

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



UNIVERSITE DE M'SILA  
FACULTE DE TECHNOLOGIE  
DEPARTEMENT DE GENIE ELECTRIQUE

MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLÔME  
DE MASTER EN GENIE ELECTRIQUE  
SPECIALITE: AUTOMATIQUE

## THEME

---

**Commande adaptative floue pour une classe de  
systèmes non linéaires à temps discret.**

---

Proposé et dirigé par :

- Mr. A. BOUKHALFA

Présenté par :

- LEFREID Kamel

Année Universitaire: 2011/2012

N° d'ordre : 012

# *Remerciements*

*Tout d'abord merci à DIEU de m'avoir donné la force  
pour  
Terminer ce travail*

Je tiens à remercier vivement monsieur **Mr. BOUKHALFA .A** pour son esprit scientifique et compréhensif, qui a consacré beaucoup de son temps à mon travail et m'a beaucoup aidé avec ses idées, ses conseils et surtout ses critiques objectives.

Je tiens également à remercier monsieur **Dr. BERKAT.S** pour son enthousiasme et son soutien sans faille durant la réalisation de ce mémoire.

Je remercie les membres de jury qui nous font l'honneur de présider et d'examiner ce modeste travail.

J'adresse aussi mes remerciements à **tous** mes amis et collègues surtout ceux qui m'ont apportés un soutien moral, patience et une amitié inoubliable. Enfin, je ne saurais terminer ce remerciement sans remercier mes parents, qui mon accompagné tout au long de mes études par leur amour inconditionnel et leur soutien constant.

# Dédicaces

*Je dédie ce modeste travail  
A tous les membres de ma famille grande et  
petite.  
A ma mère avec toute m'affection.  
A mon père avec toute ma reconnaissance.  
A mes frères Mourad, Moustafa, Rachid, Adel,  
Khalil, Fatima, roukia, Naïma, sans oublier le  
cajoleur Mohamed loai.  
A tous les enseignants qui m'ont aidé de proche  
ou de loin pour être un jour un Master.  
A mes collègues Saleh et Abdelmalek .B  
,Magique et marouf. A , mohamed .C, Ben  
kada .y, Khalil H ,Zeddani .A, Kehoul Omar  
qui m'a accompagné pendant le long de cette  
période pour réaliser ce modeste travail.  
A tous mes amis. A tous les étudiants du  
département d'électrotechnique, surtout les  
étudiants de la Master 2 Automatique  
promotion 2012.  
A tous ceux qui m'ont aidé, de près ou de loin,  
même qu'il soit un mot d'encouragement et de  
gentillesse.  
A tous ceux que j'aime et qui m'aiment.  
A tous ceux qui connaissent lefried Kamel*

*lefried Kamel*

# Sommaire

<b>REMERCIEMENTS</b> .....	<b>I</b>
<b>DÉDICACES</b> .....	<b>II</b>
<b>SOMMAIRE</b> .....	<b>III</b>
<b>TABLE DE FIGURE</b> .....	<b>V</b>
<b>LISTE DES SYMBOLES</b> .....	<b>V</b>
<b>Introduction générale</b> .....	<b>1</b>

## *Chapitre I* *Commande par logique floue*

<b>I.1 Introduction</b> .....	<b>3</b>
<b>I.2. Logique Floue</b> .....	<b>3</b>
I.2.1.Définition.....	3
I.2.2.Les Avantages de la logique floue .....	3
I.2.3.Les Inconvénients.....	4
I.2.4.Application de La Logique Floue:.....	4
<b>I.3.La Commande Floue</b> .....	<b>4</b>
I.3.1.But de La Commande floue .....	5
I.3.2.Description générale d'un contrôleur flou .....	5
I.3.3.Les Bases de La Commande Floue.....	6
I.3.3.1.Ensemble Flou.....	6
<i>I.3.3.1.1.Définition</i> .....	6

# Sommaire

---

I.3.3.1.2. Opérations sur les ensembles.....	6
I.3.3.2. Variables linguistiques .....	8
I.3.3.3. Fonction d'Appartenance.....	9
I.3.3.3.1. Formulations mathématique des fonctions d'appartenance .....	10
<b>I.4. Structure d'une commande floue.....</b>	<b>11</b>
I.4.1. Interface de fuzzification .....	11
I.4.2. Moteur d'inférence floue .....	12
I.4.2.1. Traitement numérique des inférences .....	12
I.4.3. Interface de Défuzzification.....	14
<b>I. 5. Approximation floue.....</b>	<b>16</b>
<b>I.6. Contrôleurs flous usuels.....</b>	<b>16</b>
<b>I.7. Conclusion.....</b>	<b>18</b>

## *Chapitre II*

### *Commande Adaptative*

<b>II.1. Introduction.....</b>	<b>19</b>
<b>II.2. Historique.....</b>	<b>19</b>
<b>II.3. Définition de la commande adaptative.....</b>	<b>21</b>
<b>II.4. Généralités sur la commande adaptative .....</b>	<b>23</b>
<b>II.5. Pourquoi la commande adaptative .....</b>	<b>23</b>
<b>II.6. Principe .....</b>	<b>24</b>
<b>II.7. Commande adaptative directe et indirecte .....</b>	<b>25</b>
II.7.1. commande adaptative directe.....	25
II.7.2 commande adaptative indirecte.....	26
<b>II.8. Stabilité.....</b>	<b>26</b>

# Sommaire

---

II.8.1. Méthode direct de Lyapunov .....	27
II.8.2.Méthode indirect de Lyapunov .....	27
<b>II.9. La zone morte .....</b>	<b>28</b>
<b>II.10.Conclusion .....</b>	<b>29</b>

## *Chapitre III*

### *Commande adaptative floue indirecte des systèmes non linéaires discrets*

<b>III.1.Introduction .....</b>	<b>30</b>
<b>III.2.Conception d'un contrôleur adaptatif flou indirect et analyse de stabilité.....</b>	<b>31</b>
<b>III.3.Erreurs d'estimations des paramètres .....</b>	<b>33</b>
<b>III.4. Simulation .....</b>	<b>38</b>
<b>III.5.Résultats de simulation.....</b>	<b>39</b>
<b>III.7.Conclusion .....</b>	<b>42</b>

## *Chapitre IV*

### *Commande adaptative floue directe des systèmes non linéaires discrets*

<b>IV.1.Introduction.....</b>	<b>43</b>
<b>IV.2.Commande Adaptative Floue Directe:.....</b>	<b>44</b>
<b>IV.3.Exemple de simulation.....</b>	<b>49</b>
<b>IV.4.Résultats de simulation:.....</b>	<b>51</b>
<b>IV.5. Conclusions.....</b>	<b>52</b>
<b>Conclusion générale.....</b>	<b>53</b>
<b>Bibliographie.....</b>	

## LISTES DES FIGURES

<b>Figure I.1:</b> Configuration de base d'un système de commande floue .....	05
<b>Figure I.2 :</b> Représentation floue de la variable Vitesse .....	09
<b>Figure I.3 :</b> Représentation graphique de sous-ensemble dont les ensembles de référence est infinis .....	09
<b>Figure I.4 :</b> Forme des fonctions d'appartenances usuelles .....	10
<b>Figure I.5 :</b> structure de base d'un système flou .....	11
<b>Figure I.6 :</b> Principe des différentes méthodes de défuzzification. ....	15
<b>Figure II.1 :</b> Schéma bloc d'une commande adaptative .....	22
<b>Figure II.2 :</b> Schéma de principe d'une commande adaptative .....	25
<b>Figure II.3 :</b> Commande adaptative direct .....	25
<b>Figure II.4 :</b> Commande adaptative indirect .....	26
<b>Figure II.5 :</b> La zone morte .....	28
<b>Figure III.1 :</b> La sortie de système $y(k)$ et signal de référence $r(k)$ . ....	39
<b>Figure III.2 :</b> L'erreur $e(k)$ et la taille de la zone morte $\pm (\ )$ .....	39
<b>Figure III.3 :</b> Le signal de Commande $u(k)$ .....	40
<b>Figure III.4 :</b> Loi d'adaptation $\theta_f (1 : 9)$ . ....	40
<b>Figure (III.5) :</b> Loi d'adaptation $\theta_f (10 : 18)$ .....	41
<b>Figure (III.6) :</b> Loi d'adaptation $\theta_f (19 : 27)$ .....	41
<b>Figure (III.7) :</b> Loi d'adaptation $\theta_g (1 : 9)$ .....	41
<b>Figure (III.8) :</b> Loi d'adaptation $\theta_g (10 : 18)$ .....	42
<b>Figure (III.9) :</b> Loi d'adaptation $\theta_g (19 : 27)$ .....	42
<b>Figure IV.1 :</b> Système de réservoir .....	49
<b>Figure IV.2 :</b> la sortie $y(k)$ et le signal de référence $r(k)$ .....	51
<b>Figure IV.3 :</b> la Commande $u(k)$ .....	51
<b>Figure IV.4 :</b> L'erreur $e(k)$ et la taille de la zone morte $\pm (k)$ .....	52
<b>Figure IV.5:</b> Les lois d'adaptation $\theta (k)$ .....	52

## ABRÉVIATIONS ET SYMBOLES

SISO : Système Mono Entrées Mono Sorties.

DSP : Signal Numérique de Processeur (Digital Signal Processor)

L.E.S : Linéarisation Entrée-Sortie.

$\omega(k)$  : Erreur d'approximation minimale.

$\pm\delta(k)$  : La taille de la zone morte.

$\mu_F$  : La degré d'appartenance de l'élément  $x$ .

V : Fonction de Lyapunov.

$\theta_f, \theta_g, \theta$  : Les Lois d'adaptations.

T : période de chantournage.

$x_1, x_2, x_3$  : Les variables d'entrée du système flou .

$r(k)$  : signal de référence.

$y(k)$  : La sortie de système.

$u(k)$  : La commande de système.

$\rho_f(k), \rho_g(k)$  : Les estimations.

# Introduction générale

Les techniques de commande classique (approche algorithmique) consistent dans un premier temps à construire un modèle mathématique à base d'équations différentielles ou récurrentes, du système à piloter. A partir de ce modèle, une commande est déterminée (P.I.D, commande par retour d'état, commande optimale,...) afin d'amener ce système dans les états désirés, tout en respectant des critères de performance (durée des transitoires, minimisation de l'énergie apportée,...). En effet l'obtention d'un modèle mathématique à la fois précis, pertinent et simple à exploiter, est parfois difficile et complexe. Plusieurs commandes sont apparues pour vaincre ce problème, on cite la commande adaptative floue.

La logique floue s'y développe car il s'agit d'une approche essentiellement pragmatique, efficace et générique. On dit parfois qu'elle permet de systématiser ce qui est du domaine de l'empirisme, et donc difficile à maîtriser. La théorie des ensembles flous fournit une méthode pertinente et facilement réalisable dans des applications temps réel, elle permet de transcrire et rendre dynamiques les connaissances des concepteurs ou des opérateurs.

La logique floue est bien connue des automaticiens pour ses applications dans la commande de procédés, appelé alors couramment « contrôle flou ». Tout comme un contrôleur (ou correcteur) classique, le contrôleur flou s'insère dans la boucle de régulation et calcule la commande à appliquer au procédé suivant une ou plusieurs consignes et une ou plusieurs mesures effectuées sur celui-ci.

La commande floue génère des régulateurs non linéaires qui sont en concordance avec le fait qu'ils sont des approximateurs universels [WAN.94]. Il devient alors possible de construire un régulateur flou pour n'importe quel système non linéaire. Le plus souvent les régulateurs flous sont utilisés dans des systèmes qui possèdent des variations inconnues intrinsèques. L'objectif est donc de conserver de bonnes performances du système complet en adaptant le régulateur en fonction des variations du système.

De nombreux travaux réalisés utilisent la commande adaptative floue pour les systèmes non linéaires.

Généralement, l'objectif principal de la commande adaptative est de maintenir les performances d'un système. Par conséquent, la commande avancée floue doit être adaptative.

L'objectif de ce travail est le développement de structures de commandes adaptatives floues stables pour des systèmes non linéaires SISO discrets.

L'ensemble des travaux réalisés est regroupé en quatre chapitres :

Dans le **premier chapitre**, nous définissons le formalisme de la logique floue et le raisonnement associé et nous présentons la structure générale d'un contrôleur flou avec les différents modèles utilisés dans sa conception ainsi que le théorème de l'approximation floue.

Dans le **deuxième chapitre**, on présente la commande adaptative avec ses différentes structures directe et indirecte, ces deux types de commandes seront utilisés pour commander une classe des systèmes non linéaires SISO discrets.

Dans le **troisième chapitre**, nous développons une approche adaptative indirecte floue pour commander une classe des systèmes non linéaires avec l'analyse de la stabilité par la synthèse de Lyapunov.

Le **quatrième chapitre** est consacré au développement d'une loi de commande adaptative directe floue pour une classe de système non linéaire qui utilise un contrôleur idéal implicite flou pour assurer les objectifs de commande.

Le bilan des travaux réalisés, dans le cadre de ce mémoire, et les perspectives envisagées sont donnés en conclusion.

# Chapitre I

Commande par logique floue

## I.1 Introduction

La logique booléenne, fondement de l'informatique classique, repose sur deux valeurs, le zéro et le un. Toute proposition ainsi traitée est alors considérée vraie ou fausse. Or, dans le monde réel, rares sont les catégories qui supportent une telle classification de tout ou rien. Par exemple, peut-on considérer une eau à 180°C comme étant chaude ou froide ? N'est-elle pas ni vraiment chaude, ni vraiment froide mais tout simplement tiède ? [1].

La logique floue a été introduite par le professeur Lotfi Zadeh en 1965 comme une généralisation de la logique binaire [1]. L'intérêt de la logique floue réside dans sa capacité à traiter l'imprécis, l'incertain et le vague. Elle est issue de la capacité de l'homme à décider et agir de façon pertinente malgré le flou des connaissances disponibles. En effet, la logique floue a été introduite pour rapprocher le raisonnement humain à l'aide d'une représentation adéquate des connaissances [2].

La logique floue est une technique pour le traitement de connaissances imprécises basées, sur des termes linguistiques; elle donne les moyens de convertir une commande linguistique basée sur le raisonnement humain, en une commande automatique, permettant ainsi la commande des systèmes complexes dont les informations sont exprimées d'une façon vague et mal définie.

Dans le domaine du génie électrique, la commande par logique floue a fait l'objet de plusieurs travaux : dans la commande des convertisseurs statiques et dans la commande des machines électriques, dans la navigation de robots mobiles. Toutes ces applications ont démontré qu'un régulateur par logique floue est plus robuste qu'un régulateur conventionnel.

## I.2. Logique Floue

### I.2.1. Définition

Logique floue: « logique qui substitue à la logique binaire une logique fondée sur des variables pouvant prendre, outre les valeurs « vrai » ou « faux », les valeurs intermédiaires « vrai » ou « faux » avec une certaine probabilité ». [3]

### I.2.2. Les Avantages de la logique floue [4]

- Nous n'avons pas besoin d'établir le modèle mathématique. Cela évite des années de travail sur des modèles.

- Ensuite la logique floue permet de prendre en compte les typologies linguistiques et donc toutes les nuances que l'on a créées dans le but d'imiter au mieux le réel.
- Traitement des systèmes à comportement complexe (même si on a un exemple simplifié, cela reste un système complexe en mathématique pure).
- Nous pouvons aujourd'hui servir des solutions matérielles comme des microprocesseurs uniquement dédiés aux calculs flous, ce qui permet d'accroître le rendement puisque cela réduit la programmation.
- L'ajoute ou le retrait des règles ne change quasiment rien au principe de la logique floue, alors qu'en logique bivalente (logique classique), cela peut changer du tout au tout.
- On peut augmenter facilement le nombre d'entrées sans réellement compliquer les calculs (alors qu'en logique simple la taille des calculs est exponentielle).

### **I.2.3. Les Inconvénients [4]**

- réglage empirique.
- performances en boucle fermée dépendent de l'expertise.
- il n'existe pas de théorie générale qui caractérise rigoureusement la stabilité, la robustesse (difficulté de certification dans le transport, espace).

### **I.2.4. Application de La Logique Floue:**

Les domaines d'applications de la logique floue concernent principalement les problèmes où les données qui ne peuvent être formulées de manière explicite, ainsi que des techniques de contrôle et de réglages, lorsque les moyens classiques atteignent leurs limites (exemples systèmes non linéaires, etc.). La logique floue est bien connue par des automaticiens pour ses applications dans le contrôle et commande de procédés, appelé alors couramment « contrôle flou ». Tout comme un contrôleur (ou correcteur) classique [5], Son but est de traiter les problèmes de commande de processus avec une approche différente de l'automatique classique.

## **I.3. La Commande Floue**

Les contrôleurs flous sont des contrôleurs basés sur des règles linguistiques. La commande conventionnelle commence par un modèle mathématique du système,

et les contrôleurs sont conçus pour ce modèle. La commande floue commence par l'heuristique et l'expertise humaine, et les contrôleurs sont conçus par la synthétisation de ces heuristiques et expertise humaine. Bien que les points de départ des deux approches soient différents, les produits finaux sont identiques - des contrôleurs non linéaires pour les systèmes non linéaires, à cet égard, la théorie de commande floue peut être regardés comme sous-ensemble de théorie de commande non linéaire dans lequel les contrôleurs non linéaires ont une structure basée sur des règles spéciales.

### I.3.1. But de La Commande floue

la commande floue a pour but de traiter des problèmes de commande classique de processus à partir uniquement de connaissances de comportement que les spécialistes du procédé doivent formuler sous forme linguistique (floue).

### I.3.2. Description générale d'un contrôleur flou

En générale, les contrôleurs flous sont utilisés dans les structures de commande des processus en boucle fermée (figure (I.1)).

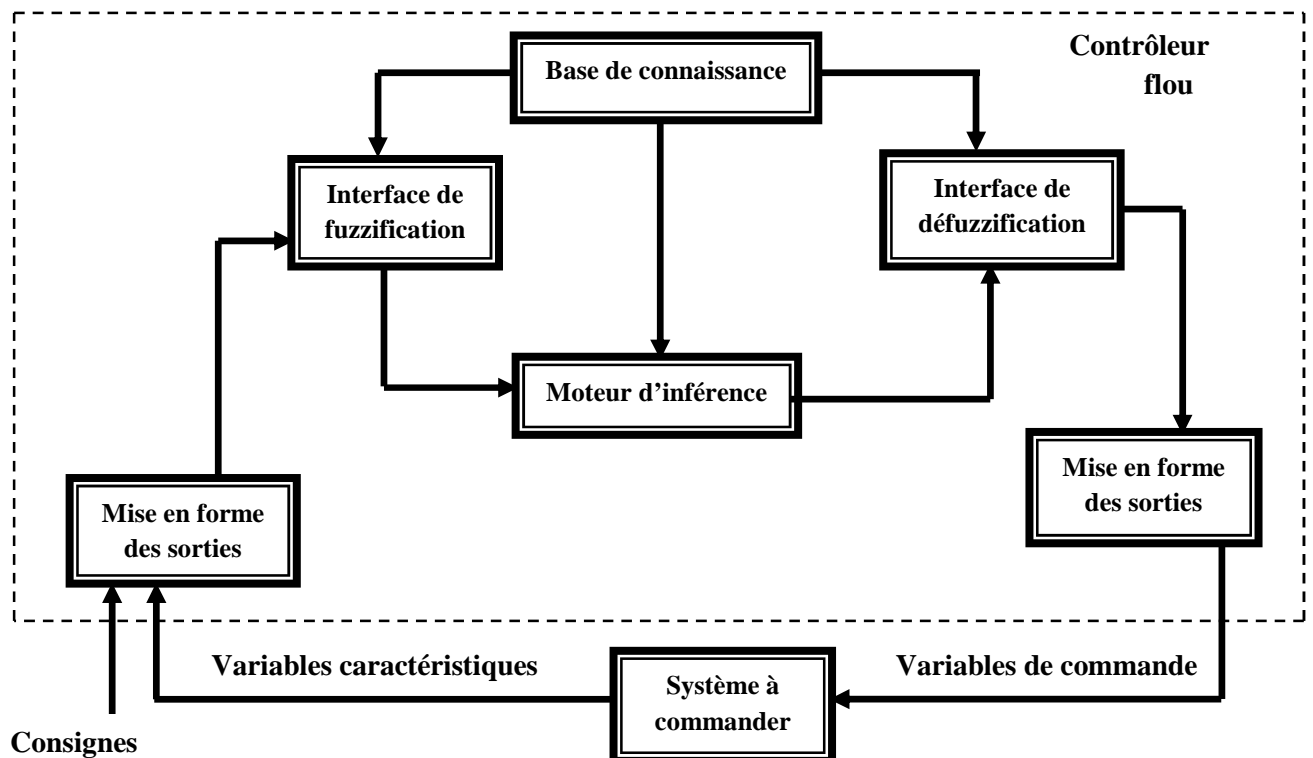


Figure (I.1): Configuration de base d'un système de commande floue.

### I.3.3. Les Bases de La Commande Floue

#### I.3.3.1. Ensemble Flou

##### I.3.3.1.1. Définition

La notion d'ensemble flou a été l'origine de l'apparition de la logique floue. Et pour mettre en évidence cette notion nous présentons dans cette section différentes définitions de cette notion [9] :

Zadeh donna la définition suivante : « Un ensemble flou est une classe d'objets qui lui appartient ayant un degré d'appartenance. Chaque ensemble flou est caractérisé par une fonction d'appartenance qui affecte à chaque objet un degré d'appartenance entre 0 et 1 »

Après Zadeh plusieurs auteurs ont donné d'autres définitions à la logique floue parmi eux Zimmerman (1990) qui a écrit :

« Un ensemble flou  $F$  est indiqué par un ensemble de paires ordonnées  $(x, \mu_F(x))$ . Le premier élément de chaque paire dénote l'élément  $x$ , et le deuxième élément  $\mu_F(x)$ , dénote le degré d'appartenance de l'élément  $x$ , où  $\mu_F$  varie dans l'intervalle  $[0,1]$ . »

##### I.3.3.1.2. Opérations sur les ensembles

Comme pour les ensembles classiques on peut associer deux ensembles flous pour obtenir un nouvel ensemble flou.

Soient les deux ensembles flous  $A$  et  $B$ , et leurs fonctions d'appartenances  $\mu_A(x), \mu_B(x)$  respectivement, Les opérations sur les ensembles flous sont définies via leurs fonctions d'appartenances comme suit :

##### Complément :

Le complément de l'ensemble flou  $A$  est défini par [6], [7], [8], [9] :

$$\bar{A} = \{(x, \mu_{\bar{A}}(x)) / x \in U\} \quad (I.1)$$

Où  $\mu_{\bar{A}}(x) = 1 - \mu_A$

**Intersection :**

L'intersection des ensembles flous  $A$  et  $B$  dans l'univers du discours  $U$  est donné par :

$$A \cap B = \left\{ \left( x, \mu_{A \cap B}(x) \right) / x \in U \right\} \quad (I.2)$$

Où  $\mu_{A \cap B}(x) = \mu_A(x) \tilde{\wedge} \mu_B(x)$

Le symbole  $\tilde{\wedge}$  dénote le « ET flou ».

L'intersection peut être réalisée par des opérations arithmétiques. En effet, Zadeh a suggéré d'utiliser les opérateurs « min » et « le produit algébrique ». Après Zadeh plusieurs chercheurs ont proposé d'autres opérateurs pour l'intersection. Nous citons ici les plus utilisés [11]:

**Min :**

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (I.3)$$

**Produit algébrique :**

$$\mu_{A \cap B}(x) = \mu_A(x) \cdot \mu_B(x) \quad (I.4)$$

**La différence bornée :**

$$\mu_{A \cap B}(x) = \max\left[0, \mu_A(x) + \mu_B(x) \mu_A - 1\right] \quad (I.5)$$

Ces opérateurs sont appelés « T-Normes ».

**Union :**

L'union des deux ensembles flous  $A$  et  $B$  dans l'univers du discours  $U$  est donné par :

$$A \cup B = \left\{ \left( x, \mu_{A \cup B}(x) \right) / x \in U \right\} \quad (I.6)$$

Où  $\mu_{A \cup B}(x) = \mu_A(x) \dot{+} \mu_B(x)$

Le symbole  $\dot{+}$  dénote le « OU flou ».

**Max :**

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (I.7)$$

**La somme algébrique :**

$$\mu_{A \cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x) \quad (I.8)$$

**La somme bornée :**

$$\mu_{A \cup B}(x) = \max\{1, \mu_A(x) \cdot \mu_B(x)\} \quad (I.9)$$

**Relation floue :**

Soient  $U_1, U_2, \dots, U_n$  des univers de discours. Une relation floue est un ensemble flou dans, exprimé par :

$$R_{U_1 \times \dots \times U_n} = \left\{ \left( (x_1, \dots, x_n), \mu_R(x_1, \dots, x_n) / (x_1, \dots, x_n) \in U_1 \times \dots \times U_n \right) \right\} \quad (I.10)$$

### I.3.3.2. Variables linguistiques

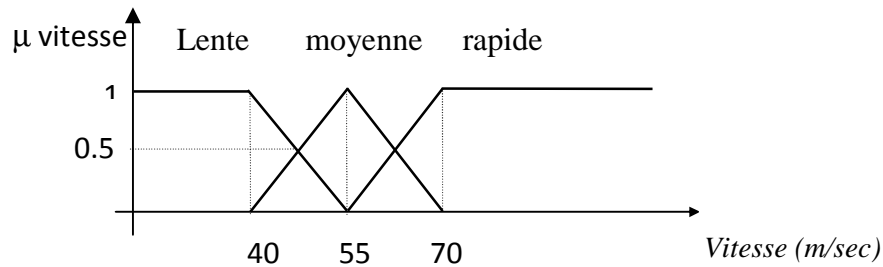
La notion de variable linguistique permet de modéliser les connaissances imprécises ou vagues sur une variable dont la valeur précise est inconnue. Une variable linguistique, ou variable floue, est donc une variable dont les valeurs appartiennent à des ensembles flous pouvant représenter des mots du langage naturel. Ainsi une variable floue peut prendre simultanément plusieurs valeurs linguistiques.

Par exemple la variable « Taille » peut appartenir aux ensembles flous " Petit, Moyen, Grand ".

La variable linguistique peut être représentée par un triplet  $(x, T(x), U)$ , dans lequel  $x$  est le nom de la variable linguistique,  $T(x)$  l'ensemble des noms des valeurs linguistiques de  $x$  et  $U$  l'ensemble de référence (univers de discours).

**Par exemple :**  $x = \text{Vitesse}$  est une variable linguistique, son ensemble de valeurs peut être :

$T(\text{Vitesse}) = [\text{Faible}, \text{Moyenne}, \text{Élevée}, \dots]$  où chaque terme dans  $T(\text{Vitesse})$  est caractérisé par un ensemble flou dans un univers de discours  $U = [0, 100]$  (Figure (I.2)).



**Figure (I.2) :** Représentation floue de la variable Vitesse.

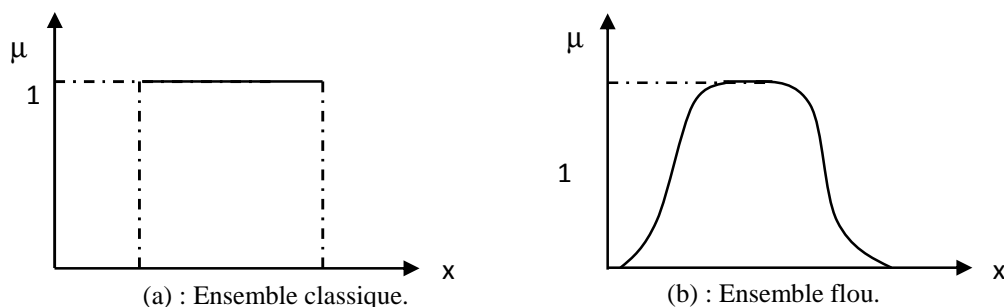
On attribue à chaque valeur de la variable linguistique des fonctions d'appartenance  $\mu$ , dont la valeur varie entre 0 et 1, en tenant compte de la classification en un certain nombre d'ensembles flous.

### I.3.3.3. Fonction d'Appartenance

N'importe quelle fonction  $A \quad x \quad [0,1]$  peut être qualifiée pour servir d'une fonction d'appartenance décrivant l'ensemble flou correspondant. En pratique, la forme des fonctions d'appartenances devrait être réfléchissante du problème pour lequel nous construisons les ensembles flous [8]. Nous présentant ici les formes de fonctions d'appartenances les plus utilisés dans la littérature [8], [11], [10]

Les valeurs affectées aux variables linguistiques ne sont pas des entiers ou des réels ou autre mais des « ensembles flous ».

Supposons que la figure (I.3.a) représente un ensemble classique des personnes dans la force de l'âge. La lettre  $\mu$  désigne l'appartenance à cet ensemble et  $x$  désigne l'âge.



**Figure (I.3) :** Représentation graphique de sous-ensemble dont les ensembles de référence est infinis.

Si l'on présente cette propriété comme une propriété floue, on doit passer à un ensemble flou et nous pouvons dessiner une courbe comme celle de la (figure (I.3.b)). Cette courbe est arbitraire, subjective, dépendante de l'instant parce que les différentes personnes jugent différemment les propriétés concernant l'âge.

### I.3.3.1. Formulations mathématique des fonctions d'appartenance

**Fonction triangulaire** : Elle est définie par trois paramètres  $\{a, b, c\}$  :

$$\mu_f(x) = \max \left[ \min \left( \frac{x-a}{b-a}, \frac{c-x}{c-b} \right), 0 \right] \quad (\text{I.11})$$

**Fonction Gaussienne** : Elle est définie par deux paramètres  $\{\delta, m\}$  :

$$\mu_f(x) = \exp \left( -\frac{(x-m)^2}{2\delta^2} \right) \quad (\text{I.12})$$

**Fonction sigmoïdale** : Une fonction sigmoïdale est définie par deux paramètres  $\{a, c\}$  :

$$\mu_f(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (\text{I.13})$$

**Fonction trapézoïdale** : Elle est définie par quatre paramètres  $\{a, b, c, d\}$  :

$$\mu_f(x) = \max \left[ \min \left( \frac{x-a}{b-a}, 1, \frac{d-x}{d-c} \right), 0 \right] \quad (\text{I.14})$$

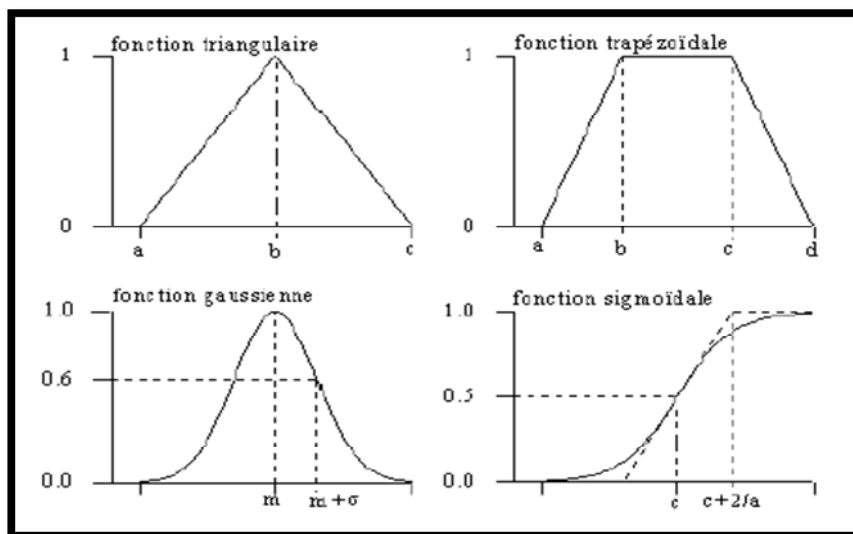


Figure (I.4) : Forme des fonctions d'appartenances usuelles.

## I.4. Structure d'une commande floue

La structure conventionnelle d'une commande floue est présentée sur la figure (I.5), elle est composée de quatre blocs distincts dont les définitions sont données ci-dessous :

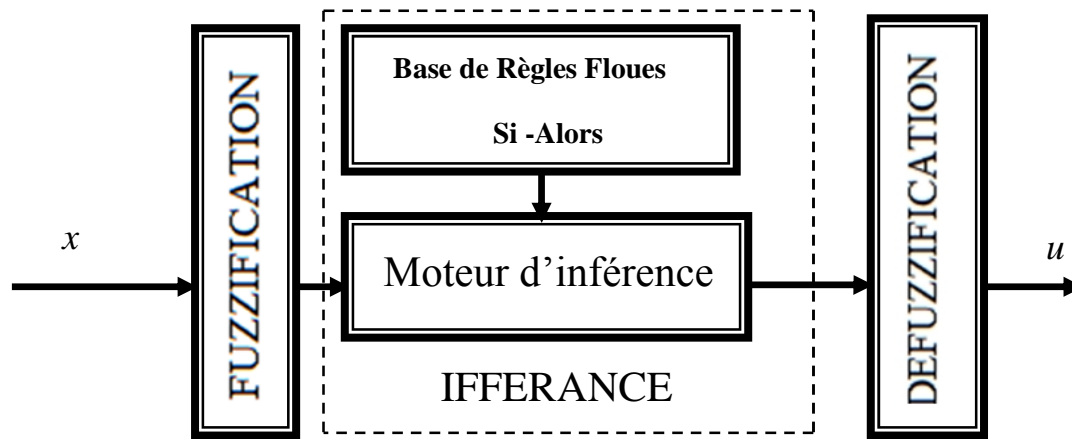


Figure (I.5) : structure de base d'un système flou.

Un contrôleur flou se compose de quatre modules principaux :

- 1) *l'interface de fuzzification* : modifie les entrées de sorte qu'elles puissent être interprétées et comparées aux règles dans la base de règle.
- 2) *le mécanisme d'inférence* : évalue quels règles de commande sont appropriées au temps actuel et puis décide quelle entrée devraient être fournis au système.
- 3) *l'interface de defuzzification* : convertit les conclusions tirées par le mécanisme d'inférence en entrées du système.

### I.4.1. Interface de fuzzification

Dans les problèmes de commande, les données observées sont habituellement physiques (réelles). Or le traitement de ces données est basé sur la théorie des ensembles flous; ceci nécessite donc une procédure de fuzzification. L'opération de fuzzification représente le passage des grandeurs réelles (ou physiques) aux valeurs floues. Cette étape nécessite souvent une conversion analogique/digitale, ainsi que le traitement des grandeurs mesurées et leur transformation en variables linguistiques avec la définition des fonctions d'appartenance.

A l'univers de discours d'une entrée  $X$  (ensemble des valeurs possibles de  $X$ ), on associera  $N$  sous-ensemble flous notés  $E_i$  (valeurs linguistiques). Chacun de ceux-ci sera défini par sa fonction d'appartenance  $\mu_{E_i}(x), 0 < \mu_{E_i}(x) < 1$ .

Le rôle du bloc de fuzzification sera de déterminer pour un  $x_i$  donné (variable observée ou mesurée) les degrés d'appartenance de  $x_i$  à chacun des sous-ensembles flous  $E_i$ .

### I.4.2. Moteur d'inférence floue

C'est une logique de prise de décision qui utilise des règles floues de la base de règle floue pour déterminer les sorties du système flou correspondantes à ces entrées *fuzzifier*. C'est le moteur d'inférence flou qui simule la procédure de prise de décisions humaine basée sur des concepts flous et des déclarations linguistiques [12].

#### I.4.2.1. Traitement numérique des inférences

En général deux ou plusieurs règles sont activées en même temps. Une règle d'inférence floue est activée lorsque le facteur d'appartenance lié à la condition de cette règle est non nul. Il existe plusieurs possibilités pour les opérateurs qui s'appliquent aux fonctions d'appartenance. On introduit alors la notion de méthode d'inférence.

En réglage par logique floue, on utilise en général des méthodes suivantes

- *Méthode d'inférence Max-Min.*
- *Méthode d'inférence Max-Prod.*
- *Méthode d'inférence Somme-Prod.*

Le nom de la méthode désigne les opérateurs utilisés respectivement pour l'agrégation et l'implication. Le tableau suivant indique la manière de leur utilisation :

Méthodes	Opérateurs sur Prémisses		Opérateur Implication	Opérateur Agrégation
	Ou	Et	Imp	
<i>Max-min</i>	Max	Min	Min	Max
<i>Max-prod</i>	Max	Min	Prod	Max
<i>Som-prod</i>	Som	Prod	Prod	Som

Tableau (I.1) : Les méthodes d'inférence.

**Exemple :**

Dans cet exemple on utilise la méthode d'inférence **somme-produit**

La méthode d'inférence somme-produit réalise, au niveau de la condition, l'opérateur **Ou** par la formation de la somme moyenne, tandis que l'opérateur **Et** est réalisé par la formation du produit. La conclusion de chaque règle, précédée par **Alors**, liant le facteur d'appartenance de la condition avec la fonction d'appartenance de la variable de sortie par l'opérateur **Et**, est réalisée par la formation du produit. L'opérateur **Ou** qui lie les différentes règles est réalisé par la formation de la somme moyenne; ainsi s'explique la désignation par **Som-Prod** de cette méthode d'inférence.

La fonction d'appartenance résultante est donnée par :

$$\mu_{RES}(x_R) = \left[ \mu_{R1}(x_R) + \dots + \mu_{Rm}(x_R) \right] / m \tag{I.15}$$

Avec:  $\mu_{Ri}(x_R) = \mu_{ci} \cdot \mu_{oi}(x_R), \quad i = 1, \dots, m$

$\mu_{oi}(x_R)$  : La fonction d'appartenance partielle (résultante de la règle i )

$\mu_{ci}$  : Facteur d'appartenance de la condition

$\mu_{oi}(x_R)$  : Fonction d'appartenance de la conclusion

$m$  : Le nombre de règles intervenant dans l'inférence.

Pour illustrer cette méthode on présente l'exemple suivant ; celle d'un régulateur flou avec deux variables d'entrée  $(x_1)$  et  $(x_2)$ , et une variable de sortie  $(x_R)$ . Chacune est décomposée en trois ensembles NG, EZ, PG. On suppose que les valeurs sont :  $x_1 = 0.44$  et  $x_2 = 0.67$  Pour l'inférence :

Si  $(x_1 \text{ PG Et } x_2 \text{ EZ})$  Alors  $x_R = \text{EZ}$  Ou  
Si  $(x_1 \text{ EZ Ou } x_2 \text{ NG})$  Alors  $x_R = \text{NG}$

Avec les facteurs d'appartenance  $\mu_{PG}(x_1 = 0.44) = 0.67$  et  $\mu_{EZ}(x_2 = 0.67) = 0.33$ , la première condition prend le facteur d'appartenance  $\mu_{Ci} = 0.22$  (produit des deux valeurs). Pour la condition de la deuxième règle, on a :

$\mu_{EZ}(x_1 = 0.44) = 0.33$  et  $\mu_{NG}(x_2 = 0.67) = 0.67$ , ce qui donne  $\mu_{Ci} = 0.5$

### I.4.3. Interface de Défuzzification

La dernière étape du contrôle, appelée défuzzification consiste à définir précisément quelle doit être l'action sur le processus. En effet, le procédé ne peut pas interpréter des ordres du type « Petit » ou « Grand », etc..... on doit lui envoyer une valeur physique.

Les méthodes d'inférences fournissent une fonction d'appartenance résultante  $\mu_{rés}(x_R)$  pour la variable de sortie  $x_R$ . L'opération de défuzzification permet de calculer à partir de cette dernière la valeur réelle de la variable de sortie à appliquer au processus. On doit souvent prévoir un traitement de signal de sortie et sa conversion numérique - analogique

Le choix d'une méthode de défuzzification est un point très délicat lors de l'élaboration d'une technique de contrôle en logique floue. Celui-ci conditionnera en effet grandement l'évolution dynamique de la commande. On distingue trois méthodes différentes (figure (I.6))

Celle du maximum, celle de la moyenne des maxima et celle du centre de gravité. Il est toutefois reconnu que la méthode de centre de gravité donne les meilleurs résultats

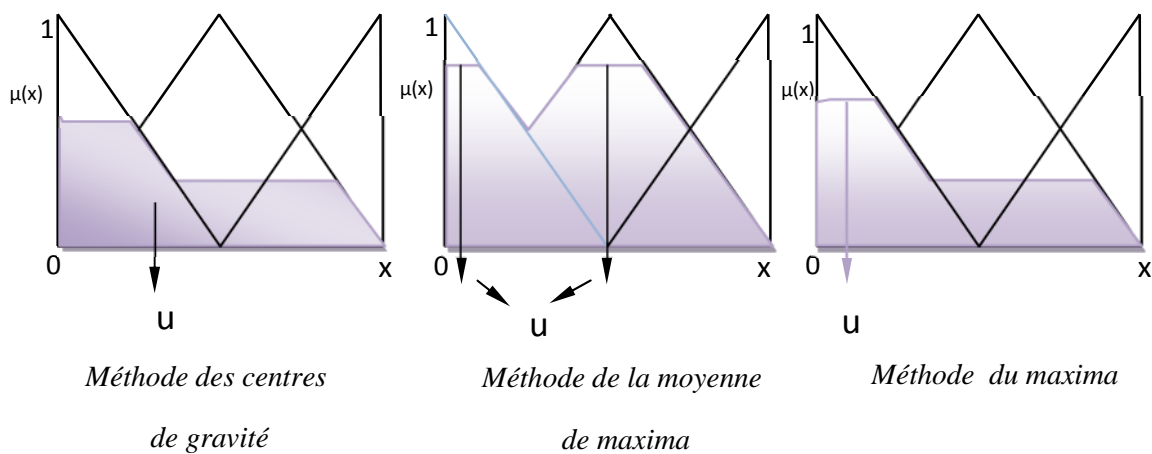


Figure (I.6) : Principe des différentes méthodes de défuzzification.

### • Méthode du maxima

Cette méthode consiste à choisir comme sortie  $x_0$  du bloc de défuzzification, une des valeurs possédant la plus grande appartenance au sous-ensemble flou  $x$ .

Il se peut que le système possède plusieurs maxima identiques, dans ce cas et afin d'éviter un choix arbitraire, on choisit d'effectuer la moyenne des maxima.

La méthode du maximum à l'avantage d'être simple, rapide et facile. Elle est malheureusement ambiguë et provoque de nombreuses discontinuités.

### • Méthode de la moyenne des maxima

Dans le cas où plusieurs sous-ensembles ont le même maximum, on réalise une commande  $u = \frac{\sum u_i}{r}$ ,  $u_i$  étant la commande issue du  $i^{\text{ème}}$  sous-ensemble flou sélectionnable.

$r$  : nombre de maxima identiques

Les avantages et inconvénients de la méthode de la moyenne des maxima restent grosso modo ceux de la méthode du maximum.

### • Méthode du centre de gravité

Cette méthode consiste à calculer le centre de gravité de la fonction d'appartenance résultante  $\mu_{\text{rés}}(x_R)$ . L'abscisse  $u$  de ce centre de gravité donne la valeur de commande à appliquer et peut être déterminée par la relation générale suivante :

$$u = \frac{\int_{-1}^1 X_R \mu_{\text{rés}}(X_R) dX_R}{\int_{-1}^1 \mu_{\text{rés}}(X_R) dX_R} \quad (\text{I.16})$$

L'intégrale au dénominateur donne la surface, tandis que l'intégrale au numérateur correspond au moment de la surface.

Cette méthode va permettre d'éviter de trop grandes discontinuités et supprimera toute ambiguïté. Elle semble donc optimale mais son implémentation est difficile et surtout coûteuse en calculs. Elle se simplifie notablement lorsqu'on utilise la méthode d'inférence Som-prod ou des singletons pour les fonctions d'appartenance des variables de sortie.

## I. 5. Approximation floue

Dans la littérature consacrée aux systèmes flous, on dispose d'un nombre important de publications montrant que les systèmes flous sont des approximateurs universels c'est-à-dire, pour toute fonction continue  $f(x)$  définie sur un ensemble compact  $\Omega$ , et pour toute constante positive  $\varepsilon$  il existe un système flou  $y(x)$  tel que

$$\sup_{x \in \Omega} |f(x) - y(x)| \leq \varepsilon \quad (\text{I.17})$$

Il y a plusieurs interprétations différentes pour les règles floues **SI- ALORS** que causent les configurations différentes du moteur d'inférence flou. Il y a également plusieurs différents types de fuzzification et de défuzzification. Beaucoup de combinaisons de ces moteurs d'inférences, fuzzification et défuzzification flous peuvent construire des systèmes flous.

Notons, cependant, que la propriété d'approximation universelle ne donne pas une méthode de construction du système flou  $y(x)$ , elle garantit seulement son existence.

De plus, pour un degré de précision quelconque, on a besoin d'un nombre important de règles floues.

## I.6. Contrôleurs flous usuels

Les contrôleurs flous sont principalement de deux types:

- Contrôleur flou de type Mamdani.
- Contrôleur flou de type Sugeno.

Pour un système à deux variables, les règles floues sont de la forme :

«**SI** x est Ai **Et** y est Bi **ALORS** z est Ci »

Où Ai et Bi sont des sous-ensembles flous, par contre Ci peut appartenir aussi bien au domaine symbolique (sous-ensemble flou) qu'au domaine numérique.

L'originalité de la méthode de Sugeno réside dans le fait que la conclusion de chaque règle n'appartient pas au domaine symbolique, mais est définie sous forme numérique comme une combinaison linéaire des entrées

Selon la méthode de Sugeno, les règles floues, dans le cas de deux variables, s'expriment donc selon la forme suivante :

«**SI** x est Ai **Et** y est Bi **Alors**  $z = p_0 + p_1 \cdot x + p_2 \cdot y$  »

On parle dans ce cas de contrôleur flou de type Sugeno d'ordre 1.

Dans la suite de ce mémoire, nous n'utiliserons qu'un raisonnement simplifié de Sugeno (contrôleur flou de type Sugeno d'ordre 0) où les règles floues utilisées sont du type :

«**SI** x est Ai **Et** y est Bi **Alors**  $z = p_0$ »

Dans le contrôleur flou type Sugeno, les étapes d'agrégation et de défuzzification des règles floues se font simultanément et la relation (I.18) devient :

$$u = \frac{\sum \mu_i z_i}{\sum \mu_i} \quad (\text{I.18})$$

Cette méthode est plus simple à mettre en œuvre et donne aussi de bons résultats en commande floue que la méthode de Mamdani. Le calcul en temps réel de cette expression ne pose pas de problème.

Une remarque peut être formulée sur le nom donné à cette étape. En effet, elle est appelée « *défuzzification* » alors qu'elle ne manipule aucune donnée floue. Ce choix a été dicté afin d'établir une similitude entre ce type de contrôleur et le contrôleur de type Mamdani

où le cheminement « *fuzzification – inférence floue – défuzzification* » a été introduit. A la place de « *défuzzification* », le terme « *agrégation* » aurait été préférable.

## **I.7. Conclusion**

Dans ce chapitre, nous avons présenté les bases nécessaires à la compréhension des méthodes à base de la logique floue la différence des méthodes issues de l'automatique classique, les contrôleurs basés sur la logique floue ne nécessitent nullement de modèle analytique de la tâche à résoudre. En effet, ils ne sont pas basés sur des équations physiques du système à contrôler mais sur le savoir-faire des opérateurs et des experts du domaine concerné. En effet, dans les cas les plus simples, leur mise au point s'effectue par le biais d'une phase d'extraction naturelle des connaissances, c'est-à-dire par un dialogue avec les experts ou par un questionnaire.

# Chapitre II

## Commande Adaptative

## II.1.INRODUCTION

La plupart des techniques actuelles de conception de systèmes de commande sont basés sur une bonne compréhension du système étudié et de son environnement.

Toutefois, dans un certain nombre de cas, le système commandé est trop complexe et ces processus physiques de base ne sont pas entièrement compris. Les techniques de conception de contrôle doivent alors être augmentées avec une technique d'identification visant à obtenir progressivement une meilleure compréhension du système contrôlé [13], ce genre de problème a mené à l'apparition de la commande adaptative.

La commande adaptative des systèmes non linéaires est un dispositif très attrayant d'un système de contrôle moderne. Aujourd'hui un grand nombre de boucles de control industrielles sont sous la commande adaptative. Celles-ci incluent une gamme étendue d'applications dans l'aérospatiale, la régulation de processus, la robotique, et les systèmes des véhicules à moteur et les systèmes biochimique [14].

Ce chapitre est une présentation de la commande adaptative .Il fournit également un certain fond et un bref historique du développement de la commande adaptative.

## II.2.Historique

La recherche en contrôle adaptatif a une longue et forte histoire. Au début des années 50 il y avait des recherches étendues sur la commande adaptative en liaison avec la conception des pilotes automatiques pour les avions à haute performance [13].

De tels avions fonctionnent sur une large plage de vitesse et d'altitude. Il a été constaté que le gain constant de commande classique de rétroaction linéaire pourrait fonctionner bien en une condition de fonctionnement mais pas au-dessus du régime de vol entier [13], [15]. Un contrôleur plus sophistiqué qui pourrait travailler bien sur une grande variété de conditions opérantes était nécessaire. Après un effort de développement significatif il a été trouvé que le gain préprogrammé (scheduling) [15] basé sur quelques mesures auxiliaires de la vitesse anémométrique était une technique convenable pour les systèmes de contrôle de vol [13]. Un autre schéma de commande adaptative à modèle de référence été proposé par *Whitaker* et ces collègues pour résoudre le problème de contrôle de pilote

automatique [15], le but de ce schéma était de construire un contrôleur auto ajustable qui rapporte une fonction de transfert en boucle fermée assortissant à un modèle de référence prescrit. Plusieurs mécanismes d'auto-ajustement des paramètres du contrôleur ont été proposées, telles que Les règles de sensibilité et la règles de M.I.T. et ont été vérifiés pour bien préformer dans de certaines conditions [13].

Finalement, en 1958 *Kalman* a met un appui analytique fermant le concept d'un contrôleur auto-réglable avec une identification explicite des paramètres d'un système linéaire, avec une seule entrée une seule sortie (SISO) et l'utilisation de ces estimations de paramètres pour mettre à jour un contrôleur quadratique linéaire optimal [13].

L'intérêt pour la commande adaptative a diminué partiellement parce qu'il était trop difficile de traiter le problème de commande adaptative utilisant les techniques qui étaient disponible au temps [15].

Pendant les années 60 il y avait beaucoup de recherche dans la théorie de contrôle qui a contribué au développement de la commande adaptative. L'espace de l'état et théorie de stabilité basé sur l'approche de Lyapunov ont été introduites [15]. La théorie de stabilité de Lyapunov a été fermement établie comme un outil pour prouver la convergence dans des schémas de commande adaptatives [13], [15]. Il y avait aussi des résultats importants dans la théorie de contrôle stochastique [13], [15]. La programmation dynamique, présentée par Bellman, a augmenté la compréhension des processus adaptatifs. Des contributions fondamentales ont été aussi faites par Tsypkin, qui a montré que beaucoup d'arrangements pour l'apprentissage et la commande adaptative pourraient être décrits dans un cadre commun. Il y avait également des développements importants dans l'identification de système. Une Renaissance de commande adaptative s'est produite pendant les années 70, lorsque les différents schémas d'estimation ont été combinées avec les diverses méthodes de conception. De nombreuses applications ont été signalées, mais les résultats théoriques ont été très limités.

A la fin des années 70 et le début des années 80, des preuves pour la stabilité des systèmes adaptatifs sont apparu, mais sous des hypothèses très restrictives.

Les efforts visant à fusionner les idées de contrôle robuste et des systèmes d'identification sont d'une importance particulière. L'investigation de nécessité de ces suppositions a suscité de nouvelles et intéressantes recherches dans la robustesse de commande adaptative, ainsi que dans la robustesse des contrôleurs d'adaptation.

Les investigations sur les systèmes non linéaires ont conduit à une compréhension accrue de manière significative de contrôle adaptatif. Dernièrement, il a également été établi que le contrôle adaptatif de solides relations d'idées sur l'apprentissage qui émergent dans le domaine de l'informatique.

Il y a eu beaucoup d'expériences sur la commande adaptative dans les laboratoires et l'industrie. Le progrès rapide dans la microélectronique était une stimulation forte [15].

L'interaction entre la théorie et l'expérimentation a eu comme conséquence un développement vigoureux du champ. En conséquence, les contrôleurs adaptatifs ont commencé à apparaître commercialement au début des années 80[15].

Au début des années 90 les nouvelles découvertes étaient appliquées aux méthodes adaptatives, telles que l'intelligence artificielle, les réseaux de neurone et les techniques flous. Toutefois, dans la deuxième moitié des années 90, les systèmes adaptatifs faisait toujours preuve d'un grand potentiel d'applications massive, bien que de nombreuses sociétés bien connues déployés les principes d'adaptation pour l'autorégulation, et parfois même pour contrôle en ligne [16].

La commande adaptative a une littérature riche complètement de différentes techniques pour la conception, l'analyse, l'exécution, et les applications. Plusieurs articles, des livres et des monographies ont été déjà édités. En dépit de la vaste littérature sur le sujet, il y a toujours un sentiment général que la commande adaptative est une collection d'indépendant techniques et outils.[15]

### **II.3. Définition de la commande adaptative**

La terminologie « commande adaptative » désigne l'ensemble de méthodes permettant l'ajustement automatique en temps réel des paramètres des régulateurs mis en œuvre dans une boucle de commande a fin de réaliser ou de maintenir un niveau de

performances désirée lorsque le processus contrôlé n'est pas trop bien connue ou ayant une non linéarité significative ou des paramètres variant dans le temps. [14]

Bien que chaque contrôleur est adaptatif dans le sens qu'il change sa sortie à chaque fois que l'erreur change, la vrai commande adaptative est lorsque le contrôleur adapte non seulement sa sortie mais aussi sa stratégie de contrôle, c'est-à-dire qu'il peut régler ces propres paramètres ou d'une autre manière sa loi de commande pour s'adapter aux changements du comportement du processus [16].

La figure (II.1), montre le block diagramme d'une commande adaptative.

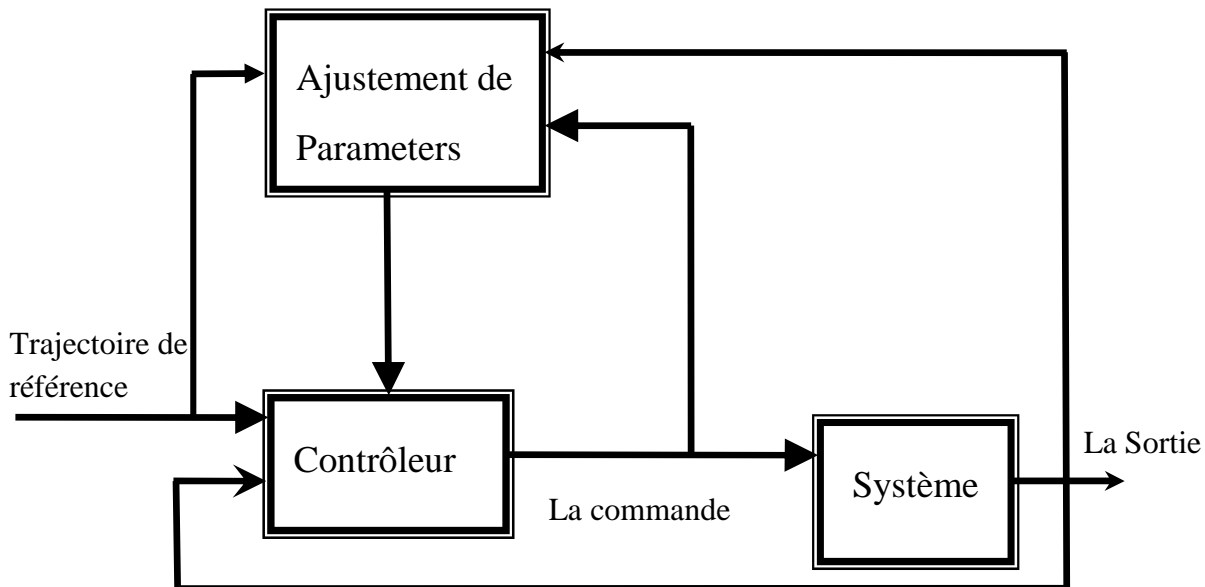


Figure (II.1) : Schéma bloc d'une commande adaptative.

La construction d'un contrôleur adaptatif contient ainsi les étapes suivantes :

- Caractériser le comportement désiré du système en boucle fermée.
- Déterminer une loi de commande appropriée avec les paramètres réglables.
- Trouver un mécanisme pour ajuster les paramètres.
- Implémenter la loi de commande.

## II.4. Généralités sur la commande adaptative :

Dans les systèmes adaptatifs, il est supposé que les paramètres sont ajustés à tout moment, ce qui implique que les paramètres changent dans le processus. Cependant, il est difficile de réaliser l'analyse de la convergence et de la stabilité du système. Pour simplifier le problème, nous pouvons supposer que le processus est constant mais les paramètres sont inconnus. Ces derniers sont estimés à tout moment, utilisant les méthodes d'estimation récursives. Différentes techniques d'estimation peuvent être utilisées :

Approximation stochastique, moindres carrés étendus et généralisés, ceci est appelé en commande adaptative, la technique d'estimation des paramètres en temps réel ou commande adaptative non linéaire.

La commande adaptative ajuste en ligne les paramètres du correcteur de manière à conserver le niveau de performance désiré lorsque les paramètres du procédé dérivent dans le temps ou sont inconnus, cette adaptation se fait au prix d'une complexité algorithmique plus élevée, Elle nécessite, la plupart du temps, des outils de calcul très puissants tels que les microprocesseurs destinés au traitement du signal (Digital Signal Processor ou DSP).

La commande adaptative prend en compte deux types de problèmes :

- Les paramètres du système constants mais inconnus sont auto-ajustés.
- Les paramètres qui dérivent dans le temps seront adaptés.

La loi de variation des paramètres des correcteurs donne un caractère non linéaire aux commandes adaptatives, la recherche de stabilité et la synthèse du correcteur reposeront donc sur des techniques spécifiques à ces systèmes en particulier, elles mettent en œuvre les critères de Lyapunov.

## II.5. Pourquoi la commande adaptative

Dans de nombreuses situations, la commande adaptative s'avère utile, parmi ces situations, nous citons [14]:

- Les changements dans la fonction de transfert du système, dans son ordre ou en valeurs de quelques paramètres dus aux variations de l'environnement.
- Les bruits de stochastiques.
- Les changements dans la nature de données.
- La propagation des bruits le long d'une chaîne de processus d'unité.
- Le comportement non linéaire comme dans le cas de réactions des complexes chimiques ou biochimiques.
- Un temps mort appréciable.

## II.6.Principe

Dans les systèmes classiques de réglage, le régulateur à paramètres fixes est utilisé pour réduire ou éliminer l'effet des perturbations agissantes sur les grandeurs à régler. Pour atteindre ce but, les variables réelles sont mesurées et comparées aux valeurs désirées, leurs différences sont injectées à l'entrée du régulateur pour générer le signal de commande. Par contre un système de commande adaptative traite l'écart entre l'indice de performance désiré et celui qui est mesuré dans le système réel. Lorsqu'il y a un écart entre ces deux grandeurs, un mécanisme d'adaptation ajuste un organe de commande qui agit sur le système de façon à éliminer cet écart.

Il y a deux structures de base pour ce type de commande :

- La structure parallèle.
- La structure série.

Du point de vue de la synthèse de la commande il y a deux méthodes :

- La méthode d'adaptation paramétrique.
- La méthode d'adaptation par synthèse de signal.

La figure (II.2) montre le schéma de principe :

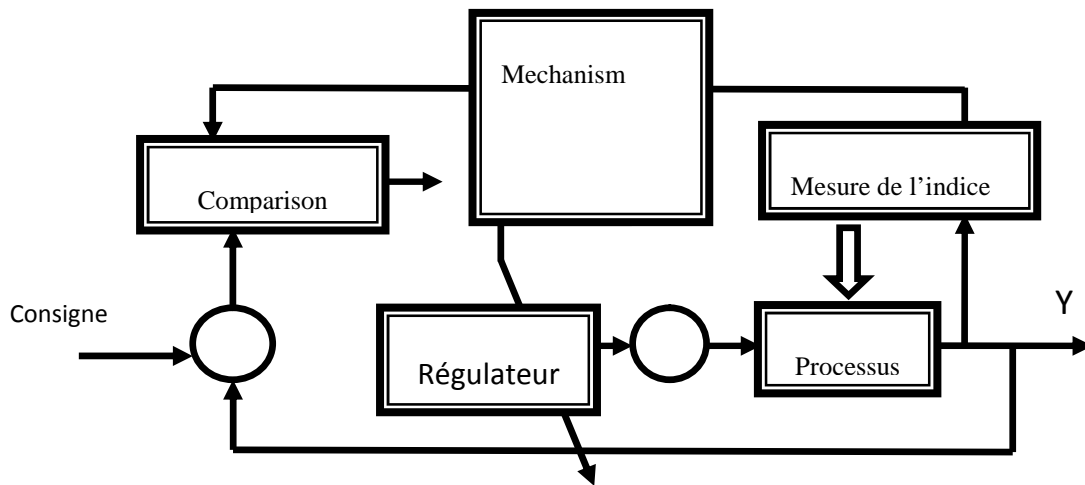


Figure (II.2) : Schéma de principe d'une commande adaptative.

## II.7. Commande adaptative directe et indirecte

La seule différence entre la commande adaptative et les régulateurs auto-ajustable est que la première approche est un schéma de commande adaptative directe et la deuxième est un schéma indirect [13].

### II.7.1. commande adaptative directe

Nous l'appelons ainsi car il y a un ajustement direct de paramètres. Cette approche repose sur le raisonnement suivant : Étant donné la structure du modèle du système (supposée connue) et la stratégie de commande choisie, on peut en déduire la structure du régulateur capable de reparamétriser l'ensemble système plus le régulateur, identifions directement ces paramètres et utilisons ces estimées dans le régulateur.

Cette approche également désigné sous le nom de la commande adaptative implicite parce que la conception est basée sur l'évaluation d'un modèle implicite du système [14].

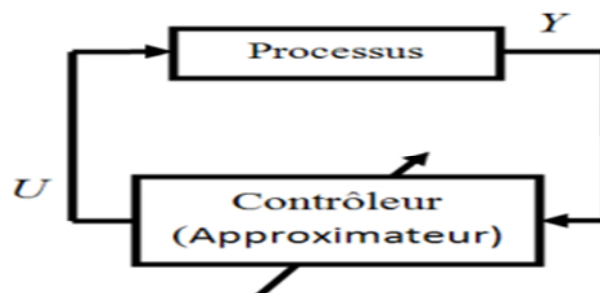


Figure (II.3) : Commande adaptative direct.

### II.7.2 commande adaptative indirecte

Dans cette approche, on décide d'identifier les paramètres du système et d'utiliser, à chaque instant, ces estimées (comme s'il s'agissait des vrais paramètres) dans le calcul de la loi de commande, c'est-à-dire dans le calcul des paramètres du régulateur adaptatif.

D'une certaine manière, on peut dire que l'on commande optimalement, à chaque instant, le modèle estimé du système [12].

Cette approche également désigné sous le nom de la commande adaptative explicite, parce que la conception est basée sur un modèle explicite du système.

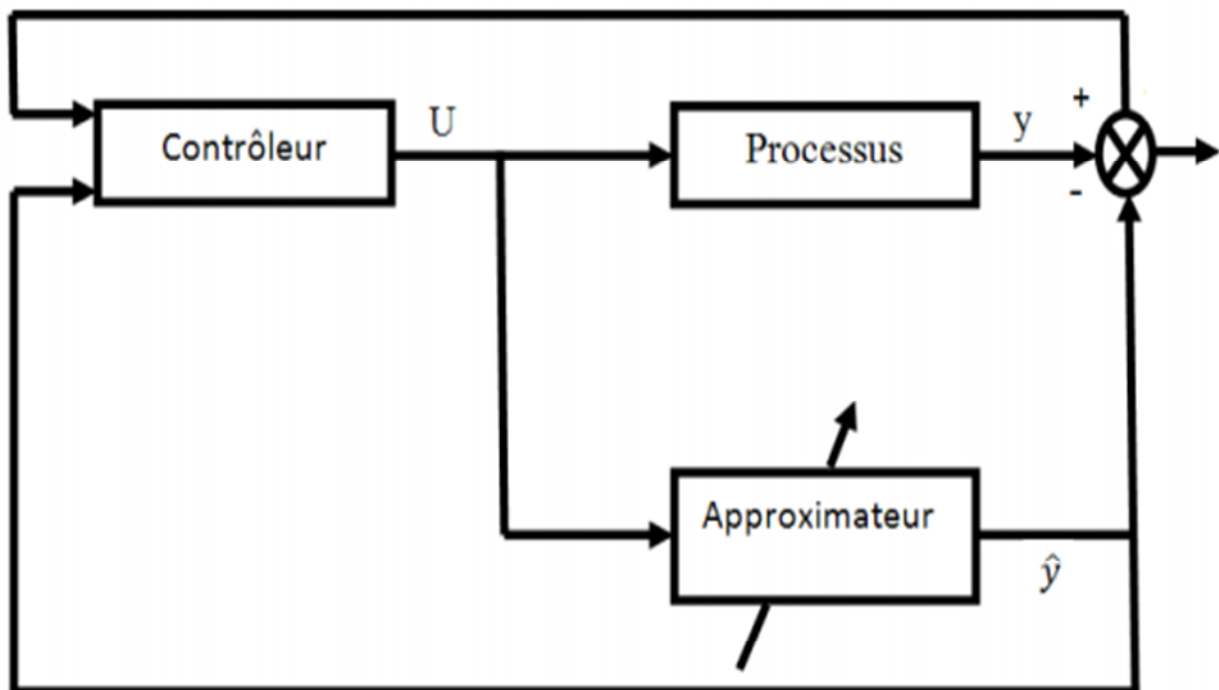


Figure (II.4) : Commande adaptative indirecte.

### II.8.Stabilité

Un système de commande tolérant aux fautes est un système capable de maintenir la stabilité et un certain degré de performance en présence de défauts [17].

### II.8.1. Méthode direct de Lyapunov

La méthode directe de Lyapunov (aussi appelée la deuxième méthode de Lyapunov) nous permet de déterminer la stabilité d'un système sans explicitement l'intégration de l'équation différentielle. La méthode est une généralisation de l'idée que s'il ya une "mesure d'énergie» dans un système, alors nous pouvons étudier le taux de variation de l'énergie du système pour vérifier, S'il existe une fonction scalaire  $V(x)$  dont les dérivées partielles d'ordre un sont continues, et telle que :

$$V(x) > 0 \quad \forall x \neq 0, V(0) = 0, \text{ Est définie positive}$$

$$\dot{V}(x) \leq 0 \quad \forall x \neq 0, \dot{V}(0) = 0, \text{ Est définie négative.}$$

$V(x)$  Est radialement non-bornée.

Alors le point d'équilibre est globalement asymptotiquement stable.

### II.8.2. Méthode indirect de Lyapunov [17]

La méthode de linéarisation de Lyapunov a été intensivement employée pour évaluer la stabilité locale des systèmes non linéaires. Fondamentalement, elle reflète l'idée intuitive qu'un système non linéaire devrait se comporter semblable à son approximation autour de proximité du point de linéarisation. Pour un système non linéaire de la forme :  $\dot{x} = f(x)$ ,  $f(0) = 0$ , où  $f(x)$  est un champ de vecteur (sans interruption différentiable), si le  $J(f(x))$  représente le jacobin de  $f(x)$  évalué à l'équilibre ( $x = 0$ ) le théorème de Lyapunov direct peut être énoncé comme suit :

Si tout les valeurs propres de  $J(f(x))$  sont à partie réelle négative, alors le point d'équilibre du système non linéaire est localement asymptotiquement stable.

Si au moins une de valeurs propres de  $J(f(x))$  à une partie réelle positive, alors l'équilibre est un point instable du système non linéaire.

Si l'approximation linéaire est marginalement stable (toutes les valeurs propres de ont  $J(f(x))$  une partie réelle plus petite ou égale a '0' et au moins une valeur propre a une partie réel égale à 0) alors aucunes conclusions sur la stabilité du point d'équilibre ne peut être obtenue.

## II.9. La zone morte

Le bloc Zone morte fournit une zone avec une sortie égale à zéro. Elle définit la gamme de variation de l'entrée pour laquelle la sortie reste inchangée. En dehors de cette étendue il y a une relation linéaire entre l'entrée de la zone morte  $u(k)$  et la sortie de la zone morte  $V(k)$  Un exemple de graphique d'une zone morte particulière est donné ci-dessous :

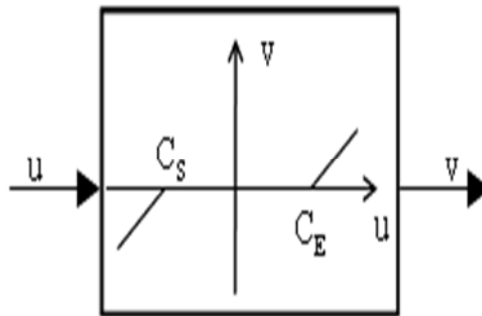


Figure (II.5) : La zone morte.

Il est définie par :

$$V(k) = \begin{cases} u(k) - c_s & \text{si } u(x) < c_s \\ 0 & \\ u(k) - c_E & \text{si } u(x) > c_E \end{cases}$$

Où  $c_E > c_s$

Quand l'entrée  $u(k)$  est entre  $c_E$  et  $c_s$ , la sortie  $V(k)$  est zéro. Quand l'entrée est supérieure ou égale à la limite haute  $c_E$ , la sortie est égale à l'entrée moins la limite haute.

Quand l'entrée est inférieure ou égale à la limite inférieure  $C_s$ , la sortie est égale à l'entrée moins la limite inférieure.

## **II.10.Conclusion**

Dans de nombreuses situations, les systèmes de commande classique sont basés sur les modèles à paramètres fixes, car ils ne peuvent prendre en charge les variations lentes ou rapides de la dynamique des systèmes, d'où la nécessité d'introduire la notion de systèmes de commande adaptative afin d'assurer les performances désirées. Pour cela, les paramètres du régulateur (contrôleur) sont ajustés automatiquement sur la base des informations recueillies du système. Dans ce chapitre, nous avons présenté quelques notions, et définition sur la commande adaptative, nous avons commencé par définir la commande adaptative, puis nous avons donné une vue générale sur son histoire, ainsi que les différents schémas de commande adaptative.

# Chapitre III

Commande adaptative floue indirecte  
des

systemes non linéaires discrets

### III.1.Introduction

Dans la dernière décennie, la commande adaptative floue non linéaire basée sur le théorème d'approximation universelle a reçu beaucoup d'attention. Cependant, la plupart des résultats sont limités aux systèmes continus, qui ne peuvent pas être directement étendus aux systèmes discrets. Dans des applications pratiques, presque tous les systèmes de commande floue sont mis en application sur calculateurs numériques. Puisque les signaux de commande peuvent seulement être appliqués à des instants fixes, quelques avantages des contrôleurs continus sont perdus au moyen de la discrétisation. Il est nécessaire de tenir compte du fait qu'en réalité le contrôleur est un système discret et non pas continu. Récemment, un contrôleur flou discret pour une classe de système dynamique non linéaire linéarisable par rétroaction a été présenté [25]. Un schéma de commande adaptative directe a été présenté où les systèmes flous de Takagi-Sugeno ont été employés comme approximateurs fonctionnels, une zone morte à temps continu a été employée pour garantir la convergence de l'erreur de poursuite à un voisinage de l'origine [25].

Dans [26], les auteurs ont présenté un schéma de commande adaptative indirecte en utilisant les systèmes flous de Takagi-Sugeno, des résultats semblables de stabilité ont été réalisés. Basé sur [25] et [26], le gain d'adaptation et la direction de la descente ont été mises à jour des manières qui cherchent à optimiser certaines fonctions coût [27], et les études expérimentales utilisant la prédiction adaptative à temps discret et des techniques de commande sont développées dans [28]. Dans [29], une commande adaptative directe pour une classe de système discret de rétroaction non linéaire était proposée. Dans [26], la loi d'adaptation a été conçue par une zone morte continue, dont sa taille a été basée sur l'erreur d'approximation du système flou, et par conséquent il est nécessaire de supposer que les bornes de l'erreur d'approximation sont connues à l'avance. Bien que l'erreur d'approximation est bornée, malheureusement, dans beaucoup de systèmes pratiques tels bornes ne pourraient pas être disponibles.

Dans ce chapitre, une méthode de commande floue adaptative indirecte est développée pour une classe de systèmes non linéaires discrets. Les systèmes flous sont employés pour approximer les fonctions inconnues des systèmes, et les paramètres inconnus dans les

systèmes flous sont ajustés par une zone - morte variable dans le temps, dont sa taille est ajustée de manière adaptative avec les bornes prévues sur l'erreur d'approximation.

La supposition des bornes connues sur l'erreur d'approximation n'est pas exigée puisque ceux sont estimés par lois d'adaptation. Le schéma de conception proposé garantit que tous les signaux dans le système en boucle fermée sont bornés, et l'erreur de poursuite converge à un petit voisinage de l'origine.

### III.2. Conception d'un contrôleur adaptatif flou indirect et analyse de stabilité

Considérons le système non linéaire discret (SISO) sous la forme suivante :

$$y(k+d) = f(x(k)) + g(x(k))u(k) \quad (\text{III.1})$$

Où  $u(k) \in R$  et  $y(k) \in R$  sont l'entrée et la sortie du système respectivement,  $d$  et le retard du système, et  $x(k) = [y(k), \dots, y(k-n), u(k-1), \dots, u(k-m)]^T$ ,  $f(x(k))$  et  $g(x(k))$  sont des fonctions lisses inconnues, et la supposition suivante est faite :

#### **Hypothèse 1:**

Il est supposé que  $g(x(k)) \neq 0$  pour tout  $x(k) \in U \subset R$ .

L'objectif de ce chapitre est de concevoir une loi de commande telle que la sortie du système  $y(k)$  suit le signal de référence  $r(k)$  alors que tous les signaux dans le système en boucle fermée demeurent bornés.

Si les fonctions  $f(x(k))$  et  $g(x(k))$  sont connues exactement, il est bien connu que pour le système (III.1), il existe un contrôleur idéal :

$$u^*(k) = \frac{-f(x(k)) + r(k+d)}{g(x(k))} \quad (\text{III.2})$$

Ce conduit la sortie du système de poursuivre parfaitement une trajectoire de référence connue  $r(k)$  c'est-à-dire :

$$e(k+d) = r(k+d) - y(k+d) = 0$$

Ceci signifie qu'après  $d$  étapes, nous avons  $e(k) = 0$ .

Puisque les fonctions  $f(x(k))$  et  $x(k)$  sont inconnues, le contrôleur idéal  $u^*(k)$  de (III.2) ne peut pas être implémenté, nous supposons que les fonctions  $f(x(k))$  et  $x(k)$  peuvent être approximées par des systèmes flous. Le système flou utilisé est une collection de règles floues SI-ALORS de la forme [18] :

$$R^{(l)} : \text{IF } x_1 \text{ is } F_1^l \text{ and } \dots \text{ and } x_n \text{ is } F_n^l \text{ THEN } y \text{ is } G^l$$

Où  $x = (x_1, \dots, x_n)^T$  et  $y$  sont l'entrée et la sortie du système flou, respectivement,  $F_1^l$  et  $G^l$  sont des ensembles flous, pour  $l = 1, \dots, m$ .

En utilisant la stratégie de fuzzification par singleton, une défuzzification par le centre de gravité, et le produit d inférence, la sortie finale du système flou est donnée comme suit :

$$y(x) = \frac{\sum_{j=1}^m y^j \left( \prod_{i=1}^n \mu_{F_i^j}(x_i) \right)}{\sum_{j=1}^m \prod_{i=1}^n \mu_{F_i^j}(x_i)} \quad (\text{III.3})$$

Où  $y^j$  est le point auquel la fonction d'appartenance de  $G^l$  atteint sa valeur maximale. En présentant le concept des fonctions floues de base  $\xi(x)$ , la sortie donnée par (III.3) peut être réécrite sous la forme compacte suivante :

$$y(x) = (x | \theta_f) = \theta_f^T \xi(x) \quad (\text{III.4})$$

Où  $\theta = (y^1, \dots, y^m)^T$ ,  $\xi(x) = (\xi^1(x), \dots, \xi^m(x))^T$

$$\xi^j(x) = \frac{\prod_{i=1}^n \mu_{F_i^j}(x_i)}{\sum_{j=1}^m \prod_{i=1}^n \mu_{F_i^j}(x_i)}$$

En utilisant deux systèmes flous  $\hat{f}((x(k) | \theta_f) = \theta_f^T \zeta_f(x(k))$  et  $\hat{g}((x(k) | \theta_g) = \theta_g^T \zeta_g(x(k))$  sous la forme (III.4) pour approximer les fonctions  $f(x(k))$  et  $g(x(k))$ .

Afin de garantir l'existence de la commande équivalente de certitude, elle peut être définie comme :

$$u(k) = \frac{-\hat{f}(x(k|\theta_f)) + r(k+d)}{\hat{g}(x(k|\theta_g)) + \varepsilon \text{sign}(\hat{g}(x(k|\theta_g)))} \quad (\text{III.5})$$

Où  $\varepsilon > 0$

$$\text{sign}(x) = \begin{cases} -1, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

En utilisant (III.1) et (III.5), l'équation d'erreur peut être écrite sous la forme:

$$e(k+d) = \hat{f} - f(x(k)) + (\hat{g} - g(x(k)))u(k) + \varepsilon u(k) \text{sign}(\hat{g}) \quad (\text{III.6})$$

### III.3. Erreurs d'estimations des paramètres

Définissons les paramètres d'approximation optimale  $\theta_f^*$  et  $\theta_g^*$  comme suit :

$$\theta_f^* = \arg \min_{\theta_f \in \Omega_f} \left[ \sup_{x \in U} |-\hat{f} - f(x(k))| \right]$$

$$\theta_g^* = \arg \min_{\theta_g \in \Omega_g} \left[ \sup_{x \in U} |-\hat{g} - g(x(k))| \right]$$

Où  $\Omega_f$  et  $\Omega_g$  sont des ensembles compact des paramètres du contrôleur. Noter que les paramètres optimaux  $\theta_f^*$  et  $\theta_g^*$  sont des quantités constantes artificielles présentées seulement pour un but d'analyse, et ses valeurs ne sont pas nécessaires pour l'implémentation. Dénoter les erreurs d'estimation des paramètres comme :

$$\Phi_f(k) = \theta_f(k) - \theta_f^*, \Phi_g(k) = \theta_g(k) - \theta_g^*, \text{ et } \omega_f = \hat{f}(x|\theta_f^*) - f(x(k)), \omega_g = \hat{g}(x|\theta_g^*) - g(x(k))$$

Comme erreurs d'approximation minimum.

Dans ce chapitre, nous supposons que le système flou utilisé ne violent pas la propriété d'approximation universelle [18] sur l'ensemble compact  $U$ , qui est assez grand assumé de sorte que les variables d'état demeurent dans  $U$  sous la commande en boucle fermée. Ainsi il est raisonnable de supposer que les erreurs minimums d'approximation sont bornées pour tout  $x(k) \in U$ .

**Hypothèse 2:**

Il existe des constantes inconnues  $\rho_f^*$  et  $\rho_g^*$  telles que:

$$|\omega_f(k)| \leq \rho_f^*, |\omega_g(k)| \leq \rho_g^*, \text{ respectivement.}$$

Avec ces définitions, l'équation d'erreur (III.6) peut être réécrite sous la forme:

$$e(k+d) = \Phi_f^T \xi_f(x(k)) + \Phi_g^T \xi_g(x(k))u(k) + \omega_f(k) + \omega_g(k)u(k) + \varepsilon u(k) \text{sign}(\hat{g}) \quad (\text{III.7})$$

Afin de répondre à l'objectif de commande, dans [26], une zone-morte continue est employée pour concevoir la loi d'adaptation, mais la borne d'erreur d'approximation est nécessaire.

Dans ce travail, nous utilisons la zone-morte variable dans le temps pour concevoir la loi d'adaptation. La taille de la zone-morte  $\delta(k)$  est ajustée de manière adaptative par loi d'adaptation suivante :

$$\delta(k) = (\varepsilon + \rho_g(k-d) |u(k-d)|) + \rho_f(k-d) \quad (\text{III.8})$$

Où  $\rho_f(k)$  et  $\rho_g(k)$  sont les estimations de  $\rho_f^*(k)$  et du  $\rho_g^*(k)$ .

La zone-morte variable dans le temps est définie comme [20]

$$e_\Delta(k) = \begin{cases} e(k) - \text{sign}(e(k)) \delta(k) & \text{si } |e(k)| > \delta(k) \\ 0 & \text{si } |e(k)| \leq \delta(k) \end{cases} \quad (\text{III.9})$$

En utilisant la loi d'adaptation suivante pour ajuster les paramètres  $\rho_f^*(k)$  et  $\rho_g^*(k)$ :

$$\rho_f(k) = \rho_f(k-d) + \eta_f \frac{|e_\Delta(k)|}{1 + \gamma u^2(k-d)} \quad (\text{III.10})$$

$$\rho_g(k) = \rho_g(k-d) + \eta_g \frac{|u(k-d)|}{1 + \gamma u^2(k-d)} |e_\Delta(k)| \quad (\text{III.11})$$

Où  $\gamma > 0$ ,  $\eta_f > 0$ ,  $\eta_g > 0$

Le vecteur de paramètres inconnus  $\theta_f(k)$  et  $\theta_g(k)$  sont mis à jour par la loi d'adaptation suivante :

$$\theta_f(k) = \theta_f(k-d) - \alpha_1 \frac{\zeta_f(x(k-d))}{1 + \gamma u^2(k-d)} |e_\Delta(k)| \quad (\text{III.12})$$

$$\theta_g(k) = \theta_g(k-d) - \alpha_2 \frac{\zeta_g(x(k-d))u(k-d)}{1 + \gamma u^2(k-d)} |e_\Delta(k)| \quad (\text{III.13})$$

Où  $\alpha_1 > 0$ ,  $\alpha_2 > 0$ .

Le théorème suivant montre les propriétés de ce contrôleur adaptatif indirect flou.

**Théorème :**

Étant donné le système défini par (III.1) satisfaisant les suppositions 1 et 2, quand

$\eta_f + \frac{\eta_g}{\gamma} + \alpha_1 + \frac{\alpha_2}{\gamma} < 2$ , la loi de commande (III.5) avec la loi d'adaptation (III.8),

(III.10) et (III.11) s'assurera que tous les signaux dans le système en boucle fermée sont bornés, et l'erreur de poursuite converge à un petit voisinage de l'origine.

**Preuve:**

Définir les paramètres d'erreurs  $\bar{\rho}_f(k) = \rho_f^* - \rho_f$ ,  $\bar{\rho}_g(k) = \rho_g^* - \rho_g$ , de (III.10) et (III.11),

$\rho_f^*(k)$  et  $\rho_g^*(k)$  peuvent être exprimés sous la forme:

$$\bar{\rho}_f(k) = \bar{\rho}_f(k-d) - \eta_f \frac{|e_\Delta(k)|}{1 + \gamma u^2(k-d)} \quad (\text{III.14})$$

$$\bar{\rho}_g(k) = \bar{\rho}_g(k-d) - \eta_g \frac{|u(k-d)|}{1 + \gamma u^2(k-d)} |e_\Delta(k)| \quad (\text{III.15})$$

De (III.12) et (III.13),  $\Phi_f(k)$  et  $\Phi_g(k)$  peuvent être exprimés comme :

$$\Phi_f(k) = \Phi_f(k-d) - \alpha_1 \frac{\zeta_f(x(k-d))}{1 + \gamma u^2(k-d)} e_\Delta(k) \quad (\text{III.16})$$

$$\Phi_g(k) = \Phi_g(k-d) - \alpha_2 \frac{\zeta_g(x(k-d))u(k-d)}{1 + \gamma u^2(k-d)} e_\Delta(k) \quad (\text{III.17})$$

Considérons la fonction

$$V(k) = V_1 + V_2 \quad (\text{III.18})$$

$$\text{Où } V_1(k) = \frac{1}{\eta_f} \bar{\rho}_f^T(k) \bar{\rho}_f(k) + \frac{1}{\eta_g} \bar{\rho}_g^T(k) \bar{\rho}_g(k), \quad V_2(k) = \frac{1}{\alpha_1} \Phi_f^T(k) \Phi_f(k) + \frac{1}{\alpha_2} \Phi_g^T(k) \Phi_g(k)$$

Soit  $\Delta V(k) = V(k) - V(k-d)$ , et considérons le cas où  $|e(k)| \leq \delta(k)$ , dans ce cas,

$e_\Delta(k) = 0$ , ainsi  $\Delta V(k) = 0$ , donc seulement la région  $|e(k)| > \delta(k)$  est considérée dans la

preuve suivante.

Si  $|e(k)| > \delta(k)$ , en utilisant (III.14) et (III.15), nous obtenons :

$$V_1(k) - V_1(k-d) = -2 \left[ \bar{\rho}_f(k-d) + \bar{\rho}_g(k-d) |u(k-d)| \right] \frac{|e_\Delta(k)|}{1 + \gamma u^2(k-d)} + \Xi_1(k-d) e_\Delta^2(k) \quad (\text{III.19})$$

Où

$$\Xi_1(k-d) = \frac{\eta_f}{(1+\gamma u^2(k-d))^2} + \frac{\eta_g u^2(k-d)}{(1+\gamma u^2(k-d))^2}$$

En utilisant (III.16) et (III.17), on trouve :

$$V_2(k) - V_2(k-d) = -2 \left[ \Phi_f^T(k-d) \zeta_f(x(k-d)) + \Phi_g^T(k-d) \zeta_g(x(k-d)) |u(k-d)| \right] \frac{|e_\Delta(k)|}{1+\gamma u^2(k-d)} + \Xi_2(k-d) e_\Delta^2(k) \quad (\text{III.20})$$

$$\text{Où } \Xi_2(k-d) = \frac{\alpha_1 |\zeta_f(x(k-d))|^2}{(1+\gamma u^2(k-d))^2} + \frac{\alpha_1 |\zeta_g(x(k-d))|^2 u^2(k-d)}{(1+\gamma u^2(k-d))^2}$$

De (III.7), on a :

$$e(k) - \omega_f(k-d) - \omega_g(k-d)u(k-d) - \varepsilon u(k-d) \text{sign}[\hat{g}(x(k-d) | \theta_f(k-d))] \quad (\text{III.21})$$

La substitution de (III.21) dans (III.20) donne :

$$V_2(k) - V_2(k-d) = -2 \frac{e(k)e_\Delta(k)}{1+\gamma u^2(k-d)} + 2 \frac{e_\Delta(k)}{1+\gamma u^2(k-d)} [(\omega_f(k-d) + \omega_g(k-d)u(k-d)) + \varepsilon u(k-d) \text{sign}(\hat{g})] + \Xi_2(k-d) e_\Delta^2(k) \quad (\text{III.22})$$

De (III.9), nous savons que :

$$e(k) = e_\Delta(k) + \text{sign}(e(k)) \delta(k) \quad (\text{III.23})$$

$$\text{sign}(e(k)) \delta(k) = |e_\Delta(k)| \quad (\text{III.24})$$

En utilisant (III.23) et (III.24), nous obtenons :

$$V_2(k) - V_2(k-d) \leq -2 \frac{e_\Delta^2(k) + |e_\Delta(k)| \delta(k)}{1+\gamma u^2(k-d)} + \frac{2|e_\Delta(k)|}{1+\gamma u^2(k-d)} (\rho_f^* + \rho_g^* |u(k-d)| + \varepsilon |u(k-d)|) + \Xi_2(k-d) e_\Delta^2(k) = \left[ \frac{-2}{1+\gamma u^2(k-d)} + \Xi_2(k-d) \right] e_\Delta^2(k) + \frac{2|e_\Delta(k)|}{1+\gamma u^2(k-d)} (\rho_f^* + \rho_g^* |u(k-d)| + \varepsilon |u(k-d)|) - \delta(k) \quad (\text{III.25})$$

En utilisant (III.19) et (III.25), nous obtenons :

$$\Delta V(k) \leq [\Xi_1(k-d) + \frac{-2}{1+\gamma u^2(k-d)} \Xi_2(k-d)] e_\Delta^2(k) + \frac{2|e_\Delta(k)|}{1+\gamma u^2(k-d)} [\rho_f(k-d) + \rho_g(k-d) |u(k-d)| + \varepsilon |u(k-d)| - \delta(k)] \quad (\text{III.26})$$

Puisque  $\eta_f + \frac{\eta_g}{\gamma} + \alpha_1 + \frac{\alpha_2}{\gamma} < 2$ , donc  $\Xi_1(k-d) + \frac{-2}{1+\gamma u^2(k-d)} + \Xi_2(k-d) < 0$

en utilisant (III.8), nous avons:

$$\Delta V(k) \leq 0 \quad (\text{III.27})$$

Ceci s'assure que la fonction  $V(k)$  est bornée, ce qui implique la bornétude de  $\theta_f(k)$ ,  $\theta_g(k)$ ,  $\rho_f(k)$ , et  $\rho_g(k)$

Soit  $2 - \eta_f - \frac{\eta_g}{\gamma} - \alpha_1 - \frac{\alpha_2}{\gamma} = \eta$ , de (III.26), nous obtenons :

$$\Delta V(k) \leq -\frac{\eta}{1 + \gamma u^2(k-d)} e_{\Delta}^2(k) \quad (\text{III.28})$$

De (III.28), on a :

$$(1 + \gamma u^2(k-d)) \Delta V(k) \leq -\eta e_{\Delta}^2(k) \quad (\text{III.29})$$

Par conséquent :

$$\Delta V(k) \leq -\eta e_{\Delta}^2(k) \quad (\text{III.30})$$

Ou

$$\Delta V(k) \leq V(k-d) - \eta e_{\Delta}^2(k) \quad (\text{III.31})$$

L'addition de (III.31) de  $d$  à  $k$  donne:

$$V(k) + V(k-1) + \dots + V(k-d+1) \leq V(0) + V(1) + \dots + V(d) - \sum_{j=d}^k \eta e_{\Delta}^2(j) \quad (\text{III.32})$$

Nous savons que pour  $k > 0$  arbitraire,  $V(k)$  est bornée, ainsi :

$$\lim_{l \rightarrow \infty} \sum_{k=d}^l \eta e_{\Delta}^2(k) < \infty \quad (\text{III.33})$$

Ceci implique que :

$$\lim_{l \rightarrow \infty} \eta e_{\Delta}^2(k) = 0 \quad (\text{III.34})$$

De (III.9), nous concluons cela  $|e(k)| \leq \delta(k)$  par conséquent, l'erreur de poursuite  $e(k)$  converge à un petit voisinage de l'origine.

**Remarque :**

Tant que les valeurs initiales pour  $\rho_f(k)$  et  $\rho_g(k)$  sont positif, de (III.10) et (III.11), nous obtenons  $\rho_f(k) > 0$  et  $\rho_g(k) > 0$ , donc  $\delta(k) > 0$ . De (III.5), nous savons que  $u(k)$  est bornée, ainsi  $\rho_f(k)$  et  $\rho_g(k)$  sont bornées. Nous concluons que  $\delta(k)$  est positif et bornée.

### III.4. Simulation :

Considérons le système non linéaire discret SISO suivant [30] :

$$y(k+1) = \frac{2.5y(k)y(k-1)}{1+y^2(k)+y^2(k-1)} + 0.7 \sin(0.5(y(k)+y(k-1))) \cos(0.5(y(k)+y(k-1))) \\ + 1.2u(k) + 1.4 u(k-1)$$

Pour s'assurer que  $y(k)$ ,  $y(k-1)$  et  $u(k-1)$  sont dans une région fixe, nous utilisons la transformation linéaire suivante [31] :

$$\bar{y}(k) = \frac{y(k)}{1 + |y(k)|}$$

Il est clair que  $\bar{y}(k) \in [-1, 1]$  pour  $y(k)$  arbitraire. Ceci peut également être employé pour  $y(k-1)$  et  $u(k-1)$ .

Le signal de référence est supposé être  $r = 0.5 + \frac{1}{4} \cos\left(\frac{k\pi T}{4}\right) + \frac{1}{4} \sin\left(\frac{k\pi T}{2}\right)$ , Soient les conditions initiales  $y(1) = 0.5$ ,  $y(0) = 0$ ,  $f(0) = 0.005$ ,  $g(0) = 0.005$ ,  $u(0) = 0$  et chaque élément de  $f(0)$  et  $g(0)$  est choisi aléatoirement dans l'intervalle  $[-0.5, 0.5]$  et  $[0.5, 1.7]$ , respectivement. D'autres paramètres sont choisis en tant que  $\gamma = 6$ ,  $\varepsilon = 0.01$ .  $\alpha_1 = 1.7$ ;  $\alpha_2 = 1.5$ ;  $\eta_f = 0.001$ ;  $\eta_g = 0.001$ ;

Les variables d'entrée du système flou sont  $x_1 = \bar{y}(k)$ ,  $x_2 = y(k-1)$ ,  $x_3 = \bar{u}(k-1)$ , respectivement. Les fonctions d'appartenance pour  $x_i$  ( $i = 1, 2, 3$ ) sont choisis comme suit :

$$\mu_{F_1^1}(x_i) = \exp\left(-\left(\frac{x_i + 1}{0.9}\right)^2\right), \mu_{F_1^2}(x_i) = \exp\left(-\left(\frac{x_i}{0.9}\right)^2\right), \mu_{F_1^3}(x_i) = \exp\left(-\left(\frac{x_i - 1}{0.9}\right)^2\right).$$

Le résultat de simulation est présenté sur les figures (III.1) et (III.2), où figure (III.1) montre la sortie du système et la trajectoire désirée de référence, figure (III.2) représente l'erreur  $e(k)$  et la taille de la zone-morte  $\pm \delta(k)$ , qui indique que l'erreur de poursuite  $e(k)$  converge à un voisinage de l'origine.

### III.5. Résultats de simulation

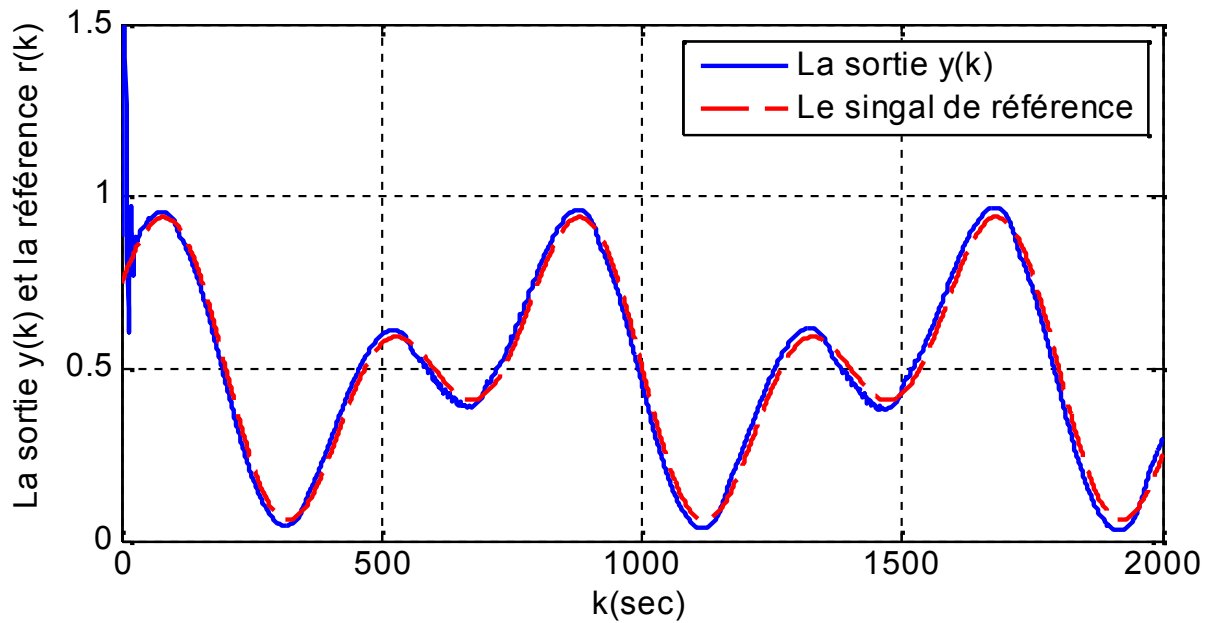


Figure (III.1) : La sortie de système y(k) et signal de référence r(k) .

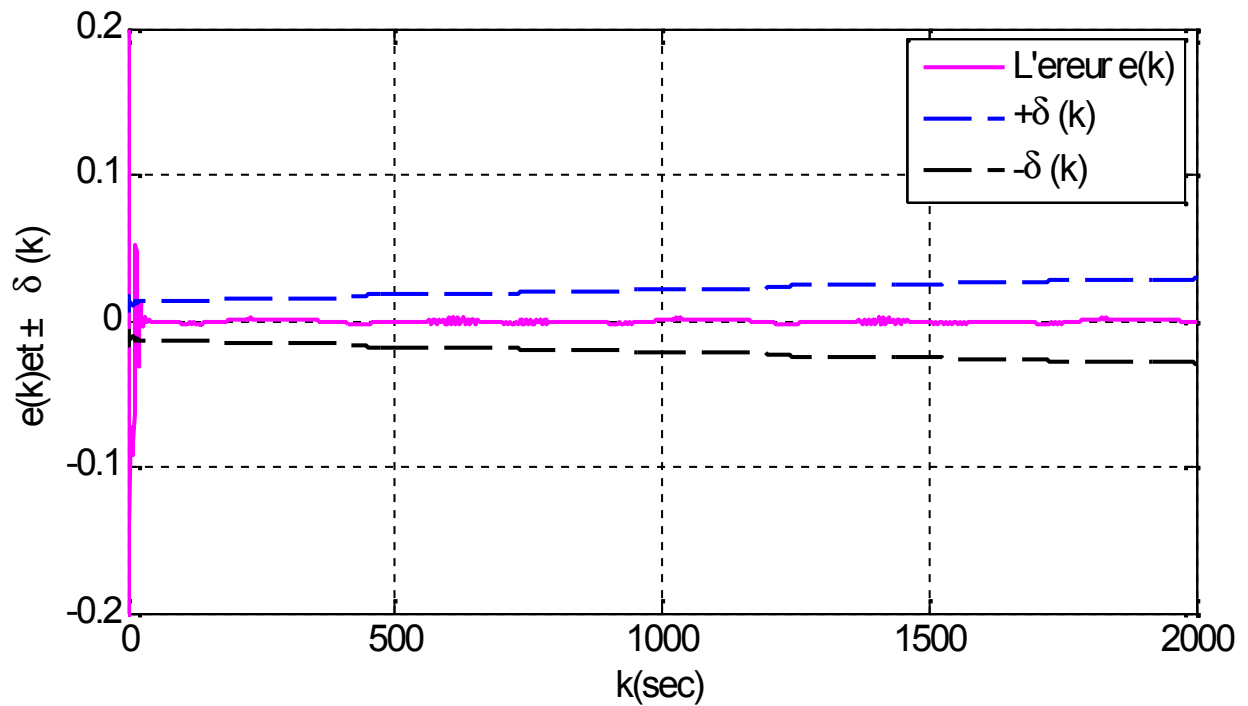


Figure (III.2) : L'erreur e (k) et la taille de la zone morte ± (k).

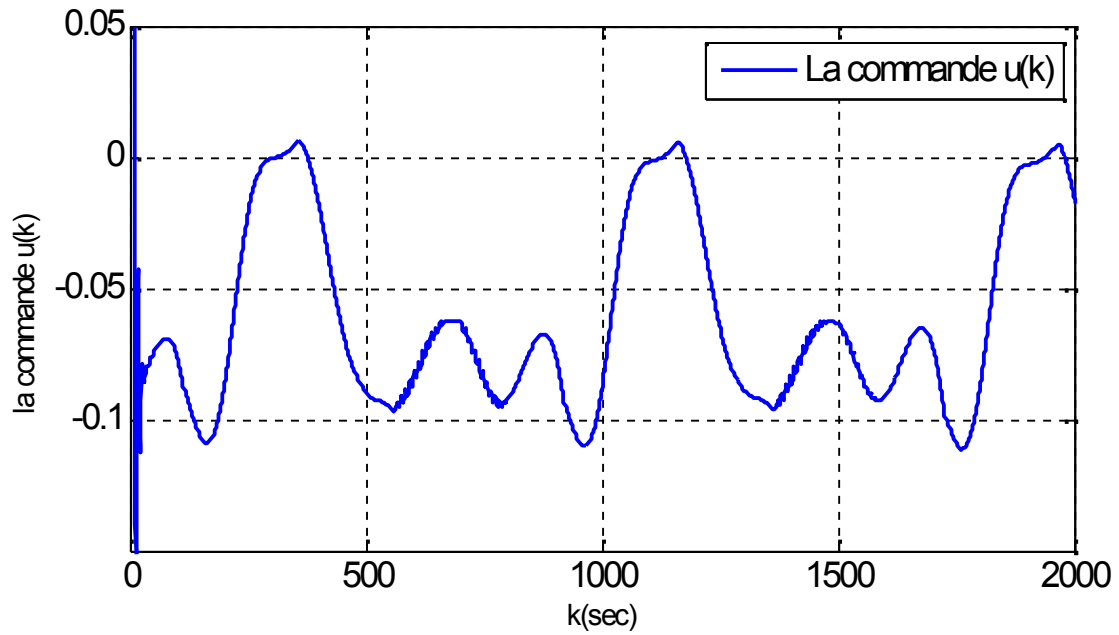


Figure (III.3) : Le signal de Commande  $u(k)$ .

Le résultat de simulation est représenté sur les figures (III.1) et (III.3), où figure (III.1) montre la sortie du système et la trajectoire désirée de référence dont on remarque une bonne poursuite, dans la figure (III.3) on a représenté le signal de la commande tandis que la figure (III.2) représente le signal de l'erreur  $e(k)$  et la taille de la zone-morte  $\pm \delta(k)$ , dont on voit que l'erreur de poursuite  $e(k)$  converge à un voisinage de l'origine.

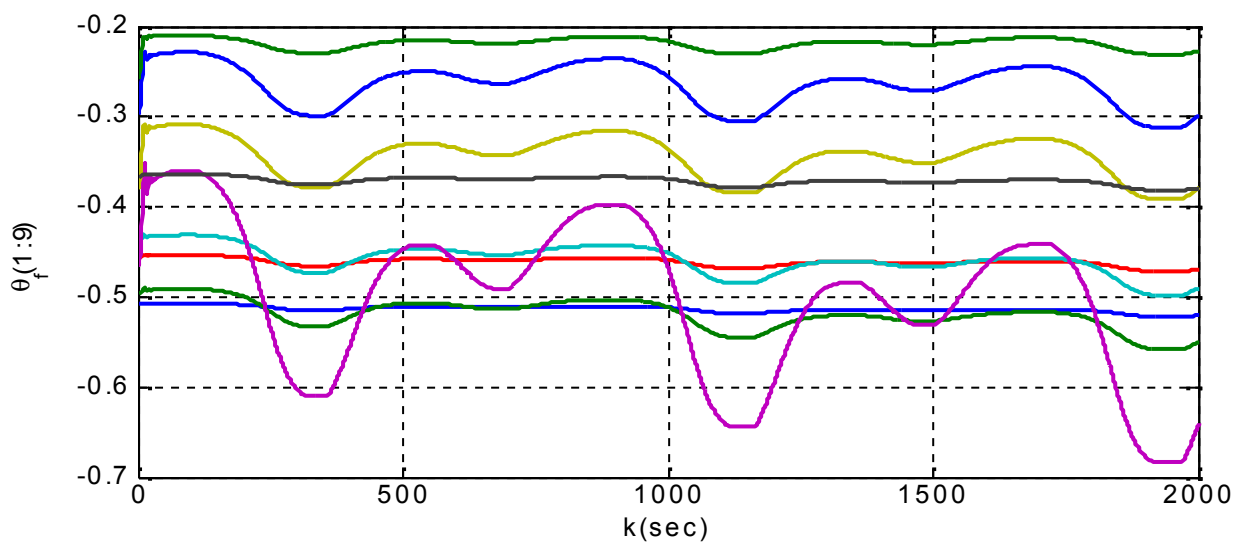


Figure (III.4) : Loi d'adaptation  $\theta_f(1:9)$ .

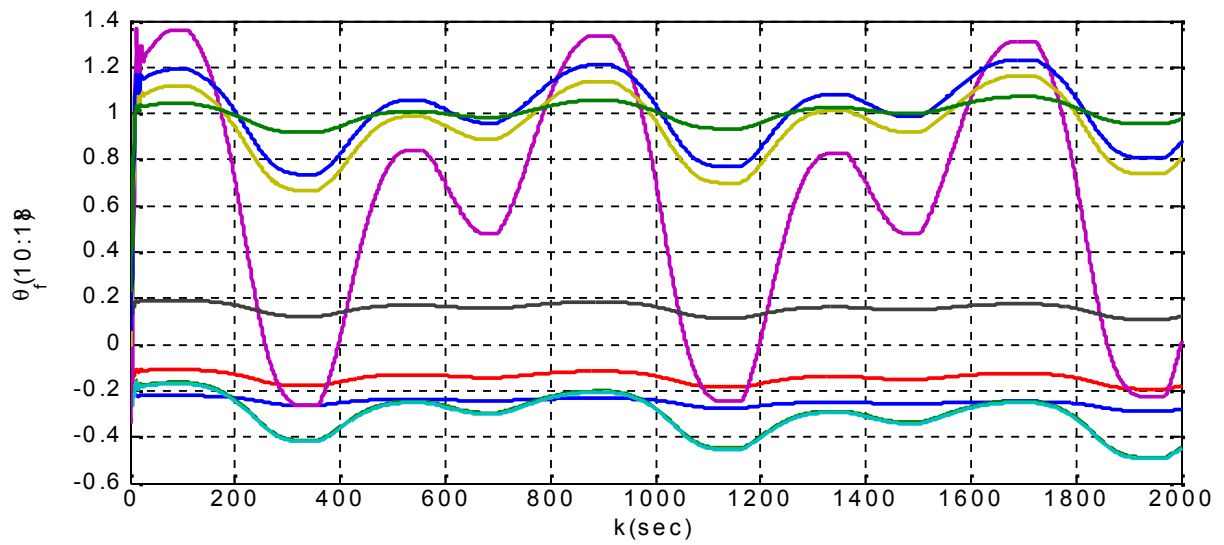


Figure (III.5) : Loi d'adaptation  $\theta_f(10 : 18)$ .

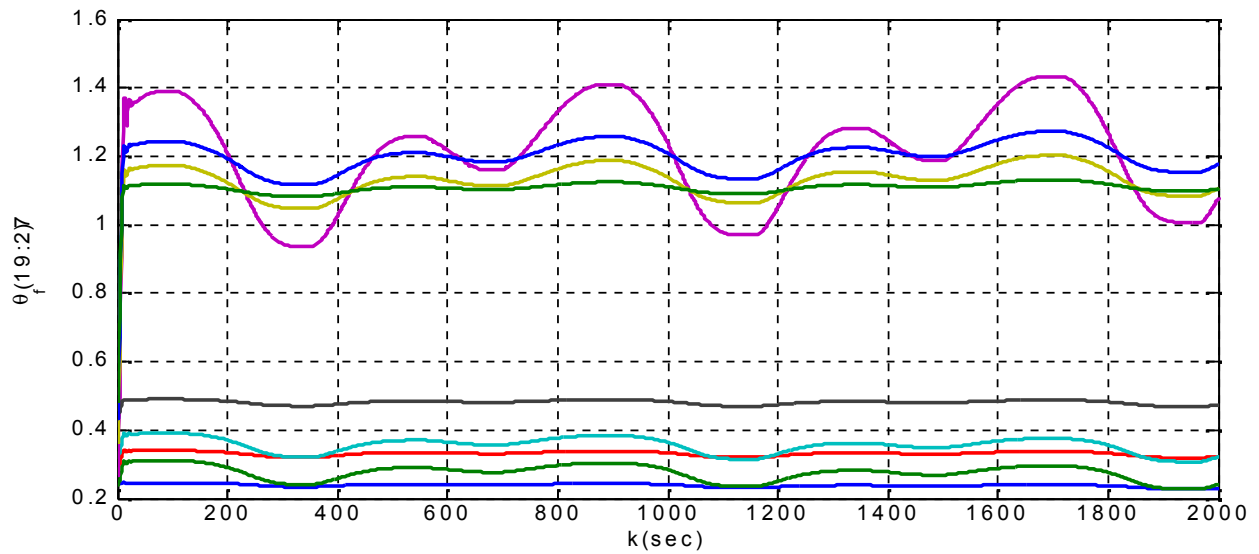


Figure (III.6) : Loi d'adaptation  $\theta_f(19 : 27)$ .

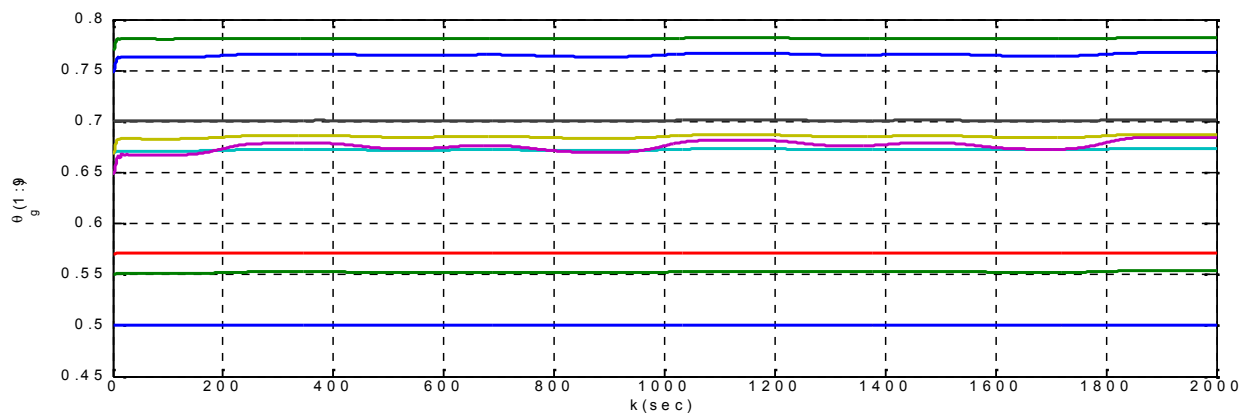


Figure (III.7) : Loi d'adaptation  $\theta_g(1 : 9)$ .

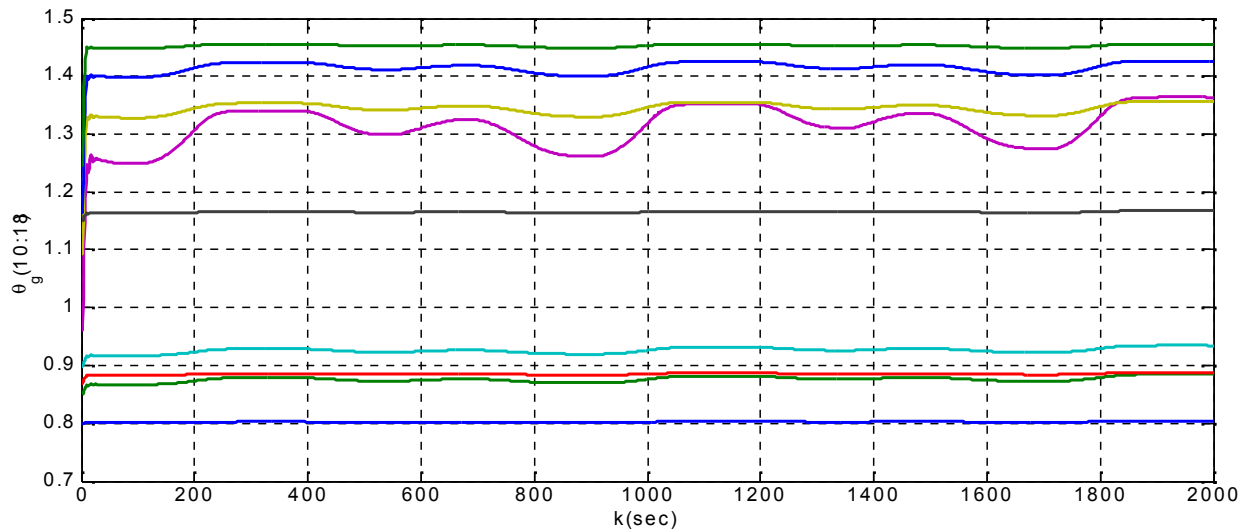


Figure (III.8) : Loi d'adaptation  $\theta_g(10 : 18)$ .

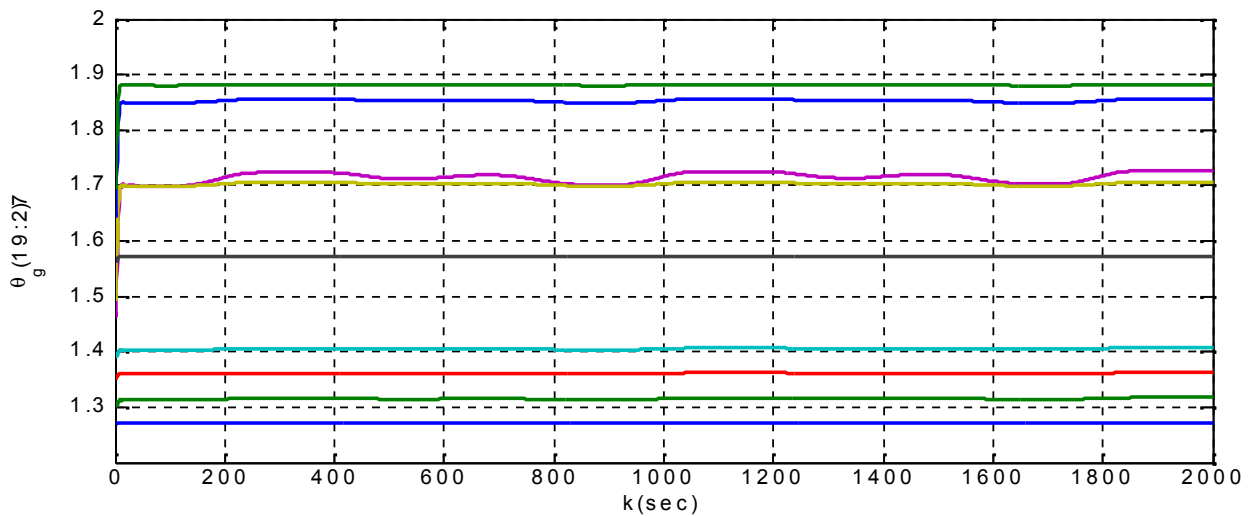


Figure (III.9) : Loi d'adaptation  $\theta_g(19 : 27)$ .

On remarque que tous les signaux des lois d'adaptation  $\theta_f(k)$  et  $\theta_g(k)$  sont bornés.

### III.7. Conclusion

Dans ce chapitre, un schéma de commande adaptative floue indirecte est développé pour une classe de systèmes non linéaires discrets. Dans ce schéma, il n'est pas nécessaire de connaître la borne supérieure de l'erreur d'approximation, car la borne inconnue est estimée en utilisant une loi d'adaptation et la taille de la zone morte est ajustée d'une manière adaptative avec la borne estimée. On a prouvé que le schéma proposé peut garantir que tous les signaux dans le système en boucle fermée sont bornés ce qui implique que l'erreur de poursuite converge vers un petit voisinage de l'origine.

# Chapitre IV

Commande adaptative floue directe  
des

systemes non linéaires discrets

## IV.1.Introduction:

Ces dernières années, diverses techniques adaptatives floues de commande ont été développées pour traiter les systèmes non-linéaires avec de dynamique mal comprise. Cependant, la plupart des résultats sont limités à des systèmes continus ce qui ne peut pas être étendu directement aux systèmes discrets. Dans des applications pratiques, presque tous les systèmes de commande floue sont implémentés sur calculateurs numériques, puisque les signaux de commande ne peuvent être appliqués qu'à des instants fixes de temps, donc quelques avantages des contrôleurs continus sont perdus au moyen de discrétisation. Il est nécessaire de tenir compte du fait que le contrôleur est en réalité un système discret et non pas continu [29]. Récemment un contrôleur flou à temps discret pour une classe de systèmes dynamique non-linéaire a été présenté.

Dans [26], un schéma de commande adaptative est présenté où les systèmes flous de Takagi Sugeno direct l'arrangement de commande adaptative a été présenté où Takas-takas-Sugeno sont utilisés comme des approximateurs fonctionnels, une zone morte continue a été utilisée pour garantir la convergence de l'erreur de poursuite à un voisinage de l'origine.

Basé sur [25] et [26], le gain d'adaptation et la direction de la descente ont été mises à jour de manière à chercher à optimiser certain fonction coût [28]. Dans [29] une commande adaptative directe pour une classe de systèmes non linéaires discrets a été proposée. Dans [27] la loi d'adaptation a été conçue par une mort-zone continue, dont sa taille a été basée sur l'erreur d'approximation du système flou, et donc il est nécessaire de supposer que les bornes de l'erreur d'approximation sont connues à l'avance. Bien que l'erreur d'approximation est bornées, malheureusement, dans beaucoup de systèmes pratiques, telles bornes ne pourraient pas être disponibles, et elle est habituellement employée, qui pourrait avoir comme conséquence un conservateur.

Dans ce chapitre, une méthode de commande adaptative directe floue est développée pour une classe de système non-linéaire discret avec la dynamique mal connu. Dans ce travail, le système flou est employé pour concevoir directement le contrôleur, et des paramètres inconnus sont ajustés par une zone morte variable, dont sa taille est ajustée de manière adaptative avec les bornes estimées sur l'erreur d'approximation du système flou.

Le schéma de conception proposé garantit que tous les signaux dans le système en boucle fermée résultant sont bornés, et l'erreur de poursuite converge vers un petit voisinage de l'origine.

## IV.2. Commande Adaptative Floue Directe:

Considérons le système non linéaire discret (SISO) sous la forme suivante :

$$y(k+d) = f(x(k)) + g(x(k))u(k) \quad (\text{IV.1})$$

Où  $u(k) \in R$  et  $y(k) \in R$  sont l'entrée et la sortie du système respectivement,  $d$  et le retard du système, et  $x(k) = [y(k), \dots, y(k-n), u(k-1), \dots, u(k-m)]^T$ ,  $f(x(k))$  et  $g(x(k))$  sont des fonctions lisses inconnues, et la supposition suivante est faite :

### Hypothèse 1:

Il existe une constante  $g_1$  tel que :  $0 < g(x(k)) < g_1 < \infty$ .

L'objectif de ce chapitre est de concevoir un contrôleur adaptatif flou direct tel que la sortie du système  $y(k)$  suit le signal de référence  $r(k)$ , alors que tous les signaux dans le système en boucle fermée demeurent bornés.

Si les fonctions  $f(x(k))$  et  $g(x(k))$  sont connues exactement, il est bien connu que pour le système (IV.1), il existe un contrôleur idéal :

$$u^*(k) = \frac{-f(x(k)) + r(k+d)}{g(x(k))} \quad (\text{IV.2})$$

Ceci conduit la sortie du système à poursuivre parfaitement une trajectoire de référence connue  $r(k)$  c'est-à-dire :

$$e(k+d) = r(k+d) - y(k+d) = 0$$

Ceci signifie qu'après  $d$  étapes, nous avons  $e(k) = 0$ .

Puisque les fonctions  $f(x(k))$  et  $g(x(k))$  sont inconnues, le contrôleur idéal  $u^*(k)$  de (IV.2) ne peut pas être implémenté, nous supposons que les fonctions  $f(x(k))$  et  $g(x(k))$

peuvent être approximées par des systèmes flous. Le système flou utilisé est une collection de règles floues SI-ALORS de la forme [18] :

$$R^{(l)} : \quad \text{IF } x_1 \text{ is } F_1^l \text{ and } \dots \text{ and } x_n \text{ is } F_n^l \text{ THEN } y \text{ is } G^l$$

Où  $x = (x_1, \dots, x_n)^T$  et  $y$  sont l'entrée et la sortie du système flou, respectivement,  $F_1^l$  et  $G^l$  sont des ensembles flous, pour  $l = 1, \dots, m$ .

En utilisant la stratégie de fuzzification par singleton, une défuzzification par le centre de gravité, et le produit d inférence, la sortie finale du système flou est donnée comme suit :

$$y(x) = \frac{\sum_{j=1}^m y^j \left( \prod_{i=1}^n \mu_{F_i^j}(x_i) \right)}{\sum_{j=1}^m \prod_{i=1}^n \mu_{F_i^j}(x_i)} \quad (\text{IV.3})$$

Où  $y^j$  est le point auquel la fonction d'appartenance de  $G^l$  atteint sa valeur maximale. En présentant le concept des fonctions floues de base  $\xi(x)$ , la sortie donnée par (IV.3) peut être réécrite sous la forme compacte suivante :

$$y(x) = \theta^T \xi(x) \quad (\text{IV.4})$$

Où  $\theta = (y^1, \dots, y^m)^T$ ,  $\xi(x) = (\xi^1(x), \dots, \xi^m(x))^T$

$$\xi^j(x) = \frac{\left( \prod_{i=1}^n \mu_{F_i^j}(x_i) \right)}{\sum_{j=1}^m \prod_{i=1}^n \mu_{F_i^j}(x_i)}$$

Soit le contrôleur idéal  $u^*(k)$  approximé sur un ensemble compact  $U$ , par un système flou comme suit :

$$u(k) = u(\bar{x}(k) | \theta) \quad (\text{IV.5})$$

Où  $\bar{x}(k) = (x^T(k), r(k+d))^T$

En utilisant (IV.1), (IV.2) et (IV.5), et l'équation d'erreur peut être écrite comme suit :

$$e(k+d) = g(x(k))(u^*(k) - u(\bar{x}(k) | \theta)) \quad (\text{IV.6})$$

Définissons le paramètre optimal du système flou comme suit :

$$\theta^* = \arg \min_{\theta \in \Omega, \bar{x} \in U} [\sup |u^*(k) - u(\bar{x}(k) | \theta)|]$$

Où  $\Omega$  est un ensemble compact de paramètre du contrôleur. Noter que le paramètre optimal  $\theta^*$  est une quantité constante artificielle présentée seulement pour un but d'analyse, et ses valeur ne sont pas nécessaires pour l'implémentation.

Dénoter l'erreur d'estimation des paramètres comme :  $\Phi(k) = \theta^* - \theta$ ,

et  $\omega(k) = u^*(k) - u(\bar{x}(k) | \theta^*)$  comme erreurs d'approximation minimum

Dans ce chapitre, nous supposons que le système flou utilisé ne violent pas la propriété d'approximation universelle [18] sur l'ensemble compact  $U$ , qui est assumé assez grand de sorte que les variables d'état demeurent dans  $U$  sous la commande en boucle fermée. Ainsi il est raisonnable de supposer que l'erreur minimum d'approximation est bornée pour tout  $\bar{x}(k) \in U$ .

### **Hypothèse 2:**

Il existe une constante inconnue  $\rho^*$  telle que  $|\omega(k)| \leq \rho^*$ , et nous définissons  $\rho$  comme l'estimation de  $\rho^*$ . Avec ces définitions, l'équation d'erreur (IV.6) peut être réécrite sous la forme:

$$e(k+d) = g(x(k))\Phi^T(k)\zeta(\bar{x}(k)) + g(x(k))(k) \quad (\text{IV.7})$$

Afin de répondre à l'objectif de commande, dans [26], une zone-morte continue est employée pour concevoir la loi d'adaptation, mais la borne d'erreur d'approximation est nécessaire.

Dans ce travail, nous utilisons une zone-morte variable dans le temps pour concevoir la loi d'adaptation. La taille de la zone-morte  $\delta(k)$  est ajustée de manière adaptative par la loi d'adaptation suivante :

$$\delta(k) = g_1 \rho(k-d) \quad (\text{IV.8})$$

La zone-morte variable dans le temps est définie comme [20]

$$e_{\Delta}(k) = \begin{cases} e(k) - \text{sign}(e(k)) \delta(k) & \text{si } |e(k)| > \delta(k) \\ 0 & \text{si } |e(k)| \leq \delta(k) \end{cases} \quad (\text{IV.9})$$

$$\text{Où, } \text{sign}(x) = \begin{cases} -1, & \text{si } x < 0 \\ 1, & \text{si } x \geq 0 \end{cases}$$

En utilisant la loi d'adaptation suivante pour ajuster le paramètre  $\rho^*(k)$

$$\rho(k) = \rho(k-d) + \beta |e_{\Delta}(k)| \quad (\text{IV.10})$$

Où  $\beta > 0$ .

Le vecteur de paramètres inconnus  $\theta(k)$  est mis à jour par la loi d'adaptation suivante :

$$\theta(k) = \theta(k-d) + \alpha \zeta(x(k-d)) e_{\Delta}(k) \quad (\text{IV.11})$$

Où  $\alpha > 0$ .

Le théorème suivant montre les propriétés de ce contrôleur adaptatif flou direct.

**Théorème :**

Étant donné le système défini par (IV.1) satisfaisant les hypothèses 1 et 2, quand

$$\alpha + \beta \leq \frac{2}{g_1}, \text{ La loi de commande (IV.5) avec les lois d'adaptation (IV.8), (IV.10) et (IV.11)}$$

s'assurera que tous les signaux dans le système en boucle fermée sont bornés, et l'erreur de poursuite converge à un petit voisinage de l'origine.

**Preuve :**

Définissons les paramètres d'erreurs  $\bar{\rho}(k) = \rho^*(k) - \rho(k)$ , de (IV.10) et (IV.11),  $\Phi(k)$  et  $\bar{\rho}(k)$  peuvent être exprimés sous la forme:

$$\bar{\rho}(k) = \bar{\rho}(k-d) - \beta |e_{\Delta}(k)| \quad (\text{IV.12})$$

$$\Phi(k) = \Phi(k-d) - \alpha \zeta(\bar{x}(k-d)) e_{\Delta}(k) \quad (\text{IV.13})$$

Considérons la fonction candidate de Lyapunov :

$$V(k) = \frac{1}{\alpha} \Phi^T(k) \Phi(k) + \frac{1}{\beta} \bar{\rho}^T(k) \bar{\rho}(k) \quad (\text{IV.14})$$

Soit  $\Delta V(k) = V(k) - V(k-d)$ , et considérons le cas où  $|e(k)| \leq \delta(k)$ , dans ce cas,  $e_{\Delta}(k) = 0$ , ainsi  $\Delta V(k) = 0$ , donc seulement la région  $|e(k)| > \delta(k)$  est considérée dans la

preuve suivante.

Si  $|e(k)| > \delta(k)$ , alors

$$\Delta V(k) = -2\Phi^T(k-d)\zeta(\bar{x}(k-d))e_{\Delta}(k) + \alpha |\zeta(\bar{x}(k-d))|^2 e_{\Delta}^2(k) - 2\bar{\rho}(k-d) - |e_{\Delta}(k)| + \beta e_{\Delta}^2(k) \quad (\text{IV.15})$$

Basé sur (IV.7), il peut être démontré que :

$$\Phi^T(k-d)\zeta(\bar{x}(k-d)) = \frac{e(k)}{g(x(k-d))} - \omega(k-d) \quad (\text{IV.16})$$

En utilisant (IV.16), (IV.15) peut être exprimée comme :

$$\Delta V(k) = -2 \frac{e(k)e_{\Delta}(k)}{g(x(k-d))} + 2\omega(k-d)e_{\Delta}(k) + \alpha |\zeta(\bar{x}(k-d))|^2 e_{\Delta}^2(k) - 2\bar{\rho}(k-d) - |e_{\Delta}(k)| + \beta e_{\Delta}^2(k) \quad (\text{IV.17})$$

De (IV.9), nous savons que :

$$e(k) = e_{\Delta}(k) + \text{sign}(e(k))\delta(k) \quad (\text{IV.18})$$

$$\text{sign}(e(k))e_{\Delta}(k) = |e_{\Delta}(k)| \quad (\text{IV.19})$$

En utilisant (IV.19), (IV.18) et l'hypothèse 1, nous obtenons :

$$\Delta V(k) = -2 \frac{e_{\Delta}^2(k)}{g_1} - 2 \frac{|e_{\Delta}(k)|}{g_1} \delta(k) + 2\rho^* |e_{\Delta}(k)| + \alpha |\zeta(\bar{x}(k-d))|^2 e_{\Delta}^2(k) - 2\bar{\rho}(k-d) - |e_{\Delta}(k)| + \beta e_{\Delta}^2(k) \quad (\text{IV.20})$$

En utilisant (IV.8) et (IV.10), (IV.20) devient :

$$\Delta V(k) \leq - \left[ \frac{2}{g_1} + \alpha |\zeta(X(k-d))|^2 + \beta \right] e_{\Delta}^2(k) \quad (\text{IV.21})$$

Puisque  $\alpha + \beta \leq \frac{2}{g_1}$ , nous obtenons :

$$\Delta V(k) \leq 0 \quad (\text{IV.22})$$

Ceci s'assure que la fonction  $V(k)$  est bornée, ce qui implique la bornétude de  $\theta(k)$ , et  $\rho(k)$ .

Soit  $\frac{2}{g_1} - \alpha - \eta$ , de (IV.21), nous obtenons :

$$V(k) \leq V(k-d) \eta e_{\Delta}^2(k) \quad (\text{IV.23})$$

L'addition de (IV.31) de  $d$  à  $k$  donne:

$$V(k) + V(k-1) + \dots + V(k-d+1) \leq V(0) + V(1) + \dots + V(d) - \sum_{j=d}^k \eta e_{\Delta}^2(j) \quad (\text{IV.24})$$

Nous savons que pour  $k > 0$  arbitraire,  $V(k)$  est bornée, ainsi

$$\lim_{l \rightarrow \infty} \sum_{k=d}^l e_{\Delta}^2(k) < \infty \quad (\text{IV.25})$$

Ceci implique que :

$$\lim_{k \rightarrow \infty} e_{\Delta}^2(k) = 0 \quad (\text{IV.26})$$

De (IV.9), nous concluons que  $|e(k)| \leq \delta(k)$  par conséquent, l'erreur de poursuite  $e(k)$  converge à un petit voisinage de l'origine.

**Remarque :**

Tant que la valeur initiale pour  $\rho(0)$  est positive, de (IV.10) nous obtenons  $\rho(k) > 0$  donc

$\delta(k) > 0$ , tant que  $\rho(k)$  est bornée, donc  $\delta(k)$  est bornée.

### IV.3.Exemple de simulation :

Considérons le modèle de réservoir qui peut être représenté par l'équation différentielle

suivante [27]: 
$$\frac{dh(t)}{dt} = \frac{-c\sqrt{2gh(t)}}{A_r} + \frac{1}{A_r(h(t))} u(t)$$

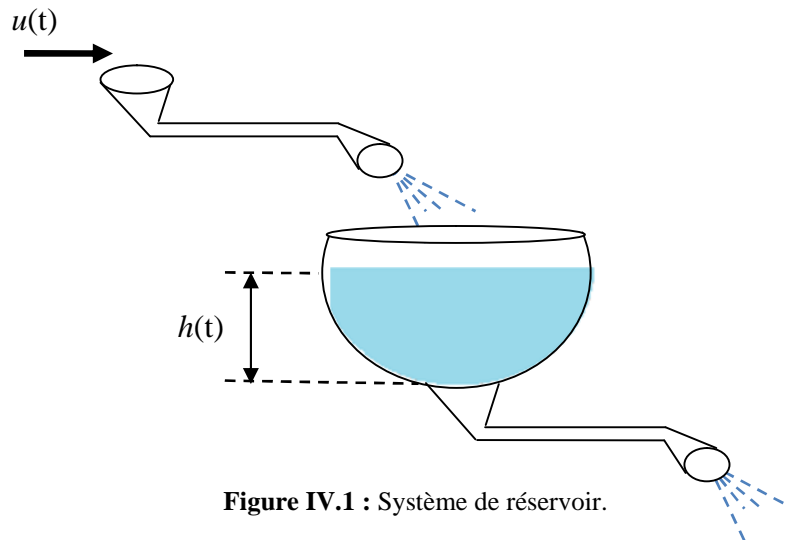


Figure IV.1 : Système de réservoir.

Où  $u(t)$  est l'entrée de système (la commande),  $h(t)$  est le niveau du liquide (sortie du système),  $A_r(h(t))$  est la surface de coupe transversale de la cuve,  $g = 9,8 \text{ m/s}^2$  est l'accélération de la pesanteur,  $d$  est la coupe connue de la sortie du tuyau.

En utilisant les paramètres donnés dans [27],  $d=1$ ,  $A_r(h(t)) = \sqrt{ah(t) + b}$ ,  $a = 1, b = 3$ , et en utilisant l'approximation d'Euler pour la discrétisation du système, nous avons:

$$h(k+1) = h(k) + T \left[ \frac{\sqrt{19.6h(k)}}{A_r(h(k))} + \frac{u(k)}{A_r(h(k))} \right] \quad (\text{IV.27})$$

Où  $T = 0,4$  est la période d'échantillonnage. Noter que le système (IV.27) présente la même forme que (IV.1) avec :

$$f(x(k)) = h(k) - T \frac{\sqrt{19.6h(k)}}{\sqrt{h(k) + 3}}, g(x(k)) = \frac{T}{\sqrt{h(k) + 3}}$$

Nous allons simuler le système pour les valeurs suivantes :

$$h(k) > 0, \text{ Donc la simulation est réalisée. Puisque } g(x(k)) = \frac{T}{\sqrt{h(k) + 3}}$$

Alors  $0 < g(x(k)) < 0.577T$ , nous obtenons  $g_1 = 0.577T$ .

Le contrôleur idéal  $u^*(x(k))$  à être approximé par le système flou de (IV.4), dont l'entrée est  $h(k)$  et  $r(k+1)$ . Faire en sorte que  $h(k)$  est dans une région fixe.

On utilise la transformation suivante :

$$\bar{h}(k) = \frac{h(k)}{1 + |h(k)|}$$

Il est clair que  $\bar{h}(k) \in [-1, 1]$  pour  $h(k)$  arbitraire. Ceci peut également être employé pour  $r(k+1)$ .

Le signal de référence est supposé être  $r(k) = 2 + \sin(k\pi / 150)$ .

Soient les conditions initiales  $y(0) = 0.5$ ,  $\rho(0) = 0.1$ , et chaque élément de (0) est choisi aléatoirement dans l'intervalle  $[-0.5, 0.5]$ ,  $\rho = 15$ ,  $\sigma = 0,002$ . Les variables d'entrée du système flou sont  $x_1 = h(k)$  et  $x_2 = r(k+1)$ , les fonctions d'appartenance pour  $x_1$  et  $x_2$  sont sélectionnés comme suit:

$$\mu F_i^1(x_i) = \exp\left(-\left(\frac{x_i + 1}{0.7}\right)^2\right), \mu F_i^2(x_i) = \exp\left(-\left(\frac{x_i}{0.7}\right)^2\right), \mu F_i^3(x_i) = \exp\left(-\left(\frac{x_i - 1}{0.7}\right)^2\right),$$

Le résultat de simulation est représenté sur les figures (IV.2) et (IV.3), où figure (IV.2), montre la sortie du système et la trajectoire désirée de référence dont on remarque une bonne poursuite, dans la figure (IV.3), on a représenté le signal de la commande tandis que la figure (IV.4) représente le signal de l'erreur  $e(k)$  et la taille de la zone-morte  $\pm \delta(k)$ , dont on voit que l'erreur de poursuite  $e(k)$  converge à un voisinage de l'origine.

#### IV.4. Résultats de simulation:

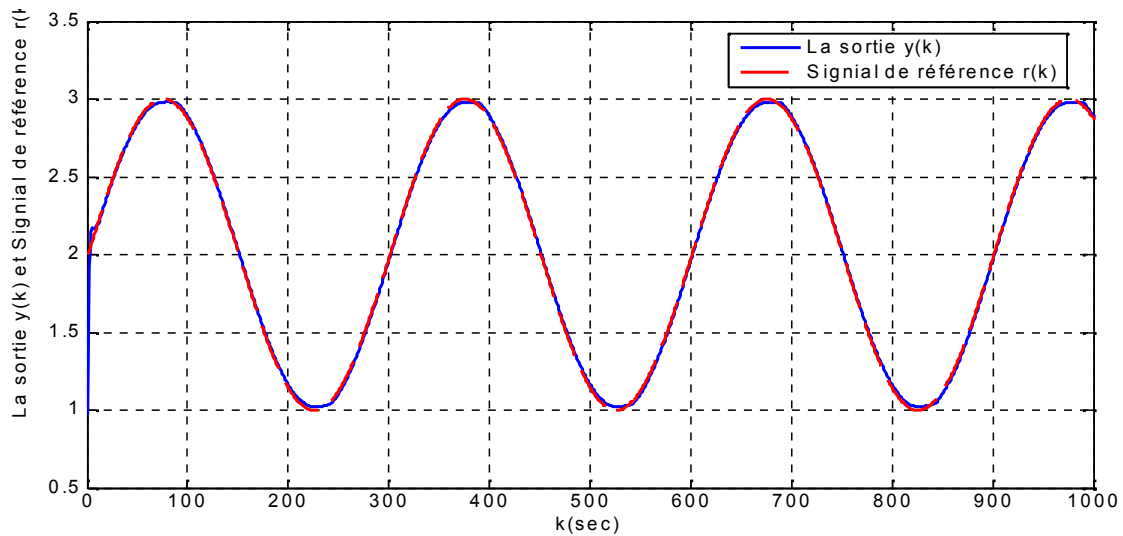


Figure IV.2 : la sortie  $y(k)$  et le signal de référence  $r(k)$ .

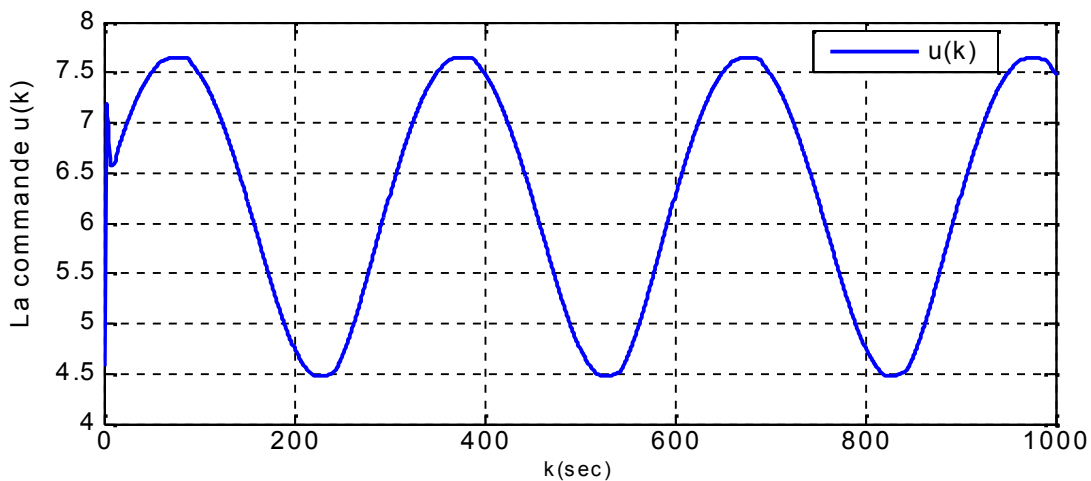


Figure IV.3 : la Commande  $u(k)$ .

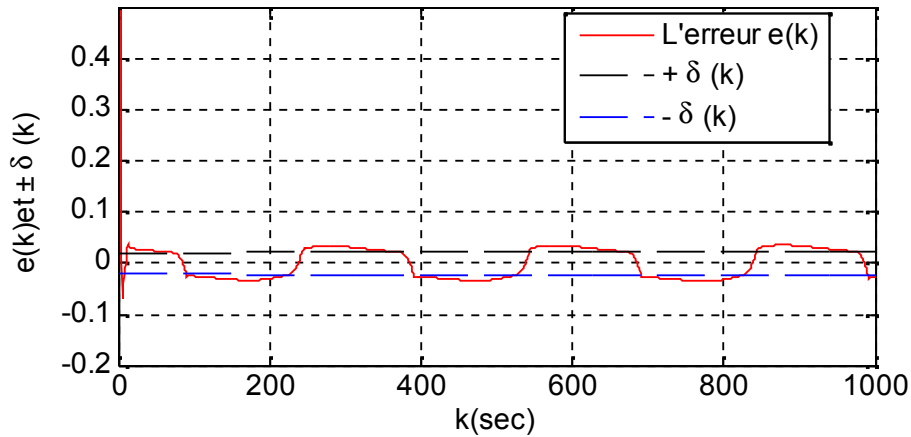


Figure IV.4 : L'erreur  $e(k)$  et la taille de la zone morte  $\pm\delta(k)$ .

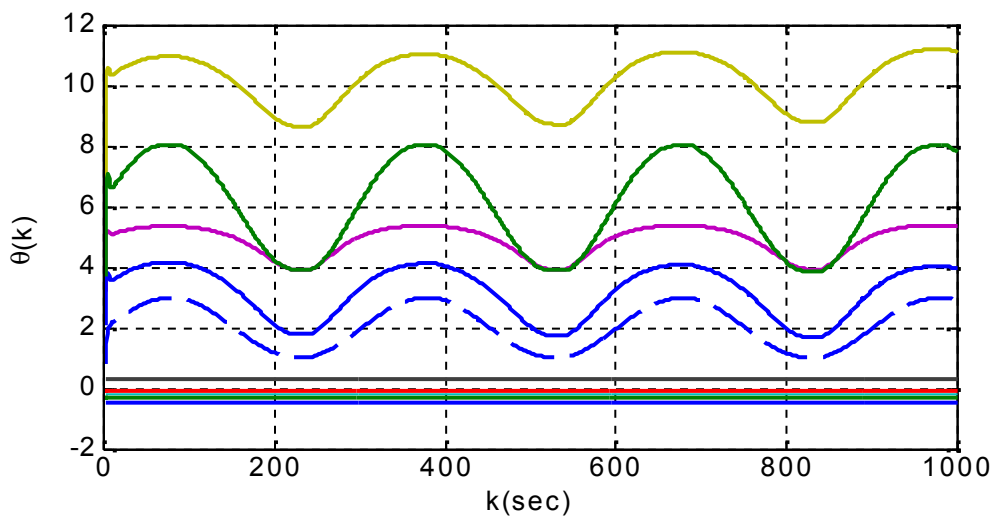


Figure IV.5: Les lois d'adaptation  $\theta(k)$ .

On remarque que tous les signaux de  $\theta(k)$  sont bornés.

## IV.5. Conclusions

Dans ce chapitre, un schéma de commande adaptative floue directe est développé pour une classe de systèmes non linéaires discrets. Dans ce travail, on n'a pas besoin de la borne de l'erreur d'approximation parce que la borne inconnue est estimée en utilisant les lois d'adaptation et la taille de la zone morte est ajustée adaptativement avec la borne estimée. On a pu prouver que le schéma peut forcer l'erreur de poursuite à converger vers un petit voisinage de l'origine.

## Conclusion générale

---

Les travaux présentés dans ce mémoire ont pour objectif principal le développement de structures de commande adaptatives floues stables et robustes, pour une classe de systèmes non linéaires non affines incertains SISO.

Dans un premier temps, nous rappelons les principes de la commande floue avec Les différentes étapes de la conception du contrôleur flou. Ensuite, nous présentons les structures de la commande floue adaptative avec ses deux variantes, directe et indirecte.

Une approche floue indirecte est utilisée pour la commande d'un système non linéaire SISO discret. La stabilité du système en boucle fermée est étudiée en utilisant l'approche de Lyapunov. La mise en œuvre de cette commande est validée par la simulation du fonctionnement d'un système non linéaire discret.

Dans un second temps, il s'agit de développer une approche adaptative directe floue pour la commande d'une classe de systèmes non linéaires discrets dont un système flou est utilisé pour construire ce contrôleur. La méthode de descente du gradient est utilisée pour obtenir les lois d'adaptation des paramètres.

La simulation, sous l'environnement Matlab, de deux systèmes non linéaires SISO, a permis de mettre en évidence les avantages des deux approches développées.

Les résultats obtenus dans le cas des systèmes non linéaires SISO, nous encouragent à étendre les deux approches développées pour le cas des systèmes non linéaires multivariables (MIMO), ainsi que l'ajout d'un terme de commande robuste via la technique de la commande par mode glissant pour avoir une commande robuste.

# Référence

- [1] S. Labiod, « Contribution à la Commande Adaptative Floue des Systèmes Non Linéaires » Thèse Doctorat, École Nationale Polytechnique, 2005.
- [2] R. Boukezzoula, « Commande Floue d'une Classe de Systèmes Non Linéaires : Application au problème de Suivi de Trajectoire », Thèse Doctorat, Université de Savoie, 2005.
- [3] : G.Huguenin, « logique floue», CH-2610 Saint-Imier, 2004.
- [4] : Douki.N « Commande Floue Adaptative par Mode Glissant d'une Classe des Systèmes Non Linéaires » Mémoire Master, Université de Ferhat .A, Setif UFAS (Alger) 2011.
- [5] : J.Lvern, « La logique floue : Concepts et définitions », Électronique radio plan N°541, pp11-18,1992.
- [6] C.C.Lee, “ Fuzzy logic in control system: Fuzzy logic controllers, Parts and “, IEEE Trans, system, Man and Cybernetics, vol.20, no.2, 1990.
- [7] E. Thro et F.M. McNeill, “Fuzzy Logic: A Practical Approach “, illustrated, 1994.
- [8] F. Gomide et W. Pedrycz, “ Fuzzy Systems Engineering: Toward Human-Centric Computing “, Wiley.IEEE, 2007.
- [9] J.Godjevac, « Neuro-Fuzzy Controllers: Design and Application », Presses Polytechniques Et Universitaires Romandes, 1997.
- [10] S. Labiod, “ Contribution à la Commande Adaptative Floue des Systèmes Non Linéaires “, Thèse Doctorat, Ecole Nationale Polytechnique, 2005.
- [11] Y. Oulmas, “ Commande Adaptative Floue de la Position et de l'Orientation des Robots Manipulateurs “, Thèse de Magister, Ecole Nationale Polytechnique, 2005.
- [12] L. Reznik, “Fuzzy Controllers “, Newnes, 1997.
- [13] M.Bodson et S.Sastry, Adaptive Control: Stability, Convergence, and Robustness. Englewood Cliffs, NJ: Prentice-hall, 1989.
- [14] V. V. Chalem, “ Adaptive Control Systems: Techniques and Applications “, Marcel Dekker, 1987.
- [15] Astorm, Bjorn, J. Karl, Wittenmark, “Adaptive Control “, Dover Publications, 2008.
- [16] J. Böhm, J. Fessl, Jan Machacek et V. Bobal, “ Digital Self-Tuning Controllers “, Springer, 2005.
- [17] R. Boukezzoula, “ Commande Floue d'une Classe de Systèmes Non Linéaires : Application au problème de Suivi de Trajéctoire “, Thèse Doctorat, Université de Savoie, 2005.

- [18] L.X.Wang. Adaptive fuzzy systems and control-design and stability analysis, New jersey:Prentice Hall,1994.
- [19] B.S.Chen,C.H.Lee and Y.C.Chang.H tracking design of uncertain nonlinear SISO systems: adaptive fuzzy approach, IEEE Trans.Fuzzy Systems, 2(4):32-43,1996.
- [20] Keun-Mo Koo.Stable adaptive fuzzy controller with timevarying dead-zone, Fuzzy Sets and Systems, 121:161-168,2001.
- [21] S.C. Tong, H.X.Li. Direct adaptive fuzzy output tracking control of nonlinear systems, Fuzzy Sets and Systems,128(1): 107-115, 2002.
- [22] Jang-Hyun Parka, Sam-Jun Seob, Gwi-Tae Parka. Robust adaptive fuzzy controller for nonlinear system using estimation of bounds for approximation errors, Fuzzy Sets and Systems, 133 :19-36,2003.
- [23] S.Jagannathan, M.W.Vandegrift and F.L.Lewis. Adaptive fuzzy logic control of discrete-time dynamical systems, Automatica,36( 2):229-241,2000.
- [24] S.Jagannathan.Adaptive fuzzy logic control of feedback linearizable discrete-time dynamical systems under persistence of excitation, Automatica, 34(11): 1295-1310,1998.
- [25] J.T.Spooner,R.Ordonez and K.M.Passin. Direct adaptive fuzzy control for a class of discrete time systems, in Proc.Amer.Control Conf., Albuquerque, June 4-6: 1814-1818,1997.
- [26] T.J.Spooner,R.Ordonez and K.M.Passin. Indirect adaptive fuzzy control for a class of discrete time systems, in Proc.Amer.Control Conf., Albuquerque, June 4-6:3311-3315,1997.
- [27] H.N.Nounou and K.M.Passin.Stable auto-tuning of adaptive fuzzy/neural controllers for nonlinear discrete-time systems, IEEE Trans.Fuzzy Systems, 12(1):70-83,2004.
- [28] R.Ordonez, J.T.Spooner and K.M.Passin. Experimental studies in nonlinear discrete-time adaptive prediction and control, IEEE Trans.Fuzzy Systems, 14(2):275-286,2006.
- [29] R.Ordonez.Direct adaptive regulation of discrete time nonlinear systems with arbitrary nonlinearities by backstepping, in Proc.the 41st CDC, Las Vegas Nevada USA:4288-4293,2002.
- [30] F.C.Chen and Khalil.Adaptive control of a class of nonlinear discrete-time systems using neural networks, IEEE Trans.Automat.Contr., 40(5):791-801,1995.
- [31] G.P.Liu,V.Kadirkamanathan and S.Billings.Predictive Control for Nonlinear Systems Using Neural Networks, Int J Control,71(6):1119-1132,1998.

MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLÔME  
DE MASTER EN GENIE ELECTRONIQUE

SPECIALITE: AUTOMATIQUE

Proposé et dirigé par : Mr. Abdelouaheb BOUKHALFA

Présenté par : LEFREID Kamel

**Thème :**

**Commande adaptative floue pour une classe de  
systèmes non linéaires à temps discret.**

**Résumé :**

Un schéma de commande adaptative indirecte floue est développé pour une classe de systèmes non linéaires discrets. Dans ce schéma, les systèmes flous sont utilisés pour approximer les dynamiques des systèmes, et les paramètres inconnus sont ajustés par une zone morte variable dans le temps dont sa taille est ajustée d'une manière adaptative avec les bornes estimés sur les erreurs de l'approximation. Le schéma de conception proposé garantie que tous les signaux en boucle fermée sont bornés, et l'erreur de poursuite converge vers un petit voisinage de l'origine.

**Mots clés :** Systèmes Discrets Non Linéaire, Commande Adaptative Floue, Zone Morte Variable dans le Temps, Fonction de Lyapunov.

N° d'ordre : 012