

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

UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DE TECHNOLOGIE

DEPARTEMENT DE GENIE ELECTRIQUE

N° : Rob-05



DOMAINE : SCIENCES ET TECHNOLOGIE

FILIERE : AUTOMATIQUE

OPTION : ROBOTIQUE

Mémoire présenté pour l'obtention  
Du diplôme de Master Professionnel en Robotique

Par :

**CHABBI Aymen**

**BOULAARES Oussama**

Intitulé :

**Commande adaptative floue de poursuite pour une classe  
de systèmes non linéaires**

**Soutenu devant le jury composé de :**

Dr.	Université Mohamed Boudiaf - M'sila	Président
Dr. BOUKHALFA Abdelouaheb	Université Mohamed Boudiaf - M'sila	Rapporteur
Dr. KHETTAB Khatir	Université Mohamed Boudiaf - M'sila	Co-Rapporteur
Dr.	Université Mohamed Boudiaf - M'sila	Examineur

**Année universitaire : 2021/ 2022**

## *Remerciements*

*Nous remercions ALLAH qui nous aide et nous avoir donné la patience et le courage durant ces années d'étude. Nous tenons à remercier Mr. Boukhalfa .A et Mr. Khettab.k, pour la proposition du sujet de ce mémoire et pour ses conseils tout au long de ce travail. Nous remercions les membres de jury, qui m'ont fait l'honneur de participer au jugement de ce travail.*

## Dédicaces

*Nous dédions ce travail à :*

*Nos parents, et le dévouement dont ils font preuve. A nos familles et amis, et à tous ceux qui nous sont chers.*

*Oussama et Aymen.*

## Sommaire

<b>Introduction générale</b> .....	Erreur ! Signet non défini.
<b>Chapitre I : logique floue</b> .....	Erreur ! Signet non défini.
I.1.Introduction : .....	Erreur ! Signet non défini.
I.2.Ensemble flou .....	Erreur ! Signet non défini.
I.3.Commande floue .....	Erreur ! Signet non défini.
I.3.1.But de La commande floue .....	Erreur ! Signet non défini.
I.4.Fonction d'appartenance .....	Erreur ! Signet non défini.
I.4.1.Fonction triangulaire .....	Erreur ! Signet non défini.
I.4.2.Fonction trapézoïdale .....	Erreur ! Signet non défini.
I.4.3.Fonction Gaussienne .....	Erreur ! Signet non défini.
I.4.4.Fonction sigmoïdale .....	Erreur ! Signet non défini.
I.5.Variables linguistiques .....	Erreur ! Signet non défini.
I.6.Opérateurs flous.....	Erreur ! Signet non défini.
I.7.Raisonnement flou .....	Erreur ! Signet non défini.
I.8.Règles floues .....	Erreur ! Signet non défini.
I.9.Structure d'un système flou .....	Erreur ! Signet non défini.
I.9.1.Base de règles .....	Erreur ! Signet non défini.
I.9.2.Fuzzification .....	Erreur ! Signet non défini.
I.9.3.Défuzzification .....	Erreur ! Signet non défini.
I.9.4.Moteur d'inférence .....	Erreur ! Signet non défini.
I.10.Système flou de Takagi-Sugeno .....	Erreur ! Signet non défini.
I.11.Approximateurs universels .....	Erreur ! Signet non défini.
I.11.1.Théorème (1) .....	Erreur ! Signet non défini.
I.12.Conclusion .....	Erreur ! Signet non défini.
<b>Chapitre II : Commande Adaptative</b> .....	16
II .1. Introduction .....	16
II.2.Principe de la commande adaptative .....	Erreur ! Signet non défini.
II.3.Système de commande adaptative.....	Erreur ! Signet non défini.
II.3.1.Commande adaptative directe et indirecte .....	Erreur ! Signet non défini.
II.4.Commande adaptative basée sur la synthèse de Lyapunov.....	Erreur ! Signet non défini.

II.4.1. Méthodes d'analyse de la stabilité des systèmes non linéaires ..	<b>Erreur ! Signet non défini.</b>
II.4.2. Théorème de Lyapunov .....	<b>Erreur ! Signet non défini.</b>
II.5. Stabilité.....	<b>Erreur ! Signet non défini.</b>
II.6. Synthèse des algorithmes de commande adaptative à modèle de référence	<b>Erreur ! Signet non défini.</b>
<b>défini.</b>	
II.6.1. Notion de critère de performance .....	<b>Erreur ! Signet non défini.</b>
II.6.2. Principe d'adaptation des paramètres .....	<b>Erreur ! Signet non défini.</b>
II.7. La commande adaptative floue .....	<b>Erreur ! Signet non défini.</b>
II.7.1. Commande adaptative floue directe .....	<b>Erreur ! Signet non défini.</b>
II.7.2. Commande adaptative floue indirecte .....	<b>Erreur ! Signet non défini.</b>
II.8. Conclusion.....	<b>Erreur ! Signet non défini.</b>
<b>Chapitre III : Commande adaptative floue de systèmes non linéaires .....</b>	<b>Erreur !</b>
<b>Signet non défini.</b>	
III .1. Introduction.....	<b>Erreur ! Signet non défini.</b>
III .2. Description du système .....	<b>Erreur ! Signet non défini.</b>
III .2.1. Description du modèle d'entrées-sorties.....	<b>Erreur ! Signet non défini.</b>
III .2.2. Description du modèle dans l'espace d'état : .....	<b>Erreur ! Signet non défini.</b>
III .2.3. Représentation et modélisation floue-neuronale : .....	30
III .3. Conception du contrôleur adaptatif.....	<b>Erreur ! Signet non défini.</b>
III .4. Exemple de simulation.....	<b>Erreur ! Signet non défini.</b>
III .5. Conclusion .....	38
<b>Conclusion Générale .....</b>	<b>39</b>
<b>Bibliographie .....</b>	<b>Erreur ! Signet non défini.</b>



**Listes des figures :**

<b>Figure (I.1)</b> : Fonction triangulaire.....	5
<b>Figure (I.2)</b> : Fonction d'appartenance trapézoïdale.....	6
<b>Figure (I.3)</b> : Fonction d'appartenance trapézoïdale.....	6
<b>Figure (I.4)</b> : Fonction d'appartenance sigmoïdale.....	7
<b>Figure (I.5)</b> : Représentation floue de la variable Vitesse.....	8
<b>Figure (I.6)</b> : Exemples de fonctions d'appartenance.....	8
<b>Figure (I.7)</b> : Structure de base d'un système flou.....	11
<b>Figure (I.8)</b> : Représentation schématique d'un système flou de Takagi-Sugeno d'ordre 1.....	13
<b>Figure (II.1)</b> : Structure du système commande adaptative.....	17
<b>Figure (II.2)</b> : Commande adaptative directe.....	18
<b>Figure (II.3)</b> : Commande adaptative indirecte.....	18
<b>Figure (II.4)</b> : Commande adaptative basée sur la synthèse de Lyapunov.....	19
<b>Figure (II.5)</b> : La commande adaptative floue directe.....	23
<b>Figure (II.6)</b> : La commande adaptative floue indirecte.....	24
<b>Figure(III.1)</b> : Configuration des réseaux flou-neuronaux.....	32
<b>Figure (III.2)</b> : Sortie $y(k)$ et signal de référence $x_2(k)$ .....	37
<b>Figure (III.3)</b> : Entrée de commande.....	38
<b>Figure (III.4)</b> : Signal d'erreur $e(k)$ .....	38

## Introduction générale

Au cours des deux dernières décennies, un grand nombre de publications a été dédié au problème de commande des systèmes non linéaires. Un problème qui présente beaucoup de challenges puisque les systèmes non linéaires, contrairement aux systèmes linéaires pour lesquels l'automatique fournit une panoplie de méthodes pour la synthèse de la commande, ne disposent pas d'outils et de méthodes généraux pour l'analyse et la synthèse des systèmes et des lois de commande non linéaires. Ceci est dû au fait que les systèmes non linéaires possèdent des structures extrêmement variées, des dynamiques complexes et peuvent présenter toutes sortes de comportement étrange.

Au début, la commande des systèmes non linéaires était classique [1], elle reposait sur la théorie de l'automatique linéaire. En fait, elle consiste à linéariser la dynamique non linéaire du système (approximation de premier ordre) autour d'un point de fonctionnement de telle sorte que les outils de la commande linéaire puissent être exploités pour la synthèse d'une loi de commande assurant les performances recherchées. Grâce à sa facilité d'implantation et de mise en œuvre, cette approche a remporté un grand succès auprès des industriels. Cependant, la nécessité d'atteindre de hautes performances dans des domaines de fonctionnement relativement larges, a imposé la prise en compte de la dynamique globale non linéaire des processus dans la synthèse de la commande.

La première application de la logique floue en commande des processus est due à E. H. Mamdani et A. Assilian en 1974 [2]. Depuis, cette technique a connu un développement croissant, qui s'est manifestait par la publication de nombreux ouvrages et papiers (voir, par exemple, [3,5]), et par la mise en place de groupes de recherches spécialisés en commande floue sur tous les continents. La technique de commande floue est souvent présentée comme une technique robuste capable de compenser quasi naturellement les incertitudes et/ou le manque de connaissances précises quant au processus à commander. Dans la majorité des applications des contrôleurs flous, ce dernier est construit à partir de connaissances linguistiques fournies par un expert. Cependant, il est parfois difficile de construire la base des règles de commande pour certains systèmes, ou le besoin peut imposer l'ajustement des paramètres du contrôleur, dans le cas de systèmes à dynamiques variables par exemple. Pour surmonter ce problème et permettre aux contrôleurs flous une acquisition automatique de connaissances à partir de données numériques, les chercheurs ont introduit les techniques de commande adaptatives [4,5]. Dans ces techniques, la propriété

d'approximation universelle des systèmes flous est exploitée [4, 6], et la stabilité et la robustesse de la structure de commande sont étudiées par l'approche de Lyapunov.

Les techniques de contrôle traditionnelles (méthodes algorithmiques) consistent d'abord à construire des modèles mathématiques basés sur des équations différentielles ou en boucle du système à contrôler. A partir de ce modèle, une commande est déterminée pour amener ce système dans l'état souhaité tout en respectant des critères de performance. En effet, obtenir un modèle mathématique précis, pertinent et simple d'utilisation est parfois difficile et complexe. Plusieurs commandes semblent pallier ce problème, on cite la commande adaptative floue.

De nombreux travaux utilisent la commande adaptative floue pour les systèmes non linéaires. Généralement, l'objectif principal de la commande adaptative est de maintenir les performances du système. Par conséquent, la commande avant floue doit être adaptative.

Le but de ce travail est de développer des structures de contrôle adaptatif flou stables pour les systèmes non linéaires.

L'ensemble des travaux réalisés est regroupé en 3 chapitres :

**Le chapitre(I)**, nous avons défini la forme de la logique floue et le raisonnement associé, nous avons introduit la structure générale d'un contrôleur flou, et les différents modèles utilisés dans sa conception, ainsi que le théorème d'approximation floue.

**Le chapitre(II)**, nous introduisons la commande adaptative avec différentes structures directes et indirectes qui seront utilisées pour commander une classe de systèmes non linéaires.

**Le chapitre(III)**, nous développons une approche adaptative floue pour commander une classe des systèmes non linéaires avec l'analyse de la stabilité par la synthèse de Lyapunov.

# **Chapitre I :**

# **Logique floue**

# Chapitre I : Logique floue

## I.1.Introduction :

La logique floue à été introduite par le professeur Lotfi Zadeh en 1965 comme une généralisation de la logique binaire [7,8]. 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 approcher le raisonnement humain à l'aide d'une représentation adéquate des connaissances in à l'aide d'une représentation adéquate des connaissances. Ces dernières années, la logique floue a été appliquée dans divers domaines. Cependant, la commande flou est le domaine qui suscite le plus de curiosité et s'avère être un champ d'application actif à travers le monde [8,9,10]. En fait, l'imitation des aspects qualitatifs et approximatifs du raisonnement humain a donné naissance à un autre type de système de commande qui trouve de puissantes applications, surtout dans les situations où le système à commander est mal défini mathématiquement et/ou en présence d'une expertise qui puisse fournir des règles de commande.

Dans ce chapitre, nous exposons un bref rappel sur la théorie des ensembles flous et un aperçu général sur le système flou, en insistant sur les idées utilisées en commande flou. Ensuite, nous présentons la structure des systèmes flous et en particulier les systèmes flous de Takagi-Sugeno.

## I.2.Ensemble flou :

La notion d'ensemble flou provient du constat que « très souvent, les classes d'objets rencontrés dans le monde physique ne possèdent pas de critères d'appartenance bien définis » [9]. Mathématiquement, un ensemble floue A est défini sur un univers de discours X, par une fonction d'appartenance  $\mu_A(x)$  qui prend ces valeur dans l'intervalle [0,1]. Cette fonction donne le degré d'appartenance de chaque élément x appartient à A .

$$A=[(x,\mu_A(x)) | x \in X] \quad (I.1)$$

D'autres représentations sont possibles. Par exemple, si X est discret, A est représenté par:

$$A=\sum_{x \in X} \mu_A(x)/x \quad (I.2)$$

et si X est continu, A est représenté comme suit:

$$A = \int_x \mu_A(x)/x \quad (I.3)$$

Par exemple, si quelqu'un veut définir un ensemble flou A par les nombres naturels proches de 6. Cet ensemble peut être défini comme suit :

$$A = 0,1/3 + 0,3/4 + 0,6/5 + 1,0/6 + 0,6/7 + 0,3/8 + 0,1/9$$

et si quelqu'un veut définir l'ensemble A par les nombres réels proche de 6. Ceci peut être accompli par le choix de la fonction d'appartenance suivante :

$$\mu_A(x) = \frac{1}{1+(x-6)^2} \quad (I.4)$$

donc A est donné par :

$$A = \int_R \mu_A(x)/x \quad (I.5)$$

### I.3. 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 [8].

#### 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.4. Fonction d'appartenance :

Au lieu d'appartenir à l'ensemble « vrai » ou à l'ensemble « faux » de la logique binaire traditionnelle, la logique floue admet des degrés d'appartenance à un ensemble donné. Le degré d'appartenance à un ensemble flou est matérialisé par un nombre compris entre 0 et 1. Une valeur précise de la fonction d'appartenance liée à une valeur de la variable est notée  $\mu$  et appelée « facteur

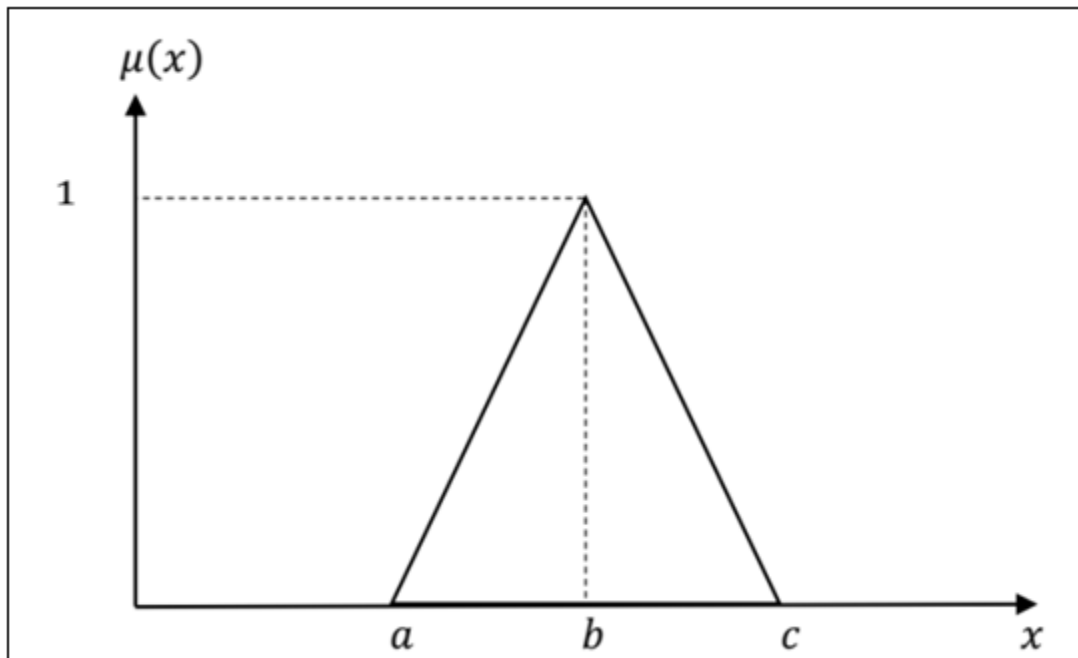
d'appartenance ». Pour des raisons de simplicité, nous introduisons les fonctions d'appartenances ci-dessous [2,7,12,18].

#### I.4.1 Fonction triangulaire :

En peut définir une fonction d'appartenance de type triangulaire avec trois paramètres

{a, b, c} :

$$u(x) = \max\left(\min\left(\frac{x - a}{b - a}, 1, \frac{c - x}{c - b}\right), 0\right) \quad (I.6)$$

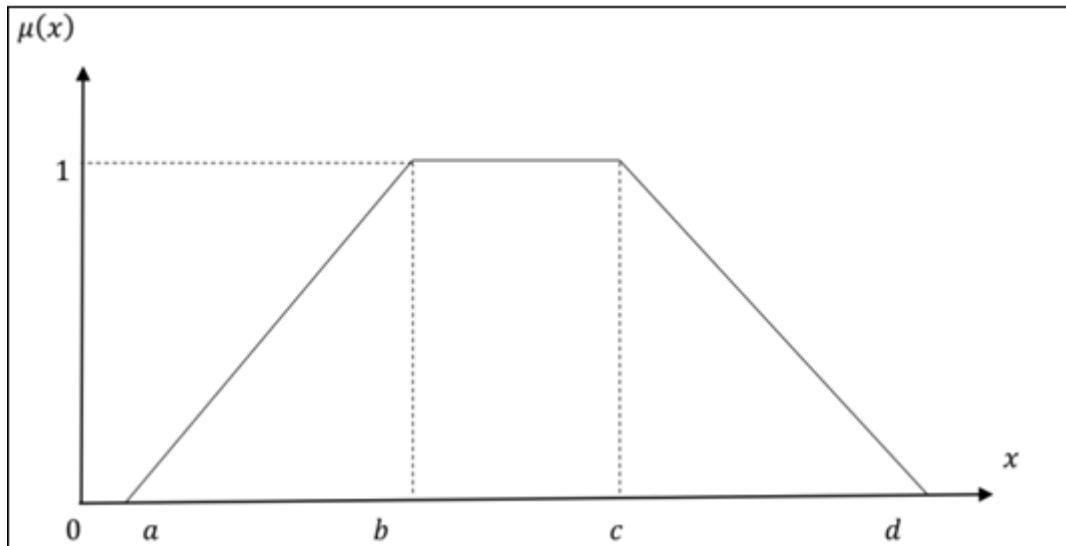


**Figure (I.9) : Fonction triangulaire**

#### I.4.2 Fonction trapézoïdale :

Elle est définie par une fonction d'appartenance avec quatre paramètres {a, b, c, d} :

$$u(x) = \max\left(\min\left(\frac{x - a}{b - a}, 1, \frac{d - x}{d - c}\right), 0\right) \quad (I.7)$$

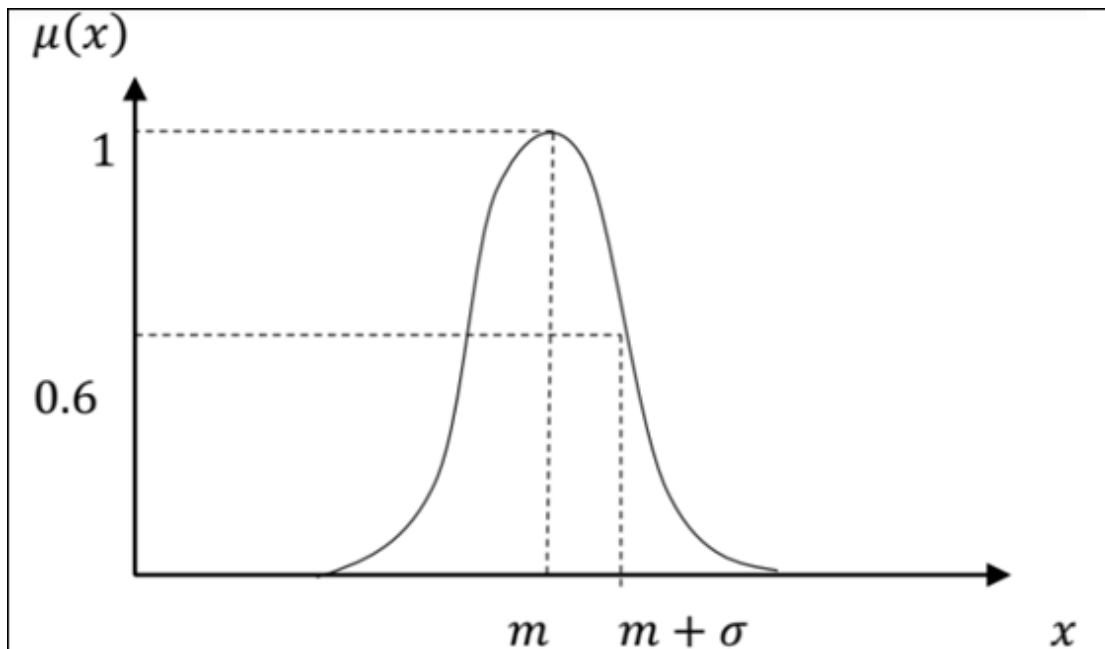


**Figure (I.10) :** Fonction d'appartenance trapézoïdale.

#### I.4.3 Fonction Gaussienne :

Elle est définie par deux paramètres  $\{\sigma, m\}$  :

$$u(x) = \exp - \left( \frac{(x-m)^2}{2\sigma^2} \right) \quad (I.8)$$



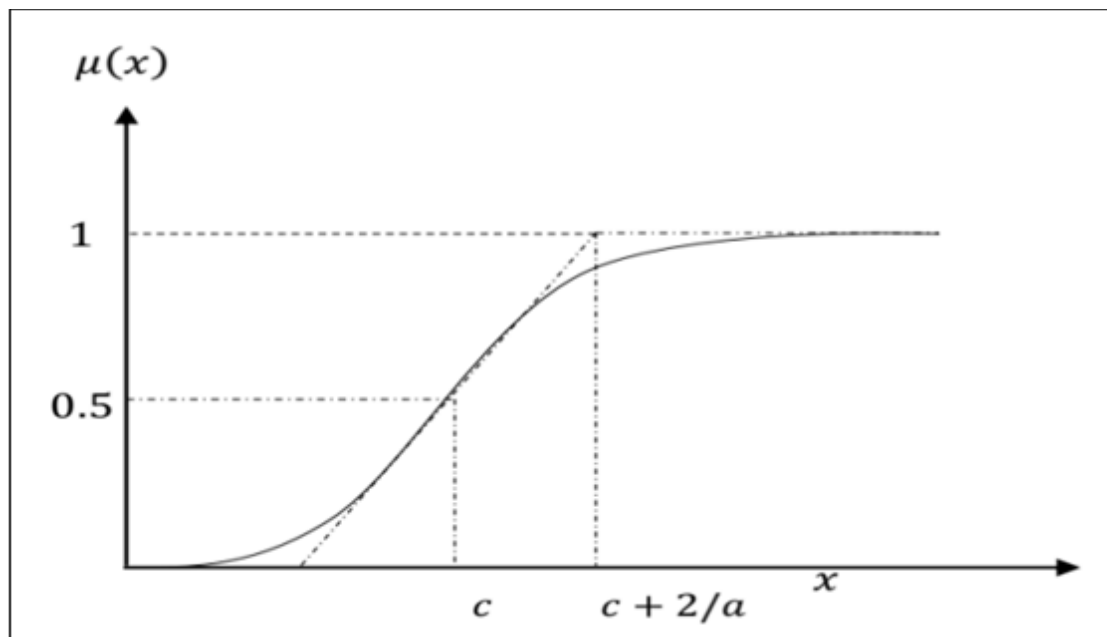
**Figure (I.11) :** Fonction d'appartenance trapézoïdale.

Notons que  $m$  est le centre,  $\sigma$  est la variance.

#### I.4.4 Fonction sigmoïdale :

Une fonction sigmoïdale est définie par deux paramètres {a, c} :

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (I.9)$$



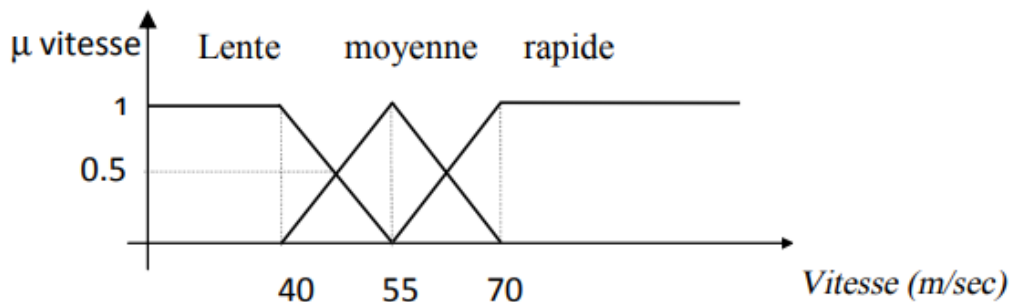
**Figure (I.12) :** Fonction d'appartenance sigmoïdale.

#### I.5. Variables linguistiques :

La notion de variables linguistiques permet de modéliser des connaissances imprécises ou Fluctue sur une variable dont la valeur exacte est inconnue. Variables linguistiques, ou Variable floue, donc une variable dont la valeur appartient à l'ensemble flou Peut représenter des mots du langage naturel. Donc une variable floue peut prendre plusieurs valeurs linguistiques en même temps, 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:  $x =$  Vitesse est une variable linguistique, son ensemble de valeurs peut être :  $T(\text{Vitesse}) = [\text{Faible}, \text{Moyenne}, \text{Élevée}, \dots]$  où

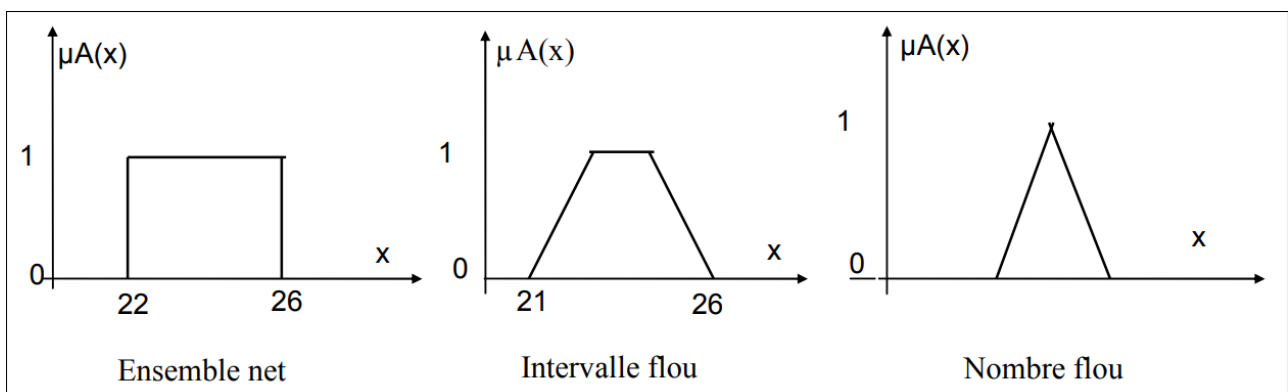
chaque terme dans  $T(\text{Vitesse})$  est caractérisé par un ensemble flou dans un univers de discours  $U = [0, 100]$ , [13].



**Figure (I.13) :** 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 floue.

Le plus souvent, on utilise pour les fonctions d'appartenance de forme trapézoïdales ou triangulaires, rectangulaires ou de type singleton. Il s'agit des formes les plus simples (Figure (I.14)).



**Figure (I.15) :** Exemples de fonctions d'appartenance.

### I.6. Opérateurs flous :

Les variables linguistiques sont liées entre elles au niveau des inférences par des opérateurs **Et** ou **Ou**. Il s'agit d'opérateurs de la logique floue qui interviennent sur les fonctions d'appartenance représentant les variables linguistiques.

Les opérateurs les plus importants sont : **l'intersection, l'union, et le complément.**

Nous allons généraliser ces fonctions fondamentales de la théorie des ensembles. On devra retrouver dans le cas classique les fonctions habituelles. Les solutions proposées devront répondre à certaines propriétés (croissance, associativité,...) spécifiques aux fonctions considérées [13].

### I.7. Raisonement flou :

On rappelle qu'en logique classique le modus ponens permet, à partir de la règle « Si x est A alors y est B » et du fait « x est A », de conclure le fait « y est B », qui sera ajouté à la base des faits. Zadeh a étendu ce principe au cas flou, principe que l'on appelle alors modus ponens généralisé [8, 10]. Le modus ponens et le modus ponens généralisé se résument comme suit :

	Modus Ponens	Modus Ponens généralisé
Fait	x est A	x est A'
Règle	Si x est A Alors y est B	Si x est A Alors y est B
Déduction	y est B	y est B'

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

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

$$\mu_{B'}(y) = \text{SUP}_x(\mu_{A'}(x) * \mu_R(x, y)) \quad (\text{I.10})$$

### I.8. Règles floues :

Une règle floue est une déclaration de la forme suivante :

**SI x est A ALORS y est B**

où x et y sont des variables linguistiques, et A et B sont des valeurs linguistiques, déterminées par les ensembles flous sur les ensembles X et Y. Une variable linguistique est une variable floue. Par exemple : La tension est haute. La variable linguistique tension prend la valeur linguistique élevée. La plage de valeurs linguistiques possibles d'une règle représente l'univers de cette variable. Un exemple de règle floue est :

**SI vitesse est lente ALORS arrêt est court**

La variable vitesse peut avoir une plage de valeurs entre 0 et 220 km/h. On peut inclure des sous-ensembles flous (très lent, lent, moyenne, rapide, très rapide) pour modifier cette règle. Chaque sous-ensemble flou représente une valeur linguistique pour la variable

La logique classique (SI – ALORS) utilise la logique binaire. La logique floue permet d'associer une plage de valeurs (un ensemble flou) à des variables linguistiques. On peut réduire le nombre de règles jusqu'à 90% en utilisant la logique floue.

### **I.9. Structure d'un système flou :**

Un système flou peut être interprété selon deux points de vue : mathématique ou logique. D'un point de vue mathématique, un système flou est une fonction non linéaire reliant un vecteur de données d'entrée à un vecteur de sortie et, de point de vue logique, un système flou est un système à base de connaissance particulière composé de quatre modules principaux, à savoir : la base de règles, la fuzzification, le moteur d'inférence et la défuzzification (Figure 7) [2,3,8,9,14,15]. :

#### **I.9.1 Base de règles :**

La base de règles floues, ou base de connaissances, contient les règles floues décrivant le comportement du système; elle est le cœur du système entier dans le sens où tous les autres composants sont utilisés pour interpréter et combiner ces règles pour former le système final. Ces règles peuvent être fournies par un expert ou peuvent être extraites de données numériques. Dans les deux cas, les règles prennent la forme « Si prémisse Alors conclusion ». D'une manière générale, la base de règles d'un système flou doit respecter les conditions de complétude et de consistance afin d'assurer le bon fonctionnement de ce dernier. Une base de règles d'un système flou est dite complète si, pour chaque vecteur d'entrée, il existe au moins une règle floue activée. Afin d'assurer cette propriété, les fonctions d'appartenance doivent couvrir tout l'espace des variables d'entrée. Une base de règles d'un système flou est dite inconsistante, s'il existe deux règles floues ayant la même prémisse mais des conclusions différentes. La propriété de consistance permet d'éviter les contradictions dans une base de règles.

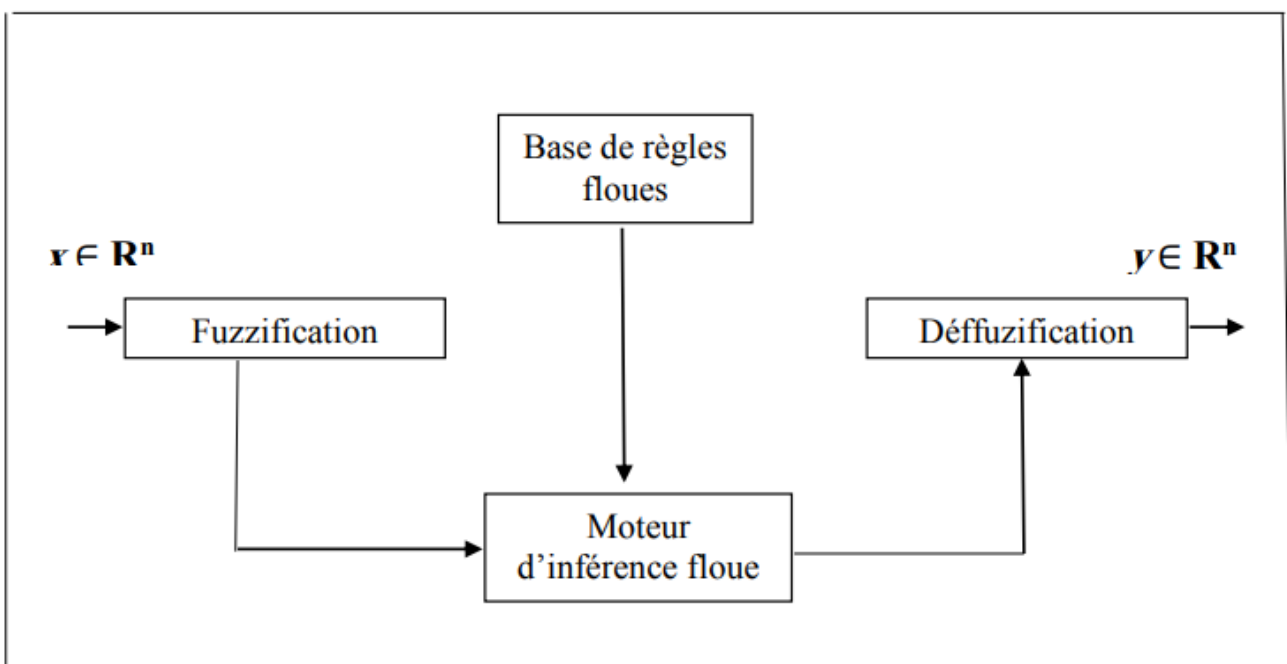
#### **I.9.2 Fuzzification :**

La fuzzification est la première étape dans la réalisation d'un système flou. Elle transforme chaque valeur réelle d'entrée (mesure) en un ensemble flou. En lui attribuant sa fonction

d'appartenance à chacune des classes préalablement définies, deux approches de fuzzification sont généralement utilisées, à savoir : la fuzzification singleton et la fuzzification non singleton.

### I.9.3 Défuzzification :

La défuzzification transforme la grandeur floue issue de l'inférence en une grandeur physique. Cependant, il n'existe pas une procédure systématique pour choisir la stratégie de défuzzification. Un critère de choix des méthodes de défuzzification en commande floue est la simplicité des calculs. Ce critère a conduit à l'utilisation, le plus souvent, de la méthode de défuzzification dite la moyenne des centres « center-average ».



**Figure (I.16) :** Structure de base d'un système flou.

### I.9.4.Moteur d'inférence :

Le moteur d'inférence floue transforme, à l'aide des techniques de raisonnement flou, la partie floue issue de la fuzzification en une nouvelle partie floue. En fait, le moteur d'inférence combine les règles floues, en utilisant les principes de la logique floue, pour effectuer une transformation à partir des ensembles flous dans l'espace d'entrée vers des ensembles flous dans l'espace de sortie

### I.10.Système flou de Takagi-Sugeno :

Dans la section précédente, nous avons proposé des systèmes flous avec des conclusions Le système de symboles est appelé le système flou standard, également connu sous le nom de

Mamdani. Ces systèmes utilisent une conclusion symbolique de même nature que les prémisses, et, En fait, les connaissances exprimées sous la forme d'une analyse qui décrit la structure interne du Le système ne peut pas être considéré directement. Kanno et ses collaborateurs [16] ont proposé une classe de systèmes flous permettant de représenter ce type de Information. De tels systèmes flous sont appelés systèmes flous de conclusion. Système flou de Takagi-Sugeno (TS)

Notons par  $x=[x_1 \dots ,x_n ]^T$  les entrées du système flou, et par  $y$  sa sortie. Pour chaque  $x_i$  est associé  $m_i$  ensembles flous  $F_i^j$  dans  $X_i$ , tel que pour  $x_i \in X_i$ , il existe au moins un degré d'appartenance  $\mu_{F_i^j}(x_i) \neq 0$  où  $i=1, 2 \dots, n$  et  $j=1,2,\dots,m_i$ . La base de règles du système flou comporte  $N= \prod$  règles floues de la forme

$R_k$  :si  $x_1$  est  $f_1^k$  et...et  $x_n$  est  $f_n^k$  alors  $y=f(x)(k=1,\dots,N)$

Où  $f_1^k \in [f_1^1, \dots, f_1^{m_1}]$  et  $f_k(x)$  est une fonction numérique dans l'espace de sortie. En général,  $f_k(x)$  est une fonction polynomiale en fonction des variables d'entrées, mais elle peut être aussi une fonction arbitraire tant qu'elle puisse décrire convenablement le comportement du système étudié. Si  $f_k(x)$  est une fonction linéaire, c'est-à-dire :

$$F_k(x) = a^k_0 + \sum_{i=1}^n a^k_i x_i \tag{I.11}$$

alors on a affaire à un système flou de Takagi-Sugeno d'ordre un (TS1). Si par contre,  $f_k(x)$  est un polynôme d'ordre zéro, i.e.

$$f_k(x) = a^k \tag{I.12}$$

on a donc un système flou de Takagi-Sugeno d'ordre zéro (TS0).

Etant donné que chaque règle possède une conclusion numérique, la sortie totale du système flou est obtenue par le calcul d'une moyenne pondérée, et de cette manière, le temps consommé par la procédure de défuzzification est évité. En fait, la sortie du système flou est donnée par la relation suivante [3,8,11,16] :

$$y(x) = \frac{\sum_{k=1}^N \mu(x) f(x)}{\sum_{k=1}^N \mu(x)} \tag{I.13}$$

avec  $\prod_{i=1}^n \mu_f(x_i)$ ,  $f_i^k \in [f_i^1, \dots, f_i^{m_i}]$

et qui représente le degré de confiance ou d'activation de la règle  $R_k$ .

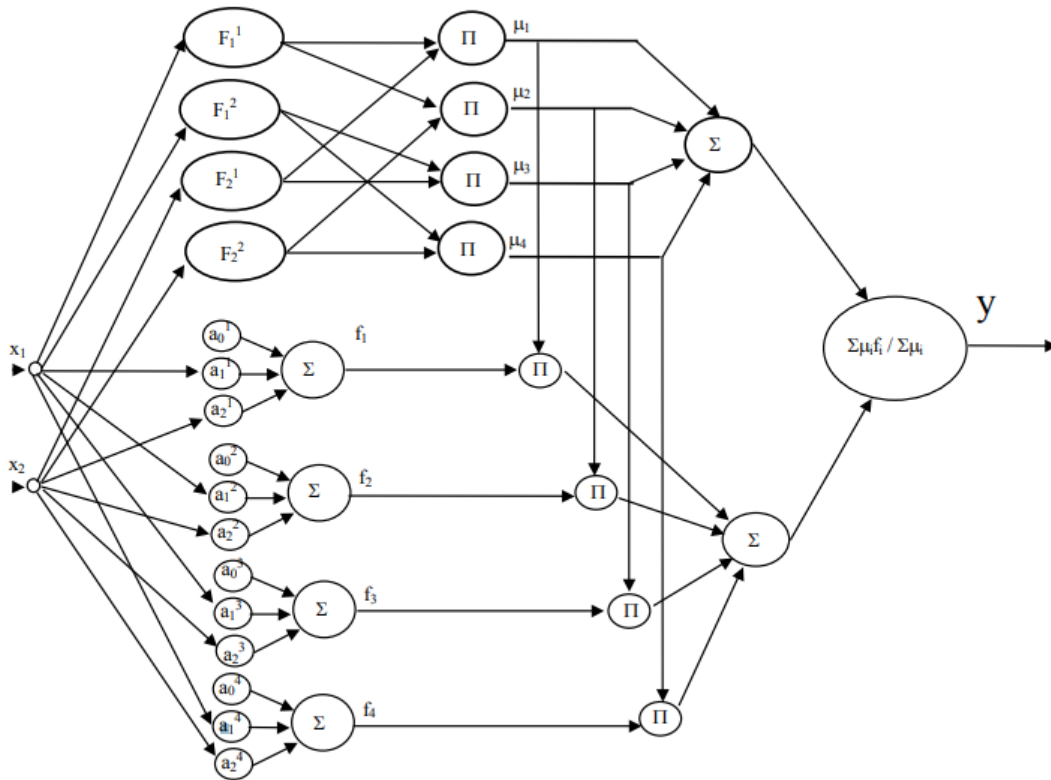
Dans le cas d'un système flou TS1, la sortie est donnée par

$$Y(x) = \frac{\sum_{k=1}^N \mu_k(x) [a_0^k + a_1^k x_1 + \dots + a_n^k x_n]}{\sum_{k=1}^N \mu_k(x)} \quad (\text{I.14})$$

et dans le cas d'un système flou TS0, la sortie se simplifie à

$$y(x) = \frac{\sum_{k=1}^N \mu_k(x) a^k}{\sum_{k=1}^N \mu_k(x)} \quad (\text{I.15})$$

montre une représentation schématique, sous forme d'un réseau, d'un système flou TS1 à deux entrées et deux ensembles flous pour chaque variable.



**Figure (I.17) :** Représentation schématique d'un système flou de Takagi-Sugeno d'ordre 1 [8].

Dans ce travail, on se limite à l'utilisation des systèmes flous à ceux qui seront directement exploités dans les approches de commande développées, à savoir les systèmes flous de Takagi-Sugeno à conclusion constante, i.e. les systèmes flous de Takagi-Sugeno d'ordre zéro. Il est à noter que le système flou TS0 (15) est équivalent au système flou de Mamdani utilisant une fuzzification singleton et la méthode de défuzzification du barycentre. En introduisant la notion de fonctions floues de base [8], la sortie du système flou de TS0 (15) peut être écrite sous la forme

$$y(x) = w^t(x) \theta \quad (\text{I.17})$$

$\theta = [a^1 \dots a^N]^T$  vecteur constitué des paramètres de la partie conclusion des règles floues :

$w(x) = [w_1(x) \dots w_N(x)]^T$  vecteur des fonctions floues de base, dont chacune des composantes est donnée par :

$$w_k(x) = \frac{\mu_k(x)}{\sum_{j=1}^N \mu_j(x)}, k=1, \dots, N \quad (\text{I.18})$$

Les fonctions d'appartenance qui caractérisent les ensembles flous  $f_i^j$  sont choisies, dans cette thèse, des fonctions Gaussiennes définies par :

$$\mu_{f_i^j}(x_i) = \exp\left[-\frac{1}{2} \left[\frac{x_i - c_i^j}{v_i^j}\right]^2\right] \quad (\text{I.19})$$

où  $c_i^j$  est la moyenne et  $v_i^j$  la variance. Notons que l'utilisation des autres formes pour les fonctions d'appartenance est possible.

### I.11. Approximateurs universels :

Les systèmes non linéaires sont très complexes et difficilement modélisables. Plusieurs techniques utilisant le principe de la boîte noire ont été développées dans les travaux antérieurs pour la modélisation de ces systèmes. Une des techniques consiste à utiliser des modèles linéaires de type auto-régressif (AR) ou auto-régressif avec entrée exogène (ARX). Cependant, ce type de structure ne permet pas d'obtenir un modèle optimal. Les réseaux de neurones, étant non linéaires par nature, ont également été utilisés pour la modélisation des processus non linéaires. Leur principal avantage réside dans leur capacité de modélisation sans aucune hypothèse préalable. Néanmoins, les variables utilisées dans ces réseaux n'ont aucune signification physique. Il est donc intéressant de garder la structure multi-couches et d'utiliser le raisonnement flou pour approximer des processus inconnus ou incertains. Plusieurs travaux ont été développés dans ce sens [19,20] et l'énoncé suivant du théorème de l'approximateur universel a été l'un de ces aboutissements :

**I.11.1 Théorème (1) :** Pour toute fonction réelle  $f(x)$  continue sur un compact  $U \subset \mathfrak{R}$  donné, il existe un système flou  $\hat{f}(x)$  tel que :

$$\sup_{x \in U} |g(x) - \hat{f}(x)| < \varepsilon \quad (\text{I.20})$$

Avec  $\varepsilon \geq 0$  est un constant arbitraire.

Les propriétés d'approximation générales ne sont pas données pour la construction  $f(x)$  qui permet seulement de garantir l'existence de ce dernier.

Nous avons choisi la structure du système flou d'une façon appropriée afin d'arriver à la notion d'approximation universelle toute en se basant sur le choix des entrées et le nombre de fonctions d'appartenance avec le type approprié pour chaque entrée du système flou, les paramètres qui constituent les fonctions d'appartenance et le nombre de règles [18,19].

### **I.12.Conclusion :**

Dans ce chapitre, nous avons présentés les bases nécessaires à la compréhension des méthodes basées sur la logique floue qui les différencient de celles issues de l'automatisation classique. Les contrôleurs basés sur la logique floue ne nécessitent pas de modèle analytique de la tâche à résoudre. En effet, elles ne reposent pas sur les équations physiques du système à contrôler, mais sur l'expertise d'opérateurs et d'experts du domaine.

## **Chapitre II :**

# **Commande Adaptative**

# Chapitre II : Commande Adaptative

## II .1. Introduction :

La commande des machines asynchrones doit prendre en compte des spécificités propres à ces machines : l'ordre du modèle, les non-linéarités de fonctionnement et le couplage entre les différentes grandeurs de commande. De plus les paramètres de la machine variaient dans le temps, soit avec la température (résistances), soit avec l'état magnétique de la machine (inductances) sans compter que la charge peut être variable. En effet l'utilisation d'un régulateur robuste à paramètres fixes ne suffit plus pour garantir les performances requises dans l'ensemble des régimes de fonctionnement possibles. Pour cela la commande adaptative trouve un grand essor pour améliorer la robustesse de la commande de la machine asynchrone [2].

Dans le présent chapitre, nous détaillons l'aspect théorique et le principe de la commande adaptative, notamment la commande adaptative à modèle de référence qui est l'objet de notre travail. Nous introduisons également la synthèse d'adaptation des paramètres qui est considérée comme l'étape la plus délicate de cette commande.

## II.2. Principe de la commande adaptative :

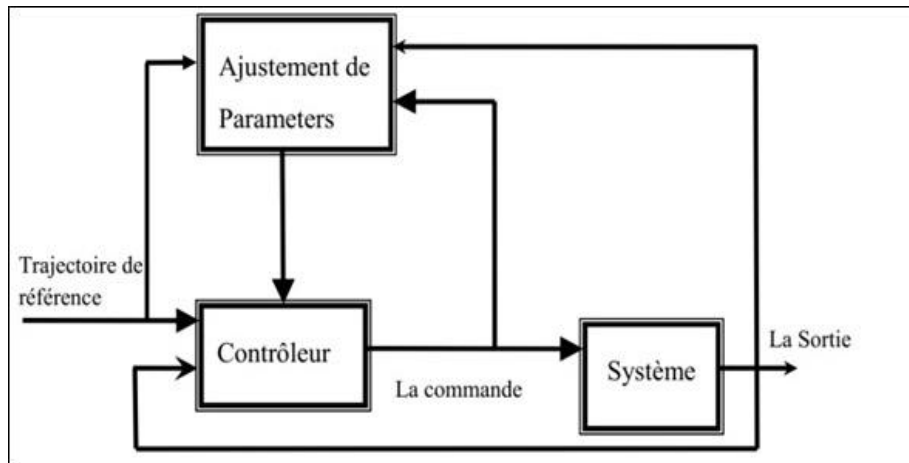
La Commande Adaptative est définie comme une commande permettant au régulateur de s'adapter de lui-même aux changements du processus. En conséquence, l'objectif de la Commande Adaptative est l'ajustement automatique en ligne et en temps réel des régulateurs, des boucles de commandes afin de réaliser ou maintenir un certain niveau de performances, quand les paramètres du procédé à commander varient dans le temps [20].

Un régulateur adaptatif peut également s'envisager comme une structure à deux boucles, une boucle principale classique qui prend en compte les variations des signaux d'entrée et de sortie, et une boucle secondaire qui réagit aux variations des paramètres du processus.

## II.3 Système de commande adaptative:

Un système de commande adaptative mesure un certain indice de performance du système de commande, et à partir de l'écart entre l'indice de performance désiré et l'indice de performance mesuré, le mécanisme d'adaptation modifie les paramètres du système ajustable. On génère des signaux de commande auxiliaire, afin de maintenir l'indice de performance du système dans le voisinage des valeurs désirées [8].

On note que les systèmes de commande adaptative, même s'il s'agit de la commande d'un système linéaire à paramètre inconnus, sont des systèmes non linéaires car les paramètres du régulateur dépendent des variables ou des procédés à travers les mécanismes d'adaptation. Ceci explique les difficultés d'analyse et de synthèse de tels systèmes.

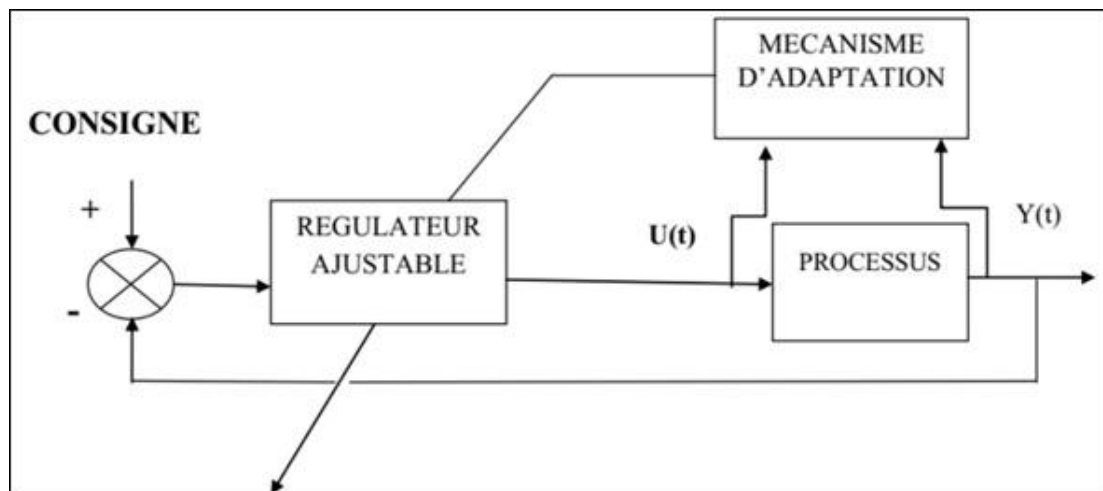


**Figure (II.1) : structure du système commande adaptative [47].**

### II.3.1 Commande adaptative directe et indirecte :

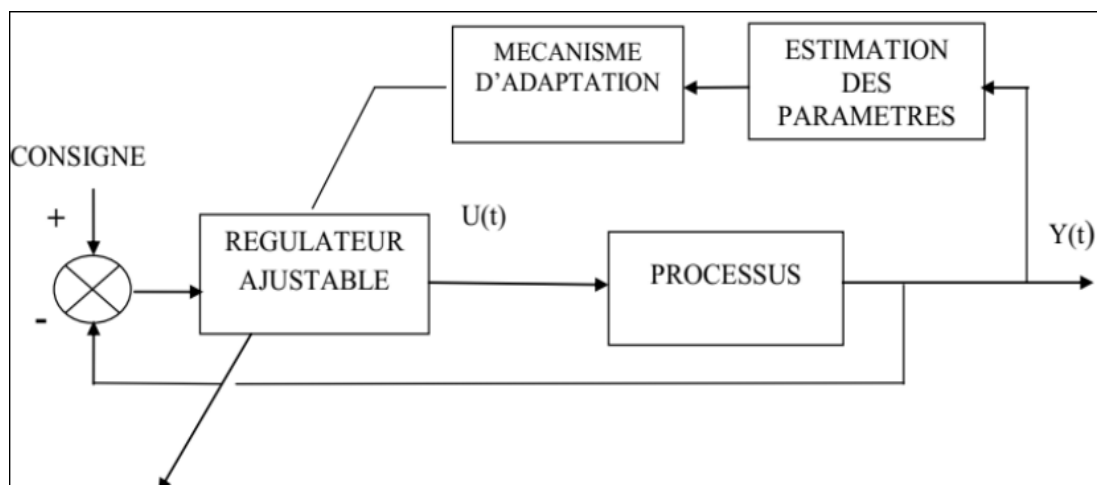
Deux approches principales existent pour la commande adaptative (des processus à paramètres inconnues ou variable dans le temps) :

a). **La commande adaptative directe** : la quelle les paramètres du régulateur sont ajustés directement et en temps réel à partir d'une comparaison entre les performances réelles et performance désirée (c'est le cas en particulier de la commande adaptative a modèle de référence (Model Reference Adaptive Control "MRAC") [21].



**Figure (II.2) : Commande adaptative directe**[47].

**b). La commande adaptative indirecte :** qui suppose une estimation des paramètres du processus par une procédure d'identification (c'est le cas des régulateurs auto-ajustables (self-tuning control "STC")) [21]:

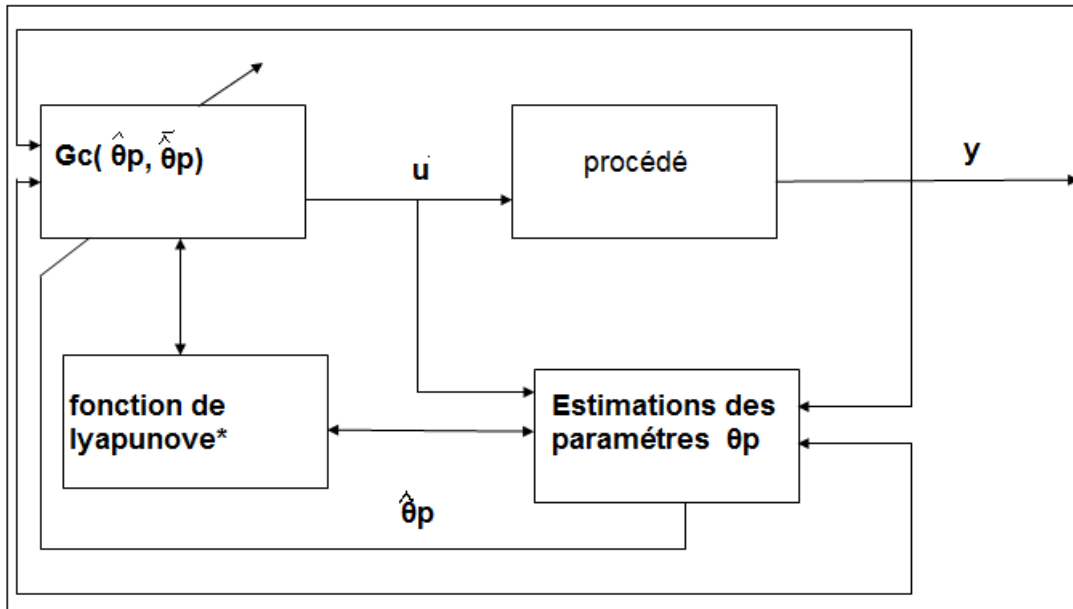


**Figure (II.3) : Commande adaptative indirecte** [47].

#### II.4. Commande adaptative basée sur la synthèse de Lyapunov :

L'idée principale de l'utilisation de la fonction de Lyapunov dans la commande adaptative, consiste à calculer une loi de commande et une loi de mise à jour des paramètres, afin de garantir que la dérivée d'une certaine fonction, définie positive et bien choisie, est non positive. L'approche consiste donc à trouver un triplé (fonction de Lyapunov, loi de commande, loi d'adaptation) (Figure. II.4) qui répond aux spécifications. Les tâches s'effectuent simultanément, et la dynamique de

l'estimation est prise en compte et/ou maîtrisée, afin d'éviter tout effet destructeur. C'est grâce à cette propriété que ce type de méthodes présentes un intérêt pour la commande des systèmes non linéaires.



**Figure (II.4) : Commande adaptative basée sur Lyapunov.**

### II.4.1. Méthodes d'analyse de la stabilité des systèmes non linéaires :

L'étude de la stabilité des systèmes non linéaires est très complexe. L'approche de Lyapunov est l'approche la plus utilisée pour étudier ce problème. Cette approche a été introduite au 19ième siècle par le mathématicien russe Alexandre Mikhailovich Lyapunov dans son travail intitulé, « The general problem of the motion stability ». On distingue deux méthodes de Lyapunov pour l'analyse de la stabilité : La méthode de linéarisation et la méthode directe. La première concerne la stabilité locale autour d'un point d'équilibre. Par contre, la deuxième détermine la stabilité des systèmes en construisant une fonction scalaire des états du système dont on examinera la variation temporelle. Par la suite on s'intéressera par la méthode directe de Lyapunov [22].

### II.4.2. Théorème de Lyapunov :

La méthode directe de Lyapunov (ou la méthode des fonctions de Lyapunov) est dérivée du critère énergétique de stabilité. En appliquant ce critère indépendamment du concept d'énergie, on remplace alors l'énergie du système par une fonction de Lyapunov qui est définie positive (comme l'énergie).

Soit le système autonome :

$$\dot{x} = f(x) \quad (\text{II. 1})$$

Ce système aura un point d'équilibre  $x_e = 0$ , globalement asymptotiquement stable, s'il existe une fonction scalaire  $V(x)$  continue avec une dérivée partielle par rapport au temps  $\dot{V}(x)$  continue ayant les propriétés suivantes :

1.  $V(0) = 0$
2.  $V(x) > 0, \forall x \neq 0$  (radialement non linéaire)
3.  $\lim_{x \rightarrow \infty} V(x) = \infty$
4.  $\dot{V} < 0, \forall x \neq 0$

### Principe :

Il s'agit, pour un système scalaire, d'utiliser la fonction de Lyapunov pour calculer une loi de commande et une dynamique d'adaptation qui assurent la stabilité, en présence d'incertitude décrit par l'équation [21] :

$$\dot{x} = u + \varphi(x)^T \theta \quad (\text{II. 2})$$

est un vecteur de paramètres constants et inconnus. On désire trouver la loi de commande  $u(x, \theta)$  qui assure la stabilité de l'origine du système (II.2). Pour ce faire on prend comme fonction de Lyapunov (fcl), la forme quadratique

$$V_1(x) = \frac{1}{2} x^2 \quad (\text{II. 3})$$

Qui est définie positive. Sa dérivée, évaluée le long de la solution de (II.02), s'écrit

$$\dot{V}_1 = x\dot{x} = x[u + \varphi(x)^T \theta]$$

Le choix de la loi de commande

$$u(x, \theta) = \varphi(x)^T \theta - k_1 x \quad (\text{II. 4})$$

où  $K_1 > 0$ : permet de rendre négative. En effet, avec un tel choix, on a

$$\dot{V}_1 = -K_1 x^2 \leq 0$$

Deux cas se présentent alors :

- $\theta$  est connu : la loi de commande (II.4) peut être réalisée, ce qui permet de stabiliser l'origine du système. L'objectif du design est ainsi atteint.

- $\theta$  est inconnu : le contrôleur décrit par (II.4) ne peut être réalisé. On se propose de le remplacer par son équivalent (basé sur le principe de l'équivalence certaine) où au lieu de  $\theta$ , on utilise son estimé  $\hat{\theta}$  dans (II.5)

$$u = -\varphi(x)^T \tilde{\theta} - k_1 \quad (II.5)$$

Non remplaçons (II.5) dans (II.2), on obtient

$$\dot{x} = -k_1 x + \varphi(x)^T (\theta - \hat{\theta}) = -k_1 x + \varphi(x)^T \tilde{\theta}$$

Où  $\tilde{\theta}$  représente l'erreur d'estimation  $(\theta - \hat{\theta})$ , La dérivée de la fonction donnée par (II.3) s'écrit avec ces choix

$$\dot{V}_1 = x[-k_1 x + \varphi(x)^T \tilde{\theta}] = -K_1 x^2 + x\varphi(x)^T \tilde{\theta}$$

Étant donné que l'expression contient le terme inconnu  $\tilde{\theta}$  son signe est indéfini. Et aucune conclusion ne peut être tirée quant à la stabilité du système. Afin d'en savoir plus sur cette stabilité, on construit un contrôleur dynamique, en augmentant (II.5) d'une loi de mise à jour pour l'estimé de  $\tilde{\theta}$ . Un bon choix de cette loi, qui doit assurer la stabilité de l'ensemble, est à déterminer. On définit alors une nouvelle fonction Lyapunov en ajoutant, à la fonction initiale donnée par (II.3), un terme quadratique en l'erreur d'estimation  $\tilde{\theta}$

$$V_2(x, \tilde{\theta}) = \frac{1}{2} x^2 + \frac{1}{2} \tilde{\theta}^T \Gamma^{-1} \tilde{\theta}$$

$\Gamma$  est une matrice (définie positive) qui représente le gain d'adaptation. La dérivée de cette fonction devient alors

$$\dot{V}_2 = x\dot{x} + \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} = -K_1 x^2 + x\varphi(x)^T \tilde{\theta} + \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} = -K_1 x^2 + \tilde{\theta}^T \Gamma^{-1} [\theta + \tau]$$

Ou  $\tau = \Gamma x \varphi(x)$

La dérivée reste toujours indéfinie, mais cette fois, le degré de liberté, offert par le libre choix de la dynamique de mise à jour, permet de choisir cette dernière de façon à annuler le second terme de l'équation, on choisit :

$$\dot{\tilde{\theta}} = \hat{\dot{\theta}} = \tau \quad (II.06)$$

On obtient

$$\dot{V}_2 = -K_1 x^2 \leq 0$$

Le système résultant est donné par (II. 2) avec la loi de commande (II.5), et la loi de mise à jour (II.6).

## II.5. 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 [25].

## II.6. Synthèse des algorithmes de commande adaptative à modèle de référence:

### II.6.1. Notion de critère de performance :

La sortie  $y_m(t)$  du modèle de référence est comparée avec le signal de sortie du système réel. L'écart d'asservissement  $e(t) = y_m(t) - y(t)$  est un signal d'entrée pour le mécanisme d'ajustement qui doit modifier les paramètres du régulateur de telle sorte que l'erreur s'annule. L'étape la plus délicate dans la synthèse des systèmes de CAMR est l'élaboration de l'algorithme du mécanisme d'ajustement destiné à réduire ou à annuler l'erreur  $(t)$ . Cet algorithme est introduit en fonction de critère de performance utilisé.

Le choix du modèle de référence s'avère très important pour la conception de la commande adaptative. Ainsi la particularité des systèmes de commande CAMR réside dans l'optimisation de l'ajustement du système de la boucle interne par rapport au modèle de référence. La qualité de commande automatique est déterminée par le critère de l'auto ajustement qui présente une certaine fonction  $J$  qui dépend de l'erreur  $e(t)$  définie précédemment :

$$J(e) = J[y_m(t) - y(t)] \quad (\text{II.07})$$

L'utilisation de critère de performance  $J(e)$  permet de déterminer les algorithmes d'adaptation des paramètres du contrôleur. Le but de ces algorithmes est de minimiser l'erreur entre la valeur désirée  $y_m(t)$  et la variable instantanée  $y(t)$ , c.-à-d  $\lim_{t \rightarrow \infty} e(t)$ .

### II.6.2. Principe d'adaptation des paramètres [23] :

Afin de synthétiser une commande adaptative à modèle de référence, il est nécessaire d'adopter certaines hypothèses:

- ✓ Le modèle de référence doit être linéaire invariant dans le temps
- ✓ Le système à commander et le modèle de référence doivent être de même ordre.
- ✓ En cas d'adaptation paramétrique, tous les paramètres du système à commander doivent être accessibles pour l'adaptation.

- ✓ Durant le processus d'adaptation, les paramètres du système à commander ajustable dépendent uniquement de mécanisme d'adaptation (condition de non stationnarité)
- ✓ Aucun vecteur autre que celui du vecteur d'entrée, n'agit sur le système.
- ✓ La différence initiale entre les paramètres du modèle et ceux du système est connue.
- ✓ Le vecteur d'état et sortie sont mesurable Cependant ces hypothèses constituent le cas idéal et elles ne sont pas toujours satisfaites dans des situations réelles.

Dans les situations réelles on a les conditions suivantes :

- ✓ Le système ajustable est un système non linéaire variable dans le temps.
- ✓ Le modèle de référence et le système à commander n'ont pas la même dimension.
- ✓ Durant le processus d'adaptation, les paramètres du système ajustable ne dépendent pas seulement du mécanisme d'adaptation mais ils sont aussi soumis à d'autres perturbations paramétriques.

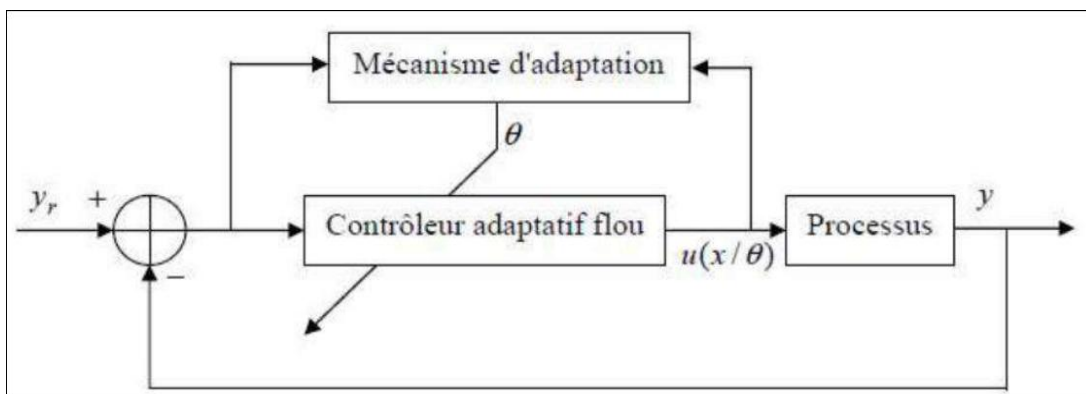
### II.7. La commande adaptative floue :

La terminologie « commande adaptative » désigne l'ensemble de méthodes permettant l'ajustement automatique en temps réel des paramètres des régulateur 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.

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 [24].

#### II.7.1. Commande adaptative floue directe :

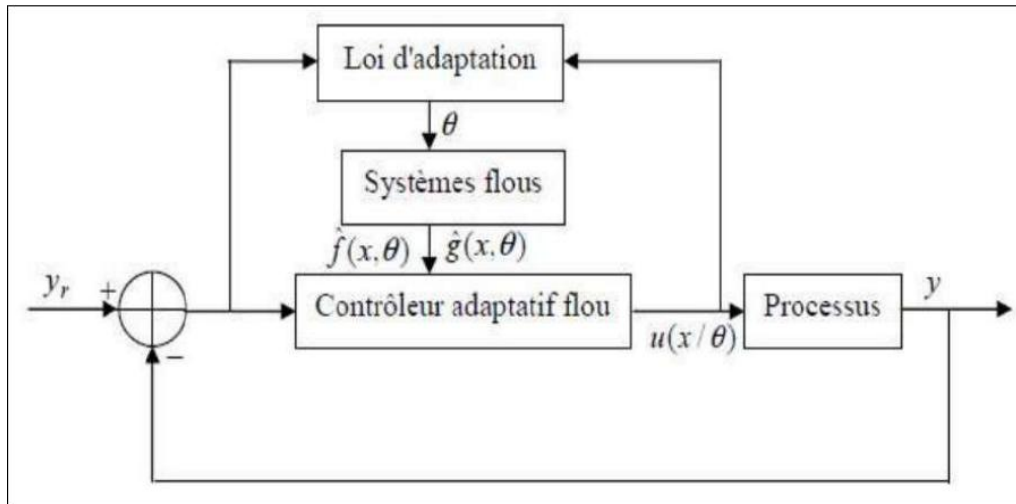
La loi de commande est directement approximée par un ou plusieurs systèmes adaptatifs flous. La structure de cette commande est montrée sur la figure (II.7)



**Figure (II.5) :** La commande adaptative floue directe.

### II.7.2. Commande adaptative floue indirecte :

Dans cette méthode on approxime d'abord le modèle du processus par des systèmes adaptatifs flous puis on synthétise la loi de commande à partir du modèle approximé. La figure (II.8) montre la structure de la commande adaptative floue indirecte.



**Figure (II.6) :** La commande adaptative floue indirecte.

### II.8. Conclusion :

Dans de nombreux cas, les systèmes de commande classiques sont basés sur des modèles à paramètres fixes car ils ne peuvent pas s'adapter aux changements lents ou rapides de la dynamique du système, de sorte que le concept de systèmes de contrôle doit être introduit. Contrôle adaptatif pour garantir les performances souhaitées. À cette fin, les paramètres du régulateur (contrôleur) sont automatiquement ajustés en fonction des informations recueillies à partir du système. Dans ce chapitre, nous avons introduit quelques concepts et définitions de la commande adaptative. Nous avons commencé par définir la commande adaptative, puis avons donné un aperçu de son histoire, ainsi que différents schémas de commande adaptative.

## **Chapitre III :**

# **Commande adaptative floue de systèmes non linéaires**

# Chapitre III : Commande adaptative floue de systèmes non linéaires

## III .1.Introduction :

Les réseaux de neurones (NNs) ou les systèmes flous ont été particulièrement utiles pour contrôler des systèmes non linéaires avec des incertitudes paramétrées de manière non linéaire, en raison de leur approximation universelle. non linéaires en raison de leur propriété d'approximation universelle d'approximation universelle [2,26]. Ces dernières années, des recherches actives ont été menées dans le domaine de la commande des systèmes NN ou à logique floue (FLS), et de nombreux travaux remarquables pour les systèmes non linéaires complexes ont été réalisés systèmes non linéaires complexes ont été réalisés dans ce domaine grâce aux progrès de la théorie géométrique de la commande non linéaire, en particulier les techniques de linéarisation par rétroaction et la conception de la stabilité de Lyapunov. et la conception de la stabilité de Lyapunov.

La commande adaptative basée sur ces approximateurs universels a suscité beaucoup d'attention, et la commande a été largement étudiée pour les systèmes à temps continu et les systèmes à temps discret. Pour les systèmes non-linéaires incertains à temps continu, de nombreux de contrôle ont été obtenus l'un après l'autre sur la base de la théorie de la stabilité de Lyapunov [27, 28,29,30,31,32,33,34]. La stabilité des systèmes en boucle fermée peut être garantie. En comparaison avec de contrôle à temps continu, le contrôle adaptatif utilisant NN ou FLS est plus difficile à concevoir dans les systèmes non linéaires à temps discret, en raison du manque d'informations sur la stabilité. En raison du manque d'outils mathématiques efficaces. Ainsi, un problème facile en temps continu peut devenir difficile en temps discret. C'est peut-être l'une des Cela peut être l'une des raisons pour lesquelles la commande NN ou FLS pour les non linéaires en temps discret a été moins étudiée que celle des systèmes non linéaires en temps continu.

Récemment, sur la base de l'approximateur universel, contrôle adaptatif pour les systèmes non linéaires incertains à temps discret a obtenu de nombreux résultats importants [23,35, 36, 37, 38, 39, 40]. Plus précisément, le problème de stabilité de la moyenne mobile autorégressive non linéaire avec entrée exogène (NARMAX), qui a été proposé pour la première fois dans [41], a suscité un intérêt accru. Par exemple, une commande adaptative de NN a été étudiée dans [42] pour une classe de systèmes non linéaires inconnus à temps discret et à entrée unique (SISO). Les systèmes dans

[42] ont été transformés en une description causale de l'espace d'état, et une commande adaptative de d'état adaptatif a été présentée sur la base de la théorie de stabilité de Lyapunov. En outre, un observateur NN adaptatif et une commande adaptative basée sur l'observateur de l'observateur ont également été présentés sous l'hypothèse d'une excitation persistante. Par la suite, une commande adaptative multicouche NN multicouche adaptative a été développée dans [43] pour une classe de systèmes non linéaires non affins à temps discrets dans le modèle NARMAX en temps discret. En utilisant le théorème de la fonction implicite, l'existence de la commande de rétroaction implicite souhaitée est démontrée. L'existence de la commande implicite à rétroaction souhaitée est prouvée. Les réseaux neuronaux multicouches sont utilisés comme émulateur de la commande de rétroaction souhaitée. Dans des commandes adaptatives par réseaux de neurones à rétroaction de sortie ont été étudiées pour deux classes de systèmes non linéaires à temps discret avec des directions de commande inconnues : (1) les systèmes non linéaires (1) les systèmes non linéaires à rétroaction pure et (2) les systèmes NARMAX [44]. Un NN d'ordre élevé est utilisé pour approximer la commande inconnue. Pour surmonter le problème non causal, les deux systèmes sont transformés en un prédicteur pour la conception de la commande de rétroaction de sortie .pour la conception de la commande de rétroaction de sortie. Le théorème de la fonction implicite est utilisé pour surmonter la difficulté de l'apparence non affine de l'entrée de contrôle .l'apparence non affine de l'entrée de commande. Le système en boucle fermée atteint la stabilité SGUUB, et l'erreur de suivi de la sortie se situe dans un voisinage de zéro. et l'erreur de suivi de la sortie se situe dans un voisinage de zéro.

Cependant, la plupart des schémas pour le modèle NARMAX doivent ajuster les paramètres adaptatifs pour l'estimation du paramètre de poids idéal [42,43,44]. Ainsi, la charge de calcul en ligne sera très lourde. Récemment, de nombreux chercheurs ont consacré des efforts importants pour réduire la charge de calcul en ligne. en ligne; voir, par exemple, et les références qui s'y rapportent [30,31,45]. Les schémas présentés dans se concentrent sur les systèmes continus non linéaires incertains. Le site est difficile de contrôler les systèmes non linéaires à temps discret en raison du problème non causal de la conception. Récemment, pour les systèmes non linéaires à temps discret, un algorithme de commande algorithme de contrôle neuronal adaptatif avec moins de paramètres adaptatifs moins de paramètres adaptatifs a été étudié sur la base du retour d'état [39] et de la sortie rétroaction [33]. Cependant, le modèle NARMAX est différent des systèmes considérés . Lorsque le modèle NARMAX est considéré, les schémas de [38,49] sont restés largement inexplorés.

Sur la base des présentations ci-dessus, ce document tente d'aborder un schéma de commande floue-neuronale adaptative pour une classe de systèmes non linéaires incertains à temps discret sous la forme la forme NARMAX. Un nouveau contrôleur adaptatif est conçu sur la base de l'approximation d'un réseau flou-neuronal. L'estimation de la norme du paramètre d'approximation paramètre d'approximation idéal est ajustée, comparé l'estimation de la norme du paramètre d'approximation idéal est ajustée, par rapport à d'autres méthodes de contrôle pour contrôler le modèle NARMAX, le schéma proposé dans ce travail réduira considérablement la charge de calcul en ligne. Tous les signaux du système en boucle fermée sont prouvés comme étant bornés, et il est garanti que l'erreur de suivi converge vers un ensemble compact borné par une sélection appropriée des paramètres de conception. L'efficacité de l'approche proposée approche proposée est vérifiée à l'aide d'un exemple de simulation exemple de simulation.

### III .2. Description du système :

#### III .2.1. Description du modèle d'entrées-sorties

Considérons le système SISO non linéaire à temps discret sous la forme NARMAX

$$Y(k+1) = f_0(y_k, \dots, y_{k-n+1}, u_{k-\tau}, \dots, u_{k-\tau-m+1}) + g_0(y_k, \dots, y_{k-n+1}, u_{k-\tau}, \dots, u_{k-\tau-m+1}) \times u_{k-\tau+1} + d_k \quad (\text{III.1})$$

où  $y \in \mathbb{R}$  est la sortie mesurée,  $u \in \mathbb{R}$  est l'entrée,  $d_k$  désigne la perturbation externe bornée par une constante connue  $d_0 > 0$ , c'est-à-dire que  $|d_k| \leq d_0$ ,  $m \leq n$ ,  $f_0(*)$  et  $g_0(*)$  sont des fonctions non linéaires inconnues, et  $\tau$  est le retard du système qui satisfait à  $1 \leq \tau \leq n$  ( $\tau$  est également appelé le degré relatif du système).

#### Remarque 1 :

Le degré relatif  $\tau$  pour les systèmes à temps discret a été bien défini et expliqué dans [37]. très différent de son homologue en temps continu. Pour un système de degré relatif un la procédure de conception du contrôleur est causale et simple, car l'entrée de commande  $u_k$  apparaît explicitement sur la droite de droite de (III.1). Cependant, pour les systèmes de degré relatif supérieur à un, nous ne pouvons pas obtenir la commande  $u_k$  directement à partir de (III.1) car  $u_k$  n'apparaît pas dans l'équation explicitement

Il est facile d'obtenir

$$y_{k+\tau} = f_0(y_{k+\tau-1}, \dots, y_{k+\tau-n}, u_{k-1}, \dots, u_{k-m}) + g_0(y_{k+\tau-1}, \dots, y_{k+\tau-n}, u_{k-1}, \dots, u_{k-m}) \times u_k + d_{k+\tau-1} \quad (\text{III.2})$$

#### Remarque 2 :

Il est évident que  $u_k$  apparaît de manière explicite. Mais nous avons encore des difficultés à concevoir  $u_k$  à cause de l'existence de futures sorties  $y_{k+1}, \dots, y_{k+\tau-1}$ , qui ne sont pas disponibles pour la rétroaction. En examinant attentivement (III.1) et (III.2), on peut voir que l'entrée de commande  $u_k$  n'affecte que la production future  $y_{k+\tau}$  et celles qui suivent, et n'a aucune influence sur les sorties futures intermédiaires  $y_{k+1}, \dots, y_{k+\tau-1}$ . Lorsque la perturbation externe  $d_{k+\tau-1}$  est ignorée, il n'y aura pas de problème non causal pour prédire les futures sorties du côté droit de (III.2) si  $f_0$  et  $g_0$  sont connus.

**Définition (1):**

La sortie future d'un système de contrôle à temps discret est dite sortie future semi-déterminée (SDFO) à l'instant  $k$ , si elle peut être déterminée sur la base des informations disponibles sur le système jusqu'à l'instant  $k$  et des commandes jusqu'à l'instant  $k - 1$ , en supposant que la dynamique de l'installation et la perturbation sont connues.

Par exemple,  $y_{k+1}$  est un SDFO car il peut être prédit à partir de (III.2) si  $f_0$ ,  $g_0$  et  $d_k$  sont connus. En fait,  $y_{k+1}, \dots, y_{k+\tau-1}$  sont tous des SDFO.

**III .2.2.Description du modèle d'espace d'état :**

Pour faciliter l'analyse, définissons les états du système :

$x(k) = [x_1(k), \dots, x_n(k)]^T$  comme  $x_i(k) = y(k - n - 1 + \tau + i)$ ,  $i = 1, \dots, n$  et réécrire (III.2) dans l'espace d'états description

$$\begin{cases} x_i(k + 1) = x_{i+1}(k), & i = 1, \dots, n - 1 \\ x_n(k + 1) = f(x(k), v_{k-1}(k) + g(x(k), v_{k-1}(k))u_k + d_{k+\tau-1} \\ y_k = x_{n-\tau+1}(k) \end{cases} \quad (III.3)$$

où  $x(k) \in \mathbb{R}^n$  et  $v_{k-1}(k) = [u_{k-1}, \dots, u_{k-m}]^T \in \mathbb{R}^m$ ,  $f(x(k), v_{k-1}(k)) = f_0(y_{k+\tau-1}, \dots, y_{k+\tau-n}, u_{k-1}, \dots, u_{k-m})$  et  $g(x(k), v_{k-1}(k)) = g_0(y_{k+\tau-1}, \dots, y_{k+\tau-n}, u_{k-1}, \dots, u_{k-m})$ .

Pour plus de clarté, définissons  $f(k) = f(x(k), v_{k-1}(k))$  et  $g(k) = g(x(k), v_{k-1}(k))$  qui sont des fonctions des états  $x(k)$  et de toutes les entrées de commande passées de  $u_{k-1}$  jusqu'à l'entrée la plus retardée  $u_{k-m}$  à droite.  $u_{k-1}$  jusqu'à l'entrée  $u_{k-m}$  la plus retardée du côté droit de (III.2)

**Hypothèse 1 :**

[44, 49] Le signe de  $g(k)$  est connu, et il existe deux constantes  $g_0, g_1 > 0$  telles que

$g_0 \leq |g(k)| \leq g_1 \quad \forall x(k) \in \Omega \subset \mathbb{R}^n$  et  $\forall u_k \in \Omega_u \subset \mathbb{R}$ . Sans perdre la généralité, nous allons supposer que  $g(k)$  est positif dans la discussion qui suit.

L'objectif de la commande est de conduire la sortie du système  $y_k$  à suivre une trajectoire désirée  $y_d(k) \in \Omega_y \subset \mathbb{R}$ . Définir le vecteur  $x_d(k)$  comme les états désirés du système  $x_d(k) = [y_d(k + \tau - n), \dots, y_d(k), \dots, y_d(k + \tau - 1)]^T$

**Hypothèse 2 :**

$y_d(k)$  est lisse et connu, et  $x_d \in \Omega_d$  avec  $\Omega_d$  étant un petit sous-ensemble connexe de  $\Omega$ .

Définir le vecteur d'erreur  $e(k)$  comme  $e(k) = x(k) - x_d(k) = [e_1(k), \dots, e_n(k)]^T$  [37,41]. Nous avons alors :

$$\begin{cases} e_i(k + 1) = e_{i+1}(k), & i = 1, \dots, n - 1 \\ e_n(k + 1) = f(k) + g(k)u_k + d_{k+\tau-1} - y_d(k + \tau) \end{cases} \quad (III.4)$$

Définissez l'erreur de suivi à un pas en boucle ouverte  $e_f(k)$  et l'erreur de vitesse à un pas en boucle ouverte  $v_e(k)$  comme suit :

$$e_f(k) = e_n(k + 1)|_{u_k=0} \quad (III.5)$$

$$v_e(k) = e_f(k) - e_n(k) \quad (III.6)$$

**Hypothèse 3 :**

$(1/g_0)|e_f(k)| \in \Omega_f \subset \Omega_u, (1/g_0)|v_e(k)| \in \Omega_v \subset \Omega_u, \forall x(k) \in \Omega \subset \mathbb{R}^n$  et  $\forall u_k \in \Omega_u \subset \mathbb{R}$ .

L'hypothèse 3 est raisonnable pour de nombreux systèmes physiques parce que les états d'un système physique ne peuvent pas changer trop rapidement dans un petit intervalle de temps dans un environnement ouvert en raison de l'inertie des systèmes[37,41].

Puisque les hypothèses 2 et 3 ne sont valables que sur les ensembles compacts  $\Omega$  et  $\Omega_u$ .

Ensembles compacts  $\Omega$  et  $\Omega_u$ , il est nécessaire de garantir que les états du système et le signal de commande restent dans  $\Omega$  et  $\Omega_u, \forall k \geq 0$  respectivement. Nous allons concevoir un commande adaptative  $u_k$  pour le système (III.3) qui fait que la sortie du système  $y_k$  suive la trajectoire désirée  $y_d(k)$  et qui garantit simultanément  $x(k) \in \Omega, u_k \in \Omega_u, \forall k > 0,$  pour  $x(0) \in \Omega$

**Définition(2) :**

La solution de (III.4) est semi-globalement uniformément bornée, si pour tout  $\Omega$ , un sous-ensemble compact de  $\mathbb{R}^n$  et tous les  $x(k_0) = x_0 \in \Omega$ , il existe un  $\varepsilon > 0$  et un nombre  $N(\varepsilon, x_0)$  tel que  $e(k) < \varepsilon$  pour tout  $k \geq k_0 + N$ . Considérons le système linéaire en temps discret variant dans le temps.

$$x(k+1) = A(k)x(k) + B_u(k), \quad y_k = Cx(k), \quad (\text{III.7})$$

où  $A(k)$ ,  $B$  et  $C$  sont des matrices de dimension appropriée,  $B$  et  $C$  étant des matrices constantes.

**Lemme 1 :**

Soit  $\Phi(k_1, k_0)$  la matrice de transition d'état correspondant à  $A(k)$  pour le système (III.7) [37].

L'état correspondant à  $A(k)$  pour le système (III.7), c'est-à-dire,

$$\Phi(k_1, k_0) = \prod_{k=k_0}^{k_1-1} A(k) \quad \text{Si } \Phi(k_1, k_0) < 1, \forall k_1 > k_0 \geq 0, \text{ le système (III.7) est (III.1)}$$

globalement exponentiellement stable. Pour le système non forcé (c'est-à-dire,  $u(k) = 0$ ), et (III.2) stable à entrée limitée et sortie bornée (BIBO).

**III.2.3.Représentation et modélisation floue-neuronale :**

Dans cette sous-section, afin de concevoir une loi de contrôle adaptative, nous introduisons des réseaux neuronaux flous pour approximer les fonctions inconnues non linéaires. La configuration de base des systèmes à logique floue (SLF) consiste en une logique floue et des réseaux neuronaux. La logique floue peut être divisée en quelques règles IF-THEN floues et un moteur d'inférence flou. À partir de un vecteur linguistique d'entrée  $z$ ,  $z = [z_1, \dots, z_n]^T \in \mathbb{R}^n$  à une variable linguistique de sortie, le moteur d'inférence floue peut effectuer une mise en correspondance  $S \in \mathbb{R}$  en utilisant les règles IF-THEN floues. La  $i^{\text{ème}}$  règle IF-THEN floue peut être écrite comme :

$$R^i: \text{Si } z_1(t) \text{ est } A_1^i \text{ et } \dots \text{ et } z_n(t) \text{ est } A_n^i, \text{ alors } S(z) \text{ est } B^i \quad (\text{III.8})$$

où  $A_1^i, A_2^i, \dots, A_n^i$  et  $B^i$  sont des ensembles flous [10, 16]. Soit  $m$  le nombre de règles IFTHEN floues. Par En utilisant l'inférence de produit, une moyenne centrale et un fuzzifier singleton, la sortie du système de logique floue peut s'exprimer comme suit :

$$s(z) = \frac{\sum_{i=1}^m y^{-i} (\prod_{j=1}^n \mu_{A_j^i}(z_j))}{\sum_{i=1}^m (\prod_{j=1}^n \mu_{A_j^i}(z_j))} = \theta^T \psi(z) \quad (III.9)$$

où  $\mu_{A_j^i}(z_j)$  est la valeur de la fonction d'appartenance de la variable floue,  $y^{-i}$  est le point auquel  $\mu_{B^i}(y^{-i})=1$ ,  $\theta^T = [y^{-1}, y^{-2}, \dots, y^{-m}]$  est le vecteur de paramètres réglable, et  $\psi^T = (\psi^1, \psi^2, \dots, \psi^m)$  est le vecteur de base vecteur de base flou, où  $\psi_i$  est défini comme suit :

$$\psi_i = \frac{(\prod_{j=1}^n \mu_{A_j^i}(z_j))}{\sum_{i=1}^m (\prod_{j=1}^n \mu_{A_j^i}(z_j))} \quad (III.10)$$

Lorsque les entrées sont entrées dans le réseau flou-neural illustré à la figure (Fig. III.1), la valeur de vérité  $\psi^i$  (couche III) de la partie antécédente de la  $i^{\text{ème}}$  implication est calculée par (III.10). Parmi les stratégies de défuzzification couramment utilisées, les sorties (couche IV) du système flou-neuronale sont exprimées par (III.9). Un approximateur de logique floue basé sur des réseaux neuronaux peut être établi. L'approximateur comporte quatre couches. Dans la couche I, nœuds, qui sont des nœuds d'entrée, représentent le vecteur linguistique d'entrée  $z = [z_1, z_2, \dots, z_n]^T$ . Dans la couche II, les nœuds représentent les valeurs de la fonction d'appartenance de variables linguistiques totales. Chaque nœud de la couche II représente une valeur de fonction d'appartenance. Dans la couche III, les nœuds représentent les valeurs du vecteur de base flou  $\psi$ . Chaque nœud de la couche III représente une règle floue. Les liens entre couche III et la couche IV sont entièrement connectés par le vecteur de pondération,  $\theta^T = [y^{-1}, y^{-2}, \dots, y^{-m}]$ , c'est-à-dire les paramètres ajustés. Au niveau de la couche IV, les sorties représentent les valeurs de la sortie  $S(z)$ . Pour une fonction désirée  $u^*(z)$ , il existe des poids idéaux  $W^*$  tels que la fonction lisse  $u^*$  puisse être approximée par un réseau flou-neuronale idéal sur un ensemble compact  $\Omega_z \subset \mathbb{R}^q$

$$u^* = W^{*T} S(z) + u_z \quad (III.11)$$

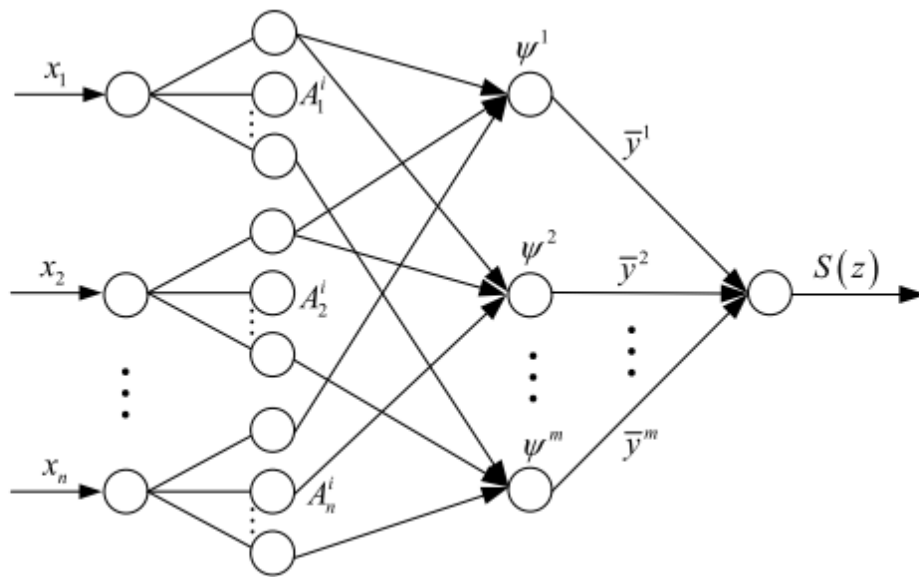
où  $u_z$  est l'erreur d'approximation bornée des réseaux flous-neuronales satisfaisant  $|u_z| \leq u_0$  sur l'ensemble compact, qui peut être réduite en augmentant le nombre de poids ajustables. La matrice de poids idéale  $W^*$  est une quantité "artificielle" nécessaire à des fins analytiques, et est définie comme minimisant  $u_z$  pour tous les éléments suivants.

$z \in \Omega_z \subset \mathbb{R}^q$  dans une région compacte, c'est-à-dire,

$$W^* = \arg \min \{ \sup |u^*(z) - W^T S(z)| \mid W \in \mathbb{R}_j, z \in \Omega_z, \Omega_z \subset \mathbb{R}_q \}.$$

**III .3.Conception du contrôleur adaptatif :**

En supposant que les fonctions non linéaires  $f(k)$  et  $g(k)$  sont connues exactement, et qu'il n'y a pas de perturbation dans le système, c'est-à-dire  $d_k = 0$ , nous présentons une commande désirée  $u^*_k$  telle que la sortie  $y_k$  suive la trajectoire désirée  $y_d$  asymptotiquement. Le lemme suivant établit l'existence de la commande idéale  $u^*_k$  qui amène la sortie du système à la trajectoire désirée.



**(Fig,III.1) :** Configuration des réseaux flous-neuraux[48].

**Lemme 2 :**

Considérons le système (3) avec les hypothèses 1 à 3 satisfaisantes et  $d_k = 0$ . Si l'entrée de commande souhaitée est choisie comme [41]:

$$u^*_k = -\frac{1}{g(k)} \left[ f(k) - y_d(k + \tau) - k_e \frac{g(x)}{g_1} e_n(k) \right] \tag{III.12}$$

où  $0 \leq k_v \leq 1$  est un paramètre de conception, alors  $\lim_{k \rightarrow \infty} ||e(k)|| = 0$

En notant l'expression (12), l'entrée de commande souhaitée  $u^*_k$  est une fonction de  $x(k)$ ,  $v_{k-1}(k)$ ,  $y_d(k+\tau)$  et  $e_n(k)$ , et elle peut être exprimée comme suit :

$$\begin{cases} u_k^* = u^*(z(k)) \\ z = [x^T(k), y_d(k + \tau), v_{k-1}^T(k), k_n e_n(k)]^T \in \Omega_z \subset \mathbb{R}^{n+m+2} \end{cases} \quad (\text{III.13})$$

où l'ensemble compact  $\Omega_z$  est défini comme suit :

$$\Omega_z = [ (x, y_d, v_{k-1}, k_v e_n) | v_{k-1}(k) = , [ u_{k-1} \cdots u_{k-m} ]^T, e(k) = x_n(k) - y_d(k + \tau - 1), x \in \Omega, y_d \in \Omega, u \in ]$$

f fonctions non linéaires f(k) et g(k) sont inconnues,

$u^*(z(k))$  n'est pas disponible. Dans la suite de l'exposé, réseaux neuronaux seront utilisés pour approximer la fonction inconnue  $u^*(z(k))$ .

**Remarque 3 :**

Lorsque  $k_v = 0$  dans (III.12), le régulateur est un régulateur de type car  $e_n(k) = 0$  est atteint après une étape. Pour faciliter l'analyse, définissons les variables

$$e_n(k) = x_n(k) - y_d(k + \tau - 1)$$

$$z(k) = [ x_1(k), \dots, x_{n-\tau+1}(k), x_{n-\tau+2}(k), \dots, x_n(k), y_d(k + \tau), v_{k-1}^T(k), k_v e_n(k) ] \quad (\text{III.14})$$

**Hypothèse 4 :**

Sur l'ensemble compact  $\Omega_z$ ,  $W^*$  satisfait à la condition suivante :  $\|W^*\| \leq \phi$

où  $\phi > 0$  est une constante inconnue. Notation  $\|\cdot\|$  désigne la norme 2.  $\hat{\phi}(k)$  est utilisé pour désigner l'estimation de  $\phi$ , et nous laissons  $\phi(k) = \hat{\phi}(k) - \phi$ .

Le contrôleur adaptatif à réseau flou-neuronal est donné par :

$$u_k = \hat{\phi}(k) \|s(z(k))\| \quad (\text{III.15})$$

$$\hat{\phi}(k + 1) = \hat{\phi}(k) - \beta [ \|s(z(k))\| e_n(k + 1) + \sigma \hat{\phi}(k) ] \quad (\text{III.16})$$

où  $\beta > 0$  et  $\sigma > 0$  sont les paramètres de conception. En substituant (III.15) dans (III.4), on obtient

$$e_n(k + 1) = f(k) - y_d(k + \tau) + g(k) \hat{\phi}(k) \|s(z(k))\| + d_{k+\tau-1} \quad (\text{III.17})$$

En ajoutant et en soustrayant  $g(k)u^*(z(k))$  du côté droit de (III.17) et en notant (III.11), nous avons

$$e_n(k + 1) = f(k) - y_d(k + \tau) + g(k) [ \hat{\phi}(k) \|s(z(k))\| - W^{*T} S(z(k)) - u_z ] + g(k)u^*(z(k)) + d_{k+\tau-1} \quad (\text{III.18})$$

En substituant (III.12) dans (III.18), nous avons

$$e_n(k + 1) = [ g(k) \hat{\phi}(k) \|s(z(k))\| - W^{*T}(k)S(z(k)) - u_z + \frac{k_v}{g_1} e_n(k) ] + d_{k+\tau-1}$$

$$= [ g(k) \hat{\phi}(k) \|s(z(k))\| - \phi \|s(z(k))\| + \phi(k) \|s(z(k))\| - W^{*T}(k)S(z(k)) - u_z + \frac{k_v}{g_1} e_n(k) ] + d_{k+\tau-1}$$

$$= [ g(k) \quad \phi(k) \parallel s(z(k)) \parallel + \phi(k) \parallel s(z(k)) \parallel -W^{*T} (k)S(z(k)) - u_z + \frac{k_v}{g_1} e_n(k) ] + d_{k+\tau-1} \quad (III.19)$$

Puisque l'approximation des réseaux flous-neuraux. (III.11) et les hypothèses 1-4 ne sont valables que sur les ensembles compacts  $\Omega$  et  $\Omega_u$ , il est nécessaire de garantir que les états du système restent dans  $\Omega$  et que le signal de commande reste dans  $\Omega_u$  pendant tout le temps.

En raison de l' $u_k \in \Omega_f \subset \Omega_u$ , il doit exister deux ensembles compacts non nuls  $\Omega_\phi \subset \mathbb{R}$  et  $\Omega_s \subset \mathbb{R}$  tels que pour tout  $\phi(k) \in \Omega_\phi$  et  $S(k) \in \Omega_s$  il est garanti. que  $u_k \in \Omega_u$ . Le théorème suivant démontre la stabilité et les performances de contrôle du système en boucle fermée.

**Théorème 1 :**

Considérons le système en boucle fermée constitué du système (III.3), du contrôleur (III.15) et de la loi d'adaptation (III.16). Il existe des ensembles compacts  $\Omega_0 \subset \Omega$ ,  $\Omega_{\phi_0} \subset \Omega_\phi$  et des constantes positives  $\alpha_1, k_v^*, I^*, \beta^*$  et  $\sigma^*$  satisfaisant  $\alpha_1 < 1, k_v^* \frac{2-\alpha}{3} \beta^* = \frac{2-\alpha-3K^*}{2+2I^*+g_1I^*}, \sigma^* = \frac{1}{\beta^*+g_1\beta^*I^*}$

où les hypothèses 1 à 4 sont satisfaites, la condition initiale  $x(0) \in \Omega_0, \phi(0) \in \Omega_{\phi_0}$ , et... les paramètres de conception sont convenablement choisis de telle sorte que :

$k_v < k_v^*, I > I^*, \sigma < \sigma^*$ , et  $\beta < \beta^*$ . Ensuite, le système en boucle fermée est SGUUB et l'erreur de suivi peut être rendue arbitrairement petite en choisissant les paramètres de conception. Rendre arbitrairement petite en choisissant les paramètres de conception de manière appropriée.

**Appendice : La preuve du théorème 1**

Preuve Sous le contrôleur (III.15), l'erreur de suivi (III.19) devient

$$\tilde{\Phi}(k) \parallel s(z(k)) \parallel = \frac{e_n(k+1) - d_{k+\tau-1}}{g(k)} + u_z - \frac{k_v}{g_1} e_n(k) - \phi \parallel s(z(k)) \parallel + W^{*T} (k)S(z(k)) \quad (III.20)$$

Choisissez le candidat de la fonction de Lyapunov suivante :

$$V(k) = \frac{\alpha_1}{g_1} \sum_{i=1}^n e_i^2(k) + \frac{2k_v}{g_1} e_n^2(k) + \beta^{-1} \tilde{\Phi}^2(k) \quad (III.21)$$

où  $\alpha_1$  est une petite constante positive et  $\alpha_1 < 1$ . Le site première différence de (III.21) le long de (III.4), (III.15), (III.16), et (III.20) est donnée par

$$\begin{aligned} \Delta v &= \frac{\alpha_1+2k_v}{g_1} e_n^2(k+1) - \frac{\alpha_1}{g_1} e_n^2(k) - \frac{2k_v}{g_1} e_n^2(k) + \beta^{-1} \tilde{\phi}^2(k+1) - \beta^{-1} \tilde{\phi}^2(k) \\ &= \frac{\alpha_1+2k_v}{g_1} e_n^2(k+1) - \frac{\alpha_1}{g_1} e_n^2(k) - \frac{2k_v}{g_1} e_n^2(k) - 2\tilde{\phi}(k) \|s(z(k))\| e_n(k+1) \\ 2\sigma \tilde{\phi}(k) \tilde{\phi}(k) + \beta \|s(z(k))\|^2 e_n^2(k+1) + 2\sigma\beta \tilde{\phi}(k) \|s(z(k))\|^2 e_n(k+1) + \sigma^2\beta\tilde{\phi}^2(k) &= \frac{\alpha_1+2k_v}{g_1} e_n^2(k+1) \\ 1) - \frac{\alpha_1}{g_1} e_n^2(k) - \frac{2k_v}{g_1} e_n^2(k) - 2\left[\frac{e_n(k+1)-d_{k+\tau-1}}{g(k)} + u_z - \frac{k_v}{g_1} e_n(k) - \phi \|s(z(k))\| + W^{*T}(k)S(z(k))\right] e_n(k+1) - 2\sigma \\ \tilde{\phi}(k) \tilde{\phi}(k) + \beta \|s(z(k))\|^2 e_n^2(k+1) + 2\sigma\beta \tilde{\phi}(k) \|s(z(k))\|^2 e_n(k+1) + \sigma^2\beta\tilde{\phi}^2(k) &\leq \frac{\alpha_1+2k_v}{g_1} e_n^2(k+1) - \\ \frac{\alpha_1}{g_1} e_n^2(k) - \frac{2k_v}{g_1} e_n^2(k) - 2\left[\frac{e_n(k+1)-d_{k+\tau-1}}{g(k)} + u_z - \frac{k_v}{g_1} e_n(k)\right] \times e_n(k+1) + 4\|s(z(k))\| |e_n(k+1)| - 2\sigma \\ \tilde{\phi}(k) \tilde{\phi}(k) + \beta \|s(z(k))\|^2 e_n^2(k+1) + 2\sigma\beta \tilde{\phi}(k) \|s(z(k))\|^2 e_n(k+1) + \sigma^2\beta\tilde{\phi}^2(k) \end{aligned}$$

En utilisant les faits suivants :

$$\beta \|s(z(k))\|^2 \leq \beta I$$

$$2e_n(k)e_n(k+1) \leq e_n^2(k) + e_n^2(k+1)$$

$$2\sigma\beta \|s(z(k))\| e_n(k+1) \leq \frac{\beta}{g_1} e_n^2(k+1) + g_1\beta I \sigma^2 \tilde{\phi}^2(k) 2\left(\frac{d_{k+\tau-1}}{g(k)} + u_z\right) e_n(k+1) \leq \frac{\beta}{g_1} e_n^2(k+1) \frac{g_1}{\beta} \phi^2$$

$$2\tilde{\phi}(k)\tilde{\phi}(k) = \tilde{\phi}(k)^2 + \tilde{\phi}(k)^2 - \phi^2$$

nous obtenons :

$$\Delta v(k) \leq \frac{\rho_1}{g_1} e_n^2(k+1) - \frac{\alpha_1}{g_1} e_n^2(k) - \frac{k_v}{g_1} e_n^2(k) - \sigma\tilde{\phi}(k)^2 - \sigma(1 - \sigma\beta - g_1\sigma\beta I) + \tilde{\phi}(k)^2 \theta$$

$$\text{Où } \rho_1 = 2 - \alpha_1 - 3k_v - 2\beta - 2\beta I - \beta I g_1, \theta_1 = \frac{g_1}{\beta} \left(u_z - \frac{d_{k+\tau-1}}{g(k)}\right)^2 + \frac{2g_1}{\beta} \phi^2 + \sigma\phi^2$$

Si nous choisissons les paramètres de conception comme

$$k_v = \frac{2 - \alpha_1}{3}, \quad \beta < \frac{2 - \alpha_1 - 3k_v}{2 + 2I + I g_1}, \quad \sigma < \frac{1}{\beta + g_1\beta I} \quad (\text{III.22})$$

alors  $V(k) \leq 0$  dès lors que l'erreur de suivi  $e_1(k)$  est supérieure à que  $\varepsilon_1 = \sqrt{\frac{g_1\theta_1}{\alpha_1}}$

Ceci implique le caractère borné de  $V(k)$  pour tout  $k \geq 0$ , ce qui conduit à la délimitation de  $e(k)$ , car  $V(k) = \sum_{j=0}^k \Delta v(j) + V(0) < \infty$

De plus, l'erreur de suivi  $e_1(k)$  convergera asymptotiquement vers l'ensemble compact  $\Omega_e$ , puisque  $e_1(k+n-1) = e_2(k+n-2) = \dots = e_n(k)$ . En raison de la négativité de  $\Delta V$ , on peut conclure que  $x(k+1) \in \Omega$  si  $x(k) \in \Omega, u(k-j) \in \Omega_u, j = 1, 2, \dots, n - \tau$  et  $\Omega_e$  est suffisamment petit. Il reste maintenant à montrer que l'estimation du poids  $\tilde{\phi}(k)$  est bornée, et  $u_k \in \Omega_u$ . La dynamique de la mise à jour des poids des réseaux flous neuronaux (III.16) peut être écrite comme suit

$$\tilde{\phi}(k + 1) = \tilde{\phi}(k) - \beta[\|s(z(k))\|e_n(k+1) + \sigma(\tilde{\phi}(k) + \phi)] = (1 - \beta\sigma)\tilde{\phi}(k) - \beta[\|s(z(k))\|e_n(k+1) + \sigma\phi]$$

qui satisfait à  $|1 - \beta\sigma| < 1$  car  $0 < \beta\sigma < \frac{1}{1+g_1} < 1$ . En notant que  $S(z(k))$ ,  $e(k + 1)$  et  $\sigma\phi$

sont bornées et d'après le lemme 1  $\tilde{\phi}(k)$ , doit être borné dans un ensemble compact noté  $\Omega_{\phi_e}$ . Ainsi, La caractéristique bornée de  $\tilde{\phi}(k)$  est obtenue.

Enfin, si on initialise l'état  $x(0) \in \Omega_0$ ,  $\tilde{\phi}(0) \in \Omega_{\phi_0}$ , et que nous choisissons des paramètres convenables  $l$ ,  $k_v$ ,  $\beta$  et  $\sigma$  selon (23) en faisant en sorte que  $\Omega_e$ ,  $\Omega_{se}$  et  $\Omega_{\phi_e}$  soient suffisamment petits, c'est-à-dire  $\Omega_d \cap \Omega_e \subset \Omega$  et  $\Omega_{\phi_e} \subset \Omega_{\phi}$ , il existe une constante  $k^*$  telle que toutes les erreurs de suivi convergent asymptotiquement vers  $\Omega_e$ , et l'erreur de poids des réseaux flous-neuraux converge asymptotiquement vers  $\Omega_{\phi_e}$ , pour tout  $k > k^*$ , et tous les autres signaux du système sont SGUUB.

### III .4. Exemple de simulation :

Pour montrer l'efficacité des schémas de contrôle proposés une étude de simulation est présentée dans cette section. Considérons une usine SISO non linéaire à temps discret décrit par :

$$\begin{cases} x_1(k + 1) = x_2(k) \\ x_2(k + 1) = x_3(k) \\ x_3(k + 1) = f(x(k), u_{k-1}) + g(x(k))u_k + d_{k+\tau-1} \end{cases} \quad (III.23)$$

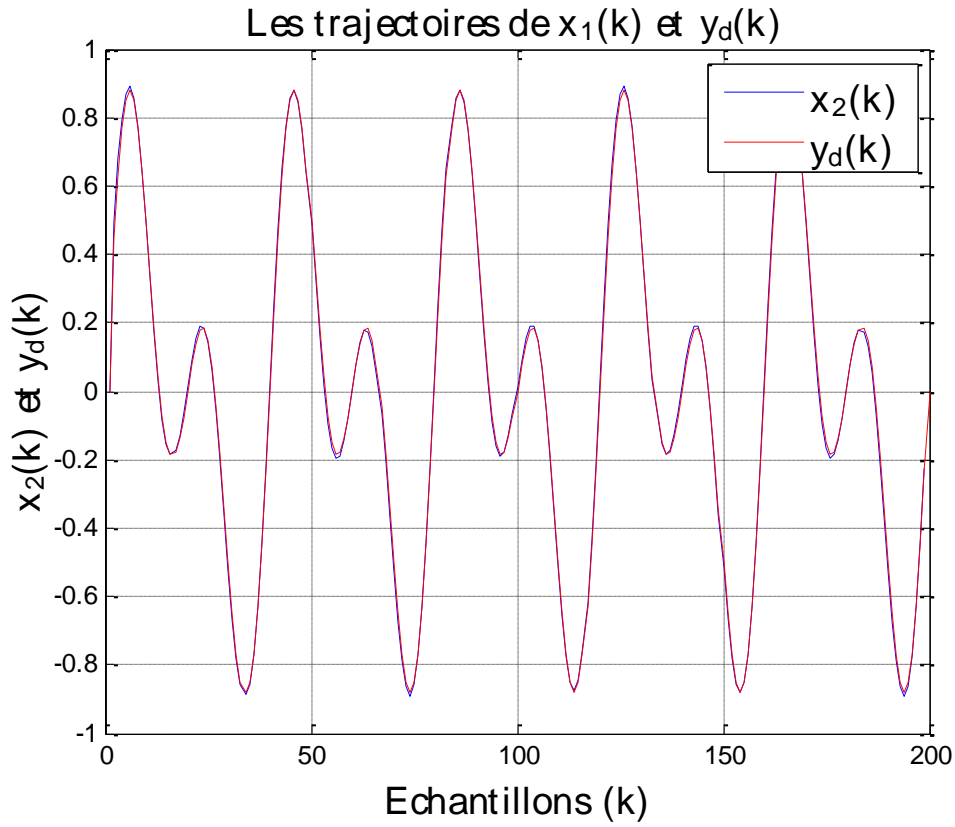
$$y_k = x_2(k)$$

$$\text{où } f(x(k), u_{k-1}) = \frac{x_1(k)x_2(k)(x_1(k)+2;5)u_{k-1}}{1+x_1^2(k)+x_2^2(k)+x_3^2(k)}, \quad g(x(k)) = \frac{2,5}{1+x_1^2(k)+x_2^2(k)+x_3^2(k)}$$

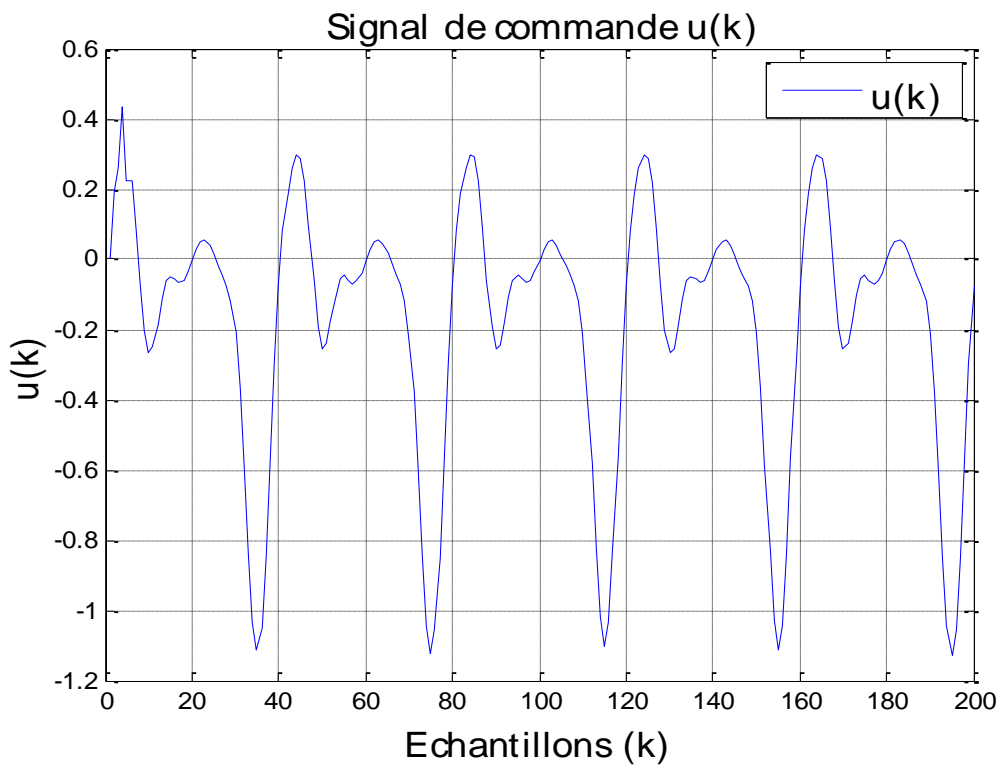
$$d_k = 0.1 \cos(0.05k) \cos(x_1(k))$$

On peut vérifier que les hypothèses 1 et 2 sont satisfaites. L'objectif de la commande est de faire en sorte que la sortie  $y_k$  suive un signal de référence désiré  $y_d(k) = \frac{1}{2} \sin(\frac{k\pi}{20}) + \sin(\frac{k\pi}{10})$  à un ensemble compact. Les paramètres de conception sont choisis comme  $\beta = 0,04$ ,  $k_v = 0,1$  et  $\sigma = 0,05$ . La valeur initiale est choisie comme  $\tilde{\phi}(0) = 0$ .

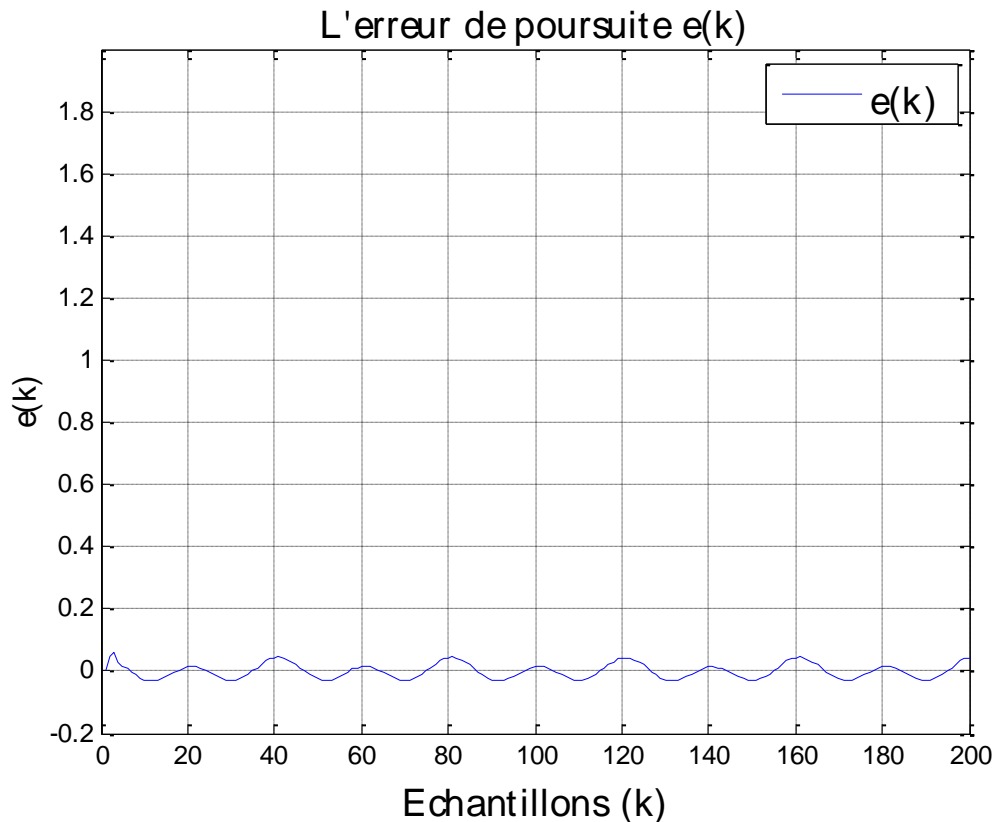
Les résultats de la simulation sont illustrés à la **figure (III.2)**, qui montre la courbe de poursuite du système en boucle fermée. On peut voir qu'une bonne performance de suivi est obtenue



**Figure (III.2) :** Sortie  $y(k)$  et signal de référence  $x_2(k)$ .



**Figure (III.3) :** Entrée de commande.



**Figure (III.4) : Signal d'erreur  $e(k)$ .**

La Figure(III.2) illustre les trajectoires de suivi du système et peut permettre d'obtenir de bonnes performances de suivi. Forme Figure (III.3), Figure (III.4) Il représente une erreur comprise entre 0,1 et -0,1.

### III .5.Conclusion

Dans ce chapitre, nous avons présentés une approche de commande adaptative pour une classe inconnue à temps discret SISO. Dans ces systèmes, le degré de relativité générale est suggéré en présence de perturbation bornée. Pour créer un contrôleur idéal, les systèmes sont d'abord transformés en une description d'espace d'état, sur la base des propriétés d'approximation universelles. Pour la commande adaptative floue, le contrôleur requis est introduit par système flou avec un nombre de paramètres adaptatifs réduit. Des résultats de simulation sont obtenus sur la base de la théorie de la stabilité Lyapunov.

## Conclusion générale

L'objectif principal du travail présenté dans ce manuscrit est de développer une structure de commande adaptative floue stable et indirecte pour une classe de systèmes non linéaires incertains.

Dans ce travail, le but de la commande est de résoudre le problème de poursuite de trajectoire. Ce dernier est considéré comme un problème de construction d'un système de contrôle capable de forcer la sortie d'un système non linéaire à suivre une trajectoire de référence autorisée. Pour résoudre ce problème, dans ce cas, nous obtenons une commande appliquée à un système non linéaire qui fait approcher l'erreur de suivi de trajectoire de zéro.

La première phase, l'étude de la logique floue, une logique qui permet le traitement mathématique de données incertaines, utilise différents modèles et théorèmes d'approximation floue dans sa conception.

La deuxième étape, nous introduisons la structure du contrôle adaptatif et ses deux variantes, contrôle décentralisé direct et indirect, et étudions l'analyse de stabilité en boucle fermée à l'aide de la méthode de Lyapunov.

La troisième étape, un problème est de développer une méthode adaptative floue décentralisée pour construire ce contrôleur pour une classe de systèmes non linéaires incertains.

Les résultats de simulation obtenus montrent l'efficacité de la structure de commande proposée en termes de performances de poursuite et de caractéristiques du signal de commande.

### Perspectives :

- l'intégration de la commande robuste via la technique de la commande par mode glissant pour avoir une commande robuste vis-à-vis des perturbations externes.
- Application de cette approche sur les systèmes réels
- Validations l'approche étudiée par des résultats expérimentaux.

## **Bibliographies**

- [1]: B. Kosko, Neural Networks and Fuzzy Systems: A Dynamical Systems Approach. Englewood Cliffs, NJ: Prentice Hall, 1992.
- [2]: L.X. Wang, Adaptive Fuzzy Systems and Control: Design and Stability Analysis. Prentice-Hall, Englewood Cliffs, NJ, 1994.
- [3]: j.S.R. Jang and C.T. Sun, "Neuro-Fuzzy modeling and control," Proc. IEEE, vol. 83, no. 3, pp. 378-406, Mar. 1995 .
- [4]: J. M. Mendel, "Fuzzy systems for engineering: A tutorial," Proc. IEEE, vol. 83, no. 3, pp. 347-377, Mar. 1995.
- [5]: K.M. Passino and S. Yurkovich, Fuzzy Control. Addison-Wesley Longman Inc., 1998.
- [6]: (commande adaptative floue décentralisée pour une classe des systèmes non linéaires incertains), mémoire de master, universite kasdi merbah ouargla, annee universitaire 2018-2019
- [7]: J.R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," IEEE Trans Syst Man, Cybern. vol. 23, no. 3, pp. 665-685, 1993.
- [8]: H. Buhler, Le réglage par logique floue. Polytechniques Rommandes, 1994
- [9]: S. Labiod, "Commande adaptative par systèmes flous: application aux robots manipulateurs, Thèse de Magister, ENP, 1998.
- [10]: S.Zeghlache ,cours , Chapitre I Commande par Logique Floue, 2A Master AUT+CSE , UNIVERSITE,M, Boudiaf , Msila.
- [11]: M.M. Heniche, Sur l'utilisation des réseaux de neurones artificiels et des systèmes flous pour la linéarisation et la commande de processus chimiques non linéaires, Thèse de Magister, ENP, 1997.
- [12]: M. Sugeno and G. T. Kang, "Fuzzy identification of systems and its applications to modeling and control," IEEE Trans. Syst., Man, Cybern., vol. 15, pp. 116-132, 1985
- [13]: Krstic M., Kanellakopoulos I. and Kokotovic P., "Nonlinear and Adaptive Control Design", John Wiley and Sons Inc., 1995.
- [14] : S. Meziane « Commande adaptative et prédictive de la machine asynchrone», Thèse doctorat en électromécanique, Université Mentouri Constantine, 2009.
- [15]: Olivier Pages, « Etude et comparaison de différentes structures de commande multi contrôleurs application à un axe robotise», L'Ecole Supérieure d'Ingénieurs d'Annecy, Université de Savoie, Juillet 2001.

- [16]: L. Reznik, "Fuzzy Controllers", Newnes, 1997.
- [17]: S. Labiod, "Contribution à la Commande Adaptative Floue des Systèmes Non Linéaires", Thèse Doctorat, Ecole Nationale Polytechnique, 2005.
- [18]: I.Boulkaibet, "Une étude en simulation des stratégies de commande non linéaire", Mémoire de Magistère, UNIV-Constantine, 2007.
- [19]: M. Zelmat., « Automatisation des processus industriels Tome 2, Commande modale et adaptative », Office des publications universitaires, Algérie 2001.
- [20]: L. Praly, "La commande adaptative des systèmes linéaires à temps discret", MINES Paristch, Center Automatique et système, 2010.
- [21]: Ge, S.S., Hang, C.C., Lee, T.H., Zhang, T.: Stable Adaptive Neural Network Control. Kluwer Academic, Boston (2002).
- [22]: W.S. Chen, Jiao, L.C., Li, J., Li, R.H.: Adaptive NN backstepping output-feedback control for stochastic nonlinear strict-feedback systems with time-varying delays. IEEE Trans. Syst. Man Cybern., Part B, Cybern. 40(3), 939–950 (2009).
- [23]: Diaz, D.V., Tang, Y.: Adaptive robust fuzzy control of nonlinear systems. IEEE Trans. Syst. Man Cybern., Part B, Cybern. 34(3), 1596–1601 (2004).
- [24]: Y.G. Leu, Lee, T.T., Wang, W.Y.: Observer-based adaptive fuzzy-neural control for unknown nonlinear dynamical systems. IEEE Trans.Syst Man Cybern , Part B, Cybern. 29(2), 583–591 (1999).
- [25]: Y.J. Liu, Wang, W., Tong, S.C., Liu, Y.S.: Robust adaptive tracking control for nonlinear systems based on bounds of fuzzy approximation parameters. IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum. 40(1), 170–184 (2010).
- [26]: Y.J. Liu, Tong, S.C., Li, T.S.: Observer-based adaptive fuzzy tracking control for a class of uncertain nonlinear MIMO systems. Fuzzy Sets Syst. 164(1), 25–44 (2011).
- [27]: H.G. Zhang, Bien, Z.: Adaptive fuzzy control of MIMO nonlinear systems. Fuzzy Sets Syst. 115(2), 191–204 (2000).
- [28]: Zhang, H.G., Cai, L.L., Bien, Z.: A fuzzy basis function vector-based multivariable adaptive fuzzy controller for nonlinear systems. IEEE Trans. Syst. Man Cybern., Part B, Cybern. 30(1), 210–217 (2000).
- [29]: Adetona, O., Garcia, E., Keel, L.H.: A new method for the control of discrete nonlinear dynamic systems using neural networks. IEEE Trans. Neural Netw. 11(1), 102–112 (2000).
- [30]: Chen, W.S.: Adaptive NN control for discrete-time purefeedback systems with unknown control direction under amplitude and rate actuator constraints. ISA Trans. 48(3), 304–311 (2009).

## Bibliographies

- [31]: Chen, F.C., Khalil, H.K.: Adaptive control of a class of nonlinear discrete-time systems using neural networks. *IEEE Trans. Autom. Control* 40, 791–801 (1995).
- [32]: Liu, Y.J., Wen, G.X., Tong, S.C.: Direct adaptive NN control for a class of discrete-time nonlinear strict-feedback systems. *Neurocomputing* 73(13–15), 2498–2505 (2010).
- [33]: Wen, G.X., Liu, Y.J., Tong, S.C.: Adaptive neural output feedback control of nonlinear discrete-time systems. *Nonlinear Dyn.* doi:10.1007/s11071-010-9874-4 (2010).
- [34]: Yang, C.G., Ge, S.S., Lee, T.H.: Output feedback adaptive control of a class of nonlinear discrete-time systems with unknown control directions. *Automatica* 45(1), 270–276 (2009).
- [35]: Yang, C., Li, Y., Ge, S.S., Lee, T.H.: Adaptive control of a class of discrete-time MIMO nonlinear systems with uncertain couplings. *Int. J. Control* 83(10), 2020–2133 (2010).
- [36]: S.Chen, Billings, S.: Representations of nonlinear systems: The NARMAX model. *Int. J. Control* 49(3), 1013–1032 (1989).
- [37]: S.S.Ge, Lee, T.H., Li, G.Y., Zhang, J.: Adaptive NN control for a class of discrete-time nonlinear systems. *Int. J. Control* 76(4), 334–354 (2003).
- [38]: Ge, S.S., Zhang, J., Lee, T.H.: Adaptive MNN control for a class of non-affine NARMAX systems with disturbances. *Syst. Control Lett.* 53, 1–12 (2004).
- [39]: C.G.Yang, S.S.Ge, C. Xiang, T. Chai, Lee, T.H.: Output feedback NN control for two classes of discrete-time systems with unknown control directions in a unified approach. *IEEE Trans. Neural Netw.* 19(11), 1873–1886 (2008).
- [40]: T.S. Li, Tong, S.C., Feng, G.: A novel robust adaptive fuzzy-tracking control for a class of nonlinear multi-input/ multi-output systems. *IEEE Trans. Fuzzy Syst.* 18(1), 150–160 (2010).
- [41]: S.C. Tong, He, X.L., Li, Y.M., Zhang, H.G.: Adaptive fuzzy backstepping robust control for uncertain nonlinear systems based on small-gain approach. *Fuzzy Sets Syst.* 161(3), 771–796 (2010).
- [42]: (Commande adaptative floue en poursuite pour une classe de systèmes non linéaires) mémoire de fin d'études en vue de l'obtention du diplôme de master génie électronique, Année Universitaire:2019/2020.
- [43]: J.E. Slotine and W. Li, *Applied nonlinear control*, Prentice Hall, Englewood Cliffs, NJ, 1991.
- [44]: E.H.Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," *IEEE Trans. Computers*, vol. 26, no. 12, pp. 1182-1191, Dec. 1977.
- [45]: T.J.Procyk and E.H.Mamdani, "A linguistic self-organizing process controller," *Automatic*, vol. 15, pp. 15-30, 1978.

## Bibliographies

- [46]: J.T.Spooner and K.M.Passino, “Stable adaptive control using fuzzy systems and neural networks,”IEEETrans.FuzzySyst.,vol.4,pp.339-359,1996.
- [47]: L.X. Wang and J.M. Mendel, “Fuzzy Basis Functions, Universal Approximation and Orthogonal Least-Squares Learning,” IEEE Trans. on Neural Networks, vol. 3, no. 5, pp. 807-814, 1992.
- [48]: Guo-Xing Wen · Yan-Jun Liu, (Adaptive fuzzy-neural tracking control for uncertain nonlinear discrete-time systems in the NARMAX form) (2011) pp:749.

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

**Proposé et dirigé par** : Dr. BOUKHALFA Abdelouaheb & Dr. KHETTAB Khatir

**Présenté par** : BOULAARES Oussama , CHABBI Aymen

**Thème :**

**« Commande adaptative floue en poursuite pour une classe de systèmes non linéaires »**

**Résumé** : Dans ce travail, un schéma de commande adaptative floue est étudié pour une classe de systèmes non linéaires à temps discret mono entrée mono sortie (SISO) en présence de perturbations bornées. Les systèmes SISO se présentent sous la forme de la moyenne mobile autorégressive non linéaire avec entrée exogène (NARMAX), qui a reçu beaucoup d'attention dans le domaine du contrôle discret. Afin de réaliser la stabilité, les systèmes sont d'abord transformés en une description d'espace d'état causal, et un contrôleur idéal est obtenu. Le contrôleur souhaité est approximé en utilisant les systèmes flous. Par rapport aux résultats précédents pour le contrôle de NARMAX, le nombre de paramètres d'adaptation est moindre, et donc, il réduit la charge de calcul. Enfin, la faisabilité de cette technique est prouvée par un exemple de simulation.

**Mots-clés** : Commande adaptative; Commande floue, Systèmes non linéaires, Stabilité de Lyapunov.

**Abstract** : In this work, an adaptive fuzzy control scheme is investigated for a class of single-input single-output (SISO) discrete-time nonlinear systems in the presence of bounded disturbances. The SISO systems are in the form of the nonlinear autoregressive moving average with exogenous input (NARMAX), which has received much attention in the discrete control field. In order to realize the stability, the systems are first transformed into a causal state space description, and an ideal controller is obtained. The desired controller is approximated by using the fuzzy systems. Compared with the previous results for controlling NARMAX, the number of adaptation parameters is less, and thus, it reduces the computational burden. Finally, the feasibility of this theory is proven in a simulation.

**Keywords** – Adaptive control, Fuzzy control, Nonlinear systems, Lyapunov stability.