\ \frac{\alpha_1 x_n}{\sum_{j=1}^m \alpha_j} a_n^1 + \frac{\alpha_2 x_n}{\sum_{j=1}^m \alpha_j} a_n^2 + \dots + \frac{\alpha_m x_n}{\sum_{j=1}^m \alpha_j} a_n^m \end{array} \right)$$

Ou encore d'une manière plus compacte :

$$y = W^T \cdot \theta \quad \text{tels que :}$$

$$W^T = \left[\begin{array}{cc|cc| \dots | cc} \frac{\alpha_1 x_1}{\sum_{j=1}^m \alpha_j} & \frac{\alpha_m x_1}{\sum_{j=1}^m \alpha_j} & \frac{\alpha_1 x_2}{\sum_{j=1}^m \alpha_j} & \frac{\alpha_m x_2}{\sum_{j=1}^m \alpha_j} & \dots & \frac{\alpha_1 x_n}{\sum_{j=1}^m \alpha_j} & \frac{\alpha_m x_n}{\sum_{j=1}^m \alpha_j} \end{array} \right]$$

$$\theta^T = [a_1^1 \ a_1^2 \ \dots \ a_1^m \quad a_2^1 \ a_2^2 \ \dots \ a_2^m \quad \dots \quad a_n^1 \ a_n^2 \ \dots \ a_n^m]$$

Cas particulier :

Modélisation d'un système de Sugeno d'ordre 0 :

En prenant le vecteur d'état $X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ et les m règles :

R_1 : Si x_1 est A_1^1 et x_2 est A_2^1 et x_n est A_n^1 alors $y_1 = a_1$

R_2 : Si x_1 est A_1^2 et x_2 est A_2^2 et x_n est A_n^2 alors $y_2 = a_2$

⋮

R_m : Si x_1 est A_1^m et x_2 est A_2^m et x_n est A_n^m alors $y_m = a_m$

On définit :

$$\alpha_j = \prod_{i=1}^n \mu_{A_i^j}(x_i)$$

$$y = \frac{\alpha_1 y_1 + \alpha_2 y_2 + \dots + \alpha_m y_m}{\alpha_1 + \alpha_2 + \dots + \alpha_m}$$

$$y = \frac{\sum_{j=1}^m \alpha_j y_j}{\sum_{j=1}^m \alpha_j} = \frac{\sum_{j=1}^m \alpha_j a_j}{\sum_{j=1}^m \alpha_j}$$

$$y = \left(\frac{\alpha_1 a_1}{\sum_{j=1}^m \alpha_j} + \frac{\alpha_2 a_2}{\sum_{j=1}^m \alpha_j} + \dots + \frac{\alpha_m a_m}{\sum_{j=1}^m \alpha_j} \right)$$

Ou d'une façon plus compacte :

$$y = W^T \cdot \theta \quad \text{tels que :}$$

$$W^T = \left[\frac{\alpha_1}{\sum_{j=1}^m \alpha_j} \quad \frac{\alpha_2}{\sum_{j=1}^m \alpha_j} \quad \dots \quad \frac{\alpha_m}{\sum_{j=1}^m \alpha_j} \right] \quad \text{et } \theta^T = [a_1 \quad a_2 \quad \dots \quad a_m]$$

I.4 Remarques sur la logique floue :

La logique floue traverse (en 1994) encore, à l'évidence, une phase de développement, mais elle a dépassé depuis longtemps le stade du laboratoire pour un bon nombre d'applications.

Dans cette optique, voici quelques remarques sur la logique floue et ses applications [9].

I.4.1 La logique floue et les appareils « grand public » :

Les systèmes logiques flous ont trouvé un large domaine d'application dans les appareils « grand public ». En général, on constate que les appareils qui utilisent la logique floue offrent plus de fonctions, sont plus faciles et plus simples à utiliser, fonctionnent mieux et consomment moins d'énergie que les appareils disponibles avant l'introduction de la logique floue. Ces applications ont été rendues possibles par le matériel spécialisé qui a été développé récemment au Japon.

Il est important dans ce genre de production de pouvoir satisfaire autant de clients que possible avec une seule chaîne de fabrication. IL semble que la logique floue soit seule à permettre d'adapter une fonction à la fois rapidement et simplement.

Il est intéressant de voir comment la logique floue est appliquée dans des appareils « grand public », à titre d'illustration des multiples possibilités d'application de la théorie. Il existe aussi des applications de la logique floue à la reconnaissance de formes, surtout dans les appareils vidéo.

1.4.1.1 Applications dans les produits bruns et les appareils photographiques :

Les applications de la logique floue sont devenues courantes, sur le marché ouest-européen, dans les caméscopes et les appareils photographiques.

Le choix des Japonais est judicieux, car l'achat de ce genre d'appareil obéit souvent à des motivations particulières : on ne veut acheter que le produit le plus récent parmi les plus récents.

On achète donc de préférence un appareil qui incorpore une technique nouvelle. L'offre de produits bruns sur le marché européen est encore limitée, car le succès de la technique est tel au Japon que toute la production est écoulee sur place.

- **La télévision :**

La régulation en logique floue est utilisée dans les téléviseurs entre autres pour exécuter automatiquement la correction des couleurs, pour adapter la luminosité de l'image à l'éclairage ambiant, pour adapter le volume sonore à l'éloignement du spectateur.

- **Les caméscopes (caméra et enregistreurs combinés):**

L'application de la logique floue aux caméscopes est en partie une application de la reconnaissance de formes. De ce fait elle est distincte de la régulation en logique floue, elle montre qu'il y a beaucoup d'autres applications de la logique floue que la simple régulation.

Ces applications, en particulier la suppression du bougé, sont du même niveau technique que l'espionnage par satellite ou la prévention des incendies.

La stabilisation électronique de l'image, telle qu'elle est réalisée dans les caméscopes de *Panasonic*, est une autre application importante de la logique floue. Elle détermine le vecteur de déplacement de l'image au moyen de 120 points de l'image. Si l'image se déplace à cause de tremblements de la main, le caméscope le sait ; il peut faire la différence entre les mouvements du sujet, les déplacements voulus par l'utilisateur et les tremblements de la main.

- **Les appareils photographiques :**

L'application aux appareils photographiques correspond en grande partie aux caméscopes ; il s'agit aussi de la mise au point et l'exposition.

1.4.1.2 La logique floue dans les produits blancs :

La logique floue est utilisée avec succès dans bon nombre de produits blancs. Elle peut être considérée sous différents aspects.

Ce qui saute aux yeux d'abord, c'est la facilité d'emploi de l'appareil. Il s'y ajoute souvent la disponibilité d'un nombre de modes de fonctionnement plus grand qu'il ne serait possible d'en gérer sans la logique floue.

Les économies d'énergie permises sont un avantage important dans beaucoup de cas. Les quelques exemples décrits ne sont qu'une partie du nombre croissant des applications.

- **La machine à laver :**

La *Aisaigo Day Fuzzy Washing Machine* utilise comme donnée d'entrée la quantité de linge, la qualité de saleté (déterminé par l'opacité de l'eau de rinçage) et la nature de la saleté (poudre, soluble) pour choisir parmi 600 modes de fonctionnement possibles. Pour chacun de ces modes, elle règle la quantité de détergent, la quantité d'eau, la durée du lavage et la durée du cycle.

La version la plus récente, la *Aisaigo Neuro-Fuzzy Washing Machine*, tient compte en plus de la nature du textile, de la dureté de l'eau, de la vitesse de dissolution du détergent et de la préférence indiquée par l'utilisateur pour un mode de travail. Cette machine choisit un mode de fonctionnement parmi 3800.

- **L'aspirateur :**

L'aspirateur *Canister Neuro-Fuzzy* de *Matsushita* connaît comme grandeurs d'entrée entre autres : la quantité de poussière, le type de revêtement de sol et le type de saleté.

La sortie s'applique à la puissance d'aspiration et à la vitesse de la brosse rotative.

L'aspiration signale le mouvement venu qu'il ne voit plus de poussière et indique même quel était le degré de saleté de la surface avant le nettoyage.

- **Le four à micro-ondes :**

Le four *Dimension 4* de *Panasonic* fait un usage astucieux d'un capteur d'humidité. Il présente une application typique de la régulation en logique floue. Il faut remarquer à ce sujet qu'il ne s'agit pas d'une application de régulation au sens du maintien d'une valeur de consigne dans une boucle fermée. Dans ce cas, c'est le déroulement d'un processus qui est déterminé par la logique floue.

Chapitre II

MODELISATION DU ROBOT

II.1 Introduction :

Un robot est une machine qui agit sur son environnement physique pour exécuter une tâche, atteindre un objectif sous certaines conditions. Il se différencie des machines automatiques rigides qui ne peuvent modifier le cycle de leurs opérations qu'elles répètent indéfiniment. Mais il reste automatique car il peut agir sans l'aide d'un être humain.

Le terme « robot » est une forme de *robota* qui signifie « travail pénible », « corvée », mot par lequel le dramaturge tchèque Karel Capek désignait dans les robots universels de Rossum (1920) des machines androïdes capables de réaliser tout travail normalement effectué par un être humain. Si le nouveau terme fit le bonheur des auteurs de science-fiction, ce n'est qu'au début des années 1960 qu'est réalisé le premier robot industriel, par unimation, pour la firme General Motors.

Quelque soit son domaine, le robot actuel n'a rien d'un androïde, si ce n'est que l'on recourt toujours à la terminologie de l'anatomie humaine pour décrire ses éléments de structures : bras, coude, avant-bras, poignet, main ; il est vraie qu'il semble difficile de faire mieux que la nature en ce qui concerne la mobilité, le positionnement, la préhension avec délicatesse ou force.

Donc, le robot industriel est l'image de notre corps en ce qu'il évolue dans un espace à 3 dimensions, dans un volume donné, et qu'il interagit avec son environnement, notamment grâce aux informations fournies par des capteurs.

La J.I.R.A (Japon Industrial Robot industry Association) décrit le robot comme étant un système versatile doté d'une mémoire et pouvant effectuer des mouvements comme ceux d'un opérateur humain.

La synthèse de la commande du robot nécessite la connaissance de son modèle dynamique. L'ensemble de ces équations constitue la modèle mathématique du robot. Si les équations sont extraites de la physique, le modèle est appelé *modèle de connaissance*, et si ces équations découlent des observations disponibles sur le système, le modèle s'appelle *modèle de représentation*.

Dans ce qui suit, nous donnons le formalisme de Lagrange pour l'établissement du modèle dynamique d'un robot à n degrés de liberté. Par la suite, nous appliquons cette méthode au cas d'un robot à structure cylindrique.

II.2 Modélisation cinématique :

II.2.1 Introduction :

L'analyse cinématique d'un système consiste à déterminer la position, la vitesse et l'accélération à tout instant l'évolution des mécanismes qui le composent. L'ensemble des paramètres qui définissent le système à un instant donné constitue son « état ». L'effet des couples de forces qui agissent sur le système, caractérisé par ses masses et inerties est plutôt pris en compte par l'étude dynamique du système [4].

En robotique, l'analyse cinématique détermine l'état de l'outil terminal en fonction de celui des articulations, des segments qui constituent le bras, ainsi que les dérivées qui caractérisent l'évolution du bras dans le temps.

La définition de vecteurs cinématiques est conservée par les opérations mathématiques relatives aux systèmes de coordonnées lorsqu'on passe d'un référentiel à un autre. Ces opérations (transformations) conservent en particulier le module des vecteurs. Denavit et Hartenberg ont développé une méthode de transformations de coordonnées pour résoudre les problèmes de chaînes cinématiques.

Les matrices de transformations définissent les relations géométriques qui existent entre deux systèmes de coordonnées consécutifs R_i et R_{i+1} , en fonction de la variable d'articulation $q_i = \theta_i$ s'il s'agit d'un axe de rotation, $q_i = d_i$ s'il s'agit d'un axe de translation.

Dans ce qui suit, la description de la rotation et de la translation des repères est bien détaillée dans l'ANNEXE 1.

II.2.2 Transformations élémentaires:

Denavit-Hartenberg ont établi une convention pour définir un repère R_{i+1} par rapport à un autre repère R_i en utilisant quatre paramètres θ, d, a et α selon le schéma de la figure 2.1 :

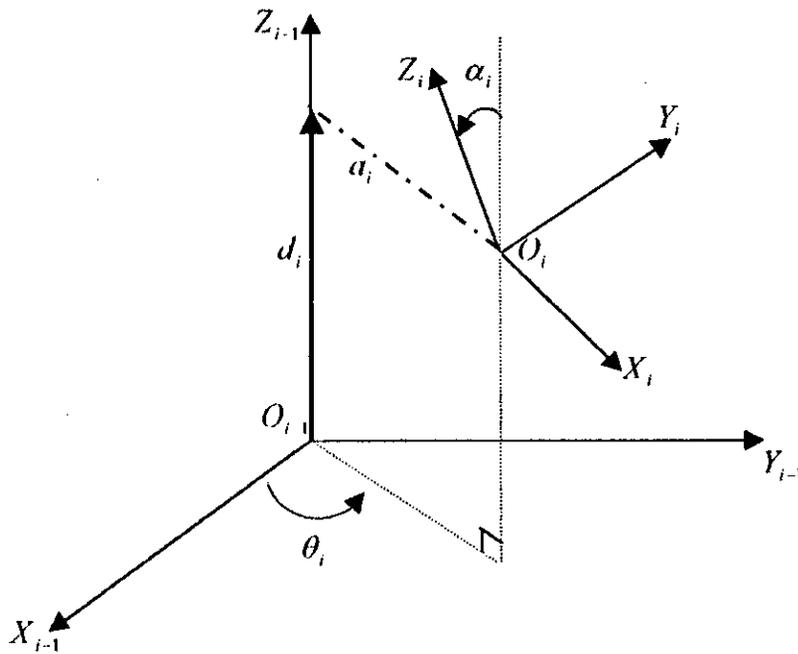


Figure 2.1 : Transformation de coordonnées entre deux système en rotation et en translation selon Denavit-Hartenberg

Le passage du repère $O_{i-1}, X_{i-1}, Y_{i-1}, Z_{i-1}$ vers O_i, X_i, Y_i, Z_i se fait par les opérations successives suivantes :

Par rotation de θ_i autour de l'axe Z_{i-1} suivi d'une translation de d_i suivant Z_{i-1} , puis une translation de l'origine du repère R_i de a_i suivant l'axe X_i , et à la fin rotation d'un angle α_i autour de X_i .

$$R_{Oz}(\theta_i) \cdot T_{Roz}(d_i) \cdot T_{Rox}(a_i) \cdot R_{Ox}(\alpha_i)$$

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

Cette transformation qui permet le passage du repère R_{i-1} au repère R_i est celle de Denavit-Hartenberg DH :

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{T_{i-1}^i} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

Ou encore :

$$R_{i-1} = T_{i-1}^i \cdot R_i$$

II.2.3 Présentation du robot à structure cylindrique :

Le bâti d'un robot cylindrique est constitué d'un bras horizontal supporté par une colonne verticale, elle-même montée sur un support rotatif, comme le montre la figure 2.2 [7]. Le bras horizontal peut s'éloigner et se rapprocher de la colonne, le support de bras peut monter et descendre le long de la colonne, et l'ensemble peut effectuer des rotation par rapport au socle support [4].

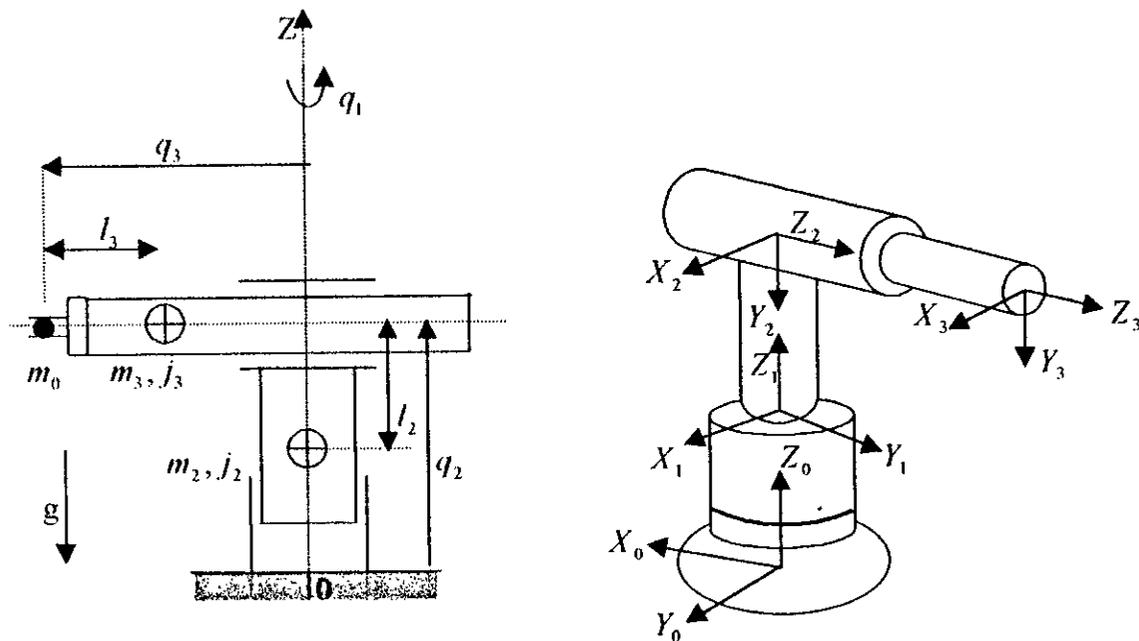


Figure 2.2 : Robot de classe 4 à structure cylindrique

Ce robot est de classe 4, son architecture est la plus rencontrée en robotique. Cette simple structure représente 45% des systèmes industriels qui simplement être actuellement utilisés.

Un robot de classe 4 est selon la J.I.R.A ; un robot copieur : ce type de robot est capable de reproduire des mouvements qui ont été exécutés sous contrôle d'un opérateur humain, et qui ont été enregistrés dans une mémoire [11].

II.2.3.1 Application de la méthode DH sur le robot à structure cylindrique:

Ce robot réalise une rotation de l'ensemble autour de l'axe vertical suivi d'un mouvement d'évolution verticale et une translation horizontale, dont les mouvements sont identifiés par les variables q_1 , q_2 et q_3 , comme le montre le schéma de la figure 2.3.

Le passage du repère R_0 vers le repère R_1 se fait par rotation d'un angle θ_1 autour de l'axe Z_0 . Pour arriver à R_2 à partir de R_1 , il faut passer par R'_1 qui s'obtient en tradant R_1 de d_2 suivant l'axe Z_1 , le repère R_2 est obtenu par rotation de R'_1 d'un angle $-\frac{\pi}{2}$ autour de l'axe X'_1 . Le passage de R_2 vers R_3 se réalise par translation de l'origine du repère R_2 de d_3 suivant l'axe Z_2 .

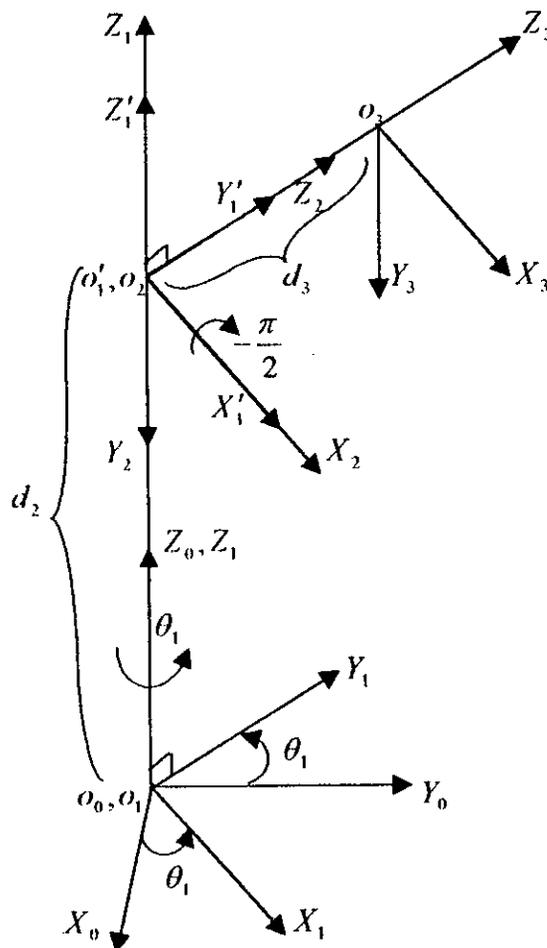


Figure 2.3 : Représentation de coordonnées généralisées du robot à structure cylindrique

Les coordonnées généralisées de ce robot sont notées par θ_1 , d_2 et d_3 caractérisant respectivement une rotation et deux translations. Le vecteur des coordonnées généralisées sera donc :

$$q = [\theta_1 \quad d_2 \quad d_3]^T$$

Le tableau suivant donne les informations géométriques de ce robot [7] :

N° de la liaison	Variable	θ_i	d_i	a_i	α_i
1	q_1	θ_1	0	0	0
2	q_2	0	d_2	0	-90°
3	q_3	0	d_3	0	0

Tableau 2.1: Information géométrique du robot à structure cylindrique

Les matrices de transformations DH appropriées à ce modèle sont :

$$T_0^1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_1^2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\left(\frac{\pi}{2}\right) & -\sin\left(-\frac{\pi}{2}\right) & 0 \\ 0 & \sin\left(-\frac{\pi}{2}\right) & \cos\left(-\frac{\pi}{2}\right) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_1^2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Notons que :

$$c_i = \cos\theta_i \quad \text{et} \quad s_i = \sin\theta_i.$$

Les matrices de transformations de DH deviennent ainsi :

$$T_0^2 = T_0^1 \cdot T_1^2 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_0^3 = T_0^2 \cdot T_2^3 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_0^3 = \begin{bmatrix} c_1 & 0 & -s_1 & -d_3 s_1 \\ s_1 & 0 & c_1 & d_3 c_1 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

II.3 Modélisation dynamique :

II.3.1 Formalisme de Lagrange :

Le formalisme de Lagrange est utilisé pour modéliser le comportement dynamique d'un robot, cette approche particulière est assez simple à mettre en œuvre et elle est bien adaptée aux techniques de calcul manuel ainsi qu'aux méthodes de calcul assistées par ordinateur.

Ce formalisme est utilisé pour décrire les équations du mouvement de plusieurs corps articulés les uns par rapport aux autres, les équations de Lagrange opèrent à partir de l'énergie cinétique et l'énergie potentielle d'un système, ce Lagrangien est défini par [11] :

$$L = E_c - E_p$$

Où E_c représente l'énergie cinétique et E_p l'énergie potentielle.

Dans la plupart des systèmes mécaniques, les frottements jouent un rôle important, on doit donc rajouter le terme qui les représente (énergie de dissipation : E_D) au niveau de l'équation de Lagrange et de ce fait, l'équation dynamique du système est donnée par :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} + \frac{\partial E_D}{\partial \dot{q}_i} = \tau_i \quad (2.1)$$

où τ_i représente la force ou le couple généralisé de la $i^{\text{ème}}$ articulation.

Pour un robot rigide à n degrés de liberté, le calcul de la force généralisée est donnée par :

$$\begin{aligned} \tau_i = & \sum_{j=i}^n \sum_{k=1}^j \text{trace}(U_{jk} J_j U_{ji}^T) \ddot{q}_k - \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{pj} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \\ & - \sum_{p=i}^n m_p g^T U_{pi} r_p^p + b_i \dot{q}_i \end{aligned} \quad (2.2)$$

(Pour le développement de cette dernière, voir ANNEXE 2).

Dans la partie qui va suivre, nous explicitons le calcul de chacun des termes de l'expression (2.1) dans le cas d'un robot rigide à n degrés de liberté (voir ANNEXE2).

II.3.2 Calcul du modèle dynamique du robot à structure cylindrique :

L'énergie potentielle est :

$$E_p = (m_0 + m_2 + m_3)(q_2 - q_{20})g \quad (2.3)$$

q_{20} : correspond à la position particulière q_2 du manipulateur pour laquelle l'énergie potentielle est nulle.

L'énergie cinétique s'exprime par :

$$E_c = \frac{1}{2} \left[(I_2 + I_3 + m_3(q_3 - l_3)^2 + m_0 q_3^2) \dot{q}_1^2 + (m_0 + m_2 + m_3) \dot{q}_2^2 + (m_0 + m_3) \dot{q}_3^2 \right] \quad (2.4)$$

Le Lagrangien est donnée par :

$$\begin{aligned} L = & E_c - E_p \\ = & \frac{1}{2} \left[(I_2 + I_3 + m_3(q_3 - l_3)^2 + m_0 q_3^2) \dot{q}_1^2 + (m_0 + m_2 + m_3) \dot{q}_2^2 + (m_0 + m_3) \dot{q}_3^2 \right] \\ & - (m_0 + m_2 + m_3)(q_2 - q_{20})g \end{aligned} \quad (2.5)$$

L'énergie de dissipation est :

$$E_D = \frac{1}{2} (f_{v1} \dot{q}_1^2 + f_{v2} \dot{q}_2^2 + f_{v3} \dot{q}_3^2) \quad (2.6)$$

L'équation d'Euler-Lagrange pour la première articulation est :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_1} \right) - \frac{\partial L}{\partial q_1} + \frac{\partial E_D}{\partial \dot{q}_1} = \Gamma_1 \quad (2.7)$$

Donc [7] :

$$\left(l_2 + l_3 + m_3(q_3 - l_3)^2 + m_0 q_3^2 \right) \ddot{q}_1 + 2(m_3(q_3 - l_3) + m_0 q_3) \dot{q}_1 \dot{q}_3 + f_{v1} \dot{q}_1 = \Gamma_1 \quad (2.8)$$

L'équation d'Euler-Lagrange pour la deuxième articulation est :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_2} \right) - \frac{\partial L}{\partial q_2} + \frac{\partial E_D}{\partial \dot{q}_2} = F_2 \quad (2.9)$$

Donc [7] :

$$(m_0 + m_2 + m_3) \ddot{q}_2 + (m_0 + m_2 + m_3)g + f_{v2} \dot{q}_2 = F_2 \quad (2.10)$$

L'équation d'Euler-Lagrange pour la troisième articulation est :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_3} \right) - \frac{\partial L}{\partial q_3} + \frac{\partial E_D}{\partial \dot{q}_3} = F_3 \quad (2.11)$$

Donc [7] :

$$(m_0 + m_3) \ddot{q}_3 - (m_3(q_3 - l_3) + m_0 q_3) \dot{q}_1^2 + f_{v3} \dot{q}_3 = F_3 \quad (2.12)$$

Chapitre III

COMMANDE FLOUE ADAPTATIVE DIRECTE DÉCENTRALISÉE BASÉE SUR LA MÉTHODE DU COUPLE CALCULÉ

III.1 Introduction générale:

La commande floue est l'application la plus réussie de la théorie des systèmes et ensembles flous en pratique. En effet, l'intérêt d'une telle commande est dû largement aux applications réussies des régulateurs logiques flous des systèmes industriels.

Pourquoi la commande floue ?

Il peut être utile de diviser les raisons du *pourquoi de la commande floue* en deux catégories : raisons théoriques et pratiques [12].

1) *Raisons théoriques pour la commande floue :*

- En règle générale, une bonne approche d'ingénierie devrait être capable de rendre l'utilisation de toute information disponible et effective. Si le modèle mathématique d'un système est très difficile à obtenir (ceci est vrai pour plusieurs systèmes pratiques), alors l'information la plus importante vient de deux sources :
 - 1) Les capteurs qui fournissent des mesures numériques pour des variables importantes.
 - 2) L'expert qui fournit les descriptions linguistiques sur le système et les instructions de commande.
- La commande floue est une approche libre du modèle en d'autre terme celle-ci nécessite pas la connaissance exacte du modèle mathématique du système.
- La commande floue permet d'accomplir les actions de commande du système non linéaire. Donc, en choisissant attentivement les paramètres du régulateur flou, il est toujours possible de concevoir un régulateur flou qui sera convenable pour le système non linéaire à commander.

2) *Raisons pratiques pour la commande floue :*

- Elle est simple à implémenter et à comprendre. Les systèmes logiques flous, qui sont au cœur de la commande floue, admettent un degré élevé d'implémentation parallèle.

- Elle n'est pas chère à développer. Du point de vue pratique, le coût de développement est l'un des critères les plus importants pour un produit. Parce que la commande floue est facile à comprendre, le temps nécessaire pour apprendre l'approche est court, c'est à dire, le coût du logiciel « software cost » est bas. Aussi, parce que la commande floue est facile à implémenter, le coût de machine « hardware cost » est aussi bas.

Pourquoi la commande floue adaptative ?

Les régulateurs flous doivent travailler dans des situations où peuvent apparaître des variations paramétriques, ou une modification de la structure du système.

Dans certaines tâches de commandes, telles que celles des robots manipulateurs, le système à commander admet des paramètres incertains au début de l'opération de commande, à moins que ces paramètres incertains soient graduellement réduits on-line par un mécanisme d'adaptation ou d'estimation, car ils pourraient causer l'imprécision ou l'instabilité.

Généralement, l'objectif de base de la commande adaptative est de maintenir la performance consistante du système en présence de ses variations incertaines, lesquelles apparaissent dans les problèmes pratiques. La commande adaptative est utilisée dans plusieurs contextes industriels ceci inclus : la manipulation du robot [10] ; les robots doivent manipuler des charges de différentes tailles, poids et distribution de masses. C'est très restreint de supposer que les paramètres d'inertie des charges sont bien connus et cela avant que le robot aille les chercher et les transporter, donc si les commandes avec des gains constants sont utilisées et les paramètres de la charge ne sont pas exactement connus, le déplacement du robot peut être soit impossible ou instable. La commande adaptative d'un autre côté permet aux robots de déplacer les charges à paramètres inconnus avec une grande vitesse et une grande précision. Donc, la commande floue avancée devrait être adaptative.

III.2 Commande floue adaptative directe décentralisée basée sur la méthode du couple calculé :

III.2.1 Introduction :

Le régulateur flou adaptatif est proposé pour la commande de poursuite de la trajectoire des robots manipulateurs. L'approche utilise une procédure de commande quantitative pour assurer la stabilité globale et la procédure de commande qualitative pour approximer n'importe quelle fonction non-linéaire causée par des perturbations, incertitudes et les interconnexions d'un système.

Avec la commande PD (telle que la stabilité locale est maintenue), nous ajoutons un retour (feed-back) non linéaire pour assurer la stabilité globale du système entier.

Le régulateur flou adaptatif est incorporé dans le système de commande du bras de robot comme un approximateur de fonction pour compenser l'effet des interconnexions, friction, force gravitationnelle, les incertitudes et les variations de la charge.

Le critère de stabilité du régulateur proposé est développé en utilisant l'approche de synthèse de Lyapunov [5].

III.2.2 Formulation du problème :

L'équation du mouvement d'un bras de robot rigide à n degrés de liberté est [5]:

$$M(q)\ddot{q} + H(q, \dot{q}) + G(q) + F_r(\dot{q}) = \tau \quad (3.1)$$

$q \in \mathfrak{R}^n$: vecteur position des articulations.

$\tau \in \mathfrak{R}^n$: vecteur du couple appliqué.

$M(q) \in \mathfrak{R}^{n \times n}$: matrice d'inertie, symétrique définie positive.

$H(q, \dot{q}) = H_m(q, \dot{q}) \cdot \dot{q} \in \mathfrak{R}^n$: couple des forces de coriolis et des forces centrifuges.

$F_r(\dot{q}) \in \mathfrak{R}^n$: couple de frottement.

Le modèle du robot à structure cylindrique est mis sous cette forme dans l'ANNEXE 3 (Ann3.1).

Il est bien établi que :

$\left(\frac{\dot{M}}{2} - H_m \right)$ est skew-symétrique c'est à dire :

$$x^T \left(\frac{\dot{M}}{2} - H_m \right) x = 0 \quad \forall x \neq 0 \quad (3.2)$$

L'erreur de poursuite en position est donnée par :

$$e = q_d - q, \quad q_d \in \mathfrak{R}^n$$

où q_d est le vecteur des trajectoires désirées.

Nous introduisons également le vecteur de l'erreur de poursuite filtrée S :

$$S = \dot{e} + Ae, \quad A = A^T \succ 0 \quad (3.3)$$

A est une matrice constante.

En utilisant S , la dynamique (3.1) du robot devient :

$$M(q)\dot{S} + H_m(q, \dot{q})S = -\tau + f \quad (3.4)$$

avec :

$$f = M(q)(\ddot{q}_d + A\dot{e}) + H_m(q, \dot{q})(\dot{q} + Ae) + G(q) + F_r(\dot{q}) \quad (3.5)$$

Pour formuler la dynamique de l'erreur de poursuite décentralisée, nous décomposons la dynamique de l'erreur globale (3.4) et (3.5) en n couples articulaires :

$$\sum_{j=1}^n M_{ij}(q) \dot{S}_j + \dot{q}^T H_i(q) S = -\tau_i + f_i \quad (3.6)$$

où :

M_{ij} : est l'élément (i, j) de la matrice $M(q)$.

f_i : la $i^{\text{ème}}$ composante de f donnée par :

$$f_i(\cdot) = \sum_{j=1}^n M_{ij}(q)(\ddot{q}_j + A_j \dot{e}_j) + \dot{q}^T H_i(q)(\dot{q}_d + Ae) + G_i(q) + F_n(q) \quad (3.7)$$

Nous appellerons (3.6), un sous-système à entrée indépendante du fait que seulement son entrée τ_i est isolée des autres sous-systèmes. Pour chaque sous-système, S est le vecteur d'état à commander et l'interconnexion f_i est bornée par un polynôme du seconde ordre :

$$|f_i| \leq \beta_{i1} + \beta_{i2} \|S\| + \beta_{i3} \|S\|^2 \quad (3.8)$$

avec : β_{i1}, β_{i2} et $\beta_{i3} \in \mathfrak{R}^+$

Dans le cas idéal, le couple exercé sur la $i^{\text{ème}}$ articulation serait :

$$\tau_i = k_i S_i + \hat{f}_i \quad (3.9)$$

avec $k_i \in K = \text{diag}\{k_1, k_2, \dots, k_n\} > 0$ est un gain PD et l'estimée \hat{f}_i est égale à f_i réelle. Sous la loi de commande de (3.9), nous obtenons une stabilité globalement asymptotique du système quand $t \rightarrow \infty$, $S \rightarrow 0$.

Cependant, pour avoir $\hat{f}_i = f_i$, cela nécessite une structure à retour globale (centralisée) et des données détaillées sur le modèle du système qui sont impossible à recueillir. Même si f_i est une fonction de variables globales, nous la décomposons en deux termes :

$$f_i = \tilde{f}_i(e_i, \dot{e}_i | \theta_i) + f_{ai} \quad (3.10)$$

où $\tilde{f}_i(e_i, \dot{e}_i | \theta_i)$ est fonction seule des variables locales et θ_i est un vecteur des paramètres à déterminer. Tous les effets restants de f_i sont laissés dans f_{ai} .

Evidemment, f_{ai} est aussi bornée par la forme quadratique (3.8) avec des coefficients différents.

Dans notre étude, nous approximos $\tilde{f}_i(e_i, \dot{e}_i | \theta_i)$ par un système flou tel que :

$$\hat{f}_i = W_i^T(e_i, \dot{e}_i) \cdot \hat{\theta}_i \quad (3.11)$$

III.2.3 Dynamique de l'erreur filtrée:

Le régulateur adaptatif flou considéré utilise la structure de réglage décentralisée par articulation, laquelle est adoptée dans la plupart des robots industriels. Ainsi pour la dynamique (3.6), nous proposons la loi de commande décentralisée de la forme :

$$\tau_i = k_i S_i + \hat{f}_i(e_i, \dot{e}_i | \hat{\theta}_i) - u_{gl}(i) \quad (3.12)$$

u_{gl} : terme de glissement.

Introduisons (3.12) dans (3.6), on obtient :

$$\sum_{j=1}^n M_{ij}(q) \dot{S}_j = -\dot{q}^T H_i(q) S - k_i S_i - \hat{f}_i(e_i, \dot{e}_i | \hat{\theta}_i) + u_{gl}(i) + f_i(e_i, \dot{e}_i | \theta_i) + f_{ai} \quad (3.13)$$

Ajoutons et retranchons $\hat{f}_i(e_i, \dot{e}_i | \theta_i^*)$ au second membre de (3.13), il vient :

$$\sum_{j=1}^n M_{ij}(q) \dot{S}_j = -\dot{q}^T H_i(q) S - k_i S_i + u_{gl}(i) + W_i^T \tilde{\theta}_i + \varepsilon_i \quad (3.14)$$

avec :

$$W_i^T \tilde{\theta}_i = \hat{f}_i(e_i, \dot{e}_i | \theta_i^*) - \hat{f}_i(e_i, \dot{e}_i | \hat{\theta}_i) \quad , \quad \tilde{\theta}_i = \theta_i^* - \hat{\theta}_i$$

$$\varepsilon_i = f_i(e_i, \dot{e}_i | \theta_i) - \hat{f}_i(e_i, \dot{e}_i | \theta_i^*) + f_{ai}$$

où ε_i est l'erreur minimale d'approximation qui est bornée par (3.8).

Pour le système global, l'expression (3.14) peut s'écrire :

$$M \dot{S} = -H_m(q, \dot{q}) S - K S - u_{gl} + W^T \tilde{\theta} + \varepsilon \quad (3.15)$$

où ε est bornée par :

$$\|\varepsilon\| \leq \beta_1 + \beta_2 \|S\| + \beta_3 \|S\|^2 \quad ; \quad \beta_1, \beta_2 \text{ et } \beta_3 \text{ des constantes positives.}$$

III.2.4 Analyse de la stabilité :

Proposition :

Soit le système (3.1) dont le modèle dynamique défini par (3.15). Dans le cas où la loi de commande est donnée par $\tau_i = k_i S_i + \hat{f}_i(e_i, \dot{e}_i | \hat{\theta}_i) - u_{gl}(i)$ où la fonction estimée $\hat{f}_i(e_i, \dot{e}_i | \hat{\theta}_i)$ est un système flou dont les paramètres sont adaptés par la loi :

$$\dot{\hat{\theta}} = \gamma W S \quad (3.16)$$

Par conséquent, l'erreur de poursuite filtrée S tend vers zéro et les paramètres $\hat{\theta}$ restent bornés.

Preuve :

Considérons la fonction de Lyapunov suivante :

$$V = \frac{1}{2} S^T M S + \frac{1}{2\gamma} \tilde{\theta}^T \tilde{\theta} \quad (3.17)$$

Sa dérivée est donnée par :

$$\dot{V} = S^T M \dot{S} + \frac{1}{2} S^T \dot{M} S + \frac{1}{\gamma} \tilde{\theta}^T \dot{\tilde{\theta}} \quad (3.18)$$

$$\dot{V} = S^T M \dot{S} + S^T \left(\frac{\dot{M}}{2} - H_m \right) S + S^T H_m S + \frac{1}{\gamma} \tilde{\theta}^T \dot{\tilde{\theta}} \quad (3.19)$$

Puisque $\left(\frac{\dot{M}}{2} - H_m \right)$ est skew-symétrique et en remplaçant $M \dot{S}$ par son expression dans (3.15) et sachant que $\dot{\tilde{\theta}} = -\dot{\hat{\theta}}$, il vient :

$$\dot{V} = -S^T K S - S^T u_{sl} + S^T W^T \tilde{\theta} + S^T \varepsilon - \frac{1}{\gamma} \tilde{\theta}^T \dot{\hat{\theta}} \quad (3.20)$$

En remplaçant $\dot{\hat{\theta}}$ par son expression dans (3.10), \dot{V} se réduit à :

$$\dot{V} = -S^T K S - S^T u_{sl} + S^T \varepsilon \quad (3.21)$$

On impose à u_{sl} la forme : $u_{sl} = k_{sl} \text{sign}(S)$, $k_{sl} \in \mathfrak{R}^+$

$$\dot{V} = -S^T K S - k_{sl} |S| + S^T \varepsilon \quad (3.22)$$

Or :

$$\dot{V} \leq -S^T K S - k_{sl} \|S\| + \|S\| \cdot \|\varepsilon\| \quad (3.23)$$

Si $k_{sl} \geq \bar{\varepsilon}$, $\|\varepsilon\| \leq \bar{\varepsilon} \quad \forall t > 0$ alors :

$$\dot{V} \leq -S^T K S < 0 \quad \forall S \neq 0 \quad (3.24)$$

En conclusion, l'erreur de poursuite filtrée S tend asymptotiquement vers zéro lorsque $t \rightarrow \infty$ et $\tilde{\theta}$ reste borné.

III.2.5 Résultats de la simulation :

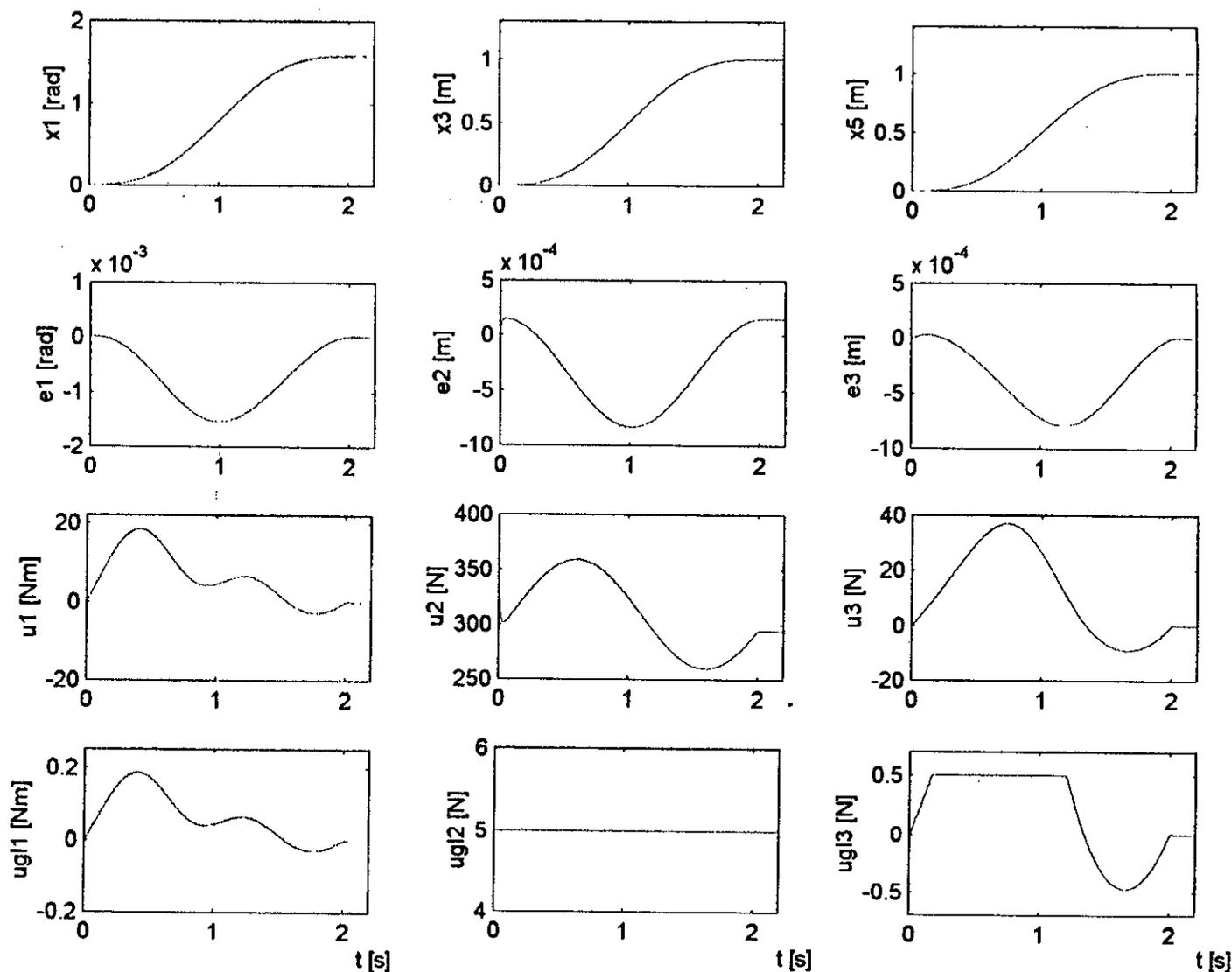


Figure 3.1 : Régime de poursuite du robot sans charge ($m_0 = 0$)

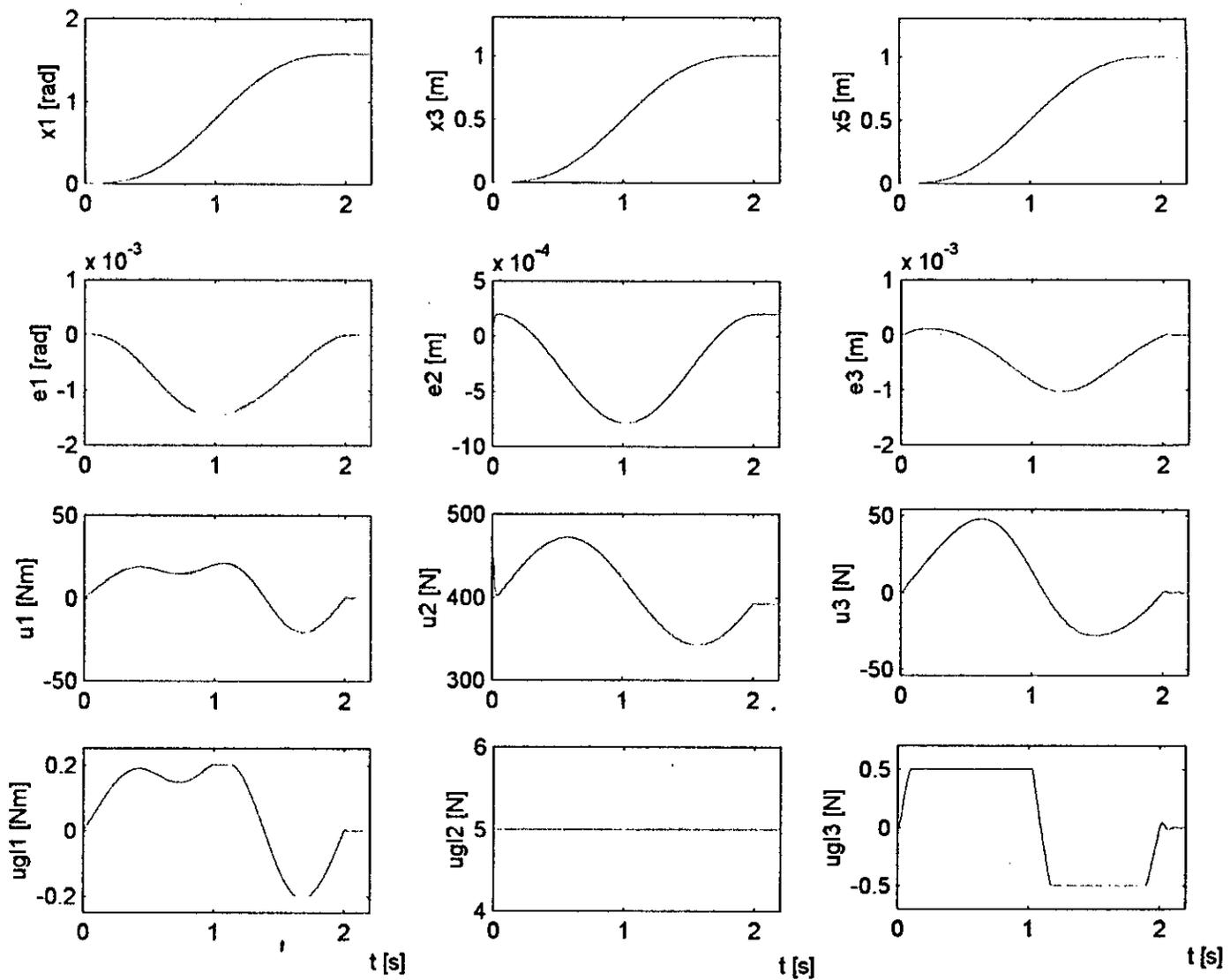


Figure 3.2 : Régime de poursuite du robot en charge ($m_0 = 10 \text{ Kg}$)

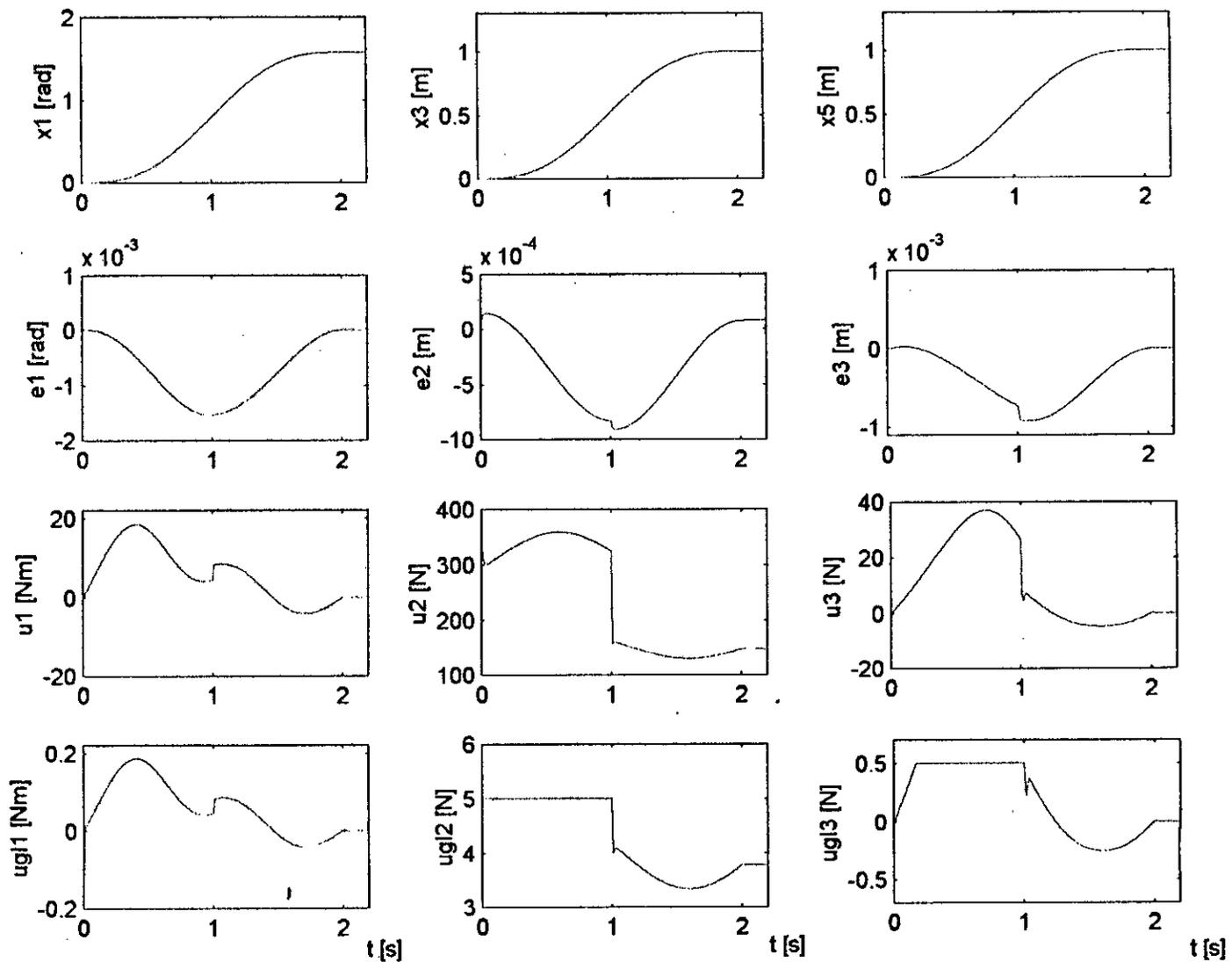


Figure 3.3 : Régime de poursuite du robot sans charge avec variations paramétriques en diminution

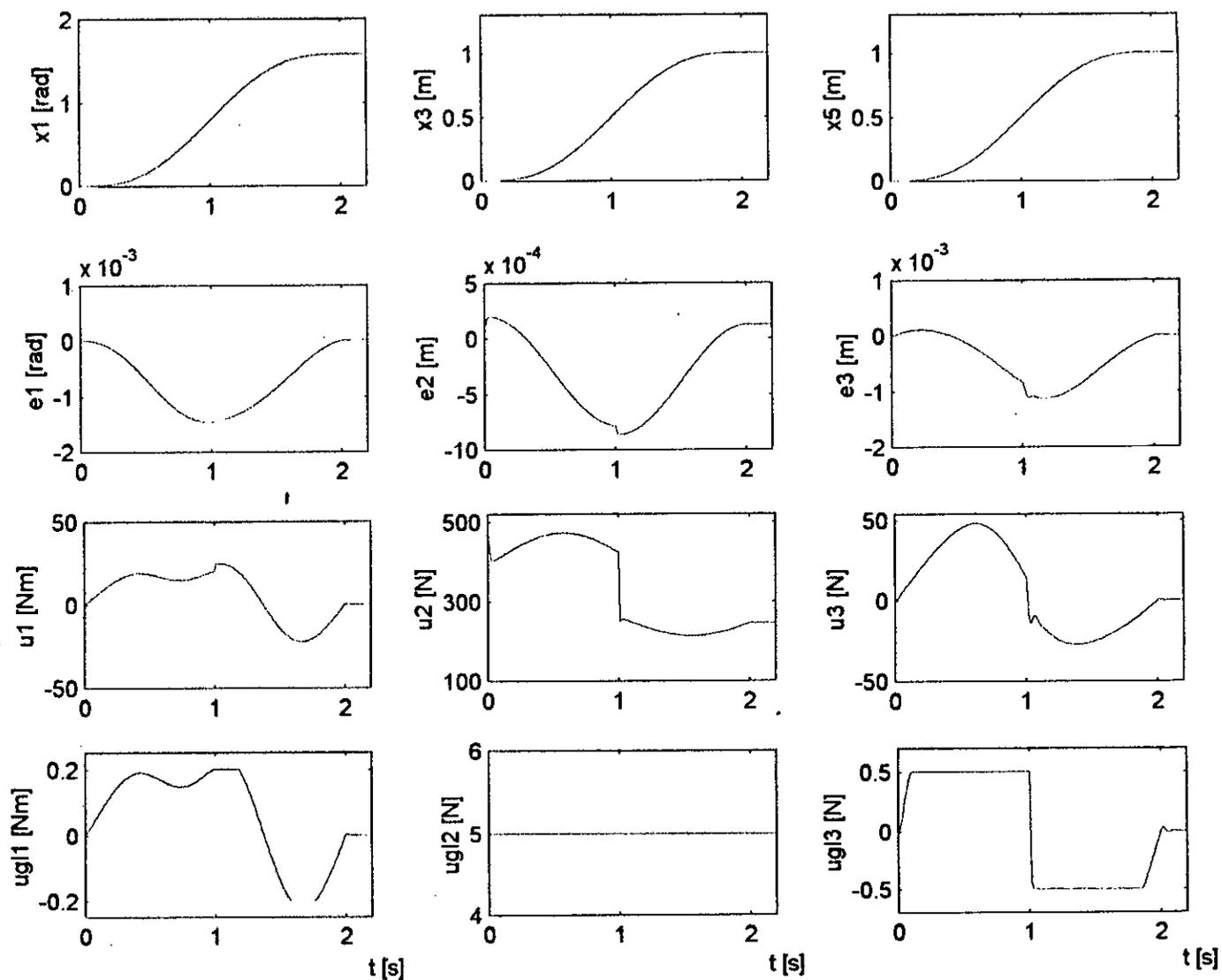


Figure 3.4 : Régime de poursuite du robot en charge avec variations paramétriques en diminution

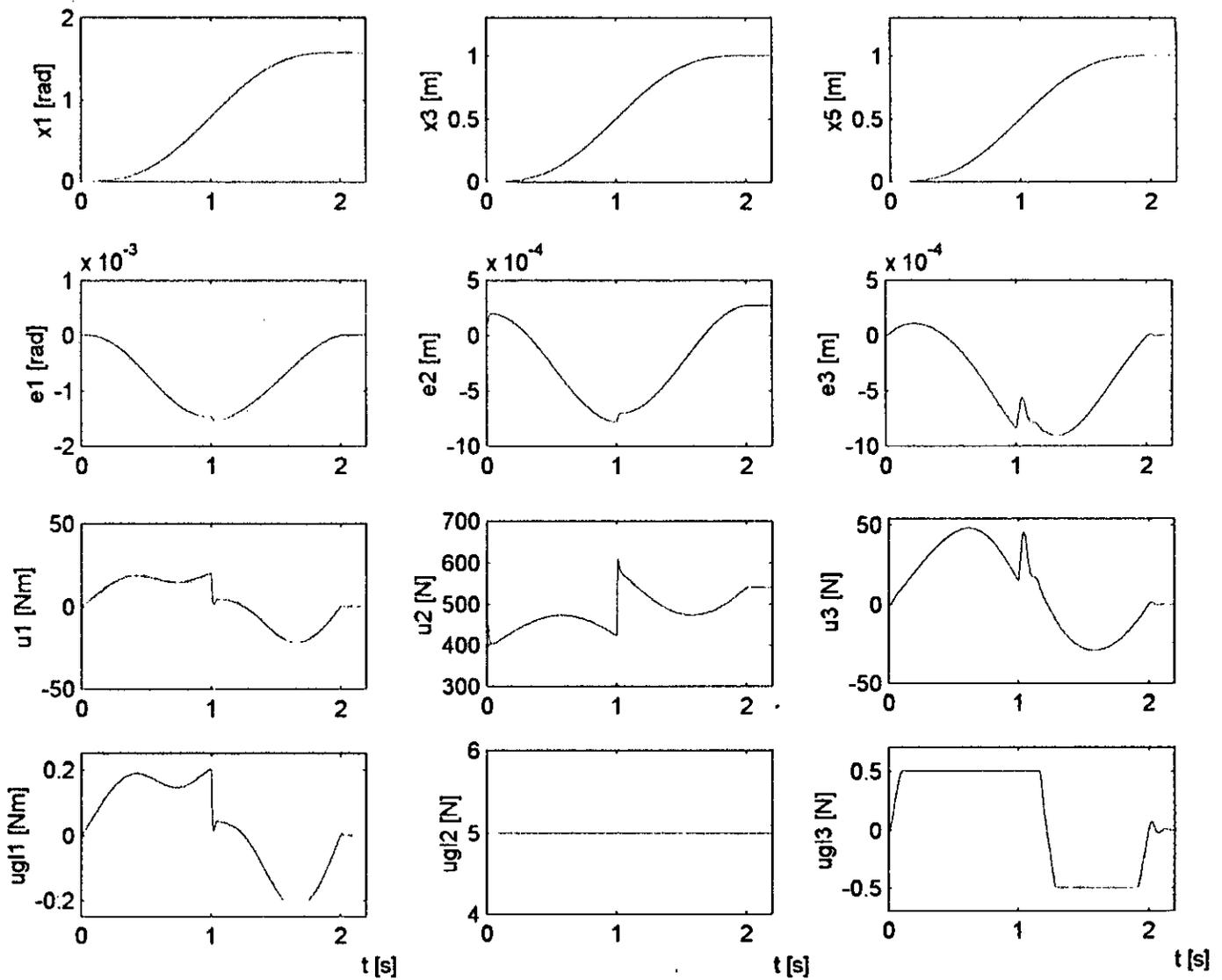


Figure 3.5 : Régime de poursuite du robot en charge avec variations paramétriques en augmentation

Interprétation des résultats :

Pour le fonctionnement du robot à vide (figure 3.1), le régime de poursuite est obtenu avec une faible erreur ($|e|_{\max} < 2 \cdot 10^{-3}$). La commande obtenue est lisse avec un terme de glissement qui reste réduit relativement à la commande.

Lors du fonctionnement en charge (figure 3.2), les réponses restent pratiquement les mêmes pour l'essai à vide, seules les commandes augmentent pour compenser les effets de la charge. En appliquant une variation des paramètres (soit une diminution ou une augmentation de 50% de tous les paramètres du robot), le régime de poursuite est toujours maintenu avec une erreur maximale ($|e|_{\max} < 2 \cdot 10^{-3}$) que ce soit en charge ou à vide. Il apparaît que le terme de glissement subit une variation juste à l'instant de l'application des perturbations, par conséquent, ce dernier intervient judicieusement.

Conclusion :

Dans ce chapitre, nous nous intéressons au modèle directe du robot ; où la commande en est directement déduite si les fonctions qui modélisent le robot sont exactement connues. Cependant, ceci n'est pas toujours vrai, ce qui nous mène à approximer les fonctions inconnues par des systèmes flous de Sugeno d'ordre zéro. La commande est rendue adaptative en déterminant les lois d'adaptation des paramètres des systèmes flous en appliquant la méthode de stabilité de Lyapunov. Cette commande a permis d'obtenir une bonne poursuite de la trajectoire avec une faible erreur, la forme de la commande est intéressante du fait qu'elle est lisse mise à part une discontinuité à l'instant de l'application des variations paramétriques, qui n'affecte pas d'ailleurs sa robustesse. L'intérêt de cette commande réside dans sa simplicité et son calcul en décentralisé.

Chapitre IV

COMMANDE FLOUE ADAPTATIVE DIRECTE LINÉARISANTE (TYPE 1)

IV.1 Introduction :

En exploitant toujours le modèle directe du robot, on développe dans ce chapitre la commande floue adaptative directe (type 1). On commence par approximer les fonctions représentant les forces généralisées par des systèmes de Sugeno d'ordre un, puis à partir de l'écriture du modèle, on déduit les fonctions floues permettant d'écrire le système sous forme linéaire.

IV.2 Formulation du problème :

Soit la classe des systèmes non linéaires de la forme :

$$\begin{cases} u_i = F_i(X) x_i^{(p)} + G_i(X) \\ y_i = x_i \quad i = 1 \dots n \end{cases} \quad (4.1)$$

où :

$$u_i \in \mathfrak{R} \quad , \quad x_i \in \mathfrak{R} \quad , \quad F_i(X) \in \mathfrak{R} \quad , \quad G_i(X) \in \mathfrak{R}.$$

$$X^T = \left[\left(x^{(p-1)} \right)^T, \left(x^{(p-2)} \right)^T, \dots, \ddot{x}^T, \dot{x}^T, x^T \right]$$

$$x = \left[x_1 \quad x_2 \quad \dots \quad x_{n+1} \right]^T$$

La commande u_i du système (4.1) peut s'écrire sous la forme :

$$u_i = f_i(X, x_i^{(p)}) \quad (4.2)$$

Pour le robot étudier, on se base sur le modèle présenté dans l'ANNEXE 3 (Ann3.1).

Si les fonctions $F_i(X)$ et $G_i(X)$ sont bien connues, la loi de commande linéarisante permet d'obtenir [6] :

$$u_i^* = F_i(X) v_i(t) + G_i(X) \quad (4.3)$$

avec :

$$v_i(t) = x_{id}^{(n)} + k_p e_i^{(n-1)} + \dots + k_1 e_i \quad (4.4)$$

où :

$$e_i = x_{id} - x_i \quad \text{et} \quad k_i > 0, \quad i = 1 \dots n \quad (4.5)$$

Si les coefficients (k_1, k_2, \dots, k_p) sont choisis tels que tous les pôles soient dans le demi-plan gauche, alors l'erreur e_i tend exponentiellement vers zéro du fait qu'en appliquant la commande u_i^* au système (4.1), il vient :

$$v_i(t) = x_i^{(n)} \quad (4.6)$$

Ce qui donne :

$$e_i^{(n)} + k_p e_i^{(n-1)} + \dots + k_1 e_i = 0 \quad (4.7)$$

Du fait que les fonctions $F_i(X)$ et $G_i(X)$ sont inconnues ou mal connues, les systèmes logiques flous sont utilisés pour approximer la commande :

$$u_i = \hat{F}_i(X) (v_i + u_{gl}) + \hat{G}_i(X) \quad (4.8)$$

Où :

$$u_i = \hat{f}_i(X, x_i^{(n)}) \quad (4.9)$$

et u_{gl} est le terme de glissement pour approximer les erreurs de reconstruction.

IV.3 Dynamique de l'erreur filtrée :

La fonction $\hat{f}_i(X, x_i^{(n)})$ est approximée par un système flou de Sugeno d'ordre 1 :

$$\hat{f}_i(X, x_i^{(n)}) = W_{f_i}^T(X, x_i^{(n)}) \cdot \hat{\theta}_{f_i} \quad (4.10)$$

La valeur exacte de $f_i(X, x_i^{(n)})$ peut s'écrire sous la forme :

$$f_i(X, x_i^{(n)}) = W_{f_i}^T(X, x_i^{(n)}) \cdot \theta_{f_i}^* + \varepsilon_{f_i} \quad (4.11)$$

Avec :

$$\varepsilon_{f_i} = f_i(X, x_i^{(n)}) - \hat{f}_i(X, x_i^{(n)}) \quad (4.12)$$

ε_{f_i} : erreur de reconstruction .

$\theta_{f_i}^*$: vecteur des paramètres optimaux .

En appliquant à l'entrée du système (4.1), la commande (4.9), on obtient :

$$v_i(t) = \hat{F}_i^{-1}(X) \left\{ \hat{F}_i(X) x_i^{(p)} + G_i(X) - \hat{G}_i(X) \right\} - u_{gl} \quad (4.13)$$

Ou encore :

$$e_i^{(p)} + k_p e_i^{(p-1)} + \dots + k_2 \dot{e}_i + k_1 e_i = -\hat{F}_i^{-1}(X) \left\{ \left(\hat{F}_i(X) x_i^{(p)} + \hat{G}_i(X) \right) - \left(F_i(X) x_i^{(p)} + G_i(X) \right) \right\} - u_{gl} \quad (4.14)$$

or :

$$\begin{aligned} \hat{F}_i(X) x_i^{(p)} + \hat{G}_i(X) &= \hat{f}_i(X, x_i^{(p)}) = W_{f_i}^T(X, x_i^{(p)}) \hat{\theta}_{f_i} \\ F_i(X) x_i^{(p)} + G_i(X) &= f_i(X, x_i^{(p)}) = W_{f_i}^T(X, x_i^{(p)}) \theta_{f_i}^* + \varepsilon_{f_i} \end{aligned} \quad (4.15)$$

d'où :

$$e_i^{(p)} + k_p e_i^{(p-1)} + \dots + k_2 \dot{e}_i + k_1 e_i = -\hat{F}_i^{-1}(X) \left\{ W_{f_i}^T (\hat{\theta}_{f_i} - \theta_{f_i}^*) - \varepsilon_{f_i} \right\} - u_{gl} \quad (4.16)$$

On définit l'erreur paramétrique :

$$\tilde{\theta}_{f_i} = \hat{\theta}_{f_i} - \theta_{f_i}^* \quad (4.17)$$

Par conséquent :

$$e_i^{(p)} = -k_p e_i^{(p-1)} - k_{p-1} e_i^{(p-2)} - \dots - k_2 \dot{e}_i - k_1 e_i - \hat{F}_i^{-1}(X) \cdot W_{f_i}^T \tilde{\theta}_{f_i} + \hat{F}_i^{-1} \cdot \varepsilon_{f_i} - u_{gl} \quad (4.18)$$

On s'intéresse à la sortie filtrée S_i définie par :

$$S_i = \left(\frac{\partial}{\partial t} + \lambda_i \right)^{(p-1)} \cdot e_i \quad i = 1 \dots p \quad (4.19)$$

On introduit le vecteur Y_i donné par :

$$Y_i = \left[e_i \quad \dot{e}_i \quad \dots \quad e_i^{(p-1)} \right]^T$$

S_i s'écrit alors la forme :

$$S_i = C^T \cdot Y_i \quad (4.20)$$

avec :

$$C^T = \left[\lambda_i^{p-1} \quad (p-1)\lambda_i^{p-2} \quad \dots \quad (p-1)\lambda_i \quad 1 \right]$$

On peut alors définir la dynamique suivante :

$$\begin{cases} \dot{Y}_i = A_i Y_i + B_i E_X(i) \\ S_i = C^T Y_i \end{cases} \quad i=1 \dots n \quad (4.21)$$

avec :

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 1 \\ 0 & 0 & \dots & \dots & 0 \\ -k_1 & -k_2 & \dots & \dots & -k_p \end{bmatrix}, \quad B_i = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$E_X(i) = -\hat{F}_i^{-1}(X) W_{fi}^T \tilde{\theta}_{fi} + \hat{F}_i^{-1}(X) \varepsilon_{fi} - u_{gi}$$

Le coefficient λ est imposé tel que la fonction de transfert $H_i(s)$ du système (4.21) donnée par :

$$H_i(s) = \frac{s + \lambda}{s^p + k_p s^{p-1} + \dots + k_2 s + k_1} \quad (4.22)$$

soit strictement réelle positive. Par conséquent, ils existent des matrices définies positives P_i et Q_i telles que :

$$\begin{cases} A_i^T P_i + P_i A_i = -Q_i \\ P_i B_i = C_i \end{cases} \quad (4.23)$$

IV.4 Analyse de la stabilité :

Proposition :

Soit le système (4.1) dont le modèle dynamique de l'erreur est donné par (4.21), si la loi de commande est de la forme (4.8) où la fonction $\hat{f}_i(X, x_i^{(p)})$ est estimée par un système flou dont les paramètres sont adaptés par :

$$\dot{\hat{\theta}}_{fi} = \gamma W_{fi} \hat{F}_i^{-1} S_i \quad (4.24)$$

Par conséquent, le vecteur d'erreur Y_i tend asymptotiquement vers zéro et $\tilde{\theta}_f$ reste borné.

Preuve :

Soit la fonction de Lyapunov suivante :

$$V = Y_i^T P Y_i + \frac{1}{\gamma} \tilde{\theta}_f^T \tilde{\theta}_f \quad (4.25)$$

Alors \dot{V} est donné par :

$$\dot{V} = \dot{Y}_i^T P Y_i + Y_i^T P \dot{Y}_i + \frac{2}{\gamma} \tilde{\theta}_f^T \dot{\hat{\theta}}_f \quad (4.26)$$

En remplaçant l'expression de \dot{Y} dans (4.21), on obtient :

$$\dot{V} = Y_i^T (A_i^T P + P A_i) Y_i + E_X^T(i) B_i^T P Y_i + Y_i^T P B_i E_X(i) + \frac{2}{\gamma} \tilde{\theta}_f^T \dot{\hat{\theta}}_f \quad (4.27)$$

Et d'après l'équation (4.23), \dot{V} peut s'écrire sous la forme :

$$\dot{V} = -Y_i^T Q_i Y_i + E_X^T(i) C^T Y_i + Y_i^T C E_X(i) + \frac{2}{\gamma} \tilde{\theta}_f^T \dot{\hat{\theta}}_f \quad (4.28)$$

Comme $Y_i^T C E_X(i) \in \mathfrak{R} \Rightarrow Y_i^T C E_X(i) = E_X^T(i) C^T Y_i = E_X^T(i) S_i$ (4.29)

$$\dot{V} = -Y_i^T Q_i Y_i + 2 S_i^T E_X(i) + \frac{2}{\gamma} \tilde{\theta}_f^T \dot{\hat{\theta}}_f \quad (4.30)$$

En remplaçant $E_X(i)$ par son expression, et du fait que $S_i^T \hat{F}_i^{-1}(X) W_f^T \tilde{\theta}_f \in \mathfrak{R}$, on peut écrire :

$$\dot{V} = -Y_i^T Q_i Y_i + 2 S_i^T \hat{F}_i^{-1}(X) \varepsilon_f - 2 S_i^T u_{gi} - 2 \tilde{\theta}_f^T W_f \hat{F}_i^{-1}(X) S_i + \frac{2}{\gamma} \tilde{\theta}_f^T \dot{\hat{\theta}}_f \quad (4.31)$$

Et vue l'expression de $\dot{\hat{\theta}}_f$ dans (4.24), \dot{V} se réduit à :

$$\dot{V} = -Y_i^T Q_i Y_i + 2 S_i^T \hat{F}_i^{-1}(X) \varepsilon_f - 2 S_i^T u_{gi} \quad (4.32)$$

et l'inégalité ci-dessous est toujours vérifiée :

$$\dot{V} \leq -Y_i^T Q_i Y_i + 2 |S_i| \left| \hat{F}_i^{-1}(X) \right| \left| \varepsilon_f \right| - 2 S_i^T u_{gi} \quad (4.33)$$

Soit $\bar{\varepsilon}_i = \sup_{t>0} |\varepsilon_i(t)|$

Si on impose à $u_{gt} = \text{sign}(S_i) \cdot \left(\left| \hat{h}_i^{-1}(X) \right| \cdot \bar{\varepsilon}_i \right)$

On aura :

$$\dot{V} \leq -Y_i^T Q_i Y_i < 0 \quad \forall Y_i \neq 0 \quad (4.34)$$

En conclusion , Y_i tend asymptotiquement vers zéro et $\tilde{\theta}_i$ reste borné.

IV.5 Résultats de la simulation :

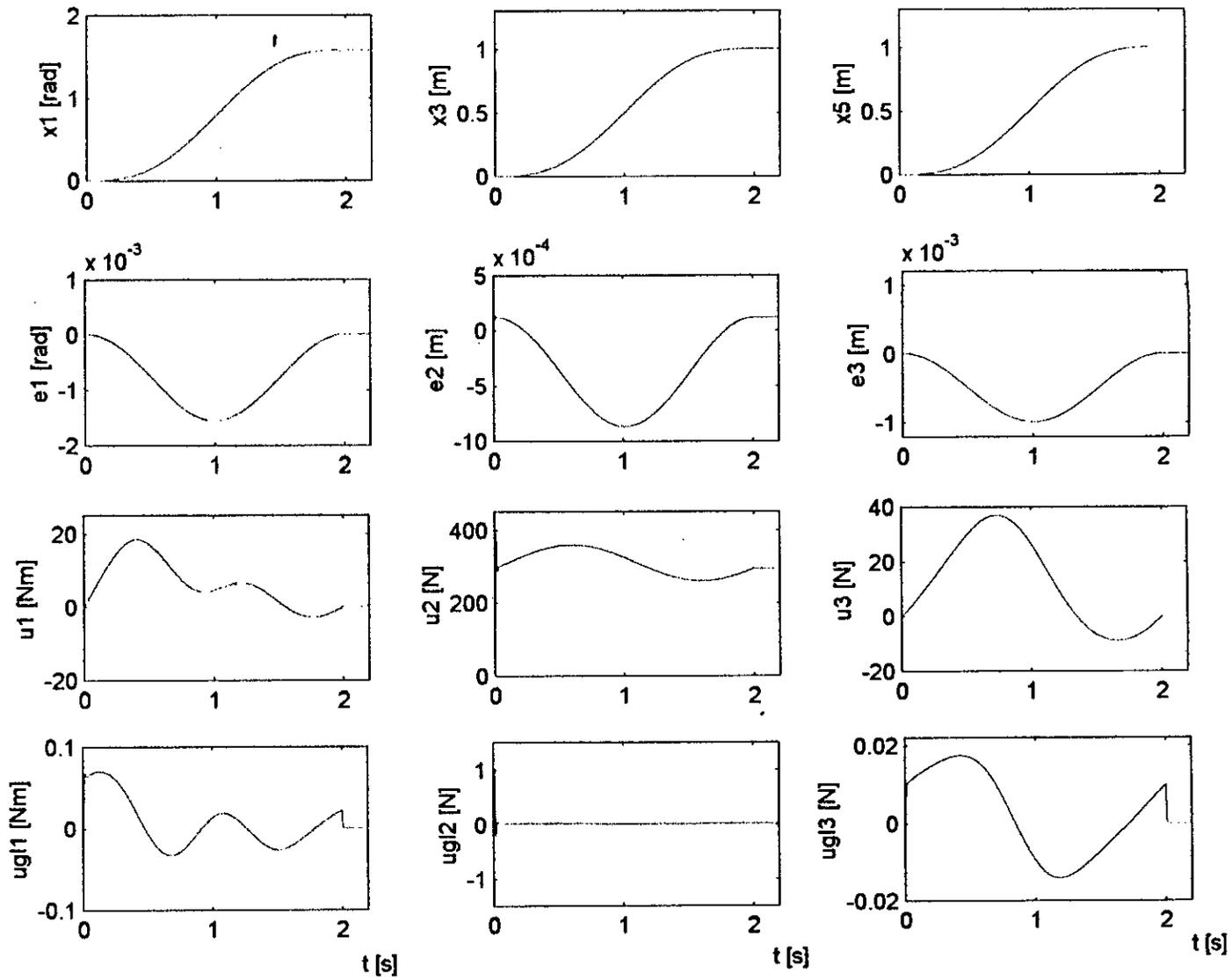


Figure 4.1: Régime de poursuite du robot sans charge ($m_0 = 0$)

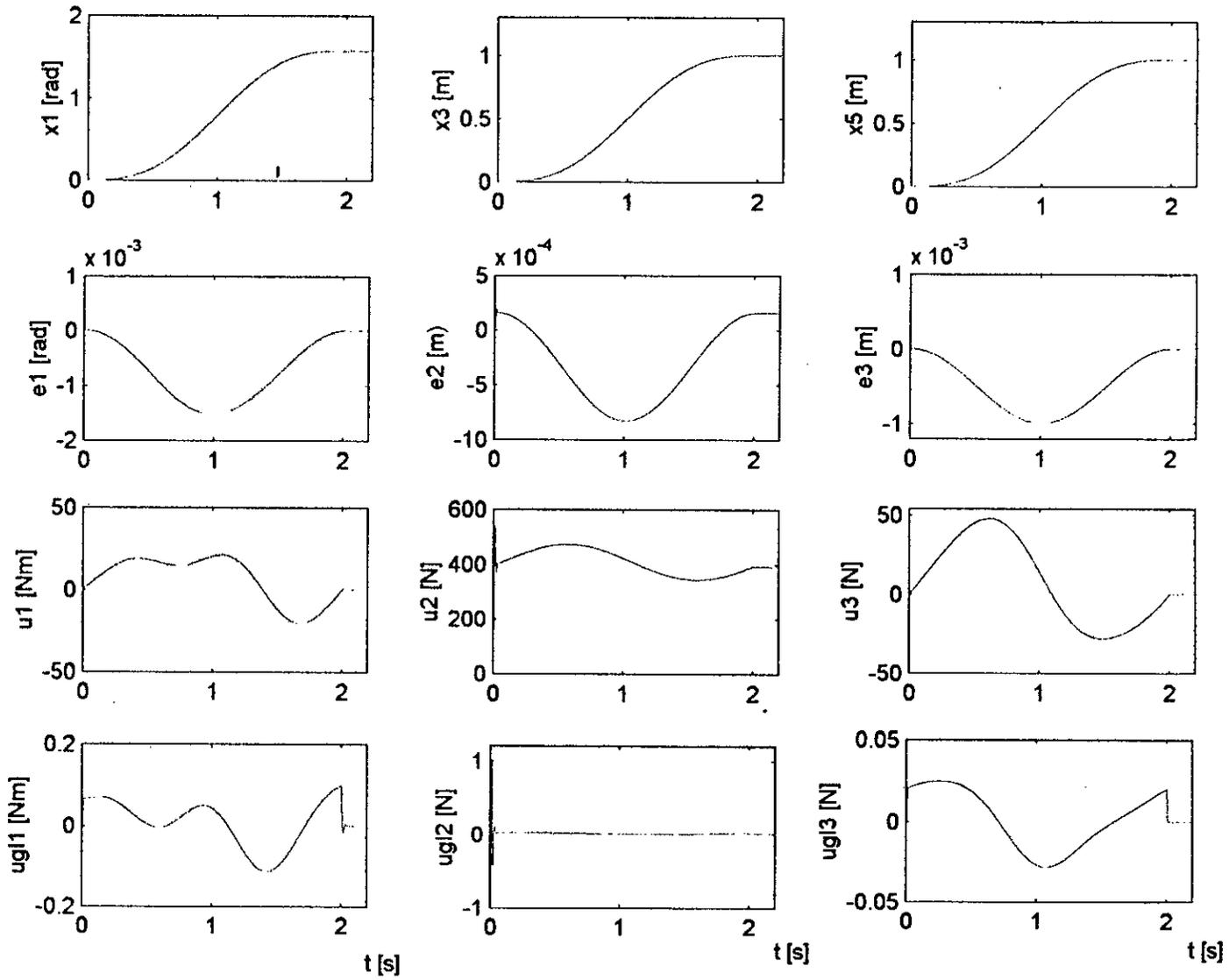


Figure 4.2: Régime de poursuite du robot en charge ($m_0 = 10$ Kg)

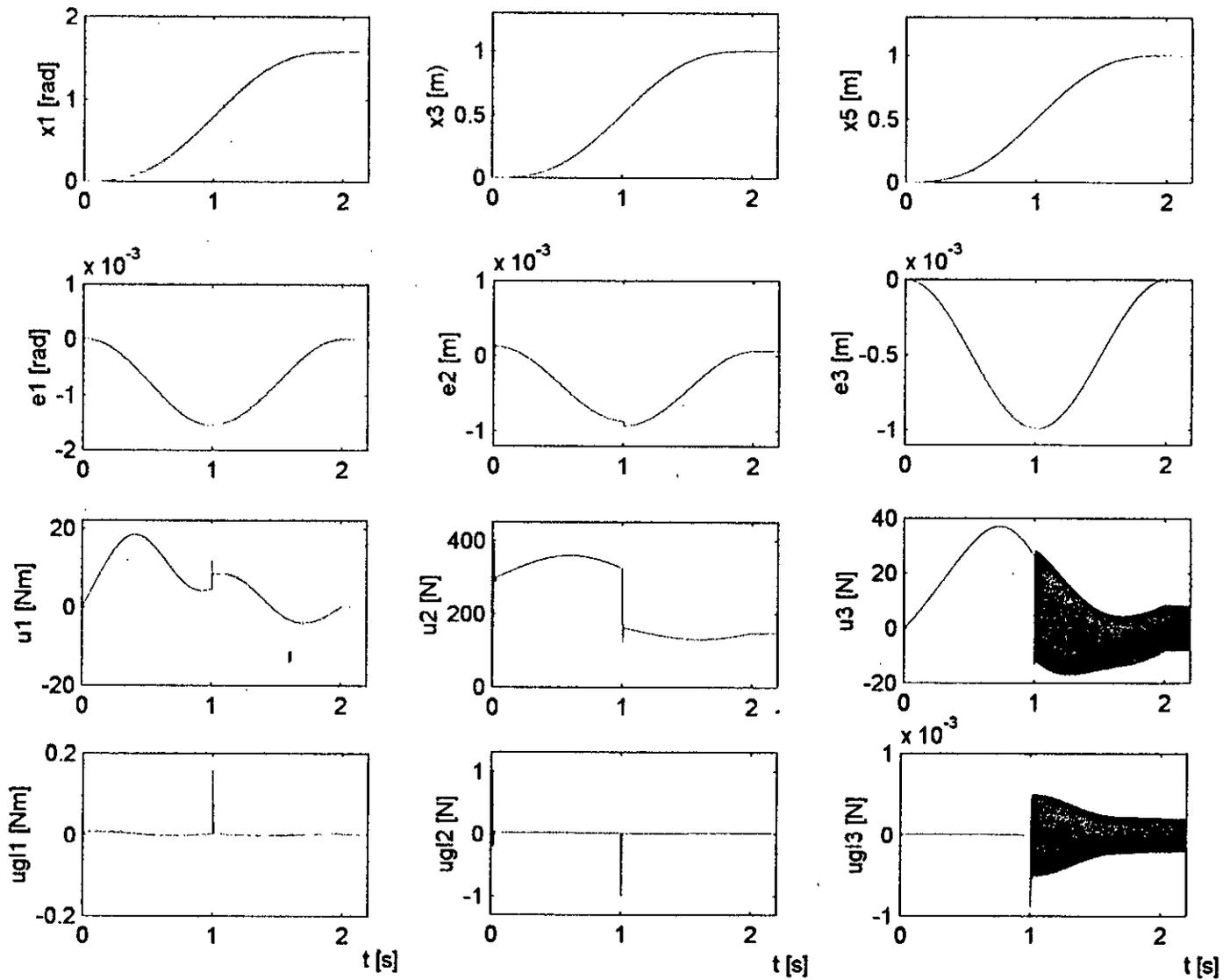


Figure 4.3: Régime de poursuite du robot sans charge avec variations paramétriques en diminution

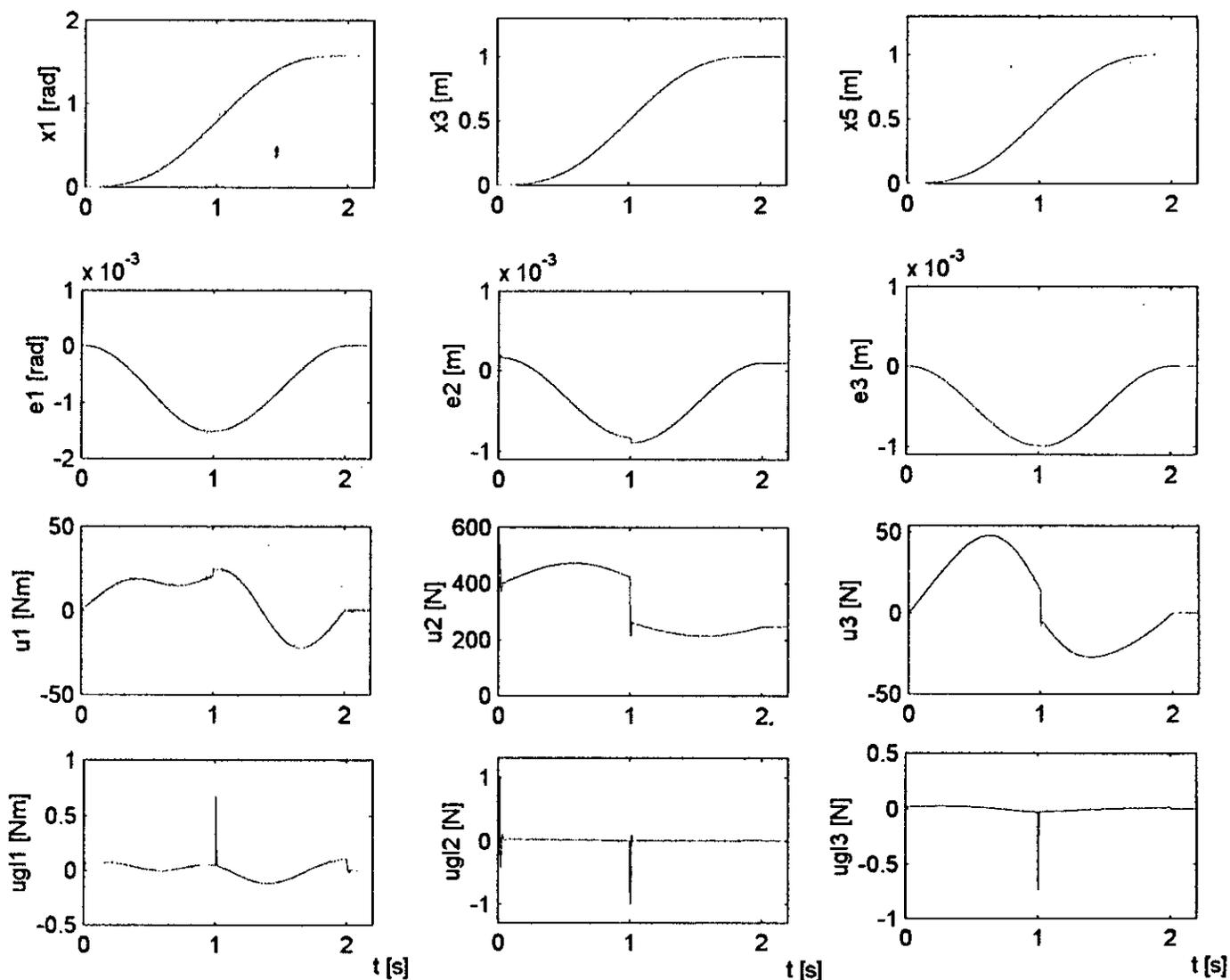


Figure 4.4: Régime de poursuite du robot en charge avec variations paramétriques en diminution.

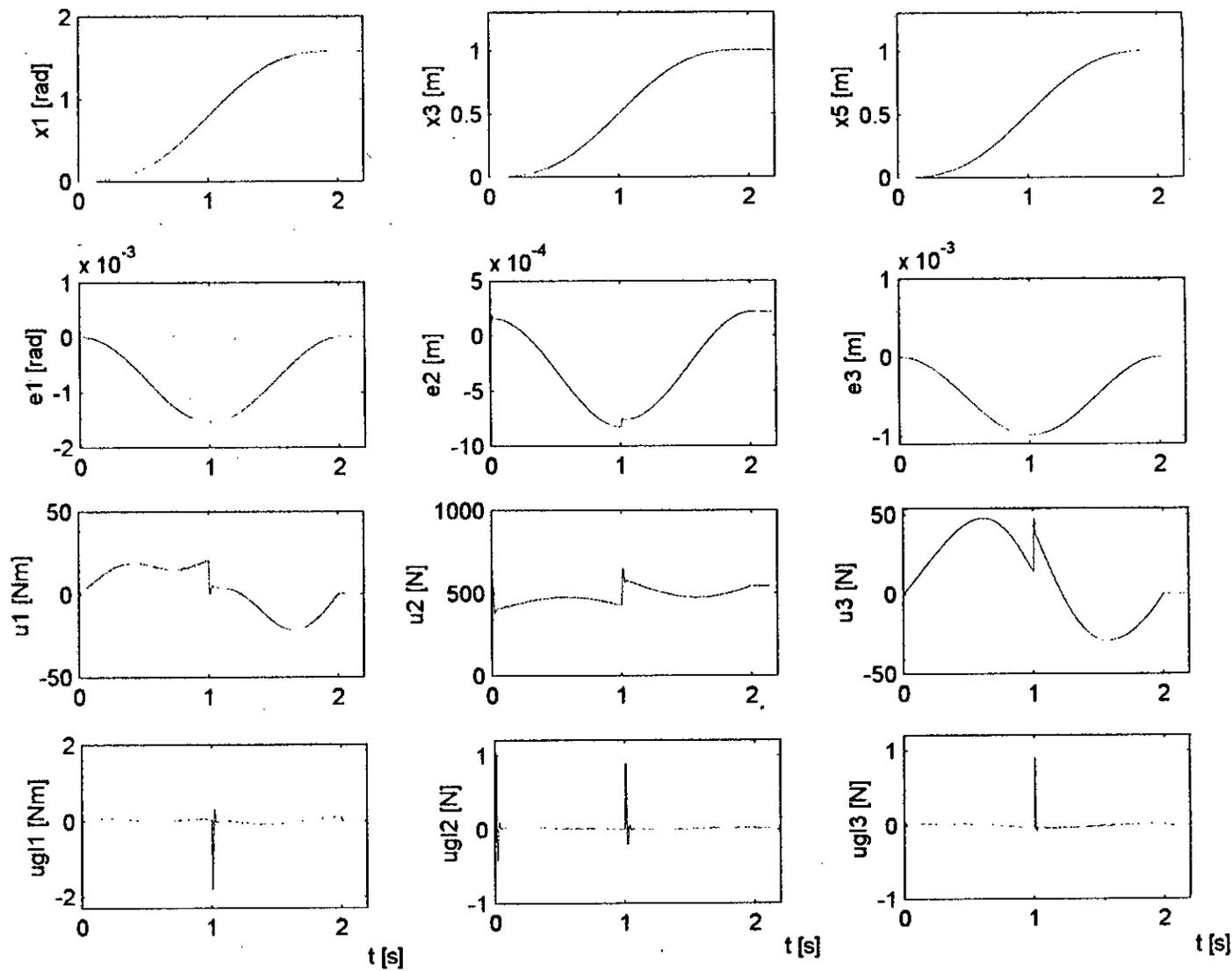


Figure 4.5: Régime de poursuite du robot en charge avec variations paramétriques en augmentation

Interprétation des résultats :

En régime de poursuite sans charge (figure 4.1), on constate une situation idéale, une erreur faible qui atteint un maximum très tolérable à $t = 1s$, le moment le plus critique vu de la trajectoire car la vitesse est à son maximum, les commandes sont lisses ce qui en pratique permet à l'actionneur de les interpréter facilement. Le terme de glissement est présent avec de faibles valeurs au niveau de la première et la troisième articulation, alors qu'il est pratiquement nul à la deuxième articulation.

Une fois le robot est chargé (figure 4.2), il apparaît une augmentation du couple et de la force de la première et la troisième articulation (car la commande doit développer ces derniers pour équilibrer le système), ainsi qu'une légère augmentation du terme de glissement.

En appliquant une variation paramétrique représentée par une diminution de 50 %, le robot étant à vide (figure 4.3), la commande de la troisième articulation présente un chattering, cela n'affecte pas la robustesse (bonne poursuite avec des erreurs dans le même intervalle), sauf que la commande est de forme différente moins intéressante que la lisse. Le terme de glissement apporte sa contribution au moment de l'application de la perturbation, les erreurs de poursuite restent dans le même intervalle.

Le système non chargé est robuste, ainsi que le système chargé (figure 4.4), dans ce cas avec une variation paramétrique en diminution, on ne remarque autre qu'une combinaison entre les simulations précédentes ; ce qui veut dire : une augmentation de la commande plus une intervention instantané du terme de glissement, il l'on est de même pour une augmentation paramétrique de 50 % des paramètres du robot (figure 4.5).

Conclusion :

La dynamique du système est mise tout d'abord sous la forme directe. La loi de commande est déterminée en se basant sur le principe de la commande linéarisante, comme les fonctions définissant le système sont inconnues, elles sont approximées par des systèmes flous de Sugeno d'ordre un, les lois d'adaptation des paramètres des systèmes flous sont déduites en utilisant à nouveau la stabilité au sens de Lyapunov. La commande a permis d'obtenir un régime de poursuite avec une erreur satisfaisante (on note que la valeur maximale des erreurs n'atteint pas $2 \cdot 10^{-3}$), de plus les essais avec variations paramétriques ont révélé que la commande s'adapte rapidement aux variations paramétriques du système du fait que les erreurs restent dans le même intervalle (même ordre de grandeur). Néanmoins, le test du robot à vide en présence des variables paramétriques montre que les commandes deviennent discontinues du fait que les paramètres évoluent rapidement entre une valeur minimale θ_{\min} et une valeur maximale θ_{\max} .

Chapitre V

COMMANDE FLOUE ADAPTATIVE DIRECTE LINÉARISANTE (TYPE 2)

V.1 COMMANDE FLOUE ADAPTATIVE DIRECTE DÉCENTRALISÉE LINÉARISANTE :

V.1.1 Introduction :

Cette méthode est basée sur une structure de régulateur adopté du domaine des réseaux de neurones. Ainsi, la commande floue adaptative stable est applicable à plusieurs classes de système non-linéaire.

Le régulateur flou adaptatif peut être vu comme un réseau obtenu par la combinaison entre la représentation des connaissances linguistiques d'un régulateur flou, et la capacité d'apprentissage des réseaux de neurones [2].

La conception du régulateur demande une petite information du système à priori ; si la connaissance experte complémentaire est disponible, elle peut facilement être utilisée pour la création de la base des règles initiales. La stabilité et la convergence de la sortie vers la référence sont garanties mathématiquement.

Pour chaque variable d'état (position et vitesse) de chaque articulation, on associe trois fonctions d'appartenance, la base des règles dans ce cas est constitué de la combinaison de tous les ensembles flous, ce qui correspond à neuf règles.

V.1.2 Formulation du problème :

Soit la classe des systèmes non linéaires SISO de la forme [2]:

$$\begin{cases} \dot{x} = F(x) + G(x).u \\ y = h(x) \end{cases} \quad (5.1.1)$$

tels que :

$$x^T = [x_1 \quad x_2 \quad \dots \quad x_n]$$

\dot{x} : dérivée temporelle de x .

y : sortie du système.

La dynamique du robot à structure cylindrique mise sous la forme (5.1.1) est présentée dans l'ANNEXE 3 (Ann3.3).

On considère les dérivées temporelles de la sortie y :

$$\dot{y} = L_f h(x) \tag{5.1.2}$$

$$\ddot{y} = L_f^2 h(x) \tag{5.1.3}$$

$$\vdots$$

$$y^{(n)} = L_f^n h(x) + L_g L_f^{n-1} h(x) \cdot u \tag{5.1.4}$$

tels que :

$$L_f h(x) = \frac{\partial h}{\partial x} F(x)$$

$$L_f^{i+1} h(x) = L_f [L_f^i h(x)] = \frac{\partial L_f^i h(x)}{\partial x} F(x)$$

$$L_g L_f^i h(x) = \frac{\partial L_f^i h(x)}{\partial x} G(x)$$

L'équation (5.1.4) est le modèle inverse du système (4.1) dans le cas où $y = x$ avec $L_g L_f^{n-1} h(x) \neq 0$.

On pose :

$$L_f^n h(x) = f(x)$$

$$L_g L_f^{n-1} h(x) = g(x)$$

d'où :

$$y^{(n)} = f(x) + g(x) \cdot u \tag{5.1.5}$$

Si $f(x)$ et $g(x)$ sont connues, alors la commande optimale sera donnée par :

$$u^* = \frac{y_d^{(n)} - k_1 e - \dots - k_n e^{(n-1)} - f(x)}{g(x)} \tag{5.1.6a}$$

En notant que y_d est la sortie désirée et e représente l'erreur tel que $e = y - y_d$; les k_i sont des constantes positives.

En remplaçant l'expression de la commande dans (5.1.5), la dynamique de la sortie sera :

$$y^{(n)} = y_d^{(n)} - k_1 e - \dots - k_n e^{(n-1)}$$

d'où :

$$e^{(n)} = -k_1 e - \dots - k_n e^{(n-1)}$$

Cette loi de commande mène à une équation différentielle de l'erreur dont le polynôme caractéristique est : $s^n + k_n s^{n-1} + \dots + k_2 s + k_1 = 0$. Si les coefficients (k_1, \dots, k_n) sont choisis tels que ce polynôme soit Hurwitz, par conséquent l'erreur e tend exponentiellement vers zéro.

Du fait que les fonctions $f(x)$ et $g(x)$ ne sont pas connues, on les remplace par leurs estimées floues $\hat{f}(x)$ et $\hat{g}(x)$, on obtient alors la commande réelle :

$$u = \frac{y_d^{(n)} - k^T e - \hat{f}(x)}{\hat{g}(x)} \quad (5.1.6b)$$

avec :

$$k^T = [k_1 \dots k_n] ; \quad \underline{e}^T = [e \quad \dot{e} \dots e^{(n-1)}]$$

V.1.3 Adaptation par logique floue :

La commande idéale présentée précédemment dans (5.1.6a) est valable pour les fonctions $f(x)$ et $g(x)$ sont connues exactement, ce qui n'est pas souvent le cas, donc il est possible de les estimer par des systèmes flous adaptatifs, tels que :

$$\begin{aligned} \hat{f}(x) &= W^T \cdot \hat{\theta}_f \\ \hat{g}(x) &= W^T \cdot \hat{\theta}_g \end{aligned} \quad (5.1.7)$$

La commande optimale u^* peut être définie en fonction des estimés optimaux $f^*(x)$ et $g^*(x)$:

$$u^* = \frac{y_d^{(n)} - k^T e - f^*(x)}{g^*(x)} \quad (5.1.8)$$

telles que :

$$\begin{aligned} f^*(x) &= W^T \cdot \theta_f^* \\ g^*(x) &= W^T \cdot \theta_g^* \end{aligned} \quad (5.1.9)$$

où θ_f^* et θ_g^* sont les paramètres optimaux d'adaptation.

V.1.4 Dynamique de l'erreur :

L'équation différentielle de l'erreur est :

$$\begin{aligned}
e^{(n)} &= y^{(n)} - y_d^{(n)} \\
&= f^*(x) + g^*(x) \cdot u - y_d^{(n)} \\
&= f^*(x) + g^*(x) \cdot u - \hat{g}(x) \cdot u + \hat{g}(x) \cdot u - y_d^{(n)}
\end{aligned} \tag{5.1.10}$$

D'après l'équation (5.1.6b) : $\hat{g}(x) \cdot u = y_d^{(n)} - k^T \underline{e} - \hat{f}(x)$ et en tenant compte des relations (5.1.7), l'expression de $e^{(n)}$ devient ainsi :

$$e^{(n)} = -k^T \underline{e} + W^T (\theta_f^* - \hat{\theta}_f) + W^T (\theta_g^* - \hat{\theta}_g) \cdot u \tag{5.1.11}$$

On définit l'erreur paramétrique comme suit :

$$\begin{aligned}
\tilde{\theta}_f &= \theta_f^* - \hat{\theta}_f \\
\tilde{\theta}_g &= \theta_g^* - \hat{\theta}_g
\end{aligned} \tag{5.1.12}$$

d'où :

$$e^{(n)} = -k^T \underline{e} + W^T \tilde{\theta}_f + W^T \tilde{\theta}_g u \tag{5.1.13}$$

On s'intéresse à la sortie S qui est l'erreur de poursuite filtrée définie par :

$$S = C^T \underline{e} \tag{5.1.14}$$

ainsi on obtient le système suivant de la dynamique de l'erreur de poursuite :

$$\begin{cases} \dot{e}^{(n)} = -k^T \underline{e} + W^T \tilde{\theta}_f + W^T \tilde{\theta}_g u \\ S = C^T \underline{e} \end{cases} \tag{5.1.15}$$

avec :

$$C^T = [p_1 \quad p_2 \quad \dots \quad p_n]$$

Les p_i sont des coefficients définis plus tard.

En posant :

$$E_X = W^T \tilde{\theta}_f + W^T \tilde{\theta}_g u \tag{5.1.16}$$

Le système suivant (5.1.15) peut encore s'écrire sous la forme :

$$\begin{cases} \dot{\underline{e}} = A \underline{e} + B E_X \\ S = C^T \underline{e} \end{cases} \tag{5.1.17}$$

avec :

$$\underline{e}^T = [e \quad \dot{e} \dots e^{(n-1)}] \quad ; \quad \underline{\dot{e}}^T = [\dot{e} \quad \ddot{e} \dots e^{(n)}]$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -k_1 & -k_2 & \dots & \dots & \dots & \dots & -k_n \end{bmatrix} ; \quad B = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

Vu la forme de la matrice A (forme compagne de commande), il existe toujours des matrices symétriques définies positives P et Q qui vérifient l'équation de Lyapunov :

$$\begin{cases} A^T P + P A = -Q \\ C^T = B^T P \end{cases} \quad (5.1.18)$$

V.1.5 Analyse de la stabilité :

Proposition :

Soit le système (5.1.1) dont le modèle dynamique de l'erreur est donné par (5.1.17), et si la commande telle que : $u = \frac{y_d^{(n)} - k^T \underline{e} - \hat{f}(x)}{\hat{g}(x)}$ où les fonctions $\hat{f}(x)$ et $\hat{g}(x)$ sont estimées par des systèmes flous dont les paramètres sont adaptés par :

$$\begin{cases} \dot{\hat{\theta}}_f = \gamma W S \\ \dot{\hat{\theta}}_g = \gamma W S u \end{cases} \quad (5.1.19)$$

Par conséquent, l'erreur de poursuite tend asymptotiquement vers zéro et les paramètres restent bornés.

Preuve :

Soit la fonction de Lyapunov suivante :

$$V = \frac{1}{2} \underline{e}^T P \underline{e} + \frac{1}{2\gamma} \tilde{\theta}_f^T \tilde{\theta}_f + \frac{1}{2\gamma} \tilde{\theta}_g^T \tilde{\theta}_g \quad (5.1.20)$$

Sa dérivée est donnée par :

$$\dot{V} = \frac{1}{2}(\underline{\dot{e}}^T P \underline{e} + \underline{e}^T P \underline{\dot{e}}) + \frac{1}{\gamma} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\gamma} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g \quad (5.1.21)$$

A partir des relations (5.1.15), (5.1.16) et (5.1.17), \dot{V} s'écrit sous la forme :

$$\dot{V} = -\underline{e}^T Q \underline{e} + \underline{e}^T P B W_1^T \tilde{\theta}_f + \underline{e}^T P B W_2^T \tilde{\theta}_g u + \frac{1}{\gamma} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\gamma} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g \quad (5.1.22)$$

Ou encore :

$$\dot{V} = -\underline{e}^T Q \underline{e} + S^T W^T \tilde{\theta}_f + S^T W^T \tilde{\theta}_g u + \frac{1}{\gamma} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\gamma} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g \quad (5.1.23)$$

Or :

$$\begin{cases} \dot{\tilde{\theta}}_f = -\dot{\hat{\theta}}_f \\ \dot{\tilde{\theta}}_g = -\dot{\hat{\theta}}_g \end{cases}$$

donc :

$$\dot{V} = -\underline{e}^T Q \underline{e} + S^T W^T \tilde{\theta}_f + S^T W^T \tilde{\theta}_g u - \frac{1}{\gamma} \tilde{\theta}_f^T \dot{\hat{\theta}}_f - \frac{1}{\gamma} \tilde{\theta}_g^T \dot{\hat{\theta}}_g \quad (5.1.24)$$

En remplaçant les paramètres d'adaptation $\dot{\hat{\theta}}_f$ et $\dot{\hat{\theta}}_g$ par leurs expressions (5.1.19), \dot{V} est réduit à :

$$\dot{V} = -\underline{e}^T Q \underline{e} < 0 \quad \forall \underline{e} \neq 0 \quad (5.1.25)$$

Ce qui fait que \underline{e} tend asymptotiquement vers 0 quand y tend vers y_d et $\tilde{\theta}_f$ et $\tilde{\theta}_g$ restent bornés.

V.1.6 Résultats de la simulation :

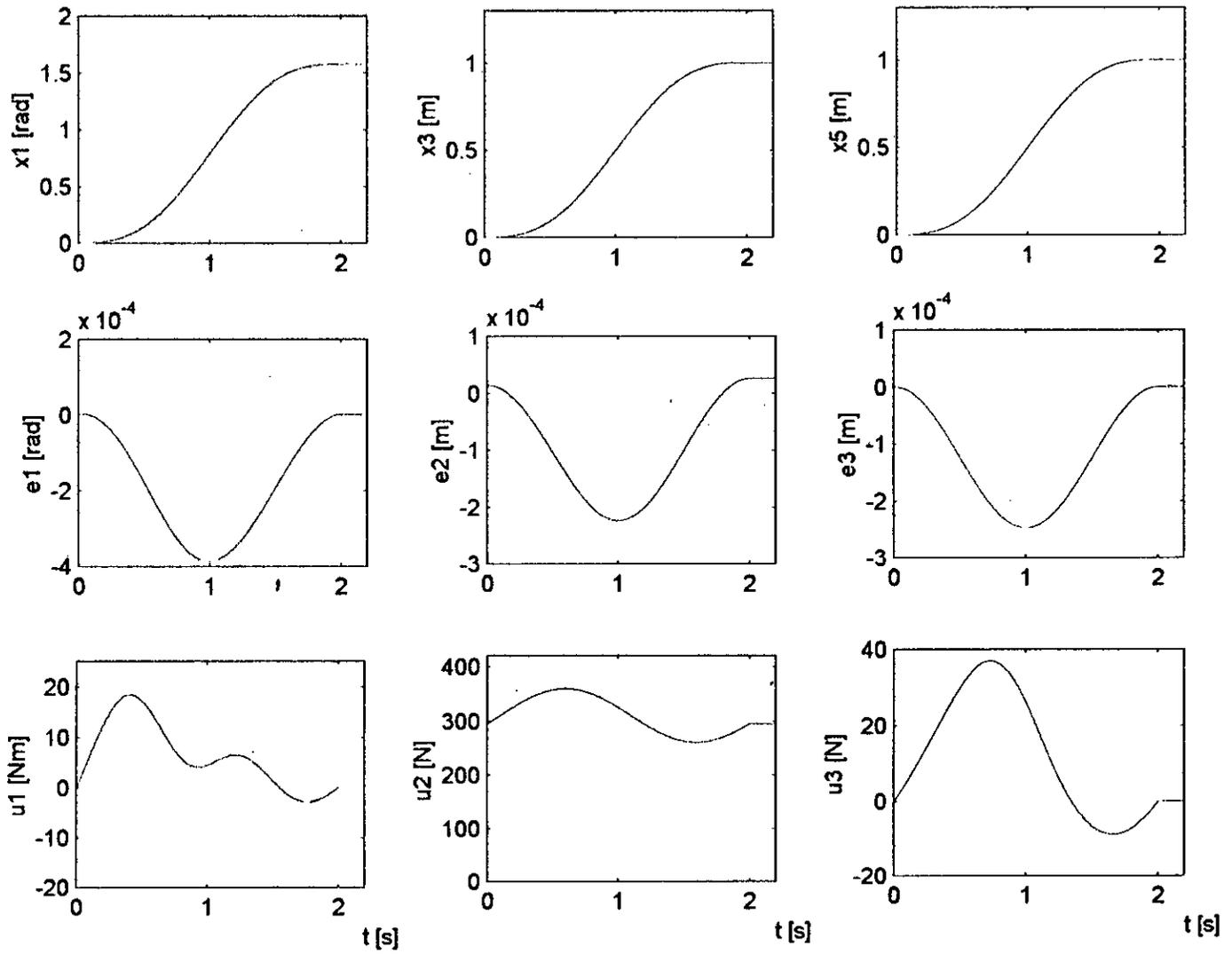


Figure 5.1.1 : Régime de poursuite du robot sans charge ($m_0 = 0$)

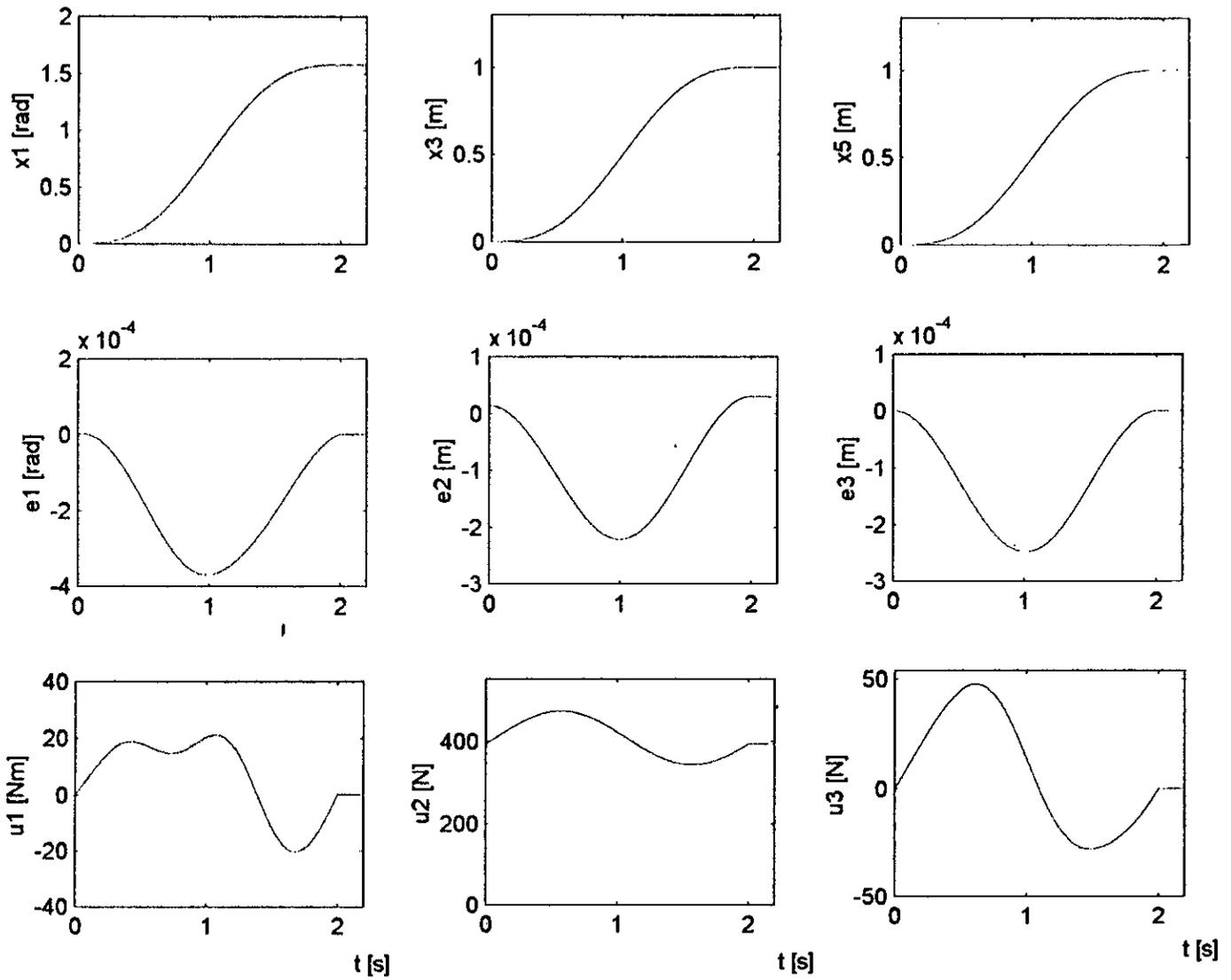


Figure 5.1.2 : Régime de poursuite du robot en charge ($m_0 = 10$ Kg)

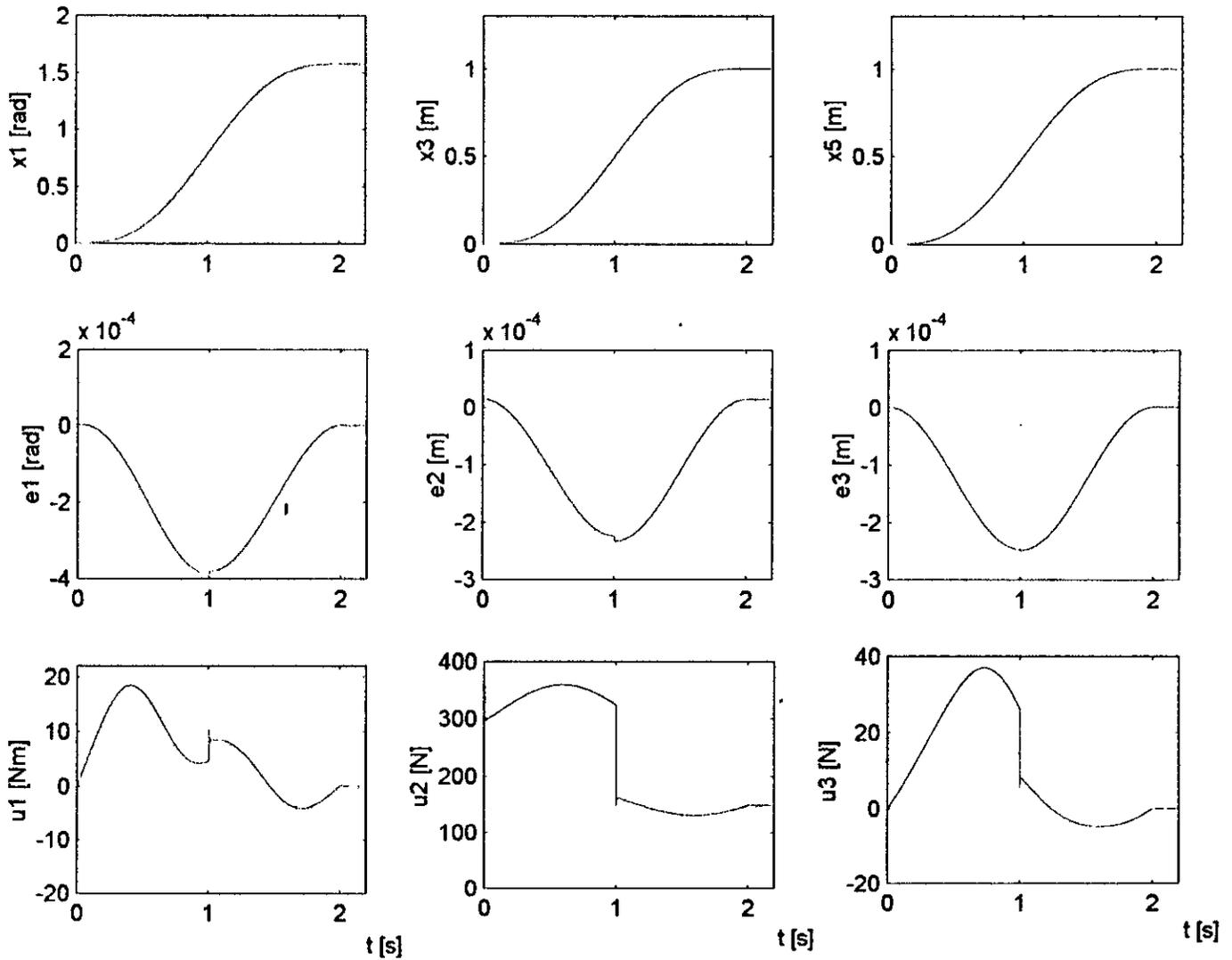


Figure 5.1.3 : Régime de poursuite du robot sans charge avec variations paramétriques en diminution

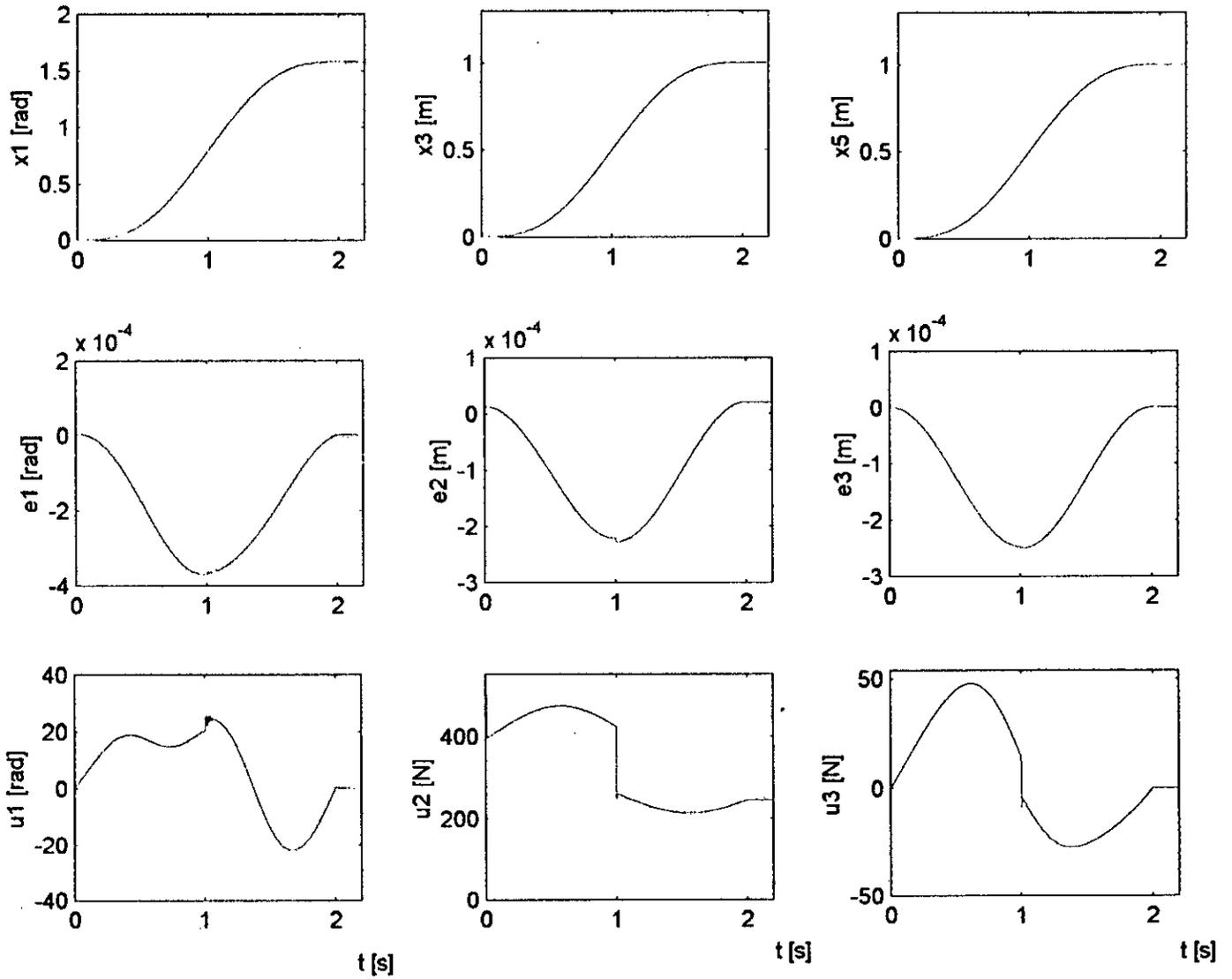


Figure 5.1.4 : Régime de poursuite du robot en charge avec variations paramétriques en diminution

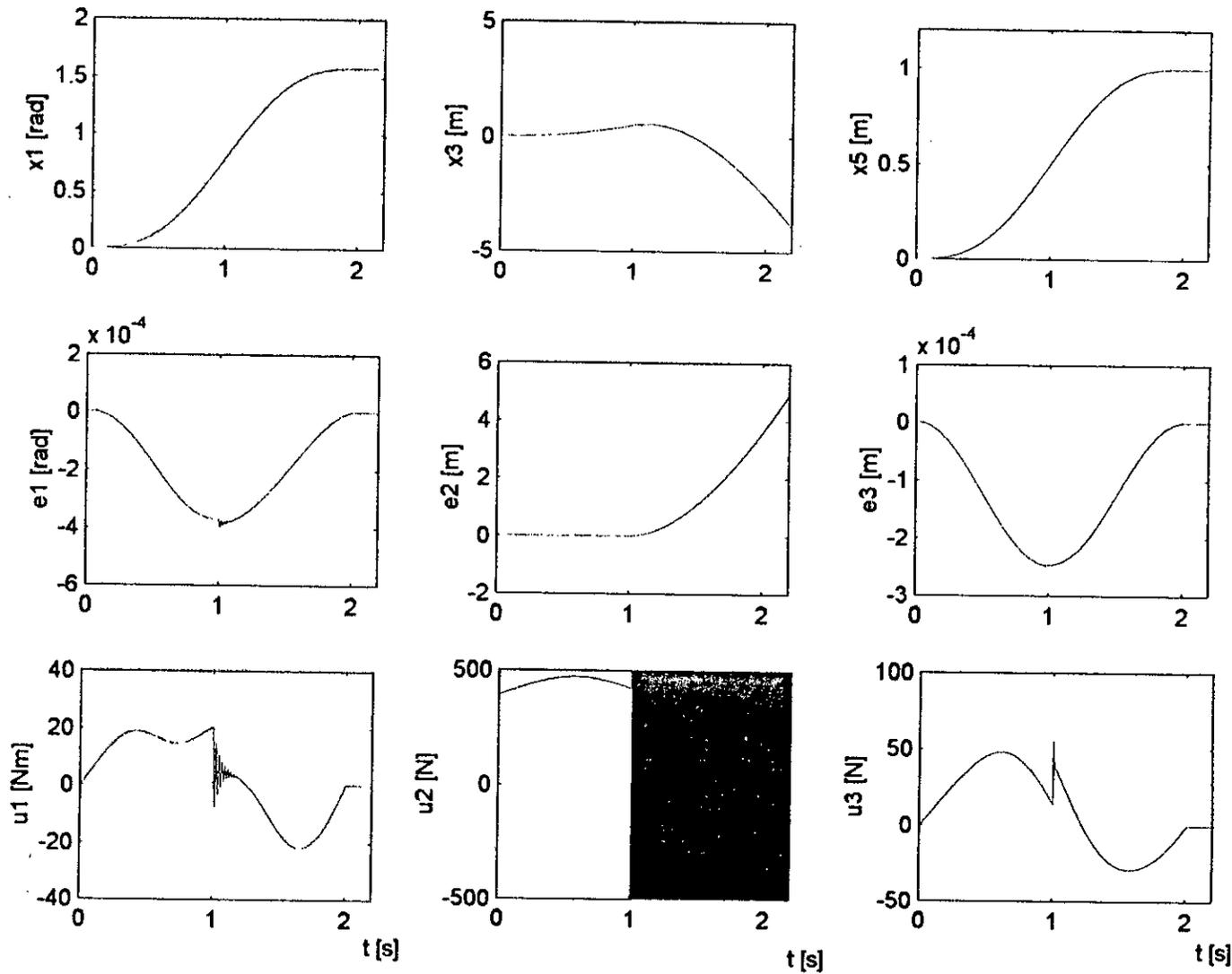


Figure 5.1.5 : Régime de poursuite du robot en charge avec variations paramétriques en augmentation

Interprétation des résultats :

En régime de poursuite à vide (figure 5.1.1), les erreurs de poursuite convergent vers zéro et les commandes restent pratiquement lisses.

En régime de poursuite en charge (figure 5.1.2), les erreurs de poursuite évoluent pratiquement de la même manière que dans le fonctionnement à vide. Seules les commandes augmentent pour compenser les effets de la charge.

L'application des variations paramétriques (dans le sens d'une diminution de 50 %) en présence ou en absence de la charge n'affecte pas le régime de poursuite (figure 5.1.3 et 5.1.4) du fait que les erreurs de poursuite restent sensiblement de même grandeur que dans l'essai à vide. Cependant, il apparaît une discontinuité sur les commandes à l'instant $t = 1s$ où sont appliquées ces variations.

Pour le cas d'une augmentation paramétrique de 50 % (le robot étant chargé (figure 5.1.5)), le régime de poursuite au niveau de la deuxième articulation ne s'effectue pas et l'erreur de poursuite correspondante diverge, donc cette commande n'est pas robuste pour une telle augmentation paramétrique, mais pour les autres essais, elle a montré de bonnes capacités d'adaptation.

V.2 COMMANDE FLOUE ADAPTATIVE DIRECTE DÉCENTRALISÉE LINÉARISANTE AVEC ADAPTATION DES FONCTIONS D'APPARTENANCE :

V.2.1 Introduction

Dans la commande précédente, on a utilisé la combinaison des ensembles flous, ce qui pouvait mener à un nombre de règles pléthorique dans le cas d'une augmentation des variables d'état. Ce qui nous permet par la suite de se contenter d'une règle pour la première articulation, une règle pour la deuxième articulation et trois règles pour la troisième articulation tout en faisant une adaptation des centres et des variances des fonctions d'appartenance. Cet algorithme d'apprentissage est fait on-line et dérivé à partir de l'analyse de stabilité de Lyapunov.

V.2.2 Position du problème

Soit la classe des systèmes non linéaires SISO de la forme [2] :

$$\begin{cases} \dot{x} = F(x) + G(x) \cdot u \\ y = h(x) \end{cases} \quad (5.2.1)$$

Si la sortie y admet un degrés relatif d'ordre n , alors :

$$y^{(n)} = f(x) + g(x) \cdot u \quad (5.2.2)$$

avec :

$$f(x) = L_f^n h(x) \quad , \quad g(x) = L_g L_f^{n-1} h(x).$$

Si $f(x)$ et $g(x)$ sont connues, alors la loi de commande optimale est donnée par :

$$u^* = \frac{y_d^{(n)} - k^T \underline{e} - f(x)}{g(x)} \quad \text{et } g(x) \neq 0 \quad (5.2.3)$$

avec :

$$\begin{aligned} e &= y - y_d \\ x^T &= [x_1 \quad x_2 \quad \dots \quad x_n] \\ \underline{e}^T &= [e \quad \dot{e} \quad \dots \quad e^{(n-1)}] \\ k^T &= [k_1 \quad k_2 \quad \dots \quad k_n] \end{aligned}$$

La dynamique de la sortie y est donc :

$$y^{(n)} = y_d^{(n)} - k^T \underline{e} \quad (5.2.4)$$

d'où :

$$e^{(n)} = -k^T \underline{e} \quad (5.2.5)$$

Si le vecteur k^T est choisi de façon à ce que le polynôme:

$$s^n + k_n s^{n-1} + k_{n-1} s^{n-2} + \dots + k_2 s + k_1 = 0$$

soit un polynôme de Hurwitz (les racines du polynôme caractéristique soient à partie réelle négative), alors $\lim_{t \rightarrow \infty} e \rightarrow 0$ exponentiellement.

Et comme $f(x)$ et $g(x)$ ne sont connues, on les remplace par leurs estimées floues $\hat{f}(x)$ et $\hat{g}(x)$. Par conséquent, la commande proposée est donnée par :

$$u = u_0 + u_{gl} \quad (5.2.6)$$

avec :

$$u_0 = \frac{y_d^{(n)} - k^T \underline{e} - \hat{f}(x)}{\hat{g}(x)} \quad (5.2.7)$$

où u_{gl} est un terme de glissement.

V.2.3 Approximation par un système flou :

On définit les fonctions $f(x)$ et $g(x)$ en fonction des paramètres optimaux θ_f^* , θ_g^* et leurs estimées $\hat{f}(x)$ et $\hat{g}(x)$ en fonction des paramètres $\hat{\theta}_f$ et $\hat{\theta}_g$ sous la forme :

$$\begin{aligned} f(x) &= W^T \theta_f^* + \varepsilon_f, & g(x) &= W^T \theta_g^* + \varepsilon_g \\ \hat{f}(x) &= \hat{W}^T \hat{\theta}_f, & \hat{g}(x) &= \hat{W}^T \hat{\theta}_g \end{aligned} \quad (5.2.8)$$

Où le vecteur W est fonction des centres et des variances optimaux $(\hat{c}, \hat{\sigma})$ des fonctions d'appartenance et le vecteur estimé \hat{W} dépend de $\hat{c}(t), \hat{\sigma}(t)$ [3].

$$\hat{W} = f(\hat{c}, \hat{\sigma}), \quad W = f(c^*, \sigma^*).$$

On effectue un développement en série de Taylor de W autour de \hat{W} , il vient :

$$W = \hat{W} + W'_c \cdot \tilde{c} + W'_\sigma \cdot \tilde{\sigma} + o^2(\tilde{c}, \tilde{\sigma}) \quad (5.2.9)$$

avec :

$$\begin{aligned} \tilde{c} &= c^* - \hat{c} & , & \quad \tilde{\sigma} = \sigma^* - \hat{\sigma} \\ W'_c &= \frac{\partial W}{\partial c} \Big|_{c=\hat{c}} & , & \quad W'_\sigma = \frac{\partial W}{\partial \sigma} \Big|_{\sigma=\hat{\sigma}} \end{aligned} \quad (5.2.10)$$

En remplaçant W par son expression dans $f(x)$ et $g(x)$, il vient :

$$f(x) = \hat{W}^T \theta_f^* + \tilde{c}^T \hat{W}'^T \theta_f^* + \tilde{\sigma}^T \hat{W}_\sigma'^T \theta_f^* + o^2(\tilde{c}, \tilde{\sigma}) \theta_f^* + \varepsilon_f \quad (5.2.11)$$

$$g(x) = \hat{W}^T \theta_g^* + \tilde{c}^T \hat{W}'^T \theta_g^* + \tilde{\sigma}^T \hat{W}_\sigma'^T \theta_g^* + o^2(\tilde{c}, \tilde{\sigma}) \theta_g^* + \varepsilon_g \quad (5.2.12)$$

Or :

$$\theta_f^* = \tilde{\theta}_f + \hat{\theta}_f \quad \text{et} \quad \theta_g^* = \tilde{\theta}_g + \hat{\theta}_g$$

Par conséquent, $f(x)$ et $g(x)$ peuvent encore s'écrire sous la forme :

$$f(x) = \hat{W}^T \tilde{\theta}_f + \hat{W}^T \hat{\theta}_f + \tilde{c}^T \hat{W}'^T \hat{\theta}_f + \tilde{\sigma}^T \hat{W}_\sigma'^T \hat{\theta}_f + \tilde{\omega}_f + \varepsilon_f \quad (5.2.13)$$

$$g(x) = \hat{W}^T \tilde{\theta}_g + \hat{W}^T \hat{\theta}_g + \tilde{c}^T \hat{W}'^T \hat{\theta}_g + \tilde{\sigma}^T \hat{W}_\sigma'^T \hat{\theta}_g + \tilde{\omega}_g + \varepsilon_g \quad (5.2.14)$$

avec :

$$\tilde{\omega}_f = \tilde{c}^T \hat{W}'^T \tilde{\theta}_f + \tilde{\sigma}^T \hat{W}_\sigma'^T \tilde{\theta}_f + o^2(\tilde{c}, \tilde{\sigma}) \theta_f$$

$$\tilde{\omega}_g = \tilde{c}^T \hat{W}'^T \tilde{\theta}_g + \tilde{\sigma}^T \hat{W}_\sigma'^T \tilde{\theta}_g + o^2(\tilde{c}, \tilde{\sigma}) \theta_g$$

V.2.4 Dynamique de l'erreur :

La dynamique de l'erreur de poursuite est donnée par :

$$\begin{aligned} e^{(n)} &= y^{(n)} - y_d^{(n)} \\ &= f(x) + g(x) \cdot u - y_d^{(n)} \\ &= f(x) + g(x)(u_0 + u_{gl}) + \hat{f}(x) - \hat{f}(x) + \hat{g}(x) \cdot u_0 - \hat{g}(x) \cdot u_0 - y_d^{(n)} \\ &= (f(x) - \hat{f}(x)) + (g(x) - \hat{g}(x)) \cdot u_0 + \hat{f}(x) + \hat{g}(x) \cdot u_0 - y_d^{(n)} + g(x) \cdot u_{gl} \end{aligned} \quad (5.2.15)$$

En tenant compte de l'expression de u_0 donnée par (5.2.7), la dynamique de l'erreur (5.2.15) se réduit à :

$$e^{(n)} = -k^T \underline{e} + \tilde{f}(x) + \tilde{g}(x) \cdot u_0 + g(x) \cdot u_{gl} \quad (5.2.16)$$

avec :

$$\begin{aligned}\tilde{f}(x) &= f(x) - \hat{f}(x) \\ \tilde{g}(x) &= g(x) - \hat{g}(x)\end{aligned}$$

On peut encore écrire $e^{(n)}$ sous la forme :

$$\dot{\underline{e}} = A\underline{e} + B E_X \quad (5.2.17)$$

où :

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ -k_1 & -k_2 & \dots & \dots & \dots & \dots & -k_n \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$E_X = \tilde{f}(x) + \tilde{g}(x) \cdot u_0 + g(x) \cdot u_g$$

Or :

$$\tilde{f}(x) = f(x) - \hat{f}(x) = \hat{W}^T \tilde{\theta}_f + \tilde{c}^T \hat{W}_c'^T \hat{\theta}_f + \tilde{\sigma}^T \hat{W}_\sigma'^T \hat{\theta}_f + \tilde{\omega}_f + \varepsilon_f \quad (5.2.18)$$

$$\tilde{g}(x) = g(x) - \hat{g}(x) = \hat{W}^T \tilde{\theta}_g + \tilde{c}^T \hat{W}_c'^T \hat{\theta}_g + \tilde{\sigma}^T \hat{W}_\sigma'^T \hat{\theta}_g + \tilde{\omega}_g + \varepsilon_g \quad (5.2.19)$$

d'où :

$$E_X = \hat{W}^T \tilde{\theta}_f + \hat{W}^T \tilde{\theta}_g \cdot u_0 + \tilde{c}^T W_c'^T (\hat{\theta}_f + \hat{\theta}_g \cdot u_0) + \tilde{\sigma}^T W_\sigma'^T (\hat{\theta}_f + \hat{\theta}_g \cdot u_0) + \tilde{\omega}_f + \tilde{\omega}_g \cdot u_0 + \varepsilon_f + \varepsilon_g \cdot u_0 \quad (5.2.20)$$

On s'intéresse à la sortie $S = C^T \underline{e}$ qui est l'erreur filtrée, on obtient donc le système :

$$\begin{cases} \dot{\underline{e}} = A\underline{e} + B E_X \\ S = C^T \underline{e} \end{cases} \quad (5.2.21)$$

Vu la forme de A , par conséquent, ils existent des matrices symétriques définies positives P et Q qui vérifient l'équation de Lyapunov telles que :

$$\begin{cases} A^T P + P A = -Q \\ C^T = B^T P \end{cases} \quad (5.2.22)$$

Donc la sortie S peut être écrite sous la forme :

$$S = B^T P \underline{e} \quad (5.2.23)$$

V.2.5 Analyse de la stabilité :

Proposition :

Soit le système (5.2.1) dont le modèle dynamique de l'erreur de poursuite est donné par (5.2.21). Si la loi de commande est de la forme (5.2.6 et 5.2.7) où les fonctions $\hat{f}(x)$ et $\hat{g}(x)$ sont estimées par des systèmes flous dont les paramètres sont adaptés comme suit :

$$\begin{aligned}\dot{\hat{\theta}}_f &= \gamma_1 \cdot \hat{W} \cdot S \\ \dot{\hat{\theta}}_g &= \gamma_2 \cdot \hat{W} \cdot S \cdot u_0 \\ \dot{\hat{c}} &= W_c'^T (\hat{\theta}_f + \hat{\theta}_g \cdot u_0) S \\ \dot{\hat{\sigma}} &= W_\sigma'^T (\hat{\theta}_f + \hat{\theta}_g \cdot u_0) S\end{aligned}\quad (5.2.24)$$

alors, l'erreur de poursuite e tend asymptotiquement vers zéro et les erreurs paramétriques $\tilde{\theta}_f, \tilde{\theta}_g, \tilde{c}$ et $\tilde{\sigma}$ restent bornés.

Preuve :

Soit la fonction de Lyapunov suivante :

$$V = \frac{1}{2} e^T P e + \frac{1}{2\gamma_1} \tilde{\theta}_f^T \tilde{\theta}_f + \frac{1}{2\gamma_2} \tilde{\theta}_g^T \tilde{\theta}_g + \frac{1}{2\gamma_3} \tilde{c}^T \tilde{c} + \frac{1}{2\gamma_4} \tilde{\sigma}^T \tilde{\sigma} \quad (5.2.25)$$

Sa dérivée est donnée par :

$$\dot{V} = \frac{1}{2} [\dot{e}^T P e + e^T P \dot{e}] + \frac{1}{\gamma_1} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\gamma_2} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g + \frac{1}{\gamma_3} \tilde{c}^T \dot{\tilde{c}} + \frac{1}{\gamma_4} \tilde{\sigma}^T \dot{\tilde{\sigma}} \quad (5.2.26)$$

A partir de (5.2.17), (5.2.20) et (5.2.22), l'expression de la dérivée de V devient :

$$\begin{aligned}\dot{V} &= -\frac{1}{2} e^T Q e + S^T \hat{W}^T \tilde{\theta}_f - \frac{1}{\gamma_1} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + S^T \hat{W}^T \tilde{\theta}_g u_0 - \frac{1}{\gamma_2} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g + \\ &\quad S^T \tilde{c}^T W_c'^T (\hat{\theta}_f + \hat{\theta}_g \cdot u_0) - \frac{1}{\gamma_3} \tilde{c}^T \dot{\tilde{c}} + S^T \tilde{\sigma}^T W_\sigma'^T (\hat{\theta}_f + \hat{\theta}_g \cdot u_0) - \\ &\quad \frac{1}{\gamma_4} \tilde{\sigma}^T \dot{\tilde{\sigma}} + S^T [(\tilde{\omega}_f + \tilde{\omega}_g \cdot u_0) + (\varepsilon_f + \varepsilon_g \cdot u_0)] + S^T g(x) \cdot u_{gl}\end{aligned}\quad (5.2.27)$$

avec :

$$\begin{aligned}\dot{\tilde{\theta}}_f &= -\dot{\hat{\theta}}_f \\ \dot{\tilde{\theta}}_g &= -\dot{\hat{\theta}}_g\end{aligned}$$

En remplaçant les paramètres d'adaptation $\hat{\theta}_f, \hat{\theta}_g, \hat{c}$ et $\hat{\sigma}$ par leurs expressions dans (5.2.27) \dot{V} devient :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + g(x) S^T \left[\frac{(\tilde{\omega}_f + \varepsilon_f) + (\tilde{\omega}_g + \varepsilon_g) u_0}{g(x)} - k_{st} \text{sign}(S) \right] \quad (5.2.28)$$

Du fait que u_{st} est pris tel que $u_{st} = -k_{st} \text{sign}(S)$.

Si $g(x) > 0, \forall x$ alors, il suffit de prendre :

$$k_{st} > \frac{(\bar{\omega}_f + \bar{\varepsilon}_f) + (\bar{\omega}_g + \bar{\varepsilon}_g) |u_0|}{[g(x)]_{\min}} \quad (5.2.29)$$

avec :

$$\tilde{\omega}_f \leq \bar{\omega}_f ; \varepsilon_f \leq \bar{\varepsilon}_f ; \tilde{\omega}_g \leq \bar{\omega}_g ; \varepsilon_g \leq \bar{\varepsilon}_g ; \forall t \geq 0$$

Dans ce cas :

$$\dot{V} \leq -\frac{1}{2} \underline{e}^T Q \underline{e} \quad (5.2.30)$$

Par conséquent, \underline{e} tend asymptotiquement vers zéro quand $t \rightarrow \infty$ et $\tilde{\theta}_f, \tilde{\theta}_g, \tilde{c}, \tilde{\sigma}$ sont bornés.

V.2.6 Résultats de la simulation :

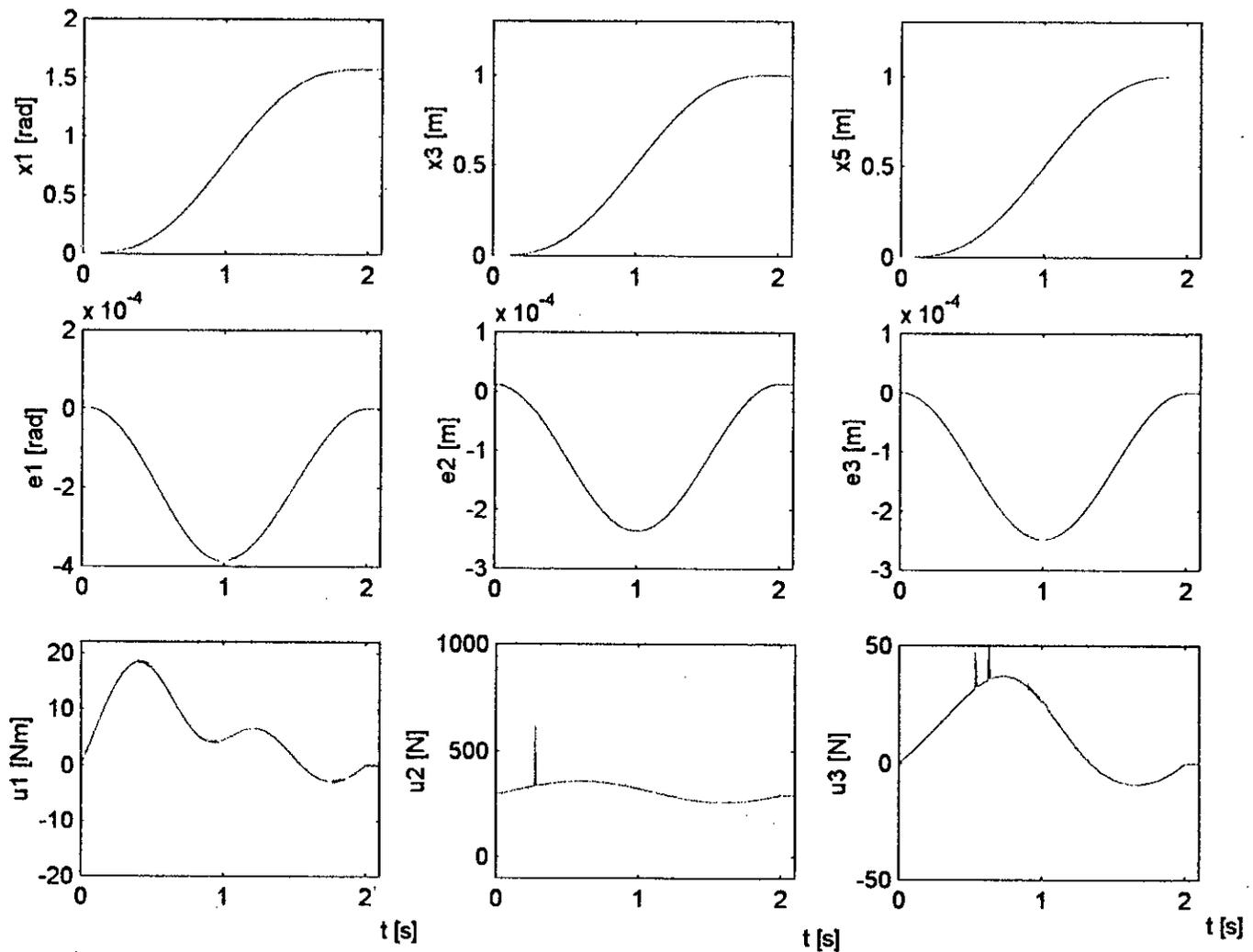


Figure 5.2.1 : Régime de poursuite du robot sans charge ($m_0 = 0$)

- 1^{ère} règle
- 2^{ème} règle
- 3^{ème} règle

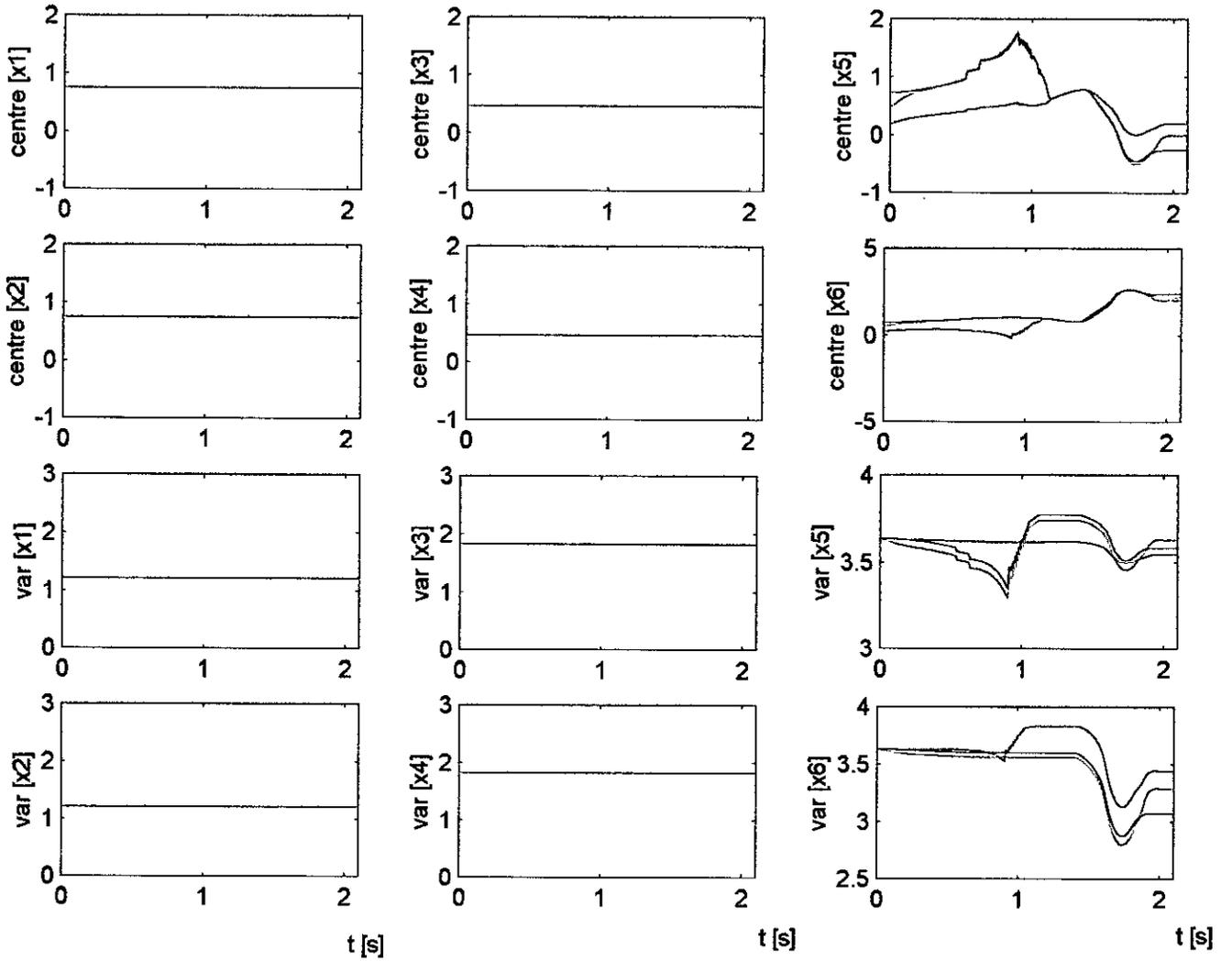


Figure 5.2.1 : (Suite)

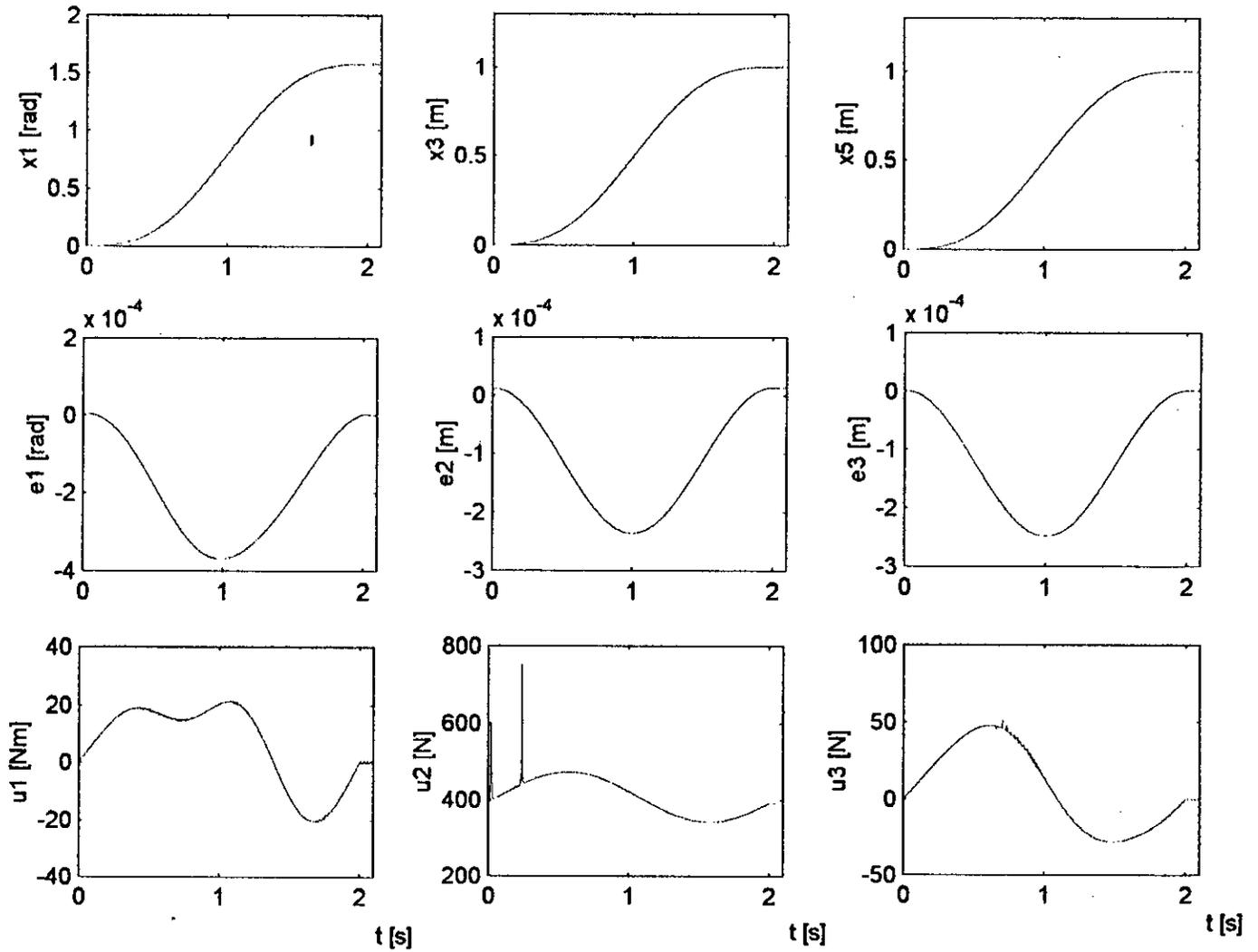


Figure 5.2.2 : Régime de poursuite du robot en charge ($m_0 = 10$ Kg)

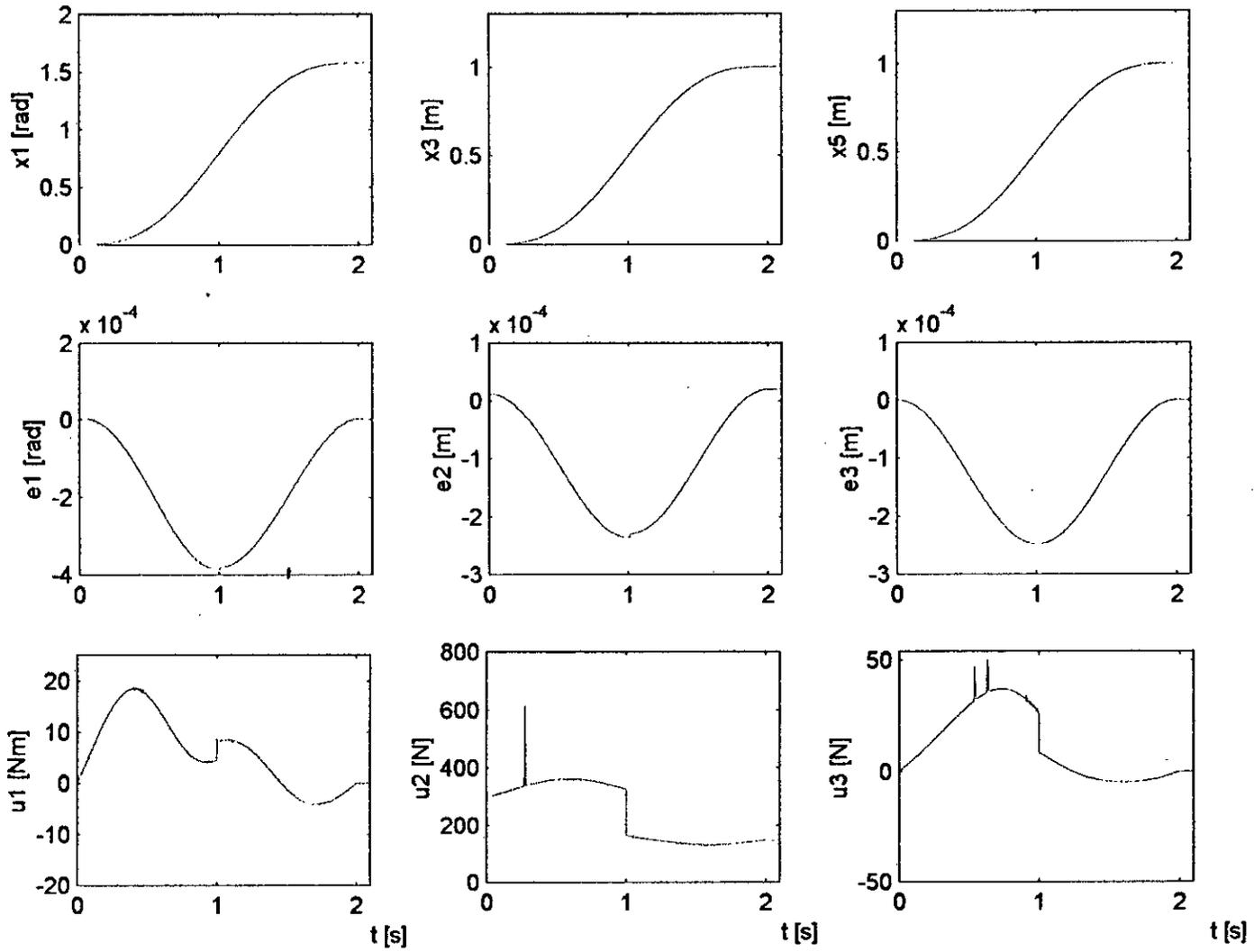


Figure 5.2.3 : Régime de poursuite du robot sans charge avec variations paramétriques en diminution

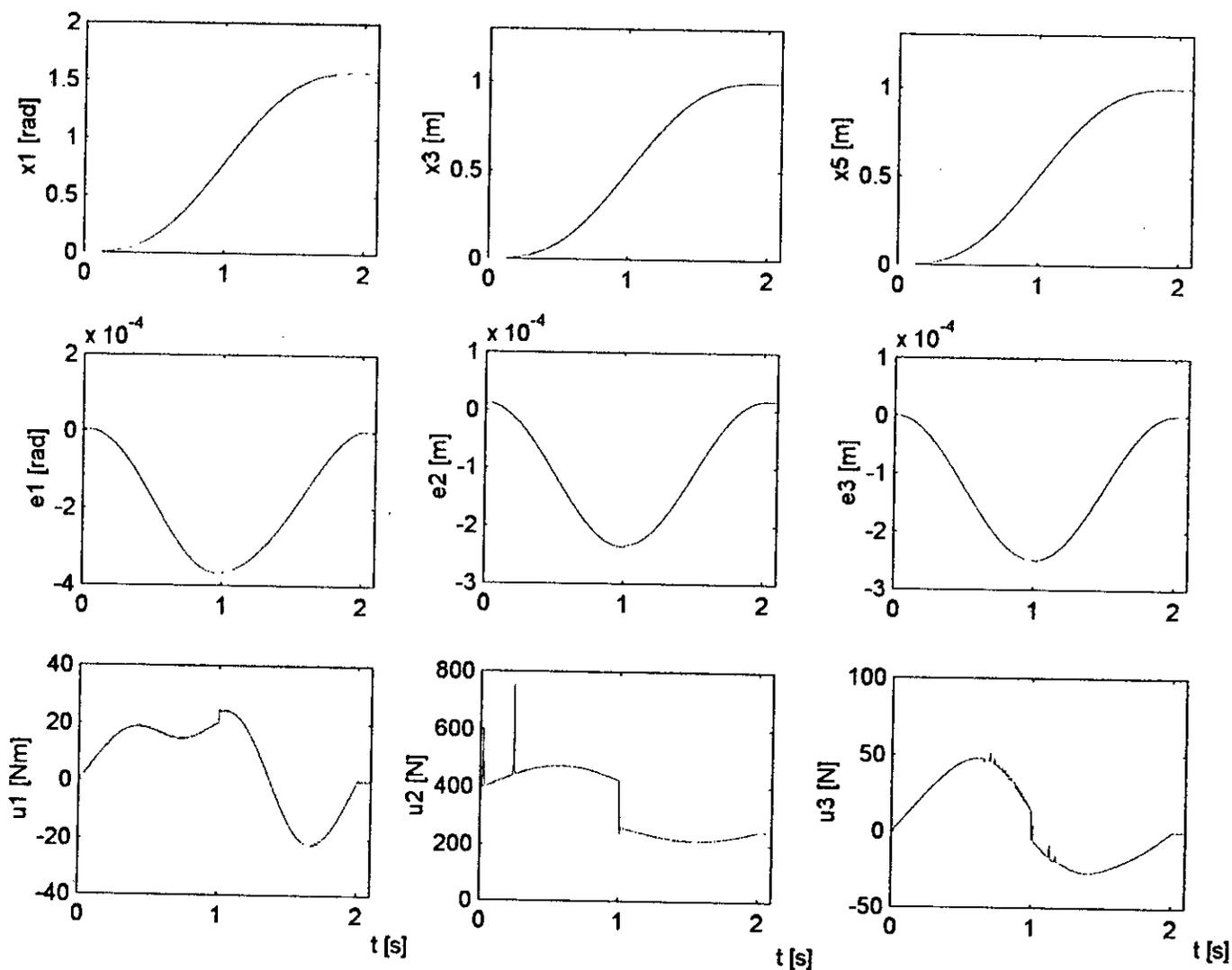


Figure 5.2.4 : Régime de poursuite du robot avec charge avec variations paramétriques en diminution

- 1^{ère} règle
- 2^{ème} règle
- 3^{ème} règle

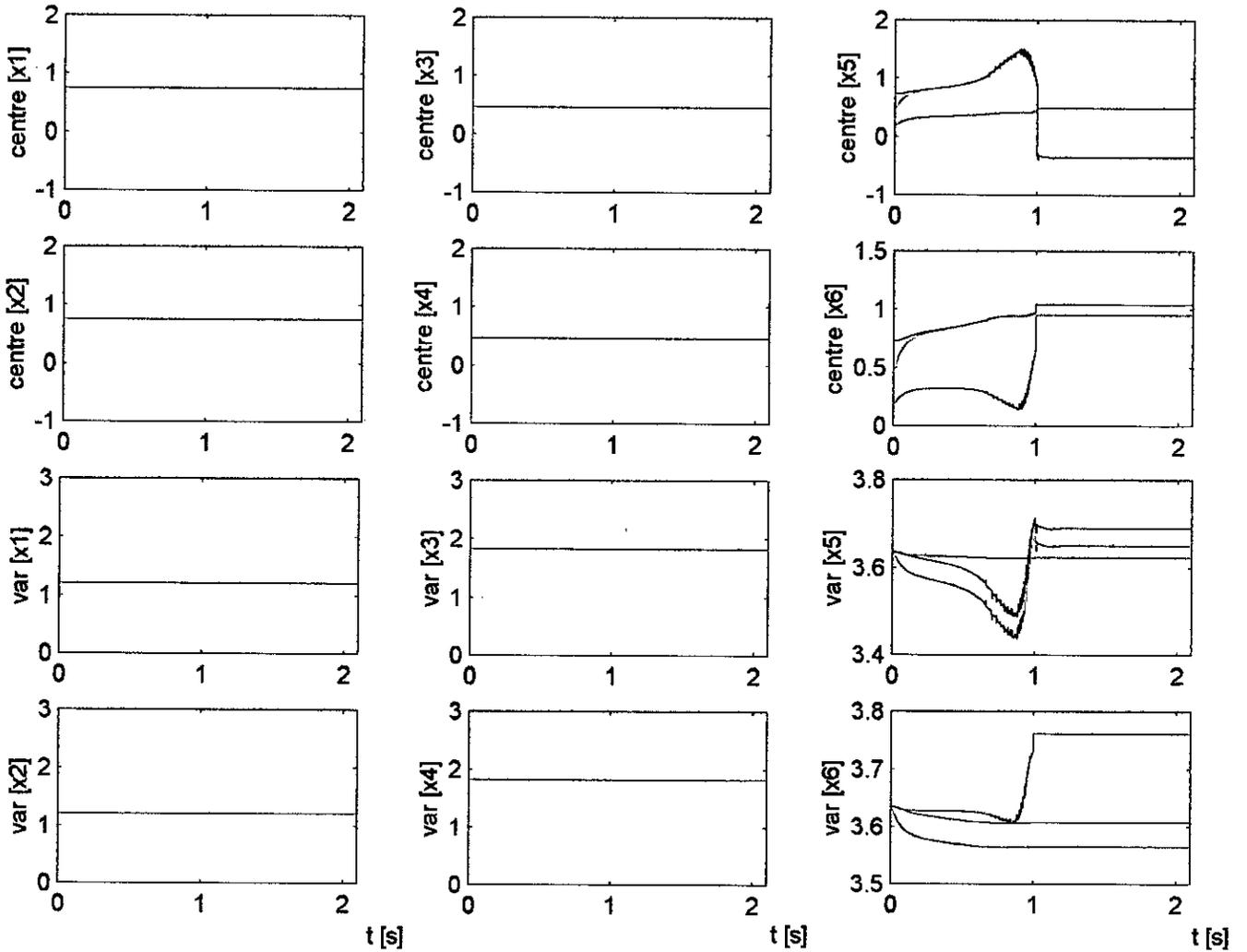


Figure 5.2.4 : (Suite)

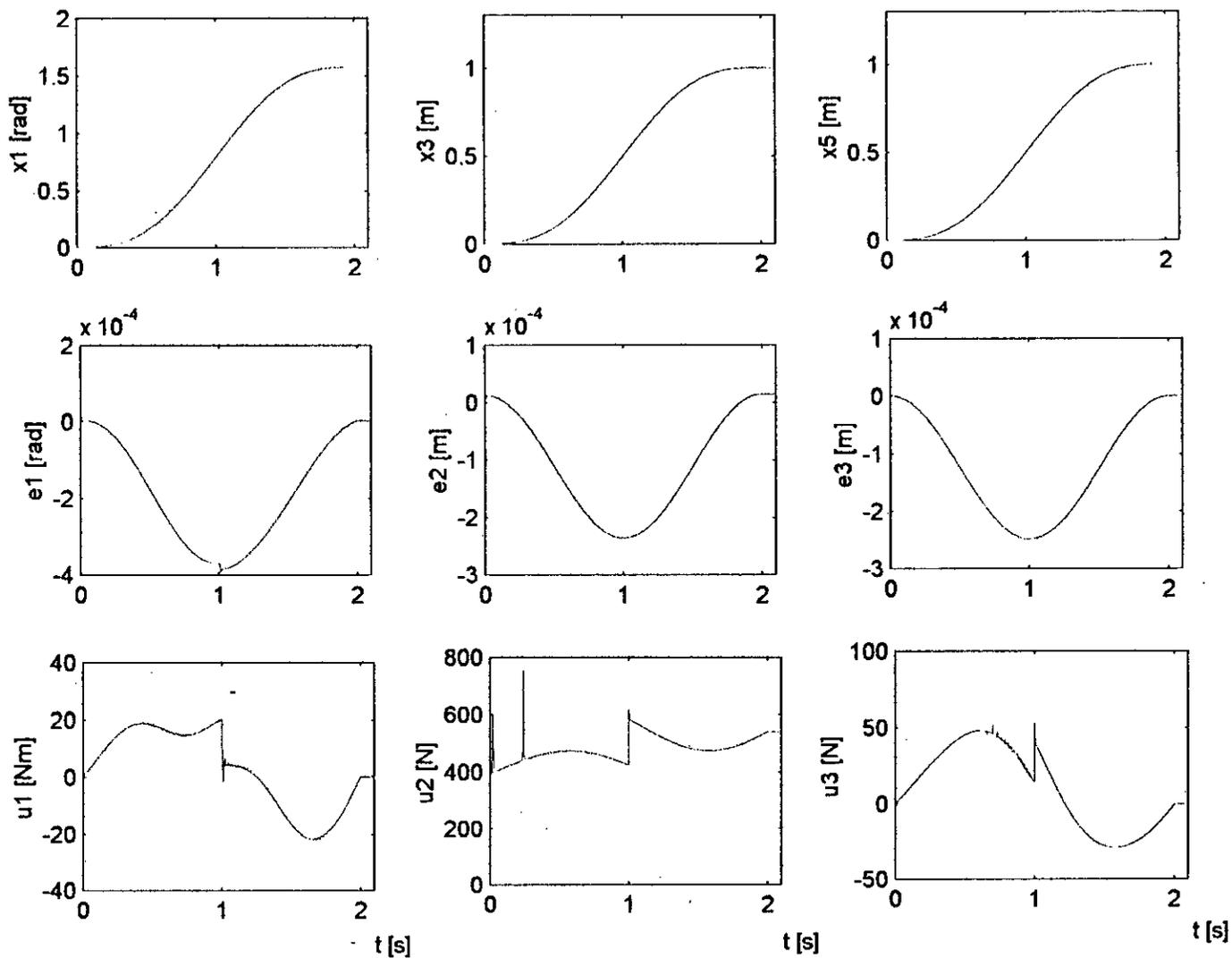


Figure 5.2.5 : Régime de poursuite du robot en charge avec variations paramétriques en augmentation

Interprétation des résultats :

Le fonctionnement à vide du robot (figure 5.2.1) permet de constater qu'il y a un bon suivi de la référence (faibles erreurs de poursuite) et que les commandes sont lisses. De plus, il est à noter que les centres et les variances des fonctions d'appartenance de la troisième articulation évoluent, les autres se stabilisent rapidement à des valeurs constantes.

Le fonctionnement en charge (figure 5.2.2) montre clairement l'augmentation du couple et des forces appliquées au niveau des articulations pour maintenir le régime de poursuite.

L'application d'une diminution de 50 % des paramètres du robot chargé ou non (figure 5.2.3 et 5.2.4) à l'instant $t=1s$ se manifeste par la discontinuité présentée au niveau des commandes. De plus, celle-ci affecte l'évolution des centres et des variances de la troisième articulation.

Ce qu'on peut constater aussi de la figure 5.2.5, c'est la façon avec laquelle la commande réagit, lorsqu'il s'agit d'augmentation au niveau des paramètres, le couple diminue et les forces s'accroissent, et le contraire se produit dans le cas de la diminution des paramètres, ceci explique la parfaite logique d'opposition des deux cas.

Conclusion :

Dans ce chapitre, nous avons développé une commande floue adaptative pour le modèle dynamique inverse du robot. Cette commande exploite les systèmes flous de Sugeno d'ordre zéro et dont l'adaptation des paramètres libres est imposée en usant de la stabilité au sens de Lyapunov.

Dans le premier cas étudié, la base des connaissances qui a servi à la modélisation des systèmes flous est formée par le produit cartésien caractérisant des ensembles flous fixés a priori. La commande proposée a permis d'obtenir un bon régime des poursuites de la trajectoire (erreur de poursuite moins de $4 \cdot 10^{-4}$), toutefois l'essai du robot en charge avec augmentation paramétrique de 50 %, où la commande a présenté une divergence de l'erreur pour la deuxième articulation. En effet, cette commande exige que les paramètres des systèmes flous aient une borne inférieure.

Dans le deuxième cas, dans un souci de réduire le nombre de règles par rapport à la méthode précédente, nous avons rendu les fonctions d'apprentissage des antécédents des règles adaptatives. Ainsi le nombre de règles est réduit en un minimum tout en améliorant les performances.

CONCLUSION GÉNÉRALE

Les robots représentent des systèmes hautement non-linéaires dont les paramètres varient en fonction du temps, de ce fait on utilise la commande adaptative pour assurer une bonne performance.

La commande floue adaptative est un concept d'autorégulation (le réglage se fait avec un choix des gains adéquats). L'algorithme d'adaptation des paramètres est basé sur l'analyse de Lyapunov en garantissant la stabilité du système et la convergence de la sortie vers le signal de référence. Cette commande a toujours montré une robustesse, précision et performance de poursuite avec une convergence asymptotique globale des erreurs de poursuite.

La commande floue adaptative directe décentralisée basée sur la méthode du couple calculé, qui est relativement simple, a permis d'imposer une bonne poursuite avec une faible erreur qui se rapproche énormément de celle de la commande floue adaptative directe linéarisante type 1. Ainsi, les performances du régime de poursuite sont pratiquement les mêmes pour les deux premières commandes et cela pour un pas de calcul de $10^{-3}s$. Ceci constitue un avantage important vis à vis de la commande adaptative directe linéarisante type 2, laquelle exige un pas de calcul $2.5 \cdot 10^{-4}s$ pour une convergence du système. Nous nous sommes intéressés à la commande floue adaptative directe linéarisante type 2 avec l'adaptation des fonctions d'appartenance, en voulant diminuer le nombre de règles ; que nous avons réduit à deux règles seulement.

Pour une meilleure comparaison des commandes, on présente le tableau ci-dessous donnant les erreurs maximales de poursuite pour l'essai en charge avec diminution des paramètres et pour le même pas de calcul de $2.5 \cdot 10^{-4}s$.

	Commande basée sur le couple calculé	Commande linéarisante type 1	Commande linéarisante type 2	Commande linéarisante type 2 avec adaptation des fonctions d'appartenance
Erreur maximale de poursuite	$4.65 \cdot 10^{-4}$	$3.9 \cdot 10^{-4}$	$3.7 \cdot 10^{-4}$	$2.37 \cdot 10^{-4}$

En effet, la commande floue adaptative directe linéarisante type 2 avec adaptation des fonctions d'appartenance de type Gaussienne donne la plus faible erreur avec le plus allégé des algorithmes.

Ainsi, notre but de réduire les algorithmes de commandes est atteint, en utilisant d'une part la méthode décentralisée (l'estimation des systèmes flous exploite uniquement les variables vitesse et position de chaque articulation) et d'autre part, en minimisant le nombre de règles (une règle pour la première articulation, une règle pour la deuxième articulation et trois règles pour la troisième articulation ont suffi à obtenir un bon régime de poursuite) par un apprentissage des fonctions d'appartenance relatives aux variables d'entrée. Ceci a permis de

réduire le temps de calcul de la loi de commande, et nous évite le recours d'un micro-processeur très puissant pour la mise en œuvre de la commande.

Lors de l'élaboration de notre travail, nous avons pris conscience d'une autre perspective qui pourrait être plus fructueuse, c'est l'utilisation des algorithmes génétiques pour la détermination des gains qui assurent la convergence et la stabilité du système commandé, et qui exige beaucoup de temps en les cherchant par tâtonnement. Effectivement, chaque gain peut être considéré comme étant un individu d'une population et moyennant un critère de minimisation à trouver, ce qui permet d'aboutir d'une génération à une autre à des valeurs adéquates.

ANNEXE 1

MODELISATION CINEMATIQUE

Description de la rotation :

La rotation d'un vecteur autour d'un axe est décrite par la matrice de transformation donnée par [8] :

$$T = \left(\begin{array}{ccc|c} & & & 0 \\ & R & & 0 \\ & & & 0 \\ \hline 0 & 0 & 0 & 1 \end{array} \right)$$

avec R une matrice décrite selon l'axe de rotation.

Dans le cas d'une rotation autour de l'axe Z_{i-1} d'un angle θ_i d'un point p de coordonnées (x_i, y_i, z_i) dans le repère R_i a pour coordonnées $(x_{i-1}, y_{i-1}, z_{i-1})$ dans le repère R_{i-1} telle que :

$$\begin{aligned} Z_{i-1} &= Z_i \\ x_i &= \rho \cos \beta \\ y_i &= \rho \sin \beta \end{aligned}$$

où ρ est le module du vecteur $\overline{op'}$ et p' la projection du point p dans le plan $X_i O Y_i$.

$$\begin{aligned} x_{i-1} &= \rho \cos(\theta_i + \beta) = \rho \cos \beta \cos \theta_i - \rho \sin \beta \sin \theta_i \\ y_{i-1} &= \rho \sin(\theta_i + \beta) = \rho \sin \beta \cos \theta_i + \rho \cos \beta \sin \theta_i \end{aligned}$$

d'où :

$$\begin{aligned} x_{i-1} &= x_i \cos \theta_i - y_i \sin \theta_i \\ y_{i-1} &= y_i \cos \theta_i + x_i \sin \theta_i \end{aligned}$$

et sous forme matricielle :

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

$R_{Z_{i-1}}(\theta_i)$: matrice de rotation

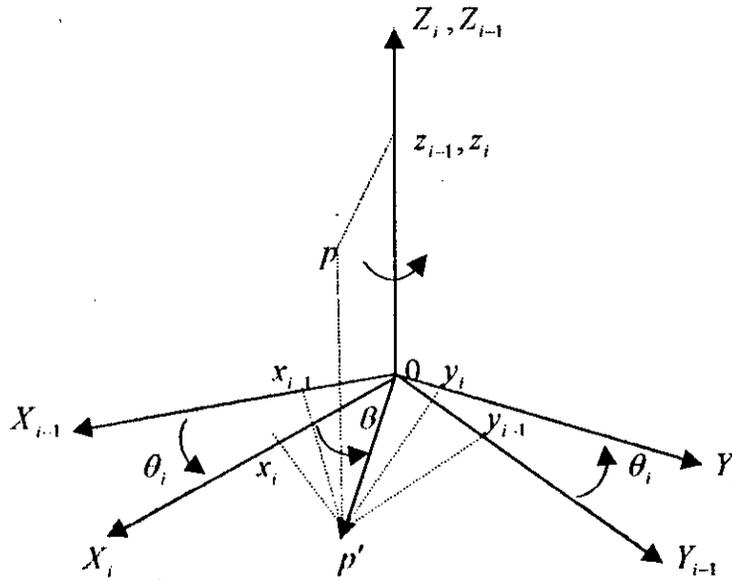


Figure Ann1.1 : Rotation du vecteur autour de l'axe Z

Dans le cas d'une rotation autour de l'axe X_{i-1} d'un angle α_i :

X_i, Y_i et Z_i jouent le même rôle que Z_i, X_i et Y_i dans la transformation $R_{Z_{i-1}}(\theta_i)$ d'où :

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{R_{X_{i-1}}(\alpha_i)} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

$R_{X_{i-1}}(\alpha_i)$: matrice de rotation

Dans le cas d'une rotation autour de l'axe X_{i-1} d'un angle α_i :

X_i, Y_i et Z_i jouent le même rôle que Z_i, X_i et Y_i dans la transformation $R_{Z_{i-1}}(\theta_i)$ d'où :

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{R_{X_{i-1}}(\alpha_i)} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

$R_{X_{i-1}}(\alpha_i)$: matrice de rotation

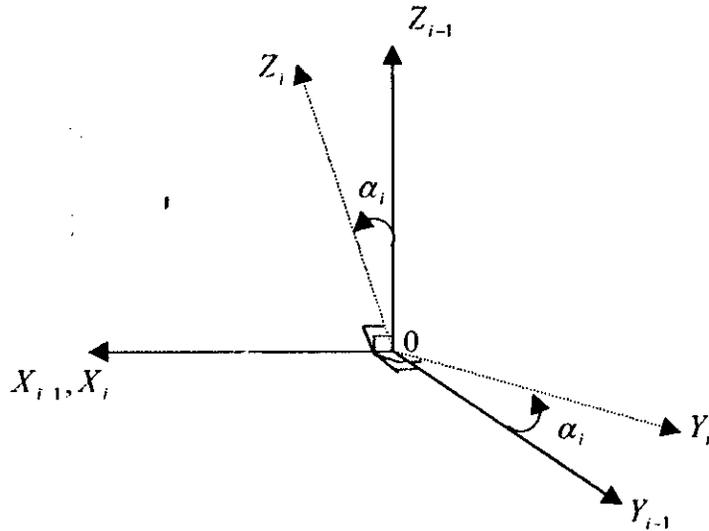


Figure Ann1.2 : Rotation du vecteur autour de l'axe X

Description de la translation (position) :

La position d'un point dans un repère R_i par rapport à un repère R_{i-1} est décrite par la matrice :

$$T = \left(\begin{array}{ccc|c} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{array} \right)$$

où (a, b, c) représentent les coordonnées de l'origine du repère R_{i-1} par rapport à R_i .

Dans le cas d'une translation de l'origine du repère de d_i dans la direction $\overline{O_{i-1}Z_{i-1}}$:

On a :

$$x_{i-1} = x_i$$

$$y_{i-1} = y_i$$

$$\overline{O_{i-1}Z} = \overline{O_{i-1}O_i} + \overline{O_iZ}$$

$$z_{i-1} = d_i + z_i$$

d'où :

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{T_{R_{z_{i-1}}}(d_i)} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

$T_{R_{z_{i-1}}}(d_i)$: matrice de translation.

d_i : coordonnée de O_i suivant l'axe $\overline{O_{i-1}Z_{i-1}}$.

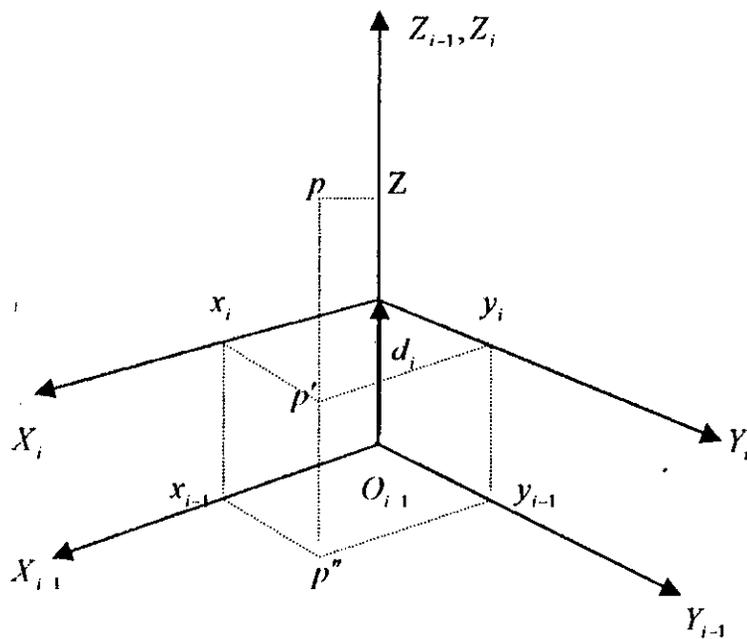


Figure Ann1.3 : Translation positive de l'origine des axes suivant l'axe Z

On peut avoir une translation négative comme la montre la figure Ann1.4 :

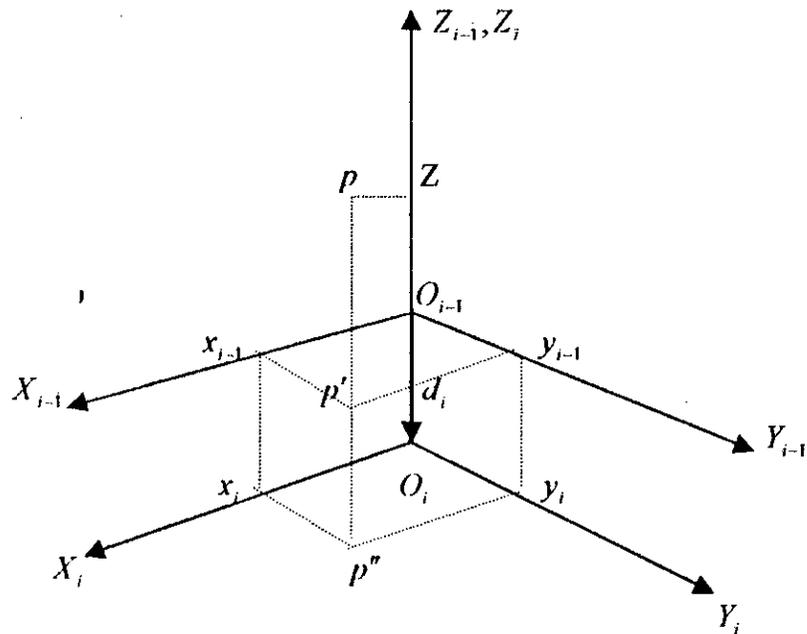


Figure Ann1.4 : Translation négative de l'origine suivant l'axe Z

Dans le cas d'une translation de l'origine du repère de a_i dans la direction $\overline{O_{i-1}X_{i-1}}$ (voir figure Ann1.5) :

On a :

$$y_{i-1} = y_i$$

$$z_{i-1} = z_i$$

$$\overline{O_{i-1}X} = \overline{O_{i-1}O_i} + \overline{O_iX}$$

$$x_{i-1} = a_i + x_i$$

et sous forme matricielle :

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{T_{R_{X_{i-1}}}(a_i)} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

$T_{R_{X_{i-1}}}(a_i)$: matrice de translation.

a_i : coordonnée de O_i suivant l'axe $\overline{O_{i-1}X_{i-1}}$.

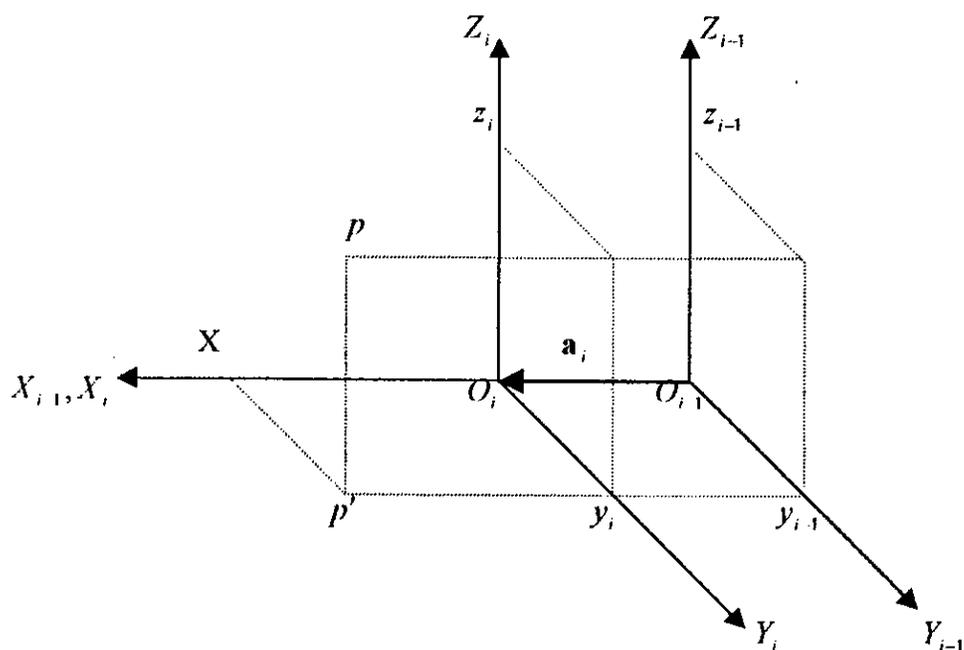


Figure Ann1.5 : Translation positive de l'origine suivant l'axe X

On peut avoir une translation négative comme la montre la figure Ann1.6 :

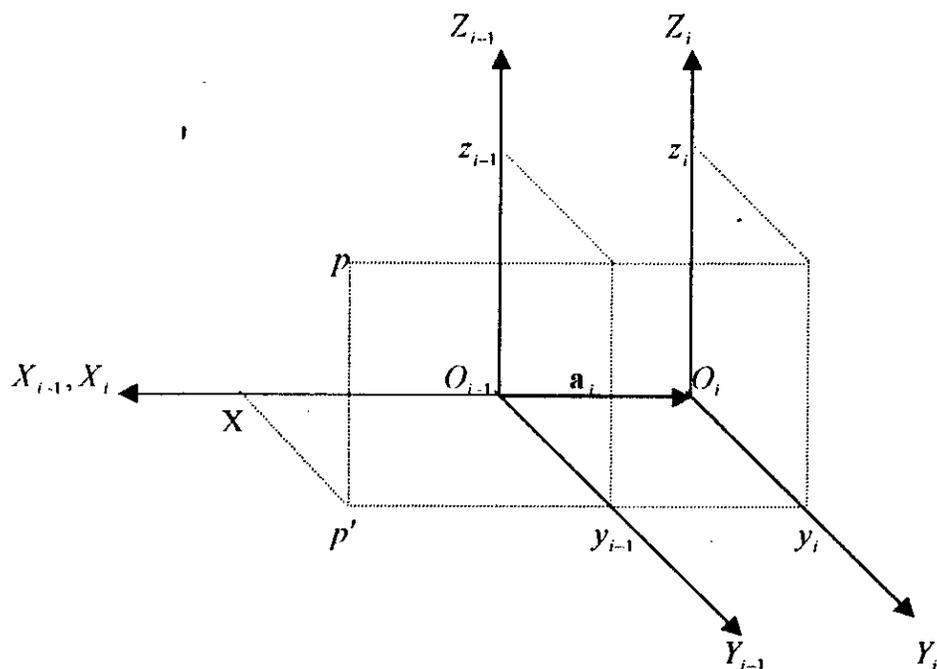


Figure Ann1.6 : Translation négative de l'origine suivant l'axe X

ANNEXE 2

MODELISATION DYNAMIQUE

Calcul du Lagrangien :

- *Vitesse des articulations dans le repère fixe R_0 :*

On note r_j^i le vecteur position de l'articulation i par rapport au repère j :

$$r_j^i = [x_i \quad y_i \quad z_i \quad 1]_j^T$$

La vitesse de l'articulation i par rapport au repère initiale est :

$$V_0^i = \frac{dr_0^i}{dt} \quad \text{avec} \quad r_0^i = T_0^i r_i^i \quad (\text{Ann2.1})$$

$$V_0^i = \frac{dT_0^i}{dt} r_i^i + T_0^i \frac{dr_i^i}{dt} \quad (\text{Ann2.2})$$

Comme la liaison n'est pas flexible $\Rightarrow \frac{dr_i^i}{dt} = 0$.

d'où :

$$V_0^i = \frac{dT_0^i}{dt} r_i^i \quad (\text{Ann2.3})$$

Or :

$$T_0^i = T_0^1 \cdot T_1^2 \cdots T_{i-2}^{i-1} \cdot T_{i-1}^i \quad (\text{Ann2.4})$$

donc :

$$\frac{dT_0^i}{dt} = \sum_{j=1}^i \frac{dT_0^i}{dq_j} \frac{dq_j}{dt} = \sum_{j=1}^i \frac{dT_0^i}{dq_j} \dot{q}_j \quad (\text{Ann2.5})$$

$$V_0^i = \left(\sum_{j=1}^i \frac{dT_0^i}{dq_j} \dot{q}_j \right) r_i^i \quad (\text{Ann2.6})$$

Comme chaque transformation ne dépend que d'une seule variable généralisée pour un robot rigide:

$$T_0^i = T_0^1 \cdot T_1^2 \cdots T_{i-2}^{i-1} \cdot T_{i-1}^i$$

$$\begin{aligned}
T_0^1 &\rightarrow q_1 \\
T_1^2 &\rightarrow q_2 \\
&\vdots \\
T_{j-1}^j &\rightarrow q_j \\
&\vdots \\
T_{i-2}^{i-1} &\rightarrow q_{i-1} \\
T_{i-1}^i &\rightarrow q_i
\end{aligned}$$

Alors : T_{j-1}^j est fonction de la variable q_j .

$$\frac{dT_0^i}{dq_j} = T_0^1 \cdot T_1^2 \cdots T_{j-2}^{j-1} \frac{dT_{j-1}^j}{dq_j} T_j^{j+1} \cdots T_{i-2}^{i-1} \cdot T_{i-1}^i \quad (\text{Ann2.7})$$

$$\frac{dT_0^i}{dq_j} = T_0^{j-1} \cdot \frac{dT_{j-1}^j}{dq_j} \cdot T_j^i \quad (\text{Ann2.8})$$

Si T_{j-1}^j est une rotation autour de l'axe Z d'un angle $\theta_j = q_j$, alors :

$$T_{j-1}^j = \begin{bmatrix} \cos \theta_j & -\sin \theta_j & 0 & 0 \\ \sin \theta_j & \cos \theta_j & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{Ann2.9})$$

$$\frac{dT_{j-1}^j}{d\theta_j} = \begin{bmatrix} -\sin \theta_j & -\cos \theta_j & 0 & 0 \\ \cos \theta_j & -\sin \theta_j & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos \theta_j & -\sin \theta_j & 0 & 0 \\ \sin \theta_j & \cos \theta_j & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\frac{dT_{j-1}^j}{d\theta_j} = Q_j \cdot T_{j-1}^j \quad \text{avec} \quad Q_j = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{Ann2.10})$$

$$\frac{dT_{j-1}^j}{dq_j} = Q_j \cdot T_{j-1}^j \quad (\text{Ann2.11})$$

L'équation (Ann2.8) devient ainsi :

$$\frac{dT_0^{i+1}}{dq_j} = T_0^{j-1} \cdot Q_j \cdot T_{j-1}^j \cdot T_j^i = T_0^{j-1} \cdot Q_j \cdot T_{j-1}^i \quad \text{et } j \leq i \quad (\text{Ann2.12})$$

Si T_{j-1}^j est une translation suivant la direction \overline{OZ} , $q_j = d_j$ alors :

$$T_{j-1}^j = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{Ann2.13})$$

$$\frac{dT_{j-1}^j}{dq_j} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{Ann2.14})$$

$$\frac{dT_{j-1}^j}{dq_j} = Q_j \cdot T_{j-1}^j \quad \text{avec } Q_j = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{Ann2.15})$$

L'équation (Ann2.8) devient alors :

$$\frac{dT_0^i}{dq_j} = T_0^{j-1} \cdot Q_j \cdot T_{j-1}^j \cdot T_j^i = T_0^{j-1} \cdot Q_j \cdot T_{j-1}^i \quad (\text{Ann2.16})$$

Par conséquent, la relation (Ann2.3) peut encore s'écrire :

$$V_0^i = \left(\sum_{j=1}^i T_0^{j-1} \cdot Q_j \cdot T_{j-1}^i \cdot \dot{q}_j \right) r_i^i \quad (\text{Ann2.17})$$

On pose :

$$U_{ij} = T_0^{j-1} \cdot Q_j \cdot T_{j-1}^i \quad (\text{Ann2.18})$$

alors :

$$V_0^i = \left(\sum_{j=1}^i U_{ij} \cdot \dot{q}_j \right) r_i^i \quad (\text{Ann2.19})$$

- *Energie cinétique :*

L'énergie cinétique d'un élément infinitésimal dm de la liaison i est :

$$dE_{Ci} = \frac{1}{2} \text{trace}(V_i V_i^T) dm \quad (\text{Ann2.20})$$

avec :

$$(V_i^T)^T = (r_i^T)^T \sum_{k=1}^i U_{ik}^T \dot{q}_k \quad (\text{Ann2.21})$$

$$dE_{Ci} = \frac{1}{2} \text{trace} \left[\left(\sum_{j=1}^i U_{ij} \dot{q}_j \right) r_i^T (r_i^T)^T \sum_{k=1}^i U_{ik}^T \dot{q}_k \right] dm$$

$$dE_{Ci} = \frac{1}{2} \text{trace} \left[\left(\sum_{j=1}^i U_{ij} \dot{q}_j \right) (r_i^T (r_i^T)^T dm) \sum_{k=1}^i U_{ik}^T \dot{q}_k \right]$$

$$dE_{Ci} = \frac{1}{2} \text{trace} \left[\left(\sum_{j=1}^i U_{ij} \dot{q}_j \right) dJ_i \sum_{k=1}^i U_{ik}^T \dot{q}_k \right]$$

$$dE_{Ci} = \frac{1}{2} \text{trace} \left[\sum_{j=1}^i \sum_{k=1}^i U_{ij} dJ_i U_{ik}^T \dot{q}_j \dot{q}_k \right] \quad (\text{Ann2.22})$$

En posant :

$$dJ_i = r_i^T (r_i^T)^T dm \quad (\text{Ann2.23})$$

$$r_i^T = \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \quad \text{et} \quad (r_i^T)^T = [x_i \quad y_i \quad z_i \quad 1] \quad (\text{Ann2.24})$$

$$r_i^i(r_i^i)^T = \begin{bmatrix} x_i^2 & x_i y_i & x_i z_i & x_i \\ y_i x_i & y_i^2 & y_i z_i & y_i \\ z_i x_i & z_i y_i & z_i^2 & z_i \\ x_i & y_i & z_i & 1 \end{bmatrix} \quad (\text{Ann2.25})$$

$$dJ_i = \begin{bmatrix} x_i^2 dm & x_i y_i dm & x_i z_i dm & x_i dm \\ y_i x_i dm & y_i^2 dm & y_i z_i dm & y_i dm \\ z_i x_i dm & z_i y_i dm & z_i^2 dm & z_i dm \\ x_i dm & y_i dm & z_i dm & dm \end{bmatrix} \quad (\text{Ann2.26})$$

$$J_i = \begin{bmatrix} \int x_i^2 dm & \int x_i y_i dm & \int x_i z_i dm & \int x_i dm \\ \int x_i y_i dm & \int y_i^2 dm & \int y_i z_i dm & \int y_i dm \\ \int x_i z_i dm & \int y_i z_i dm & \int z_i^2 dm & \int z_i dm \\ \int x_i dm & \int y_i dm & \int z_i dm & \int dm \end{bmatrix} \quad (\text{Ann2.27})$$

d'où :

$$E_{Ci} = \frac{1}{2} \text{trace} \left[\sum_{j=1}^i \sum_{k=1}^i U_{ij} J_i U_{ik}^T \dot{q}_j \dot{q}_k \right] \quad (\text{Ann2.28})$$

L'énergie cinétique des actionneurs est donnée par:

$$E_{Ca} = \frac{1}{2} \sum_{i=1}^n I_i \dot{q}_i^2 \quad (\text{Ann2.29})$$

où I_i représente le moment d'inertie dans le cas d'une rotation ou bien la masse dans le cas d'une translation de l'actionneur.

L'énergie cinétique totale sera :

$$E_C = \sum_{i=1}^n E_{Ci} + E_{Ca} \quad (\text{Ann2.30})$$

L'énergie potentielle est :

$$E_p = \sum_{i=1}^n -m_i g^T r_0^i = - \sum_{i=1}^n m_i g^T T_0^i r_i^i \quad (\text{Ann2.31})$$

avec :

$$g^T = \begin{bmatrix} 0 & 0 & -|g| & 1 \end{bmatrix}$$

Par conséquent, le Lagrangien s'exprime par :

$$L = \frac{1}{2} \sum_{p=1}^n \left[\sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{pj} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \right] + \sum_{p=1}^n m_p g^T T_0^p r_p^p \quad (\text{Ann2.32})$$

Calcul explicite de la formule d'Euler-Lagrange :

Appliquons la formule d'Euler-Lagrange à la fonction Lagrangienne :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} + \frac{\partial E_D}{\partial \dot{q}_i} = \tau_i \quad i = 1 \dots n \quad (\text{Ann2.33})$$

- Calcul de $\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right)$:

On considère le premier terme de L :

$$L = \frac{1}{2} \sum_{p=1}^n \sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{pj} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \quad (\text{Ann2.34})$$

Qui s'écrit :

$$\begin{aligned} L_1 = & \frac{1}{2} \sum_{j=1}^1 \sum_{k=1}^1 \text{trace}(U_{1j} J_1 U_{1k}^T) \dot{q}_j \dot{q}_k + \frac{1}{2} \sum_{j=1}^2 \sum_{k=1}^2 \text{trace}(U_{2j} J_2 U_{2k}^T) \dot{q}_j \dot{q}_k + \dots \\ & \dots + \frac{1}{2} \sum_{j=1}^i \sum_{k=1}^i \text{trace}(U_{ij} J_i U_{ik}^T) \dot{q}_j \dot{q}_k + \dots + \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \text{trace}(U_{nj} J_n U_{nk}^T) \dot{q}_j \dot{q}_k \end{aligned} \quad (\text{Ann2.35})$$

Pour la $i^{\text{ème}}$ articulation, les dérivées partielles sont effectuées par rapport à la variable \dot{q}_i , par conséquent, on ne garde de L_1 que les termes contenant \dot{q}_i .

$$\begin{aligned} L_1' = & \frac{1}{2} \sum_{j=1}^i \sum_{k=1}^i \text{trace}(U_{ij} J_i U_{ik}^T) \dot{q}_j \dot{q}_k + \frac{1}{2} \sum_{j=1}^{i+1} \sum_{k=1}^{i+1} \text{trace}(U_{(i+1)j} J_{i+1} U_{(i+1)k}^T) \dot{q}_j \dot{q}_k + \dots \\ & \dots + \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \text{trace}(U_{nj} J_n U_{nk}^T) \dot{q}_j \dot{q}_k \end{aligned} \quad (\text{Ann2.36})$$

En fixant $j = i$; on obtient :

$$L'_{1a} = \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{ii} J_i U_{ii}^T) \dot{q}_i^2 + \dots + \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{mi} J_n U_{ni}^T) \dot{q}_i^2 + \frac{1}{2} \sum_{k=i+1}^n \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i \quad (\text{Ann2.37})$$

Les termes restants contenant \dot{q}_i sont obtenus en fixant $k = i$ et j variable, avec $j \neq i$:

$$L'_{1b} = \frac{1}{2} \sum_{j=1}^{i-1} \text{tr}(U_{ij} J_i U_{ii}^T) \dot{q}_j \dot{q}_i + \dots + \frac{1}{2} \sum_{j=1}^{i-1} \text{tr}(U_{nj} J_n U_{ni}^T) \dot{q}_j \dot{q}_i + \frac{1}{2} \sum_{j=i+1}^n \text{tr}(U_{mj} J_n U_{mi}^T) \dot{q}_j \dot{q}_i \quad (\text{Ann2.38})$$

L'_1 se réduit encore une fois à $L'_1 = L'_{1a} + L'_{1b}$

$$L'_1 = \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{ii} J_i U_{ii}^T) \dot{q}_i^2 + \dots + \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{mi} J_n U_{ni}^T) \dot{q}_i^2 + \frac{1}{2} \sum_{k=i+1}^n \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \sum_{j=1}^{i-1} \text{tr}(U_{ij} J_i U_{ii}^T) \dot{q}_j \dot{q}_i + \dots \dots + \frac{1}{2} \sum_{j=1}^{i-1} \text{tr}(U_{nj} J_n U_{ni}^T) \dot{q}_j \dot{q}_i + \frac{1}{2} \sum_{j=i+1}^n \text{tr}(U_{mj} J_n U_{mi}^T) \dot{q}_j \dot{q}_i \quad (\text{Ann2.39})$$

En remplaçant j par k , et en utilisant la propriété de la transposée ; $\text{tr}(A) = \text{tr}(A)^T$, il vient :

$$L'_1 = \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{ii} J_i U_{ii}^T) \dot{q}_i^2 + \dots + \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{mi} J_n U_{ni}^T) \dot{q}_i^2 + \frac{1}{2} \sum_{k=i+1}^n \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k \dot{q}_i + \dots \dots + \frac{1}{2} \sum_{k=1}^{i-1} \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \sum_{k=i+1}^n \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i \quad (\text{Ann2.40})$$

En effectuant la somme des termes identiques, on obtient :

$$L'_1 = \sum_{k=1}^{i-1} \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{ii} J_i U_{ii}^T) \dot{q}_i^2 + \dots + \sum_{k=1}^{i-1} \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i + \frac{1}{2} \text{tr}(U_{mi} J_n U_{ni}^T) \dot{q}_i^2 + \sum_{k=i+1}^n \text{tr}(U_{mi} J_n U_{nk}^T) \dot{q}_k \dot{q}_i \quad (\text{Ann2.41})$$

$$\frac{\partial L'_1}{\partial \dot{q}_i} = \sum_{k=1}^{i-1} \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k + \text{tr}(U_{ii} J_i U_{ii}^T) \dot{q}_i + \dots + \sum_{k=1}^{i-1} \text{tr}(U_{ni} J_n U_{nk}^T) \dot{q}_k + \text{tr}(U_{ni} J_n U_{ni}^T) \dot{q}_i + \sum_{k=i+1}^n \text{tr}(U_{ni} J_n U_{nk}^T) \dot{q}_k \quad (\text{Ann2.42})$$

$$\frac{\partial L'_1}{\partial \dot{q}_i} = \sum_{k=1}^i \text{tr}(U_{ii} J_i U_{ik}^T) \dot{q}_k + \dots + \sum_{k=1}^n \text{tr}(U_{ni} J_n U_{nk}^T) \dot{q}_k \quad (\text{Ann2.43})$$

$$\frac{\partial L'_1}{\partial \dot{q}_i} = \sum_{j=i}^n \sum_{k=1}^j \text{tr}(U_{jj} J_j U_{jk}^T) \dot{q}_k \quad (\text{Ann2.44})$$

qui peut encore s'écrire :

$$\frac{\partial L'_1}{\partial \dot{q}_i} = \sum_{j=i}^n \sum_{k=1}^j \text{tr}(U_{jk} J_j U_{ji}^T) \dot{q}_k \quad (\text{Ann2.45})$$

$$\frac{d}{dt} \left(\frac{\partial L'_1}{\partial \dot{q}_i} \right) = \sum_{j=i}^n \sum_{k=1}^j \text{tr}(U_{jk} J_j U_{ji}^T) \ddot{q}_k \quad (\text{Ann2.46})$$

- Calcul de $\frac{\partial L}{\partial q_i}$:

$$L = L_1 + L_2 \quad \text{avec :}$$

$$L_1 = \frac{1}{2} \sum_{p=1}^n \left[\sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{pj} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \right] \quad (\text{Ann2.47})$$

$$L_2 = \sum_{p=1}^n m_p \mathbf{g}^T T_0^p \mathbf{r}_p^p \quad (\text{Ann2.48})$$

$$\frac{\partial L}{\partial q_i} = \frac{\partial L_1}{\partial q_i} + \frac{\partial L_2}{\partial q_i} \quad (\text{Ann2.49})$$

L_1 peut s'écrire comme suit :

$$\begin{aligned}
L_1 = & \frac{1}{2} \sum_{j=1}^1 \sum_{k=1}^1 \text{tr}(U_{1j} J_1 U_{1k}^T) \dot{q}_j \dot{q}_k + \frac{1}{2} \sum_{j=1}^2 \sum_{k=1}^2 \text{tr}(U_{2j} J_2 U_{2k}^T) \dot{q}_j \dot{q}_k + \\
& \frac{1}{2} \sum_{j=1}^3 \sum_{k=1}^3 \text{tr}(U_{3j} J_3 U_{3k}^T) \dot{q}_j \dot{q}_k + \dots + \frac{1}{2} \sum_{j=1}^i \sum_{k=1}^i \text{tr}(U_{ij} J_i U_{ik}^T) \dot{q}_j \dot{q}_k + \dots \\
& \dots + \frac{1}{2} \sum_{j=1}^{n-1} \sum_{k=1}^{n-1} \text{tr}(U_{(n-1)j} J_{n-1} U_{(n-1)k}^T) \dot{q}_j \dot{q}_k + \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \text{tr}(U_{nj} J_n U_{nk}^T) \dot{q}_j \dot{q}_k
\end{aligned} \quad (\text{Ann2.50})$$

Du fait que U_{ij} s'écrit :

$$U_{ij} = \begin{cases} T_0^{j-1} Q_j T_{j-1}^{i-1} T_{i-1}^j T_i^p & 0 < j \leq i \leq p \\ T_0^{i-1} T_{i-1}^j T_{j-1}^{i-1} Q_j T_j^p & 0 < i \leq j \leq p \\ 0 & 0 < p < j < i \end{cases} \quad (\text{Ann2.51})$$

et comme seul T_i^i est en fonction de q_i , les termes de L_1 contenant q_i sont ceux pour $p \geq i$

$$L'_1 = \frac{1}{2} \sum_{p=i}^n \left[\sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{ij} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \right] \quad (\text{Ann2.52})$$

$$\frac{\partial L'_1}{\partial q_i} = \frac{1}{2} \sum_{p=i}^n \left[\sum_{j=1}^p \sum_{k=1}^p \text{trace} \left(\frac{\partial U_{ij}}{\partial q_i} J_p U_{pk}^T \right) \dot{q}_j \dot{q}_k \right] + \frac{1}{2} \sum_{p=i}^n \left[\sum_{j=1}^p \sum_{k=1}^p \text{trace} \left(U_{ij} J_p \frac{\partial U_{pk}^T}{\partial q_i} \right) \dot{q}_j \dot{q}_k \right] \quad (\text{Ann2.53})$$

Et comme :

$$\text{trace} \left(U_{ij} J_p \frac{\partial U_{pk}^T}{\partial q_i} \right) = \text{trace} \left(\frac{\partial U_{pk}}{\partial q_i} J_p U_{ij}^T \right) \quad (\text{Ann2.54})$$

Et du fait que :

$$\sum_{j=1}^p \sum_{k=1}^p (x_j y_k) = \sum_{k=1}^p \sum_{j=1}^p (x_k y_j) \quad (\text{Ann2.55})$$

Il vient que :

$$\frac{1}{2} \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{tr} \left(U_{ij} J_p \frac{\partial U_{pk}^T}{\partial q_i} \right) \dot{q}_j \dot{q}_k = \frac{1}{2} \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{tr} \left(\frac{\partial U_{ij}}{\partial q_i} J_p U_{pk}^T \right) \dot{q}_j \dot{q}_k \quad (\text{Ann2.56})$$

d'où :

$$\frac{\partial L'_1}{\partial q_i} = \frac{1}{2} \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{tr} \left(\frac{\partial U_{ij}}{\partial q_i} J_p U_{pk}^T \right) \dot{q}_j \dot{q}_k + \frac{1}{2} \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{tr} \left(\frac{\partial U_{ij}}{\partial q_i} J_p U_{pk}^T \right) \dot{q}_j \dot{q}_k$$

$$\frac{\partial L'_1}{\partial q_i} = \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{tr} \left(\frac{\partial U_{ij}}{\partial q_i} J_p U_{pk}^T \right) \dot{q}_j \dot{q}_k \quad (\text{Ann2.57})$$

avec :

$$\frac{\partial U_{ij}}{\partial q_i} = \begin{cases} T_0^{j-1} Q_j T_{j-1}^{i-1} \frac{\partial T_{i-1}^i}{\partial q_i} T_i^p & 0 < j \leq i \leq p \\ T_0^{i-1} \frac{\partial T_{i-1}^i}{\partial q_i} T_i^{j-1} Q_j T_j^p & 0 < i \leq j \leq p \\ 0 & 0 < p < j < i \end{cases} \quad (\text{Ann2.58})$$

Or :

$$\frac{\partial T_{i-1}^i}{\partial q_i} = Q_i T_{i-1}^i \quad (\text{Ann2.59})$$

On a finalement :

$$\frac{\partial U_{ij}}{\partial q_i} = \begin{cases} T_0^{j-1} Q_j T_{j-1}^{i-1} Q_i T_i^p & 0 < j \leq i \leq p \\ T_0^{i-1} Q_i T_{i-1}^{j-1} Q_j T_j^p & 0 < i \leq j \leq p \\ 0 & 0 < p < j < i \end{cases} \quad (\text{Ann2.60})$$

On pose :

$$U_{ij} = \begin{cases} T_0^{j-1} Q_j T_{j-1}^{i-1} Q_i T_i^p & 0 < j \leq i \leq p \\ T_0^{i-1} Q_i T_{i-1}^{j-1} Q_j T_j^p & 0 < i \leq j \leq p \\ 0 & 0 < p < j < i \end{cases} \quad (\text{Ann2.61})$$

d'où :

$$\frac{\partial L_1}{\partial q_i} = \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{trace} (U_{ij} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \quad (\text{Ann2.62})$$

- Calcul du terme $\frac{\partial L_2}{\partial q_i}$:

Comme L_2 est donnée par :

$$L_2 = m_1 g^T T_0^1 r_1^1 + m_2 g^T T_0^2 r_2^2 + \dots + m_i g^T T_0^i r_i^i + \dots + m_n g^T T_0^n r_n^n \quad (\text{Ann2.63})$$

Les termes de la somme qui contiennent q_i sont tel que $p \geq i$ d'où L_2 est réduite à :

$$\begin{aligned} L'_2 &= \sum_{p=i}^n m_p g^T T_0^p r_p^p \\ \frac{\partial L'_2}{\partial q_i} &= m_i g^T \frac{\partial T_0^i}{\partial q_i} r_i^i + \dots + m_n g^T \frac{\partial T_0^n}{\partial q_i} r_n^n \\ &= m_i g^T T_0^{i-1} \frac{\partial T_{i-1}^i}{\partial q_i} r_i^i + \dots + m_n g^T T_0^{i-1} \frac{\partial T_{i-1}^n}{\partial q_i} T_i^n r_n^n \\ &= m_i g^T T_0^{i-1} Q_j T_{i-1}^i r_i^i + \dots + m_n g^T T_0^{i-1} Q_j T_{i-1}^i T_i^n r_n^n \\ &= m_i g^T U_{ii} r_i^i + \dots + m_n g^T U_{ni} r_n^n \\ &= \sum_{p=i}^n m_p g^T U_{pi} r_p^p \end{aligned} \quad (\text{Ann2.64})$$

Et finalement :

$$\frac{\partial L}{\partial q_i} = \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{pi} J_p U_{pk}^T) \dot{q}_j \dot{q}_k + \sum_{p=i}^n m_p g^T U_{pi} r_p^p \quad (\text{Ann2.65})$$

- Calcul de $\frac{\partial E_D}{\partial \dot{q}_i}$:

L'énergie de dissipation est donnée par :

$$E_D = \frac{1}{2} \sum_{p=1}^n b_p \dot{q}_p^2 = \frac{1}{2} (b_1 \dot{q}_1^2 + b_2 \dot{q}_2^2 + \dots + b_i \dot{q}_i^2 + \dots + b_n \dot{q}_n^2) \quad (\text{Ann2.66})$$

Où b_i désigne le coefficient de frottement de la $i^{\text{ème}}$ articulation.

$$\frac{\partial E_D}{\partial \dot{q}_i} = b_i \dot{q}_i \quad (\text{Ann2.67})$$

d'où l'expression de la force ou du couple généralisé :

$$\begin{aligned} \tau_i = & \sum_{j=i}^n \sum_{k=1}^j \text{trace}(U_{jk} J_j U_{jk}^T) \ddot{q}_k - \sum_{p=i}^n \sum_{j=1}^p \sum_{k=1}^p \text{trace}(U_{pj} J_p U_{pk}^T) \dot{q}_j \dot{q}_k \\ & - \sum_{p=i}^n m_p g^T U_{pi} r_p^p + b_i \dot{q}_i \end{aligned} \quad (\text{Ann2.68})$$

ANNEXE 3

Modèle d'état du robot à structure cylindrique :

Les équations dynamiques du robot sont [7] :

$$\begin{cases} \Gamma_1 = [I_2 + I_3 + m_3(q_3 - l_3)^2 + m_0 q_3^2] \ddot{q}_1 + 2[m_3(q_3 - l_3) + m_0 q_3] \dot{q}_1 \dot{q}_3 + f_{v1} \dot{q}_1 \\ F_2 = (m_0 + m_2 + m_3) \ddot{q}_2 + (m_0 + m_2 + m_3)g + f_{v2} \dot{q}_2 \\ F_3 = (m_0 + m_3) \ddot{q}_3 - [m_3(q_3 - l_3) + m_0 q_3] \dot{q}_1^2 + f_{v3} \dot{q}_3 \end{cases} \quad (\text{Ann3.1})$$

En inversant le système :

$$\begin{aligned} \ddot{q}_1 &= \frac{1}{[I_2 + I_3 + m_3(q_3 - l_3)^2 + m_0 q_3^2]} \{- 2[m_3(q_3 - l_3) + m_0 q_3] \dot{q}_1 \dot{q}_3 - f_{v1} \dot{q}_1 + \Gamma_1\} \\ \ddot{q}_2 &= -\frac{f_{v2}}{(m_0 + m_2 + m_3)} \dot{q}_2 - g + \frac{F_2}{(m_0 + m_2 + m_3)} \\ \ddot{q}_3 &= \frac{[m_3(q_3 - l_3) + m_0 q_3]}{(m_0 + m_3)} \dot{q}_1^2 - \frac{f_{v3}}{(m_0 + m_3)} + \frac{F_3}{(m_0 + m_3)} \end{aligned} \quad (\text{Ann3.2})$$

Pour obtenir le modèle d'état, on introduit la notation suivante :

$$\begin{bmatrix} q_1 \\ \dot{q}_1 \\ q_2 \\ \dot{q}_2 \\ q_3 \\ \dot{q}_3 \end{bmatrix} \longleftrightarrow \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} ; \quad u^T = [\Gamma_1 \quad F_2 \quad F_3]$$

Le modèle d'état est alors :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{1}{[I_2 + I_3 + m_3(x_5 - l_3)^2 + m_0 x_5^2]} \{-2[m_3(x_5 - l_3) + m_0 x_5]x_2 x_6 - f_{v1} x_2 + u_1\} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -\frac{f_{v2}}{(m_0 + m_2 + m_3)} x_4 - g + \frac{u_2}{(m_0 + m_2 + m_3)} \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = \frac{[m_3(x_5 - l_3) + m_0 x_5]}{(m_0 + m_3)} x_2^2 - \frac{f_{v3}}{(m_0 + m_3)} x_6 + \frac{u_3}{(m_0 + m_3)} \end{cases} \quad (\text{Ann3.3})$$

Ou encore :

$$\begin{aligned} \Gamma_1 &= (I_2 + I_3 + m_3(x_5 - l_3)^2 + m_0 x_5^2) \dot{x}_2 + 2(m_3(x_5 - l_3) + m_0 x_5) x_2 x_6 + f_{v1} x_2 \\ F_2 &= (m_0 + m_2 + m_3) \dot{x}_4 + (m_0 + m_2 + m_3) g + f_{v2} x_4 \\ F_3 &= (m_0 + m_3) \dot{x}_6 - (m_3(x_5 - l_3) + m_0 x_5) x_2^2 + f_{v3} x_6 \end{aligned} \quad (\text{Ann3.4})$$

On peut écrire :

$$\begin{cases} \tau_1 = F_1(x_5) \dot{x}_2 + G_1(x_2, x_5, x_6) \\ \tau_2 = F_2(0) \dot{x}_4 + G_2(x_4) \\ \tau_3 = F_3(0) \dot{x}_6 + G_3(x_2, x_5, x_6) \end{cases} \quad (\text{Ann3.5})$$

Ou encore :

$$\begin{cases} \tau_1 = f_1(\dot{x}_2, x_2, x_5, x_6) \\ \tau_2 = f_2(\dot{x}_4, x_4) \\ \tau_3 = f_3(\dot{x}_6, x_2, x_5, x_6) \end{cases} \quad (\text{Ann3.6})$$

Les paramètres du robot sont :

$$\begin{array}{llll}
 l_3 = 0.75 \text{ m} & m_2 = 20 \text{ kg} & m_3 = 10 \text{ kg} & I_2 = 2 \text{ kg m}^2 \\
 I_3 = 0.2 \text{ kg m}^2 & f_{v1} = 7.825 \text{ N rad}^{-1} \text{ s} & f_{v2} = 30 \text{ N m}^{-1} \text{ s} & f_{v3} = 20 \text{ N m}^{-1} \text{ s}
 \end{array}$$

Référence cycloïde :

Soient $q_{\hat{a}}$, $\dot{q}_{\hat{a}}$, $\ddot{q}_{\hat{a}}$ les références ou les consignes à suivre (position, vitesse, accélération) ; telles que :

$$q_{\hat{a}}(t) = \begin{cases} q_{\hat{a}}(0) + \frac{D_i}{2\pi} \left[2\pi \frac{t}{t_f} - \sin\left(2\pi \frac{t}{t_f}\right) \right] & 0 \leq t \leq t_f \\ q_{\hat{a}}(t_f) & t > t_f \end{cases}$$

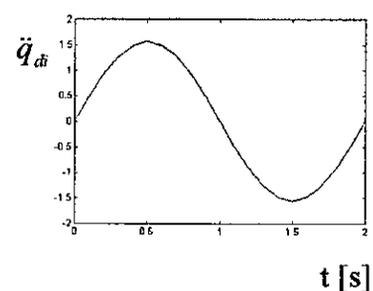
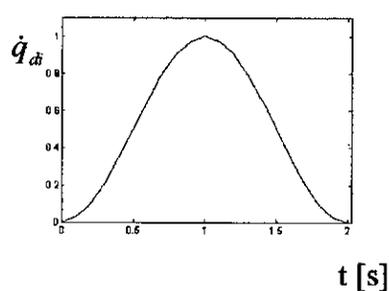
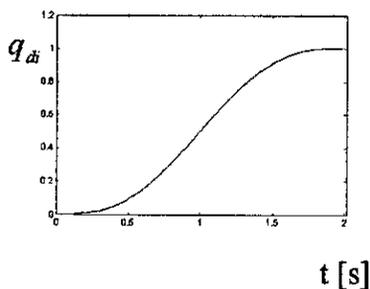
$$\dot{q}_{\hat{a}}(t) = \begin{cases} \frac{D_i}{2\pi} \left[\frac{2\pi}{t_f} - \frac{2\pi}{t_f} \cos\left(2\pi \frac{t}{t_f}\right) \right] & 0 \leq t \leq t_f \\ 0 & t > t_f \end{cases}$$

$$\ddot{q}_{\hat{a}}(t) = \begin{cases} \frac{D_i}{2\pi} \left[0 + \left(\frac{2\pi}{t_f}\right)^2 \sin\left(2\pi \frac{t}{t_f}\right) \right] & 0 \leq t \leq t_f \\ 0 & t > t_f \end{cases}$$

avec :

$$D_i = q_{\hat{a}}(t_f) - q_{\hat{a}}(0)$$

où $q_{\hat{a}}(t)$, $q_{\hat{a}}(0)$ sont respectivement la position initiale et la position finale.



ANNEXE 4

THEOREME DE LYAPOUNOV :

Les théorèmes de LYAPOUNOV de base comprennent deux méthodes introduites par LYAPOUNOV, la méthode directe et la méthode indirecte.

La méthode indirecte ou linéarisante énonce que les propriétés de la stabilité d'un système non linéaire dans un voisinage non fermé d'un point d'équilibre sont les mêmes que ceux de son approximation linéarisée, la méthode sert comme une justification théorique à l'usage des systèmes linéaires pour l'étude des systèmes physiques qui sont toujours naturellement non linéaire.

La méthode directe est un outil puissant pour l'analyse des systèmes non linéaires, c'est pourquoi ce qui est appelé analyse de LYAPOUNOV souvent fait allusion à la méthode directe.

La méthode directe est une généralisation du concept de l'énergie associé à un système mécanique ; le mouvement d'un système mécanique (ou électrique) est stable si son énergie totale décroît dans le temps, en effet si elle se dissipe d'une manière continue, le système qu'il soit linéaire ou non, doit éventuellement s'établir en un point d'équilibre, ainsi on peut analyser la stabilité en examinant la variation d'une seule fonction scalaire.

En comparant les définitions de la stabilité et de l'énergie mécanique, on peut facilement constater des relations entre elles [10] :

- L'énergie égale à zéro correspond à un point d'équilibre.
- La stabilité asymptotique implique la convergence de l'énergie mécanique à zéro.
- L'instabilité est reliée à l'augmentation de l'énergie mécanique.

Ces relations indiquent que la valeur de la quantité scalaire, l'énergie mécanique, reflète indirectement la grandeur du vecteur d'état, et de plus, que les propriétés de la stabilité d'un système peuvent être caractérisées par la variation de son énergie mécanique.

Dans l'usage de la méthode directe, l'idée est de construire une fonction scalaire semblable à celle de l'énergie, ou plus exactement une fonction de LYAPOUNOV et de voir si elle décroît.

La puissance de cette méthode vient de ses généralités, elle est appliquée à toutes sortes de commandes, qu'elles soient variantes ou invariantes dans le temps, de dimension finies ou infinies, réciproquement la limitation de la méthode est liée au fait que c'est souvent difficile de trouver une fonction de LYAPOUNOV pour un système donné.

Même si la méthode de LYAPOUNOV directe est une méthode à l'originale destinée à l'analyse de la stabilité, elle peut être utilisée dans d'autres problèmes de synthèse de la commande non linéaire, c'est d'ailleurs une des approches les plus fructueuses .

THEOREME DE LYAPOUNOV POUR LA STABILITE GLOBALE :

Supposons qu'il existe une fonction scalaire V fonction des éléments du vecteur d'état x , avec une dérivée du premier ordre \dot{V} tel que :

- $V(x)$ est défini positive
- $\dot{V}(x)$ est défini négative $\forall x \neq 0$
- $V(x) \rightarrow \infty$ quand $\|x\| \rightarrow \infty$

Alors l'équilibre à l'origine est globalement asymptotiquement stable.

ETUDE DE LA STABILITE AU SENS DE LYAPOUNOV :

Après avoir supposer la forme de la fonction de LYAPOUNOV $V(x)$, le but est d'arriver à $\dot{V}(x)$ négative pour vérifier la condition du théorème précédant, la troisième condition étant vérifiée, de là on distingue trois cas différents selon l'écriture de la dérivée $\dot{V}(x)$:

- $\dot{V}(x)$ est fonction de tous les éléments du vecteur d'état, de ce fait ces derniers tendent tous vers zéro asymptotiquement.
- $\dot{V}(x) = -\lambda \cdot V(x)$, $\lambda \in \mathbb{R}^+$ les variables d'état tendent exponentiellement vers zéro.
- $\dot{V}(x)$ s'écrit en fonction de quelques éléments du vecteur d'état, ce qui fera tendre ces derniers vers zéro asymptotiquement les autres seront seulement bornées et ceci d'après le lemme de BARBALAT.

Bibliographie

- [1] Alienne Yves ; « Systèmes de contrôle : Logique flou »
Décembre 1994 (Internet)
- [2] Fischle.K , Schröder.D ; « An Improved Adaptive Fuzzy Control Method»
Fuzzy systems, IEEE Transaction on Published , Feb 1999
Vol.71 Pages : 27 – 40
- [3] KIM.Y.H , LEWIS.F.L. ; « Direct-reinforcement-adaptive-learning fuzzy logic
control for a class of non linear systems»
Intelligent control, 1997.Proceedings of the 1997
IEEE International Symposium on, 1997 pages : 281 – 286
- [4] Yoram Koren ; « La robotique pour ingénieurs »
Copyright © 1986, by McGraw-Hill, Paris pour l'édition française
ISBN : 2 – 7042 – 1133 – 7
(Edition original ISBN : 0 – 07 – 035399 – 9 McGraw-Hill, Newyork)
- [5] KWAN.E , MING Liu ; « An Adaptive Fuzzy Approach for Robot Manipulators
Tracking »
Computational Intelligence in Robotics and Automation, 199. CIRA.
Proceedings .1999 IEEE International Symposium on, 1999.
Pages : 53 – 58
- [6] LABIOD (S) ; « Commande adaptative par les systèmes flous. Application aux robots
manipulateurs »
Alger : ENP, 1998.- 97 f
Mag. : Auto. : Alger : 1998
- [7] MADANI (T) , AMRATI (S) ; « Commande décentralisée à structure variable.
Application en robotique »
P.F.E. 1997.- 128 f
- [8] Alain PRUSKI ; « Robotique générale »
Edition MARKETING 1988
- [9] HENK SCHOLTEN ; « Logique floue et régulateur PID »
© 1994 Elektor, première édition, deuxième édition
impression août 1995
Publitronic

- [10] Jean-Jacques, E.SLOTINE, « Applied Nonlinear Control »
Weiping Li Prentice Hall 1991
Englewood cliffs, NewJersy 07632
- [11] Claude Vibert ; « Robots Principes et Contrôle »
Edition MARKETING (éditeur des préparations grandes écoles de médecine)
Copyright 1987.
- [12] L.X.Wang USA ; « Stable Adaptive Fuzzy Control of Nonlinear Systems »
IEEE Transaction on Fuzzy systems
Vol.1, no.2, pp. 146 – 55, May 1993

ملخص

نقدم في هذا العمل 4 طرق للتحكم المبهم المتكيف المباشر للذراع الآلي ذو البنية الاسطوانية، الاوليتين قائمة على التمثيل المباشر للذراع الآلي أما التاليتين فعلى التمثيل المعكوس. التحكم كان بالطريقة المباشرة، أي أن عوامل التحكم تضبط مباشرة، بحيث يزول خطأ الخروج نحو الصفر، هذه العوامل تنتمي للنظام المبهم لـ Sugeno من الدرجة 0 إلى الدرجة 1 مع اساس قواعد مبهمه مشكلة من كل الترتيبات الممكنة لمجموعات الانتماء من نوع Gaussienne ماعدا الطريقة الاخيرة أين يستعمل فيها ضبط مجموعات الانتماء مما إعادة البناء . هدفنا يتمحور في تحقيق اتباع جيد لمملك المرجع Cycloïde مع ضمان قيم للأخطاء، قمنا بعدة إختبارات لقوة النظام، اثبتت سرعة إنضباط العوامل، كما إستطعنا تخفيف الغورتم التمثيل الآلي للنظام بالعمل لامركزيا .

الكلمات المفاتيح : تحكم مبهم ، تحكم متكيف مباشر ، ذراع آلي ، بنية أسطوانية

ABSTRACT :

In this work, we present 4 methods of fuzzy direct adaptive control applied to the robot with cylindrical structure, the two first methods are based on the direct model of the robot, the followings on the reverse model. The development of the control law has been made by the direct method; the controller's parameters are adjusted in such way to bring back the tracking error to 0, based on the Lyapunov stability. These parameters are those of the fuzzy systems of Sugeno, with the fuzzy rules base organized on the combination of the fuzzy sets (Gaussienne), for the last control law the fuzzy sets are adapted, this allows to reduce considerably the rules number.

Our goal is to achieve a good tracking of the reference trajectory (Cycloïde), with a weak error value. Furthermore the robust tests have shown that the controller's parameters adaptation is fast. The control law algorithms are reduced in decentralized case.

Keywords: fuzzy control, direct adaptive control, robot manipulator, cylindrical structure.

RESUME:

Dans ce travail, nous présentons 4 commandes floues adaptatives directes appliquées au robot à structure cylindrique, les deux premières se basent sur le modèle direct du robot les deux autres sur le modèle inverse. Les commandes sont du type direct ; les paramètres du contrôleur sont ajustés de telle façon à ramener l'erreur de poursuite à 0, en se basant sur la stabilité de Lyapunov. Ces paramètres sont ceux des systèmes flous de Sugeno avec une base des règles floues constituée de la combinaison des ensembles flous du type Gaussien. En ce qui concerne la dernière commande on effectue une adaptation des ensembles flous ce qui permet de réduire considérablement la base des règles.

Notre but est de réaliser une bonne poursuite de la trajectoire de référence (Cycloïde), en ayant une valeur minimale de l'erreur. Les tests de robustesse mis en œuvre ont montré une adaptation rapide des paramètres de la commande, mieux que cela nous somme parvenus à alléger les algorithmes de commandes en utilisant la méthode décentralise.

Mots clés : commande floue, commande adaptative directe, robot manipulateur, structure cylindrique.