

République algérienne démocratique et populaire
Ministère de l'enseignement supérieur et de la recherche scientifique

Université de Constantine
Faculté des Sciences de l'Ingénieur
Département d'Electronique

N° d'ordre :

Série :

THESE

Présentée pour l'obtention du grade de

Docteur en Sciences

En Electronique, Option : Contrôle des Systèmes
Par

BAHITA Mohamed

Thème

**Commande Adaptative Floue des Systèmes Non Linéaires
Basée sur les Réseaux à Mémoire Associative
à Fonction de Base Radiale**

Soutenu le 12 / 07 / 2011 devant le jury composé de:

Président	S. Filali,	Professeur	Université de Constantine
Directeur de thèse	K. Belarbi,	Professeur	Université de Constantine
Examineur	S. Djennoune,	Professeur	Université de Tizi-Ouzou
Examineur	A. Khelassi,	MC. A	Université de Boumerdès
Examineur	B. Boutamina	MC. A	Université de Constantine

Constantine, 2011

الإهداء

الى عائلتي
الى الذين حملوا شعلة العلم
الى الذين ياتمسون الطريق المستقيم لبناء جيل جديد على
أساس من الإيمان و العلم و المعرفة ...
الى هؤلاء جميعا.
أهدي هذا العمل .

ذو العلم يشقى في النعيم بعقله
و اخو الجهالة في الشقاوة ينعم

Remerciement

Je remercie infiniment mon directeur de thèse Monsieur Khaled BELARBI, professeur à l'Université de Constantine pour sa confiance, sa bonne humeur, et la liberté qu'il m'a donné et pour ses conseils et son grand effort déployé tout au long de ce travail.

Je tiens à exprimer mes vifs remerciements à Mr. Salim Filali, professeur à l'Université de Constantine pour avoir accepté de présider le jury de ma soutenance.

Je tiens aussi à remercier vivement les membres du jury, Prof. S. Djennoune, Dr. A. Khelassi,, et Dr. B. Boutamina pour avoir accepté d'évaluer mon travail de thèse.

Je tiens aussi à remercier tous mes amis et tout les membres de ma famille qui m'ont toujours soutenu, dans les meilleurs moments comme dans les pires.

De tout mon cœur, Je remercie tout le monde

Tables des matières:

Chapitre I: Introduction générale	1
I.1 Motivation et formulation du problème.....	1
I.2 Objectif de la thèse	3
I.2.1 L'approche indirecte.	3
I.2.2 L'approche directe	4
I.3 Organisation de la thèse	4
Chapitre 2: Préliminaires	5
II.1 Introduction	5
II.2 Linéarisation par retour d'état (Feedback linearization).....	5
II.3 Linéarisation par retour d'état adaptative	7
II.4 Les approximateurs universels	8
II.4.1 Approximation des fonctions par les réseaux RBF.....	8
II.4.1.a La structure des réseaux RBF.....	8
II.4.1.b Entraînement et placement des centres dans un réseau RBF.....	10
i Ajustement des centres.....	10
ii La règle d'adaptation instantanée.....	11
II.4.2 Approximation des fonctions par les systèmes d'inférence flous.....	11
II.4.2.a Modèle de Mamdani-Zadeh.....	12
II.4.2.b Modèle de Takagi-Sugeno	12
Chapitre 3: Commande adaptative floue et neuronale des systèmes non linéaires affines avec gain de commande constant	13
III.1 Introduction.....	13
III.2 Formulation du problème.....	14
III.2.1 Le modèle.....	14
III.2.2 La loi de commande idéale.....	14
III.3 Commande adaptative floue directe	16

III.3.1 Analyse de la stabilité et lois d'adaptation du contrôleur TS.....	16
Théorème.....	19
Démonstration	19
Remarque 1.....	21
Remarque 2	21
<i>Etape 1: Calcul hors ligne.....</i>	21
<i>Etape 2: Adaptation en ligne.....</i>	21
III.3.2 Résultats de simulation.....	22
III.3.2.1 Exemple 1.....	22
III.3.2.2 Exemple 2.	26
III.3.3 Conclusion.....	28
III.4 Commande adaptative neuronale directe	29
III.4.1 Analyse de la stabilité et lois d'adaptation du contrôleur RBF.....	29
<i>Etape 1: Calculs hors ligne.....</i>	30
<i>Etape 2: Adaptation en ligne.....</i>	31
III.4.2 Résultat de simulation.....	31
III.4.2.1 Exemple 1.....	31
III.4.2.2 Exemple 2.....	33
III.4.3 Conclusion.....	35

**Chapitre 4: Commande adaptative floue et neuronale indirecte et directe
des systèmes non linéaires affines**

IV Introduction	36
IV.1 L'approche indirecte.....	37
IV.1.1 Analyse de la stabilité et lois d'adaptation des paramètres.....	37
Théorème.....	43
Démonstration	44
Remarque	47
IV.1.2 Conception du contrôleur adaptatif indirect TS-RBF	47
Etape1: Calculs hors ligne.....	47
Etape2: Adaptation en ligne.	48
IV.1.3 Résultats de simulation.....	48
IV.1.3.1 Exemple 1.....	49
IV.1.3.2 Exemple 2.....	55

IV.1.4 Conclusion	60
IV.2 L'approche directe.....	60
IV.2.1 Le contrôleur adaptatif flou TS.....	60
IV.2.2 L'estimateur flou de l'erreur de commande.....	62
IV.2.3 Analyse de la stabilité et lois d'adaptation des paramètres.	64
Théorème.....	65
Démonstration	66
IV.2.4 Conception du contrôleur adaptatif direct TS-Mamdani	67
<i>Etape1: Calculs hors ligne.....</i>	67
<i>Etape2: Calcul en ligne.....</i>	68
IV.2.5 Résultats de simulation.....	68
IV.2.5.1 Exemple 1.....	68
IV.2.5.2 Exemple 2	73
IV.2.6 Conclusion.....	77
Conclusion générale.....	78
Bibliographie.....	80
Annexe.....	84

Chapitre I:

Introduction générale

I.1 Motivation et formulation du problème

La recherche dans la théorie de la commande non linéaire a été motivée par les caractéristiques non linéaires propres aux systèmes dynamiques. Si on ajoute à la nature non linéaire, le fait que beaucoup des systèmes ne sont pas bien connus et non exactement modélisés, il est clair que la commande linéaire n'est pas suffisante pour garantir la stabilité et les performances des systèmes non linéaires. La commande adaptative semble aujourd'hui une stratégie naturelle pour la stabilisation et la poursuite des systèmes dynamiques hautement incertains. En réalité, depuis la fin des années cinquante, les théoriciens de la commande ont développé des lois de commande qui garantissent la stabilité en boucle fermée en présence des dynamiques non modelées et des perturbations extérieures. L'architecture de la commande adaptative à modèle de référence (ou Model Reference Adaptive Control (MRAC)) [1, 2, 3] était premièrement proposée pour les systèmes linéaires [4]. L'approche du MRAC était basée sur la loi heuristique du gradient ou la règle Delta, aussi connue comme la loi du M.I.T.

L'extension des techniques de commande linéaires aux systèmes non linéaires a été possible grâce au développement de la commande géométrique non linéaire. De récentes recherches incluant les méthodes géométriques différentielles [3], [5, 6] ont rendu la conception des contrôleurs pour une classe des systèmes non linéaires plus systématique. Cette théorie de commande non linéaire est basée sur des transformations coordonnées par lesquelles une classe des systèmes non linéaires peut être transformée en systèmes linéaires à travers le retour d'état, c'est la fameuse *feedback linearization*. Cette linéarisation par retour exacte s'est avérée assez restrictive pour des applications pratiques dû à son manque de robustesse envers les dynamiques non modelées. Des séries de résultats étaient rapportées dans la littérature essayant d'adresser des moyens pour surmonter ce problème par des méthodes de linéarisation par retour approximées [7,

8, 9, 10]. La commande adaptative des systèmes linéarisables était premièrement proposée dans [11, 12]. Plus tard, Kanellakopoulos et al [13] développent le **back-stepping control** laquelle devenue un outil puissant et très populaire dans la commande adaptative non linéaire [6]. Une extension aux systèmes non linéarisables était proposée dans [14]. Souvent, dans la littérature de la commande adaptative classique, il est commun de supposer des dynamiques inconnues avec une structure connue et des paramètres inconnus entrant linéairement dans les dynamiques. La paramétrisation linéaire des dynamiques inconnues pose des obstacles sérieux dans l'adoption des algorithmes de commande adaptative dans les applications pratiques, parce qu'il est difficile de fixer la structure des non linéarités inconnues. Ce fait a été le facteur motivant derrière l'intérêt des approximateurs de fonctions en ligne pour estimer et apprendre les fonctions inconnues. Les approximateurs de fonctions les plus couramment utilisés dans la commande adaptative sont les structures des réseaux de neurones artificiels et de la logique floue.

L'aptitude des réseaux de neurones d'approximer uniformément les fonctions continues a été prouvée dans plusieurs articles [15, 16, 17]. Les réseaux de neurones pour l'identification et la commande étaient premièrement proposés par Narendra et Parthasarathy [18] mais aucune preuve de stabilité dans la boucle fermée était fournie. Sanner et Slotine ont proposé en premier lieu des réseaux de neurones à fonction de base radiales (Radial Basis Function (RBF)) pour la commande des systèmes affines en la commande avec une analyse de la stabilité en introduisant une composante à mode glissant dans la commande [19]. De la même façon, il est montré [20, 21, 22] que les systèmes flous sont capables d'approximer n'importe quelle fonction non linéaire lisse autour d'une région compacte convexe, ce qui a permis leur utilisation dans le cadre de la commande adaptative.

L'approche générale utilisant ces approximateurs de fonctions est alors basée sur la technique de linéarisation par retour discutée au-dessus [3], [5, 6]. Deux méthodologies ont été utilisées, la méthode directe et la méthode indirecte. Dans la méthode indirecte, les approximateurs sont utilisés pour approximer les fonctions non linéaires inconnues apparentes dans le modèle [23, 24, 25, 26, 27, 28, 29], dans la méthode directe, ils approximent la loi de commande complète [23], [26], [29, 30, 31, 32, 33, 34, 35]. Cependant, dans la plus part des cas, un terme complémentaire appelé : terme superviseur est ajouté à la sortie du système d'inférence flou dans le but de garantir la stabilité globale en utilisant la théorie de Lyapunov. La différence

entre les différentes méthodes réside généralement dans la structure du modèle et dans la forme du terme superviseur. En particulier, les systèmes non linéaires sous la forme canonique sont considérés dans [25], [27, 28], [33] par contre les auteurs dans [23], [26], [29], [31] étudient les systèmes sous la forme affine. Dans [24], [35], la loi de commande est obtenue à travers la méthodologie **backstepping**. Plus généralement, des systèmes non affines sont étudiés dans [30], [32, 33].

Dans la plus part des études incluant la commande adaptative, les lois d'adaptation pour l'ajustement des paramètres sont obtenues à travers la théorie de stabilité de Lyapunov et l'erreur de poursuite est utilisée comme un signal d'adaptation. Quelques auteurs ont proposé l'utilisation de l'erreur de commande. Cependant le signal de commande idéal étant inconnu, cette erreur n'est pas directement disponible. L'erreur de commande qui apparaît dans la loi d'adaptation est calculée en utilisant un contrôleur hors ligne [36]. Dans [37] un estimateur flou de l'erreur de commande est utilisé pour l'adaptation en ligne d'un contrôleur adaptatif à base d'un réseau MLP, Multi layer Perceptron. Plus récemment [32], une commande floue adaptative stable basée sur l'erreur de commande est obtenue en utilisant une approximation linéaire du modèle non affine.

I.2 Objectif de la thèse

Dans cette thèse, quelques approches originales de la commande adaptative des systèmes non linéaires en utilisant la linéarisation par retour d'état sont introduites. Ce travail considère le cas des systèmes non linéaires à une seule entrée, une seule sortie (SISO) et affines en l'entrée de commande. Deux approches sont considérées : l'approche indirecte et l'approche directe.

I.2.1 L'approche indirecte

Dans cette partie, on présente une approche de commande adaptative indirecte. La méthode combine les performances d'une commande à logique floue et les propriétés d'approximation des systèmes à réseaux de neurones. Les réseaux à fonction de base radiale RBF sont adoptés pour l'approximation en ligne du gain de commande virtuel qui est une fonction non linéaire inconnue dans le système non linéaire contrôlé. Le contrôleur à logique floue du type (Takagi-Sugeno (TS)) est utilisé pour approximer la loi de commande idéale qui ne peut pas être

implémentée à cause du manque des connaissances des dynamiques du système. Dans la plus part des travaux discutés ci-dessus, la convergence de la stabilité est établie et les lois de commande sont dérivées en utilisant la théorie de stabilité de Lyapunov et en se basant sur quelques suppositions. Dans cette première approche, la stabilité asymptotique du système résultant en boucle fermée est établie et les lois de commande pour les paramètres (les poids de connections du réseau RBF et les paramètres des conséquences du contrôleur TS) sont dérivées en utilisant toujours la théorie de stabilité de Lyapunov, mais en se basant sur quelques développements mathématiques additionnels. L'algorithme des *k-means* [38, 39] sera utilisé pour ajuster en ligne les centres des fonctions de base du réseau RBF.

I.2.2 L'approche directe

Dans la deuxième partie de ce travail, on considère le développement d'une commande adaptative directe floue en utilisant un système d'inférence flou du type TS pour approximer la loi de commande de linéarisation par retour d'état sans l'addition du terme superviseur. Un deuxième système d'inférence flou du type Mamdani est utilisé pour estimer l'erreur de commande qui n'est pas disponible. Les paramètres du contrôleur TS sont mis à jour en utilisant la loi de la descente du gradient et en se basant sur l'erreur de commande estimée. La base de règle de l'estimateur flou Mamdani est construite en se basant sur quelques connaissances à priori du système. La stabilité asymptotique du système en boucle fermée résultant est analysée en utilisant la théorie de Lyapunov.

I.3 Organisation de la thèse

Cette thèse est organisée comme suit: dans le chapitre II, la technique de base (linéarisation par retour d'état) est introduite, ainsi que les définitions et concepts utilisés. Dans le chapitre III, la formulation du problème est introduite et deux solutions seront proposées pour traiter et résoudre un cas particulier des deux approche principales (qui seront traitées dans le chapitre IV) avec le gain de commande virtuel constant. Dans le chapitre IV, un premier algorithme de commande adaptatif indirect TS-RBF pour la première approche avec l'analyse de la stabilité sera développé, les lois d'adaptation pour les paramètres seront dérivées et des résultats de simulation seront présentés. Dans le reste du chapitre IV, le deuxième algorithme de commande flou TS-Mamdani adaptatif direct pour la deuxième approche est présenté.

Chapitre 2:

Préliminaires

II.1 Introduction

Le but de ce chapitre est de donner les définitions principales et les concepts utilisés dans cette thèse. Nous commencerons par donner les définitions de base concernant la linéarisation par retour d'état d'un système non linéaire. Ensuite nous introduirons, les éléments de base des réseaux de neurones artificiels et quelques notions sur la théorie de la logique floue.

II.2 Linéarisation par retour d'état (Feedback linearization)

La linéarisation par retour d'état (Feedback Linearization) [3], [5, 6] est une approche de conception de commande non linéaire dont l'idée principale est de transformer les dynamiques d'un système non linéaire en un système complètement ou partiellement linéaire, pour que les techniques de commande linéaire puissent être appliquées. Cette approche a été utilisée avec succès pour quelques problèmes de commande pratiques. Cela inclut la commande des hélicoptères, des avions à hautes performances, des robots industriels et du matériel médical [3]. L'idée de la linéarisation par retour d'état peut être appliquée facilement à une classe de systèmes non linéaires sous forme compagne ou canonique de contrôlabilité. Lorsque les dynamiques non linéaires ne sont pas représentées sous cette la forme, on doit utiliser des transformations pour mettre les dynamiques sous forme canonique avant d'utiliser la technique de linéarisation par retour d'état, ou considérer une linéarisation partielle des dynamiques originales, au lieu de la linéarisation complète.

Un système non linéaire est dit sous la forme canonique si ses dynamiques sont représentées par :

$$x^{(n)} = f(x) + b(x)u \quad y = x \quad (\text{II.1})$$

où $u \in R$ est l'entrée de commande scalaire, $x \in R$ est la sortie scalaire, $\underline{x} = (x_1, x_2, \dots, x_n)^T = (x, \dot{x}, \dots, x^{(n-1)})^T \in R^n$ est le vecteur d'état du système, et $f(x)$ et $b(x)$ sont des fonctions non linéaires des états. Notant que dans la représentation de l'espace d'état, l'équation (II.1) peut être écrite :

$$\frac{d}{dt} \begin{bmatrix} x_1 \\ \dots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} x_2 \\ \dots \\ x_n \\ f(\underline{x}) + b(\underline{x})u \end{bmatrix} \quad (\text{II.2})$$

Supposant que $b(x)$ est différent de zéro, alors de (II.2), on peut choisir la loi de commande suivante :

$$u \dot{=} \frac{1}{b(\underline{x})} \cdot (v - f(\underline{x})) \quad (\text{II.3})$$

Avec v étant une entrée équivalente à spécifier. Remplaçant par (II.3) dans (II.1), on obtient :

$$x^{(n)} = v \quad (\text{II.4})$$

De cette façon, les non linéarités sont éliminées et une simple relation entrée-sortie (II.4) est obtenue. L'étape suivante consiste à choisir la nouvelle commande v sous une forme qui doit assurer des dynamiques du système exponentiellement stables. Alors,

$$v = -k_0 \cdot x - k_1 \cdot \dot{x} - \dots - k_{n-1} \cdot x^{(n-1)} \quad (\text{II.5})$$

avec les k_i choisis tels que les racines du polynôme $p^n + k_{n-1} \cdot p^{n-1} + \dots + k_0 = 0$ sont à partie réelle négative menant à des dynamiques exponentiellement stables :

$$x^{(n)} + k_{n-1} \cdot x^{(n-1)} + \dots + k_0 \cdot x = 0 \quad (\text{II.6})$$

Cela signifie que $x(t) \rightarrow 0$. Pour les tâches englobant la poursuite d'une sortie désirée $x_d(t)$, la loi de commande peut être choisie de la même manière que précédemment (sous forme d'un contrôleur à placement de pôles) :

$$v = x_d^{(n)} - k_0.e - k_1.\dot{e} - \dots - k_{n-1}.e^{(n-1)} \quad (\text{II.7})$$

$$\text{Où } e(t) = x(t) - x_d(t) \quad (\text{II.8})$$

représente l'erreur de poursuite, et donc le vecteur d'erreur est défini comme :

$$\underline{e} = (e, \dot{e}, \dots, e^{(n-1)})^T \in R^n \quad (\text{II.9})$$

Remplaçant (II.7) dans (II.3), puis l'équation résultante dans (II.1) et utilisant (II.8), cela mène à des dynamiques en boucle fermée exponentiellement stables :

$$e^{(n)} + k_{n-1}.e^{(n-1)} + \dots + k_0.e = 0 \quad (\text{II.10})$$

Ce qui implique que $\lim_{t \rightarrow \infty} e(t) \rightarrow 0$, et qui est l'objectif principal de la commande.

II.3 Linéarisation par retour d'état adaptative

Les applications de l'approche de la linéarisation par retour d'état sont limitées parce qu'elles reposent sur la connaissance complète des non linéarités et des paramètres du système. Pour remédier à cet inconvénient, une commande alternative est la linéarisation par retour d'état adaptative. Plusieurs schémas adaptatifs ont été introduits pour résoudre le problème des incertitudes paramétriques [40]. Cependant, la linéarisation par retour d'état adaptative ne peut pas être appliquée à tout système qui peut être linéarisé par retour. La raison est que les dérivés élevés des estimés des paramètres vont apparaître dans la loi de commande pour les systèmes d'ordre élevé. De plus, les lois de commande non linéaires augmentent la complexité et la difficulté générale des applications en temps réel. Tout cela a forcé les chercheurs à trouver d'autres méthodes applicables. La commande adaptative utilisant des approximateurs de fonctions en ligne pour les systèmes linéarisables par retour d'état s'est avérée très efficace. Les approximateurs de fonctions les plus couramment utilisés dans la commande adaptative sont les structures de la logique floue et les réseaux de neurones artificiels.

En effet, il a été prouvé que les réseaux de neurones artificiels et la logique floue peuvent approximer un domaine vaste de fonctions non linéaires à n'importe quel degré de précision sous certaines conditions [41].

Dans ce qui suit, nous donnons des notions sur les structures des réseaux de neurones et de la logique floue, et puis on montre comment ces deux structures peuvent être utilisées comme approximateurs dans le schéma de linéarisation par retour d'état adaptatif.

II.4 Les approximateurs universels

II.4.1 Approximation des fonctions par les réseaux RBF

Dans la théorie de la commande, le réseau de neurones est considéré comme un approximateur de fonctions qui émule la fonction donnée avec une petite tolérance d'erreur. Il a été prouvé [42] qu'une fonction continue peut être uniformément approximée par une combinaison linéaire de fonctions gaussiennes, et que les réseaux à fonction de base radiales RBF peuvent approximer n'importe quelle fonction continue à plusieurs variables dans un domaine compact, pour un nombre suffisant de neurones [43]. Les réseaux à fonctions de base radiales sont utilisés dans la classification, l'approximation et la reconnaissance de parole. Leur but est d'approximer un comportement désiré par une collection de fonctions, appelées noyaux ou fonctions de base radiales caractérisées par un centre c_i et un champ récepteur σ . Les réseaux à fonctions de base radiales ont été étudiés et appliqués dans différentes disciplines [19].

II.4.1.a La structure des réseaux RBF

Le réseau RBF est un réseau à deux couches dans lesquelles la couche cachée accomplit une transformation non linéaire fixe sur l'espace d'entrée et le transforme en un espace intermédiaire et la couche de sortie combine linéairement les sorties des couches intermédiaires. La structure d'un réseau de neurones à fonctions de base (noeuds) radiales est montrée dans la figure II.1. Plus spécifiquement, le réseau RBF accomplit la transformation suivante :

$$F : x \in R^n \rightarrow R : , \quad F(\underline{x}, \underline{\theta}) = \sum_{i=1}^{nr} \xi_i \theta_i = \underline{\theta}^T \underline{\xi}(\underline{x}), \quad \xi_i = \psi(\|\underline{x} - c_i\|_2) \quad (\text{II.11})$$

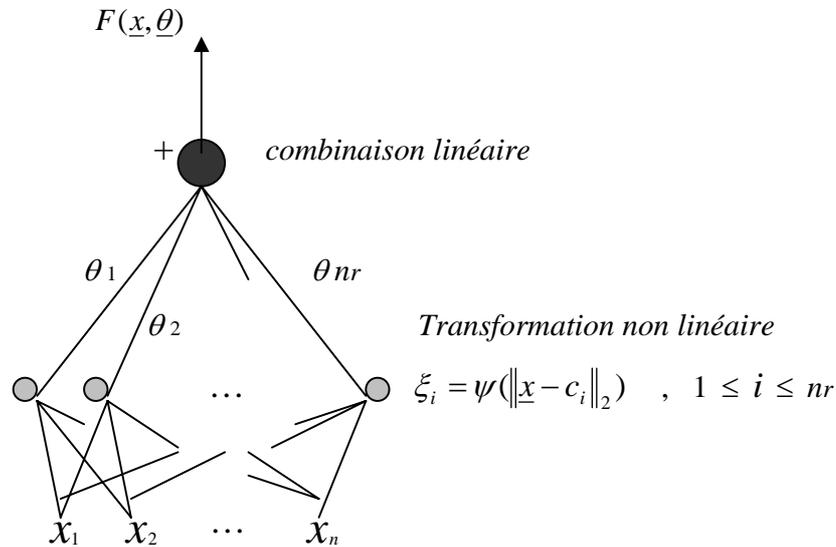


Figure II.1. Réseau de neurones à fonctions de base radiales

\underline{x} est le vecteur d'entrée, ψ est une fonction non linéaire appelée fonction de base radiale, θ_i sont les poids des connections (paramètres) entre la couche cachée et la couche de sortie, c_i sont les centres des fonctions de base, nr est le nombre de fonctions de base. La fonction de base la plus utilisée est la fonction gaussienne. Il est montré dans [42, 45] que la fonction de base gaussienne possède la meilleure propriété d'approximation. Cette fonction est donnée par :

$$\psi(r) = \exp\left(\frac{-r^2}{2\sigma^2}\right) \quad (\text{II.12})$$

Avec $r = \|\underline{x} - c_i\|_2$, c_i sont les centres de $\psi(r)$, σ est une constante associée à la fonction gaussienne $\psi(r)$ et représente sa largeur. Alors, pour approximer un comportement donné, les fonctions de base sont assemblées pour couvrir de leurs champs récepteurs l'ensemble des données d'entrées. Ces fonctions sont ensuite pondérées, et leurs valeurs sont sommées pour produire une valeur de sortie.

II.4.1.b Entraînement et placement des centres dans un réseau RBF

L'apprentissage des RBF consiste à déterminer les poids des connexions et les paramètres des fonctions de base. Comme les sorties dépendent linéairement des poids de connexions, la détermination de ces derniers est simplement un problème d'optimisation linéaire [44]. D'un autre côté, la plus part des algorithmes d'apprentissage des réseaux RBF procèdent à une sélection aléatoire d'un certain nombre d'échantillons de l'espace d'entrée comme étant les centres des fonctions de base. Cependant, dans la plus part des cas, il est évident qu'une sélection aléatoire des centres n'est pas satisfaisante. Pour remédier à cet inconvénient, d'autres algorithmes ont été utilisés dans lesquels la procédure d'apprentissage est divisée en deux étapes : un entraînement non supervisé pour l'ajustement des centres des fonctions de base dans la couche cachée, suivi par un entraînement supervisé pour l'ajustement des poids des connexions entre la couche cachée et la couche de sortie. L'algorithme des (*k-means*) [38, 39] et la méthode des moindres carrés récursifs (ou la règle Delta instantanée) sont souvent utilisés pour l'ajustement des centres et l'adaptation des poids des connexions respectivement.

i Ajustement des centres

L'algorithme des (*k-means*) [38, 39] est une méthode d'entraînement non supervisée pour le classement des données. La version la plus couramment utilisée est la classification par *k-means* adaptatif basé sur la distance euclidienne [38], [46]. Elle consiste à diviser l'espace d'entrée en *k* classes comme suit :

- 1- choisir un nombre *k* de classes (*k* fonctions de base dans notre cas).
- 2- Initialiser les centres des fonctions de base.
- 3- Calculer la distance euclidienne entre les centres de chaque fonction de base et le vecteur d'entrée \underline{x} , i.e. $dist(i) = \|\underline{x} - c_i\|_2$, $i=1$ to nr (II.13)

Puis, ajuster le vecteur des centres c_i qui correspond à la distance minimale $dist(j) = \min\|\underline{x} - c_i\|_2$ en utilisant la loi d'adaptation suivante [38] :

$$c_j(t) = c_j(t-1) + \delta(t) \cdot (\underline{x}(t) - c_j(t-1)) \quad (II.14)$$

où j est l'indice du vecteur des centres le plus proche du vecteur d'entrées $\underline{x}(t)$ (en d'autre terme, j est l'indice de la fonction de base qui correspond à la distance euclidienne minimale) et $c_j(t-1)$ représente le centre à l'étape précédente ($t-1$). Le taux d'adaptation $\delta(t)$ est un gain appartenant à l'intervalle $[0 \ 1]$, et peut être sélectionné par plusieurs méthodes. Chen et al [46] utilisent un taux d'adaptation qui est ajusté à chaque pas et qui tend vers zéro que $t \rightarrow \infty$ selon la loi suivante :

$$\delta(k) = \frac{\delta(t-1)}{\sqrt{1 + \text{int}(\frac{t}{nr})}} \quad (\text{II.15})$$

Où int est la partie entière de $(\frac{t}{nr})$

ii La règle d'adaptation instantanée [44]

La règle d'apprentissage instantanée est donnée par la formule suivante :

$$\underline{\theta}(t) = \underline{\theta}(t-1) + \gamma \cdot \xi \cdot (y_d(t) - \xi^T \cdot \underline{\theta}(t-1)) \quad (\text{II.16})$$

avec $\underline{\theta}$ le vecteur des paramètres (poids du réseau) à adapter, ξ le vecteur qui contient les sorties ξ_i des fonctions de base du réseau, ξ^T est son vecteur transpose, γ est le taux d'apprentissage, y_d est la sortie désirée du réseau. Cette règle est appelée la descente du gradient instantanée.

II.4.2 Approximation des fonctions par les systèmes d'inférence flous [41], [47]

Les systèmes d'inférence flous sont construits à partir des règles floues du type IF-THEN en utilisant des inférences floues spécifiques et des stratégies de fuzzification et de defuzzification, ainsi les informations linguistiques des experts humains (sous forme de règles floues du type IF-THEN) peuvent être directement incorporées dans ces systèmes [20], [41]. Il est montré [20, 21, 22] qu'un système flou est capable d'approximer n'importe quelle fonction non linéaire lisse autour d'une région compacte convexe. Un système d'inférence flou ou encore modèle de raisonnement approximatif flou comprend donc un ensemble de règles qui associent les variables linguistiques décrivant les entrées (cause) aux variables linguistiques de sortie (conséquence).

Plusieurs modèles de raisonnement ont été développés. Les plus courants sont : le modèle dit de Mamdani-Zadeh et le modèle dit de Takagi-Sugeno.

II.4.2.a Modèle de Mamdani-Zadeh [20]

Le modèle de raisonnement approximatif flou de Mamdani-Zadeh (MZ) est composé de règles de la forme :

R^i : *if x_1 is A_1^i and ... x_n is A_n^i then y is B^i*

$x_1 \dots x_n$ sont les variables linguistiques d'entrée, y est la variable linguistique de sortie, A_j^i et B^i sont des ensembles flous (ou des valeurs linguistiques) de la règle R^i .

II.4.2.b Modèle de Takagi-Sugeno [41]

Le modèle de raisonnement approximatif flou de Takagi-Sugeno (TS) est composé de règles de la forme :

R^i : *if x_1 is A_1^i and ... x_n is A_n^i then $y = a_0^i + a_1^i x_1 + \dots + a_n^i x_n$*

$x_1 \dots x_n$ sont les variables linguistiques d'entrée, y est la variable de sortie, A_j^i sont des ensembles flous (ou des valeurs linguistiques) et $a_0^i, a_1^i, \dots, a_n^i$ sont les paramètres ajustables des conséquences de la règle R^i .

Chapitre 3:

Commande adaptative floue et neuronale des systèmes non linéaires affines avec gain de commande constant

III.1 Introduction

Dans ce chapitre, nous proposons deux schémas de commande adaptative directe pour une classe des systèmes non linéaires. L'architecture est basée sur les approximateurs universels [41]: les systèmes à logique floue et les réseaux de neurones artificiels pour approximer une loi de commande à linéarisation par retour d'état [3], [5, 6]. Le premier schéma de commande utilise comme contrôleur un système d'inférence flou, (Fuzzy Inference System: FIS) du type Takagi-Sugeno (TS) et le deuxième schéma de commande utilise un système de réseau de neurone à fonctions de base radiales (Radial Basis Function: RBF). Les paramètres des conséquences du système flou TS et les poids de connections entre la couche cachée et la couche de sortie du réseau RBF sont adaptés selon des lois dérivées en utilisant la théorie de stabilité de Lyapunov. Dans la plus part des cas de commande adaptative basée sur les approximateurs, un terme complémentaire appelé superviseur est superposé à la sortie du contrôleur pour garantir la stabilité globale. Lorsque le système fonctionne dans un intervalle prédéfini entourant le point de fonctionnement le contrôleur superviseur n'est pas activé, et il le sera seulement lorsque le système est hors de cet intervalle. Dans ce travail, la stabilité asymptotique du système résultant en boucle fermée est garantie sans utilisation du terme superviseur dans la loi de commande.

Ce chapitre est organisé comme suit: dans la section III.2, la formulation du problème est introduite, dans la section III.3, la commande adaptative basée sur un système flou est présentée tandis que la section III.4 décrit la commande adaptative basée sur un réseau RBF. Dans les deux cas l'analyse de la stabilité est développée, les lois d'adaptation sont dérivées et des tests de simulations sont présentés.

III.2 Formulation du problème

III.2.1 Le modèle

Considérons le système non linéaire donné par (II.1) et qui peut être affecté par des perturbations extérieures:

$$\dot{x}^{(n)} = f(x, \dot{x}, \dots, x^{(n-1)}) + b(x, \dot{x}, \dots, x^{(n-1)})u + d, \quad y = x \quad (\text{III.1})$$

où, $u \in R$ et $y \in R$ sont l'entrée et la sortie du système respectivement, $f(\underline{x})$ et $b(\underline{x})$ sont des fonctions non linéaires inconnues et d est une perturbation extérieure. On assume que le vecteur d'état $\underline{x} = (x_1, x_2, \dots, x_n)^T = (x, \dot{x}, \dots, x^{(n-1)})^T \in R^n$ est mesurable. On définit le vecteur d'erreur :

$$\underline{e} = (e, \dot{e}, \dots, e^{(n-1)})^T \in R^n \quad (\text{III.2})$$

L'objectif de la commande est de forcer la sortie $y(t)$ à suivre un signal de référence $y_d(t)$ donné et borné, sous les contraintes que tous les signaux fournis doivent être bornés, ou telles que les conditions suivantes sont vérifiées.

1) Le système en boucle fermé doit être au moins asymptotiquement stable dans le sens que toutes les variables doivent être uniformément bornées.

2) L'erreur de poursuite, $e = y - y_d = x - x_d$, (parce que de III.1, on a : $y = x$) doit être la plus petite possible sous les contraintes dans 1).

Dans ce chapitre, nous considérons le cas particulier où le gain de commande virtuel est une constante : $b(\underline{x}) = b$. Le cas où $b(\underline{x})$ n'est pas une constante sera détaillé dans le chapitre IV.

III.2.2 La loi de commande idéale

Nous choisissons une commande u basée sur la linéarisation par retour d'état pour annuler les non linéarités dans le système non linéaire (III.1) afin d'obtenir une forme linéaire des dynamiques du système en boucle fermée [3], [5, 6]. Alors, si $f(\underline{x})$ et b sont connues et la

perturbation extérieure d n'existe pas, et supposant que b est non nulle, alors de (III.1), la loi de commande est :

$$u^{\bullet} = \frac{1}{b} \cdot (v - f(\underline{x})) \quad (\text{III.3})$$

Avec v étant une entrée équivalente à spécifier. Remplaçant (III.3) dans (III.1), on peut éliminer les non linéarités et obtenir la relation simple :

$$\begin{aligned} x^{(n)} &= f(\underline{x}) + b \left(\frac{1}{b} (v - f(\underline{x})) \right) \\ x^{(n)} &= f(\underline{x}) + v - f(\underline{x}) \\ x^{(n)} &= v \end{aligned} \quad (\text{III.4})$$

La nouvelle commande v est ensuite choisie sous une forme qui doit assurer des dynamiques du système exponentiellement stables. Alors,

$$v = x_d^{(n)} - K^T \cdot \underline{e} \quad (\text{III.5})$$

$$\text{Où } e(t) = x(t) - x_d(t) = y(t) - y_d(t) \quad (\text{III.6})$$

représente l'erreur de poursuite.

Remplaçant (III.5) dans (III.3), alors la loi de commande devient :

$$u^{\bullet} = \frac{1}{b} \cdot (x_d^{(n)} - K^T \cdot \underline{e} - f(\underline{x})) \quad (\text{III.7})$$

Remplaçant maintenant (III.7) dans (III.1) et utilisant (III.6), on obtient :

$$e^{(n)} + K^T \cdot \underline{e} = 0 \quad (\text{III.8})$$

Si on choisit le vecteur :

$$K = (k_0, k_1, \dots, k_{n-1})^T \in R^n \quad (\text{III.9})$$

telle que toutes les racines du polynôme :

$$p^n + k_{n-1} \cdot p^{n-1} + \dots + k_0 = 0 \quad (\text{III.10})$$

Sont à partie réelle négative, cela mène à des dynamiques en boucle fermée exponentiellement stables, c'est à dire: $\lim_{t \rightarrow \infty} e(t) \rightarrow 0$ qui est l'objectif principal de la commande.

Cependant lorsque $f(\underline{x})$ et b sont inconnues et la perturbation extérieure d existe, la commande dans (III.7) ne peut pas être implémentée. La solution la plus courante pour résoudre ce problème est d'utiliser des approximateurs universels pour approximer cette commande dans un schéma

directe ou indirect (cf. chapitre II). Dans la suite de ce chapitre, nous développons une commande adaptative directe du système (III.1) avec une approximation par FIS et par RBF.

III.3 Commande adaptative floue directe

Dans cette section, nous proposons d'approximer la commande u^* de (III.7) par un système flou du type TS adapté en ligne. Les paramètres du contrôleur adaptatif TS sont changés selon une loi dérivée en utilisant la théorie de stabilité de Lyapunov. La stabilité asymptotique est établie telle que l'erreur de poursuite converge vers de l'origine. Cela est assuré sans l'utilisation du terme de supervision dans la loi de commande.

III.3.1 Analyse de la stabilité et lois d'adaptation du contrôleur TS

Un système d'inférence flou TS avec des conséquences linéaires est composé de règles de la forme :

$$R^i : \text{if } x_1 \text{ is } A_1^i \text{ and } \dots x_n \text{ is } A_n^i \text{ then } u_i = a_0^i + a_1^i x_1 + \dots + a_n^i x_n$$

A_j^i sont les ensembles flous. Si on prend $\theta_i^T = [a_0^i \ a_1^i \dots a_n^i]$ comme vecteur de paramètres ajustables des conséquences de la règle R^i , la sortie du système flou TS peut être mise sous la forme suivante :

$$u(x) = \underline{\theta}^T \cdot \underline{\xi}(x) \tag{III.11}$$

Où $\underline{\xi}(x)$ est le vecteur des fonctions de base floues.

Il a été prouvé que la loi (III.11) peut approximer autour d'un ensemble compact Ω_z , n'importe quelle fonction lisse à un degré de précision donné [20, 21, 22], ce qui permet d'utiliser ce système pour approximer la loi de commande u^* de (III.7).

Dans ce qui suit, nous dérivons la loi d'adaptation pour les paramètres du système TS garantissant la stabilité asymptotique du système en boucle fermée en utilisant l'approche de Lyapunov.

Pour obtenir l'équation de l'erreur en boucle fermée, nous remplaçons la commande u dans (III.1) par $u(x, \theta)$, ce qui donne :

$$x^{(n)} = f(x) + b.u(x, \theta) + d \tag{III.12}$$

Ajoutant et soustrayant $b.u^*$ à (III.12), on obtient :

$$\dot{x}^{(n)} = f(\underline{x}) + b.u(\underline{x}, \underline{\theta}) + d + b.u^* - b.u^* \quad (\text{III.13})$$

Remplaçant (III.7) dans (III.13), on aura :

$$\dot{x}^{(n)} = f(\underline{x}) + b.u(\underline{x}, \underline{\theta}) + d - b.u^* + x_d^{(n)} - K^T . \underline{e} - f(\underline{x}) \quad (\text{III.14})$$

De cette façon :

$$\dot{x}^{(n)} - x_d^{(n)} = -K^T . \underline{e} + b.(u(\underline{x}, \underline{\theta}) - u^*) + d \quad (\text{III.15})$$

Utilisant (III.2) et (III.6) on obtient l'équation de l'erreur :

$$\dot{\underline{e}} = A_c . \underline{e} + b_c [b.(u(\underline{x}, \underline{\theta}) - u^*) + d] \quad (\text{III.16})$$

avec

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ & & & & & \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -k_0 & -k_1 & -k_2 & \dots & -k_{n-2} & -k_{n-1} \end{bmatrix}, \quad b_c = \begin{bmatrix} 0 \\ 0 \\ . \\ 0 \\ 1 \end{bmatrix} \quad (\text{III.17})$$

L'étude de la stabilité du système se fera à partir de l'équation de l'erreur III.16.

Nous commençons par définir le vecteur de paramètres optimal $\underline{\theta}^*$ comme le vecteur qui correspond au meilleur (optimal) terme d'approximation $u(\underline{x}, \underline{\theta}^*)$ du signal de commande u^* de (III.7), pour cela, on définit l'erreur d'approximation minimale :

$$w = u(\underline{x}, \underline{\theta}^*) - u^*(\underline{x}). \quad (\text{III.18})$$

Alors, l'équation d'erreur (III.16) peut être écrite comme :

$$\dot{\underline{e}} = A_c . \underline{e} + b_c [b.(u(\underline{x}, \underline{\theta}) - u(\underline{x}, \underline{\theta}^*)) + b.w + d] \quad (\text{III.19})$$

En se servant maintenant de (III.11), on peut écrire :

$$u(\underline{x}, \underline{\theta}) = \underline{\theta}^T . \underline{\xi}(\underline{x}), \quad \text{et} \quad u(\underline{x}, \underline{\theta}^*) = \underline{\theta}^{*T} . \underline{\xi}(\underline{x}) \quad (\text{III.20})$$

Soit $\underline{\varphi} = \underline{\theta} - \underline{\theta}^*$ et en utilisant (III.20), alors (III.19) devient :

$$\dot{\underline{e}} = A_c . \underline{e} + b_c . b . \underline{\varphi}^T . \underline{\xi}(\underline{x}) + b_c . (b.w + d) \quad (\text{III.21})$$

Pour étudier la stabilité et dériver les lois d'adaptation, nous considérons la fonction de Lyapunov candidate suivante :

$$V = \frac{1}{2} \underline{e}^T . P . \underline{e} + \frac{b}{2\gamma} \varphi^T . \varphi \quad (\text{III.22})$$

Où γ est une constante positive et P est la solution de l'équation de Lyapunov :

$$A_c^T . P + P . A_c = -Q \quad \text{avec} \quad Q > 0 : \text{un paramètre de conception.} \quad (\text{III.23})$$

En dérivant V par rapport au temps, on obtient :

$$\dot{V} = \frac{1}{2} \dot{\underline{e}}^T . P . \underline{e} + \frac{1}{2} \underline{e}^T . P . \dot{\underline{e}} + \frac{b}{2\gamma} \dot{\varphi}^T . \varphi + \frac{b}{2\gamma} \varphi^T . \dot{\varphi} \quad (\text{III.24})$$

Utilisant (III.21) et (III.23), on aura :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \underline{e}^T P b_c b . \varphi^T \xi(\underline{x}) + \underline{e}^T P b_c (b . w + d) + \frac{b}{\gamma} \varphi^T \dot{\varphi} \quad (\text{III.25})$$

Soit P_n la dernière colonne de P , et utilisant (III.17) on obtient :

$$\underline{e}^T P b_c = \underline{e}^T P_n \quad (\text{III.26})$$

Remplaçant (III.26) dans (III.25), on a :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \frac{b}{\gamma} \varphi^T [\gamma . \underline{e}^T P_n \xi(\underline{x}) + \dot{\varphi}] + \underline{e}^T P b_c (b . w + d) \quad (\text{III.27})$$

Si on choisit la loi d'adaptation :

$$\dot{\underline{\theta}} = -\gamma . \underline{e}^T P_n \xi(\underline{x}) \quad (\text{III.28})$$

Cela nous donne :

$$\frac{b}{\gamma} \varphi^T (\gamma . \underline{e}^T P_n \xi(\underline{x}) + \dot{\varphi}) = 0 \quad (\text{III.29})$$

En considérant le fait que $\dot{\varphi} = \dot{\underline{\theta}} - \dot{\underline{\theta}}^* = \dot{\underline{\theta}}$, parce que le vecteur de paramètres optimal $\underline{\theta}^*$ est constant et il est évident que sa dérivée est zéro, c'est à dire $\dot{\underline{\theta}}^* = 0$, donc (III.27) devient :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \underline{e}^T P b_c (b . w + d) \quad (\text{III.30})$$

Ou bien :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} - \underline{e}^T P b_c . w_1 \quad (\text{III.31})$$

avec

$$w_1 = -(b . w + d) \quad (\text{III.32})$$

La loi d'adaptation obtenue pour le vecteur de paramètres $\underline{\theta}$ du système flou TS est :

$$\dot{\underline{\theta}} = -\gamma \cdot \underline{e}^T P_n \xi(\underline{x}) \quad (\text{III.33})$$

Le théorème suivant montre les propriétés du contrôleur TS adaptatif direct.

Théorème

Considérant le système non linéaire (III.1) avec la loi de commande $u = u(\underline{x}, \underline{\theta})$ donnée par (III.11), et la loi d'adaptation donnée par (III.33) pour le vecteur de paramètres $\underline{\theta}$, alors le système global est tel que :

i) L'erreur de poursuite $\underline{e}(t)$ converge à un ensemble Ω_e défini par :

$$\Omega_e = \left\{ \underline{e} : \|\underline{e}\| \leq \sqrt{\frac{\delta}{\mu}} \right\} \quad (\text{III.34})$$

Où δ et μ sont deux constantes positives.

ii) si w_1 dans (III.31) est sommable carrée (squared integrable), c'est-à-

$$\text{dire } \int_0^{\infty} \|w_1(t)\|^2 dt < \infty, \text{ alors } \lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$$

Démonstration

i) Soit $\lambda_{Q_{\min}}$ la valeur propre minimale de Q alors, de (III.31) on a

$$\dot{V} \leq -\frac{1}{2} \lambda_{Q_{\min}} \|\underline{e}\|^2 - \underline{e}^T P b_c w_1 \quad (\text{III.35})$$

Qui peut être écrite ainsi :

$$\begin{aligned} \dot{V} &\leq -\frac{1}{2} \lambda_{Q_{\min}} \|\underline{e}\|^2 - \underline{e}^T P b_c w_1 + \frac{1}{2} \|\underline{e}\|^2 - \frac{1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P b_c w_1\|^2 - \frac{1}{2} \|P b_c w_1\|^2 \\ &= -\frac{\lambda_{Q_{\min}} - 1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P b_c w_1\|^2 - \frac{1}{2} \|(e + P b_c w_1)\|^2 \end{aligned} \quad (\text{III.36})$$

Qui peut être simplifiée :

$$\dot{V} \leq -\frac{\lambda_{Q_{\min}} - 1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|Pb_c w_1\|^2 \quad (\text{III.37})$$

Parce que le terme $\frac{1}{2} \|(\underline{e} + Pb_c w_1)\|^2$ est supérieur ou égale à 0.

Si on choisit Q telle que $\lambda_{Q_{\min}} > 1$, alors :

$$\dot{V} \leq -\mu \|\underline{e}\|^2 + \delta \quad (\text{III.38})$$

$$\text{Où } \mu = \frac{\lambda_{Q_{\min}} - 1}{2}, \text{ et } \delta = \frac{1}{2} \|Pb_c w_1\|^2 \quad (\text{III.39})$$

De (III.38), on peut conclure que : $\dot{V} < 0$ si $\|\underline{e}\| > \sqrt{\frac{\delta}{\mu}}$. On définit l'ensemble :

$$\Omega_e = \left\{ \underline{e} : \|\underline{e}\| \leq \sqrt{\frac{\delta}{\mu}} \right\} \quad (\text{III.40})$$

De (III.32), on a : $w_1 = -(b.w + d)$, où l'erreur d'approximation minimale w dans (III.18) peut être construite arbitrairement petite par utilisation d'éléments d'approximation flous appropriés [20, 21, 22], [41]. La perturbation d étant supposée bornée, la quantité w_1 est bornée et donc de (III.39) δ est bornée, ce qui implique que l'ensemble Ω_e dans (III.40) est borné. Maintenant, \dot{V} est négative tant que $\underline{e}(t)$ est en dehors de l'ensemble Ω_e , selon la théorie de stabilité de Lyapunov, nous concluons que l'erreur $\underline{e}(t)$ est bornée et va converger vers Ω_e . Comme premier résultat, on peut voir que l'équation (III.40) correspond à l'équation (III.34)

ii) On utilise le lemme de Barbalat suivant pour prouver la partie *ii)* du théorème.

Lemme de Barbalat [3], [20], [48] :

Si $\underline{e}(t) \in L_2$ (sommable carrée, c'est-à-dire, $\int_0^{\infty} \|\underline{e}(t)\|^2 dt < \infty$),

Et $\underline{e}(t) \in L_{\infty}$ (bornée), et $\dot{\underline{e}}(t) \in L_{\infty}$ (bornée), alors $\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$.

Intégrant les deux côtés de (III.37), on obtient :

$$[V(t) - V(0)] \leq -\frac{\lambda_{Q_{\min}} - 1}{2} \int_0^t \|\underline{e}(\tau)\|^2 d\tau + \frac{1}{2} \|Pb_c\|^2 \int_0^t \|w_1(\tau)\|^2 d\tau \quad (\text{III.41})$$

Donc :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{2}{\lambda_{Q_{\min}} - 1} [V(0) - V(t)] + \frac{1}{\lambda_{Q_{\min}} - 1} \|Pb_c\|^2 \cdot \int_0^t \|w_1(\tau)\|^2 d\tau \quad (\text{III.42})$$

Cela nous donne :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{1}{\lambda_{Q_{\min}} - 1} [2(\|V(0)\| + \|V(t)\|) + \|Pb_c\|^2 \cdot \int_0^t \|w_1(\tau)\|^2 d\tau] \quad (\text{III.43})$$

Comme montré par [20], cela implique que si $w_1 \in L_2$ (c'est à dire, sommable carrée), alors de (III.43) $\underline{e}(t) \in L_2$, et en se basant sur la conclusion ci-dessus (conclusion après preuve de la première partie du théorème), selon la théorie de stabilité de Lyapunov, $\underline{e}(t)$ est bornée ($\underline{e}(t) \in L_\infty$). D'un autre côté de (III.21) $\dot{\underline{e}}(t) \in L_\infty$ (bornée), parce que tous les éléments de son côté droit sont bornés. Utilisant le Lemme de Barbalat mentionné ci-dessus, on conclut que $\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$.

Remarque 1

Dans le développement ci-dessus, les résultats de la stabilité asymptotique sont fournis en utilisant la théorie de Lyapunov sans utilisation du terme superviseur.

Remarque 2

L'application de l'algorithme de commande adaptative proposé est résumée dans les étapes suivantes :

Etape 1: Calcul hors ligne

- Spécifier les paramètres: k_0, \dots, k_{n-1} , telles que toutes les racines du polynôme $p^n + k_{n-1} \cdot p^{n-1} + \dots + k_0 = 0$, soient dans le demi- plan gauche complexe.
- Spécifier une matrice $Q (n \times n)$ définie positive.
- Résoudre l'équation de Lyapunov (III.23) pour obtenir une matrice symétrique $P > 0$.
- Spécifier les paramètres des prémisses du contrôleur flou TS : les centres, les largeurs et les formes de chaque fonction de base (fonctions d'appartenance) pour chaque entrée.
- Initialiser

Etape 2: Adaptation en ligne

- Appliquer la loi de commande (III.11) au système (III.1), où u dans (III.11) est la sortie du système contrôleur flou TS.
- Utiliser la loi d'adaptation (III.28) pour ajuster le vecteur des paramètres $\underline{\theta}$ du contrôleur TS.

III.3.2 Résultats de simulation:

III.3.2.1 Exemple 1

Dans cet exemple, nous appliquons en simulation la commande adaptative floue directe proposée pour contrôler le niveau dans le système à trois bacs (Three-Tank System). Les résultats seront comparés avec ceux obtenus par l'application d'un régulateur proportionnel intégral PI. Le système à trois bacs illustré par la figure III.1, est composé de trois cylindres (bacs) T1, T2 et T3 avec une surface de section transversale identique A interconnectés en série. Le liquide quittant T2 est collecté dans un réservoir à partir duquel les pompes 1 et 2 (actionnées par deux moteurs à courant continu) alimentent les réservoirs T1 et T2 par des débits Q_1 et Q_2 . Les réservoirs sont couplés par deux conduites cylindriques de section transversale S et un coefficient d'écoulement $\mu_1 = \mu_3$. L'écoulement nominal situé au réservoir T2, a une section transversale circulaire S et un coefficient d'écoulement μ_2 . Dans cet exemple, on considère le système à trois bacs comme un système monovariante : on s'intéressera au contrôle du niveau L_2 dans le réservoir T2 par le débit Q_2 .

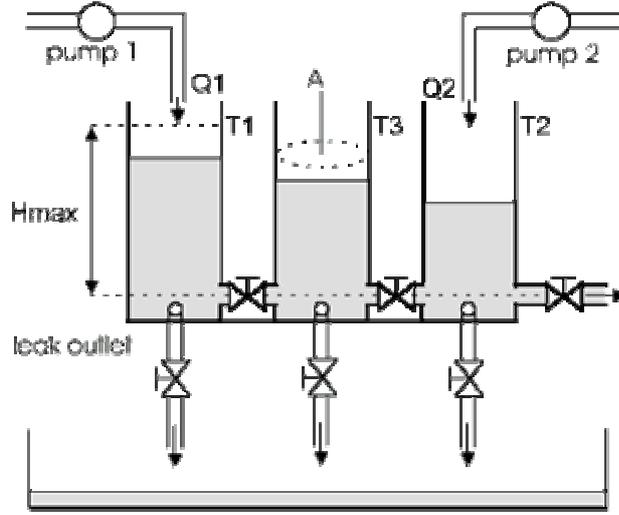


Figure III.1: La structure du système à trois bacs

L'équation dynamique décrivant le système à trois bacs est :

$$A \frac{dL_2}{dt} = Q_2 - \mu_2 S \sqrt{2gL_2} \quad (\text{III.44})$$

où $S = 0.5 (cm^2)$, $\mu_2 = \mu_3 = 0.4896$, $A = 154 (cm^2)$, $g = 9.81 * 100 (cm/s^2)$ est la gravitation universelle et $y = x = L_2$ est le niveau dans le bac (réservoir) T2. Le signal de référence est $x_d = y_d = L_{d2}$. Ce système est simulé avec un pas d'échantillonnage $dt = 1$.

Les paramètres de conception sont $\gamma = 0.008$, $k = k_0 = 1.5$ (pour avoir toutes les racines du polynôme $p + k_0 = 0$ dans le demi-plan gauche complexe), $Q = 2 > 0$, où la valeur propre minimale de Q : $\lambda_{Q_{\min}} = 2 > 1$ (pour satisfaire la transition de (III.37) à (III.40) dans la preuve du théorème). Par résolution de (III.23), nous obtenons $P = 0.6667$.

Le contrôleur flou TS a trois entrées $\underline{z} = [z_1 \ z_2 \ z_3] = [L_2 \ L_{d2} \ (\dot{L}_{d2} - k^T \underline{e})]$ avec $\underline{e} = [L_2 - L_{d2}]$. Chacune des trois entrées possède deux ensembles flous et des fonctions d'appartenance gaussiennes données par :

$$\mu_N(x) = \exp(-(x - c_N)^2 / 2.\sigma_N) \quad (\text{III.45})$$

$$\mu_P(x) = \exp(-(x - c_P)^2 / 2.\sigma_P) \quad (\text{III.46})$$

Où z_i est l'entrée i .

Les paramètres du FIS sont choisis comme suit :

$\sigma_p = \sigma_N = 6$ pour la première et la deuxième entrée $z_1 = L_2$, $z_2 = L_{d2}$ et $\sigma_p = \sigma_N = 3.5$ pour la troisième entrée $z_3 = \dot{L}_{d2} - k^T \underline{e}$.

$c_N = 10$, $c_p = 20$ pour z_1 et z_2 , et $c_N = -1$, $c_p = 3$ pour z_3 .

Cela donne huit règles de la forme :

$$R^i : \text{if } z_1 \text{ is } A_1^i \text{ and } z_2 \text{ is } A_2^i \text{ and } z_3 \text{ is } A_3^i \text{ then } u_i = a_1^i z_1 + a_2^i z_2 + a_3^i z_3, \text{ avec } i = 1 \text{ à } 8 \quad (\text{III.47})$$

On a 24 paramètres à ajuster. Tous les paramètres (a_1^i , a_2^i et a_3^i) sont initialisés à zéro. La condition initiale est $L_2(0) = 0$.

Les paramètres du PI utilisés sont $k_p = 7.5$ (l'action proportionnelle) et $T_i = 6.5$ (l'action intégrale).

Le scénario de simulation est réalisé comme suit :

De $t=0$ à $t=900$ s la perturbation $d=0$.

De $t=900$ à $t=1200$ on introduit une perturbation d .

Cette perturbation est obtenue en fermant de 50 % la valve d'écoulement nominal du bac T2. En d'autre terme, dans le bac T2, la section transversale S de la valve de l'écoulement nominal va prendre la valeur $S = 0.5/2$ (cm^2) à $t = 900$ s au lieu de la valeur nominale $S = 0.5$ (cm^2). Alors, l'équation du système (III.44) peut être écrite ainsi :

$$A \frac{dL_2}{dt} = Q_2 - \mu_2(S/2)\sqrt{2gL_2} \quad (\text{III.48})$$

Qui peut être écrite:

$$A \frac{dL_2}{dt} = Q_2 - \mu_2 S \sqrt{2gL_2} + \mu_2(S/2)\sqrt{2gL_2} \quad (\text{III.49})$$

On compare maintenant la nouvelle équation du système perturbé (III.49) avec l'équation du système global dans (III.1), alors la perturbation extérieure d peut être exprimée comme :

$$d = \mu_2(S/2)\sqrt{2gL_2} \quad (\text{III.50})$$

la perturbation extérieure d est donc bornée.

Les résultats de simulation sont montrés dans les figures III.2 et III.3 où les résultats correspondants au contrôleur TS sont en pointillé et ceux du contrôleur PI sont en continu et le signal de référence est en trillé. La figure III.2 montre l'évolution du niveau L_2 dans le réservoir T2 et La figure III.3 montre les signaux de commande pour les deux contrôleurs. Sur la figure III.2, on peut voir que le niveau au réservoir T2 avec le contrôleur TS a atteint la référence rapidement, avec un temps de réponse inférieur à celui du contrôleur PI. Le dépassement à chaque changement de référence est inférieur à celui obtenu avec le contrôleur PI. On peut voir que les perturbations sont supprimées rapidement avec moins d'amplitude pour le contrôleur TS que pour le contrôleur PI. En d'autre terme, les perturbations sont supprimées au temps $t = 970s$ avec moins d'amplitude pour le contrôleur flou TS, et au temps $t = 1040s$ pour le contrôleur PI comme montré par la figure III.2.

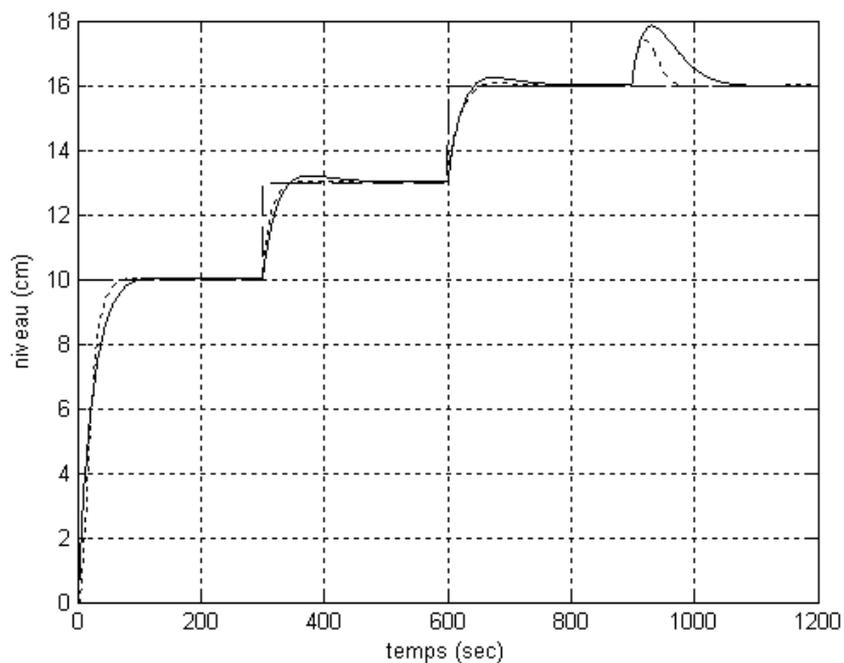


Figure III.2: L'évolution du niveau dans le réservoir T2 avec le contrôleur TS (.....) et avec le contrôleur PI en continu

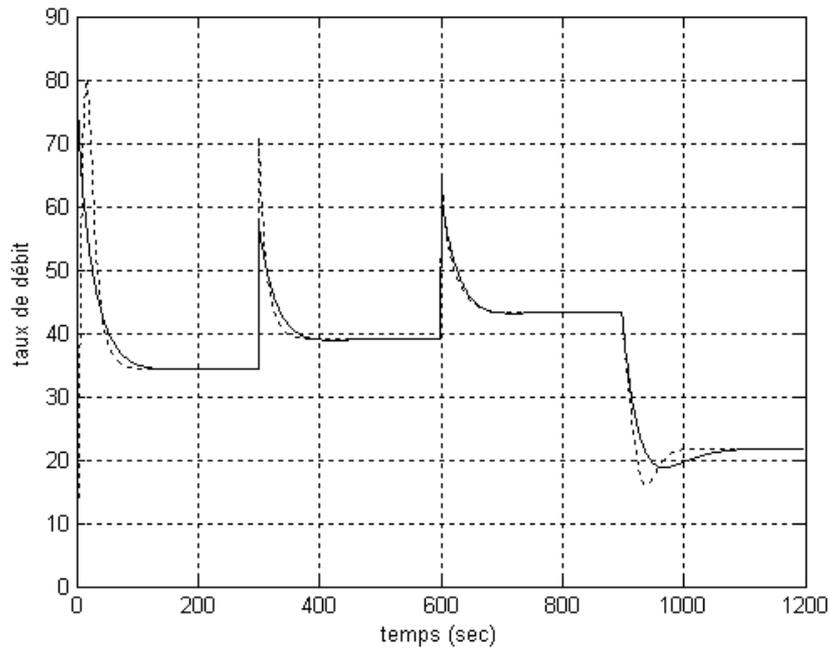


Figure III.3: L'évolution du signal de commande correspondant au niveau dans le réservoir T2 avec le contrôleur TS (.....) et avec le contrôleur PI en continu

III.3.2.2 Exemple 2

Dans cet exemple, on applique le contrôleur flou adaptatif direct pour réguler à l'origine la sortie d'un système instable [20], [49].

$$\dot{x}(t) = \frac{1 - e^{-x(t)}}{1 + e^{-x(t)}} + u(t) \quad (\text{III.51})$$

$$y = x$$

De (III.51), si l'entrée $u(t) = 0$, on a : $\dot{x}(t) = \frac{1 - e^{-x(t)}}{1 + e^{-x(t)}} > 0$ pour $x(t) > 0$, et $\dot{x}(t) = \frac{1 - e^{-x(t)}}{1 + e^{-x(t)}} < 0$

pour $x(t) < 0$, le système (III.51) est donc instable.

Le signal de référence sera $y_d = 0$.

La condition initiale est $x(0) = 1$ et la période d'échantillonnage est $dt = 0.2$.

Les paramètres de conception sont choisis comme suit :

$\gamma = 0.8$, $k = k_0 = 7.5$ pour avoir toutes les racines de $p + k_0 = 0$ dans le demi- plan gauche complexe, $Q = 5 > 0$, $\lambda_{Q_{\min}} = 5 > 1$, et par résolution (III.23), on obtient $P = 0.3333$.

Le contrôleur flou (TS) possède trois entrées $\underline{z} = [z_1 \ z_2 \ z_3] = [x \ x_d \ (\dot{x}_d - k^T \underline{e})]$ avec $\underline{e} = x - x_d$. La structure du contrôleur TS est la même comme dans l'exemple précédent avec :

$\sigma_p = \sigma_N = 0.7$ pour la première et la deuxième entrée z_1, z_2

$\sigma_p = \sigma_N = 0.2$ pour la troisième entrée z_3 .

$c_N = -1$, $c_p = 1$ pour z_1, z_2 , et z_3 .

La figure III.4 montre l'état du système $x(t)$ et la position désirée $x_d(t)$. On peut voir de cette figure que le contrôleur flou adaptatif direct ramène rapidement la sortie du système à l'origine. La figure III.5 montre le signal de commande $u(t)$. Par comparaison aux résultats dans [20], [49], une bonne amélioration peut être observée en particulier sur le temps de réponse : 1.6 s dans notre système et 8 s dans [20] et [49].

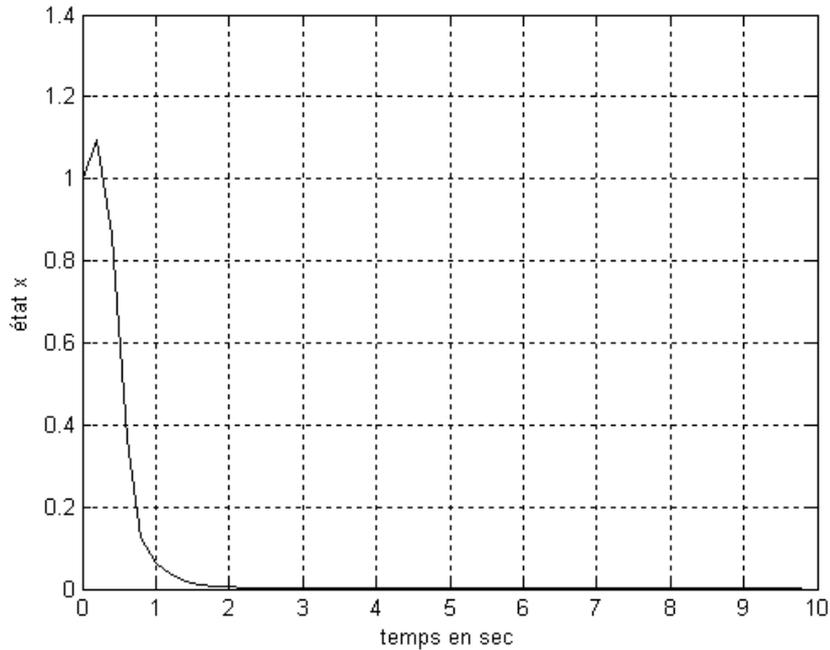


Figure III.4: l'état du système $x(t)$ et la position désirée $x_d(t)$

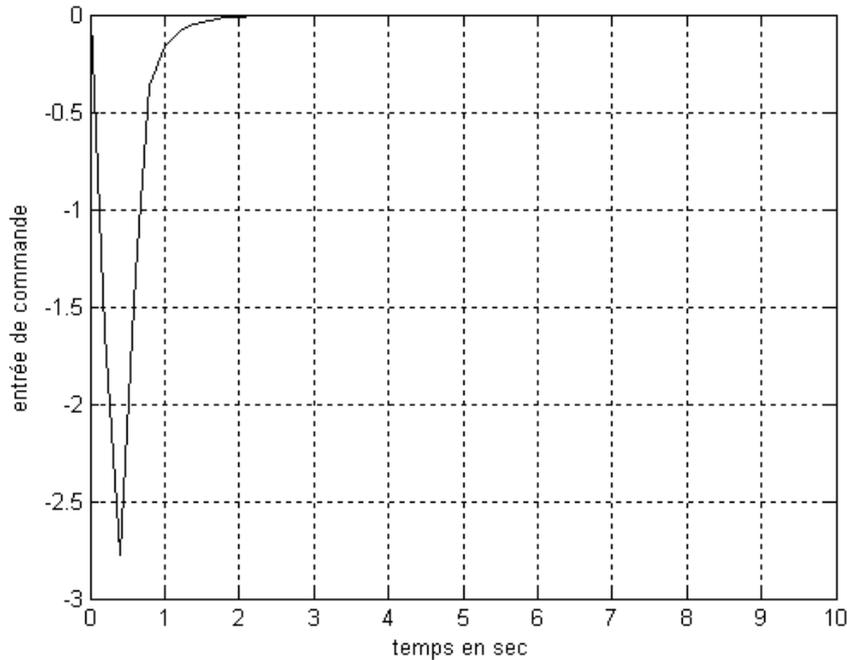


Figure III.5: le signal de commande correspondant $u(t)$

III.3.3 Conclusion

Dans cette partie, nous avons développé un contrôleur stable adaptatif direct pour une classe de systèmes non linéaires basé sur la théorie de linéarisation par retour d'état. Nous avons utilisé un système d'inférence flou du type TS pour approximer le signal de commande idéal. Les paramètres de conséquence du contrôleur flou utilisé sont adaptés selon une loi dérivée en utilisant la théorie de stabilité de Lyapunov. La méthode proposée a pu garantir la stabilité asymptotique du système en boucle fermée dans le sens que tous les signaux sont uniformément bornés sans l'utilisation du terme superviseur dans la loi de commande. Nous avons testé en simulation l'algorithme flou TS adaptatif direct pour contrôler le niveau dans le système à trois bacs et pour la régulation d'un système instable. Les résultats de simulation ont montré les bonnes performances et la robustesse de la commande proposée.

III.4 Commande adaptative neuronale directe

Dans cette deuxième approche, on propose comme alternative de la première approche un système de réseau de neurones du type RBF pour approximer la commande u^* dans (III.7), avec toujours $b(\underline{x}) = b$ (constante). Les paramètres du contrôleur RBF adaptatif sont adaptés selon une loi dérivée en utilisant la théorie de stabilité de Lyapunov et les centres du réseau RBF sont ajustés en utilisant l'algorithme des *k_means* [38, 39]. La stabilité asymptotique est établie avec la convergence de l'erreur de poursuite au voisinage de l'origine sans l'utilisation du terme de supervision dans la loi de commande. Les résultats de simulation sont présentés pour montrer les performances de la méthode RBF proposée.

III.4.1 Analyse de la stabilité et lois d'adaptation du contrôleur RBF

Le principe d'un réseau RBF est d'approximer un comportement désiré par une collection de fonctions, appelées fonctions de base radiales. Le réseau RBF accomplit la transformation suivante :

$$F: R^n \rightarrow R: \text{ avec : } F(\underline{x}, \underline{\theta}) = \sum_{i=1}^{nr} \xi_i \theta_i = \underline{\theta}^T \underline{\xi}(\underline{x}), \quad \xi_i = \psi(\|\underline{x} - c_i\|_2) \quad (\text{III.52})$$

\underline{x} est le vecteur d'entrée, ψ est la fonction de base radiale, θ_i sont les poids des connections entre la couche cachée et la couche de sortie et nr est le nombre des fonctions de base possédant la forme :

$$\psi(r) = \exp\left(\frac{-r^2}{2\sigma^2}\right) \quad (\text{III.53})$$

avec $r = \|\underline{x} - c_i\|_2$, c_i et σ sont respectivement le vecteur des centres et la largeur (champ récepteur) de la fonction de base gaussienne $\psi(r)$. Alors, pour approximer un comportement donné, les fonctions-noyau (fonctions de base) sont assemblées pour couvrir de leurs champs récepteurs l'ensemble des données d'entrées. Alors la sortie d'un réseau RBF peut être mise sous la forme suivante :

$$u(\underline{x}) = \underline{\theta}^T \underline{\xi}(\underline{x}) \quad (\text{III.54})$$

Où le vecteur $\underline{\theta}^T = [\theta_1 \theta_2 \dots \theta_n]$ contient les paramètres ajustables qui sont les poids de connections entre la couche cachée et la couche de sortie et $\underline{\xi}(x)$ est le vecteur des fonctions de base RBF. Il a été prouvé [43] que la loi (III.54) peut approximer autour d'un ensemble compact Ω_z , n'importe qu'elle fonction lisse à un degré de précision donné pour un nombre suffisant de neurones, ce qui permet d'utiliser ce réseau pour approximer la loi de commande u^* de (III.7). On peut voir que l'équation (III.54) qui représente la sortie d'un système RBF correspond à l'équation (III.11) donnant la sortie d'un système flou, et comme il a été prouvé dans la littérature des systèmes de commande que les deux systèmes (flou et RBF) sont des approximateurs universels [41], alors l'analyse de la stabilité et la dérivation de la loi de commande pour le système RBF restent les mêmes que pour le système flou déjà développé dans la section III.3.1, incluant le théorème et sa démonstration. L'adaptation des poids de connections du réseau RBF peut alors être exprimée par :

$$\dot{\underline{\theta}} = -\gamma \cdot \underline{e}^T P_n \underline{\xi}(x) \quad (III.55)$$

Où γ est une constante positive, P est la solution de l'équation de Lyapunov : donnée par (III.23), P_n est la dernière colonne de P , \underline{e} est le vecteur d'erreurs donné par (III.2) et $\underline{\xi}(x)$ est le vecteur des fonctions de base radiales qui remplace le vecteur des fonctions de base floues dans (III.11).

L'algorithme des *k_means* [38, 39] décrit dans le chapitre II est utilisé en ligne pour l'ajustement des centres du réseau RBF. La conception du contrôleur RBF adaptatif direct peut être résumée dans les étapes suivantes

Etape 1: Calculs hors ligne

- Définir le nombre de fonctions de base avec des centres uniformément repartis dans l'intervalle de variation de données.
- Spécifier les paramètres k_0, \dots, k_{n-1} tel que toutes les racines du polynôme $p^n + k_{n-1} \cdot p^{n-1} + \dots + k_1 \cdot p^1 + k_0 = 0$ soient à partie réelle négative.

- Spécifier une matrice Q définie positive et de rang $n \times n$, où n est le degré du système.
- Résoudre l'équation de Lyapunov (III.23) pour obtenir une matrice symétrique $P > 0$.

Etape 2: Adaptation en ligne

- Appliquer la commande (III.54) sur le système (III.1).
- Utiliser la loi d'adaptation (III.55) pour ajuster les poids des connections $\underline{\theta}$ de l'approximateur RBF ($u(\underline{x}, \underline{\theta}) = \underline{\theta}^T \cdot \xi(\underline{x})$) du signal de commande u^* de (III.7).
- Utiliser l'algorithme des *k-means* détaillé dans le chapitre II pour ajuster les centres des fonctions de base radiales.

III.4.2 Résultat de simulation

III.4.2.1 Exemple 1

Dans cet exemple, on applique en simulation la commande adaptative neuronale directe proposée pour contrôler le niveau dans le système à trois bacs (The Three Tank System) étudié ci-dessus et on compare les résultats de simulation avec ceux obtenus par l'application d'un régulateur proportionnel intégral PI. Le signal de référence est $x_d = y_d = L_{d2}$.

Les paramètres sont choisis comme suit :

$\gamma = 0.07$, le pas d'échantillonnage est $dt = 1$, $k_0 = 1$, $Q = 25 > 0$, et donc $\lambda_{Q_{\min}} = 25 > 1$, et par résolution de (III.23), nous obtenons $P = 12.5$.

Dans les applications de commande, le nombre de fonctions de base radiales est souvent choisi entre quatre et dix, nous choisissons cinq fonctions de base. Les paramètres $\underline{\theta}$ du contrôleur RBF sont initialisés aléatoirement dans l'intervalle $[0 \quad 1]$. Les centres des fonctions de base sont uniformément distribués dans l'intervalle $[aa \quad 14aa]$, où $aa = 1.13$ est une constante et $\sigma = 5$ pour toutes les fonctions de base.

Le vecteur d'entrée du réseau RBF est composé de deux entrées $\underline{x} = [e \quad de]$ avec $e = x - x_d$ est l'erreur, et de est la variation de l'erreur. La condition initiale est $x(0) = L_2(0) = 0 \text{ cm}$.

On considère le même scénario que dans la section précédente avec la même perturbation. Les résultats de simulation sont montrés dans les figures III.6 et III.7. Les résultats correspondants au contrôleur RBF sont en pointillé, et ceux du régulateur PI sont en continu et le signal de référence est en trillé. La figure III.6 montre l'évolution du niveau L_2 dans le réservoir T2 pour les deux contrôleurs (RBF et PI) et la figure III.7 montre les signaux de commande correspondants. Au début de l'intervalle de temps de la figure III.6, on peut voir que la réponse du système commandé par notre contrôleur RBF est plus rapide que celle du PI. Dans le deuxième et troisième intervalle de temps, le niveau qui correspond au contrôleur RBF évolue avec moins de dépassement que celui correspondant au contrôleur PI. On peut aussi remarquer que les perturbations sont supprimées plus rapidement et avec moins de dépassement que pour le contrôleur PI.

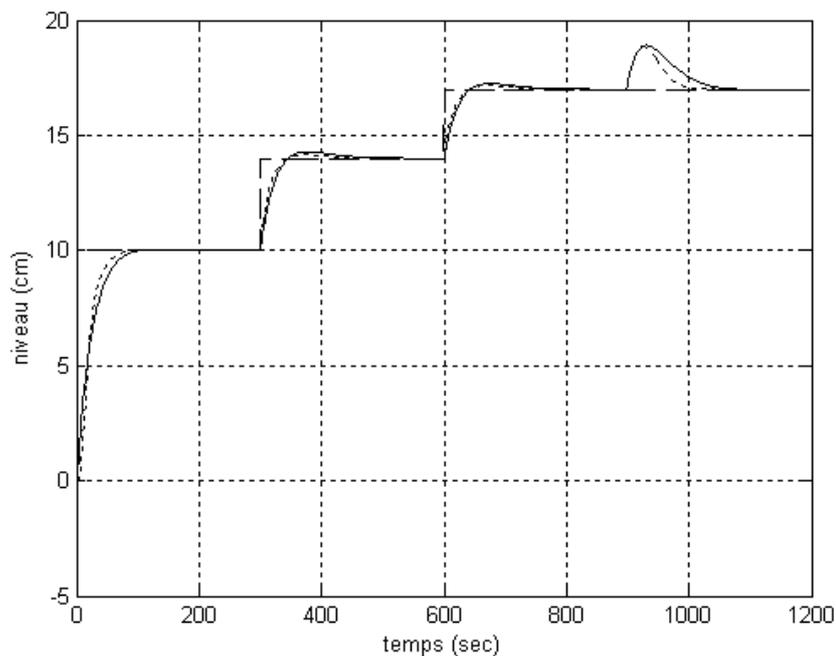


Figure III.6: L'évolution du niveau dans le réservoir T2 avec le contrôleur RBF (.....) et avec le contrôleur PI en continu

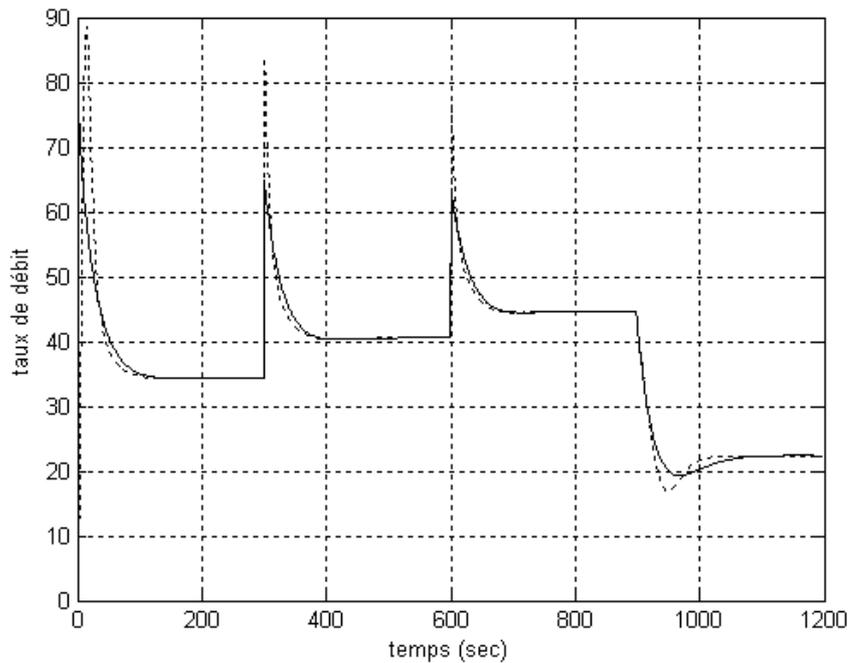


Figure III.7: L'évolution du signal de commande correspondant au niveau dans le réservoir T2 avec le contrôleur RBF (.....) et avec le contrôleur PI en continu

III.4.2.2 Exemple 2

Dans cet exemple, on a appliqué en simulation le contrôleur RBF adaptatif direct pour ramener à zéro la sortie du système non linéaire instable [20], [49] de la section précédente (exemple 2) décrit par l'équation (III.51). La condition initiale est $x(0) = 1$ et $dt = 0.2$.

Le contrôleur RBF a cinq fonctions de base radiales. Ses centres sont distribués uniformément dans l'intervalle $[-2 \ 2]$ et $\sigma = 0.7$ pour toutes les fonctions de base. On met $k_0 = 2.2$, $Q = 12$, et donc $\lambda_{Q_{\min}} = 12 > 1$ et par résolution de (III.23), nous obtenons $P = 2.7273$.

Le pas d'adaptation des poids de connections $\underline{\theta}$ est $\gamma = 2.2$. Les paramètres $\underline{\theta}$ sont initialisés à 0. Le vecteur d'entrée du réseau RBF est composé de deux entrées $\underline{x} = [e \ de]$ avec $e = x - x_d$ est

l'erreur, et de est la variation de l'erreur. La figure III.8 montre l'état du système $x(t)$ et la position désirée $x_d(t)$ et la figure III.9 montre le signal de commande correspondant $u(t)$.

A partir de ces figures, on peut voir que le contrôleur adaptatif direct RBF a pu réguler la sortie du système à l'origine. Il est bien clair que l'état et le signal de commande sont bornés. Par comparaison aux résultats dans [20], [49], une bonne amélioration sur notre système est observée, en particulier le temps de réponse (environ 2.3 s dans notre system et 8 s dans [20] et [49]).

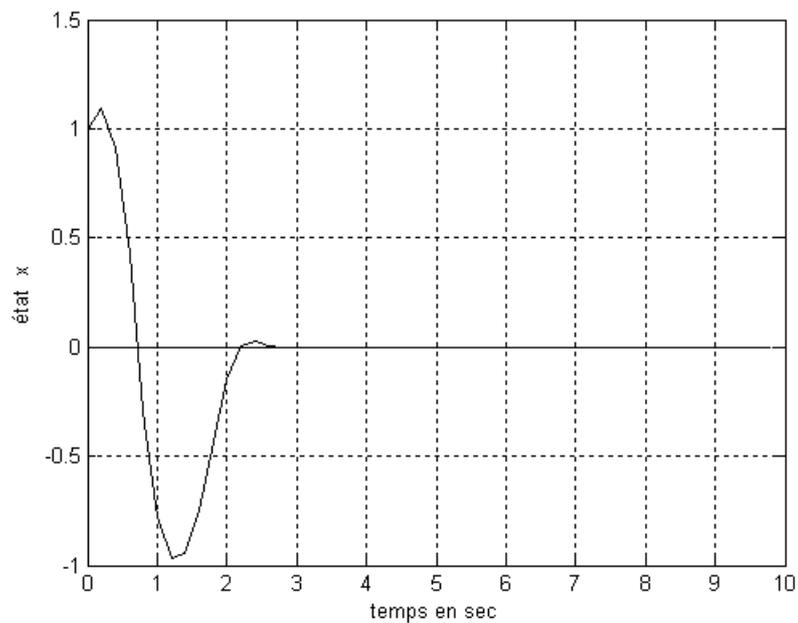


Figure III.8: l'état du système $x(t)$ et la position désirée $x_d(t)$

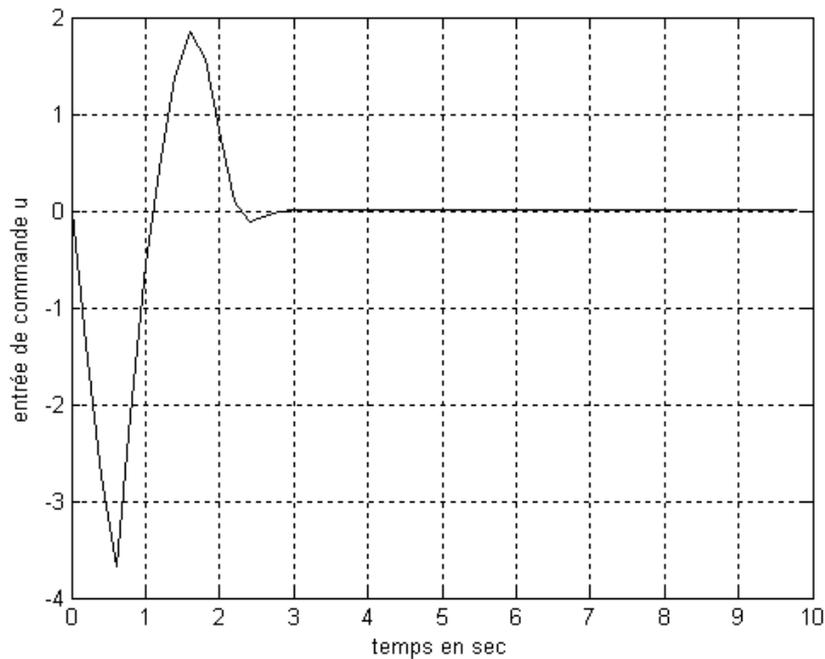


Figure III.9: le signal de commande correspondant $u(t)$

III.4.3 Conclusion

Dans cette section, nous avons proposé un schéma de commande pour une classe de systèmes non linéaires. On a utilisé pour cela un système de réseau de neurone RBF en ligne basé sur la linéarisation par retour d'état pour approximer le signal de commande idéal qui ne peut pas être implémenté. L'algorithme des *k-means* est utilisé pour l'ajustement des centres des fonctions de base radiales et les poids de connections sont adaptés selon une loi dérivée en utilisant la théorie de stabilité de Lyapunov pour garantir la stabilité asymptotique du système global résultant en boucle fermée sans l'utilisation du terme superviseur dans la loi de commande. Finalement, on a utilisé le contrôleur stable adaptatif direct RBF pour contrôler le niveau dans le système à trois bacs et un système non linéaire instable. Les résultats de simulation ont montré l'efficacité et la propriété de robustesse du système RBF proposé.

Chapitre 4:

Commande adaptative floue et neuronale indirecte et directe des systèmes non linéaires affines

IV Introduction

Soit le système non linéaire donné par (III.1) dans le chapitre III :

$$x^{(n)} = f(x, \dot{x}, \dots, x^{(n-1)}) + b(x, \dot{x}, \dots, x^{(n-1)})u + d, \quad y = x \quad (\text{IV.1})$$

Dans ce chapitre, nous considérons le cas où le gain virtuel n'est plus constant. Ainsi, de la même façon, la loi de commande idéale obtenue par application de la linéarisation par retour d'état est donnée par :

$$u^* = \frac{1}{b(x)} \cdot (x_d^{(n)} - K^T \cdot \underline{e} - f(x)) \quad (\text{IV.2})$$

Supposant que $f(x)$ et $b(x)$ sont connues et la perturbation extérieure d n'existe pas, et que $b(x)$ est non nulle, avec $v = x_d^{(n)} - K^T \cdot \underline{e}$, et $e(t) = x(t) - x_d(t)$ et le vecteur $K = (k_0, k_1, \dots, k_{n-1})^T \in R^n$ est choisi telles que toutes les racines du polynôme $p^n + k_{n-1} \cdot p^{n-1} + \dots + k_0 = 0$ sont à partie réelle négative.

Selon la formulation du problème dans la section III.2 du chapitre précédent, comme $f(\underline{x})$ et $b(\underline{x})$ sont inconnues et la perturbation extérieure d existe, la commande u^* de (IV.2) (donnée par (III.7) dans le chapitre précédent) ne peut pas être implémentée. Notre objectif dans ce chapitre, est de proposer deux schémas de commande adaptative indirecte et directe. Dans la première approche dite indirecte, un système FIS du type TS sert à approximer la loi de commande u^* de (IV.2) et un réseaux RBF estimera le gain de commande virtuel $b(\underline{x})$. Cette approche est basée sur le signal de l'erreur de poursuite pour l'adaptation des paramètres. Dans la deuxième approche dite directe, un système FIS du type TS approximera la loi de commande u^* de (IV.2) et un système FIS du type Mamdani estimera le signal de l'erreur de commande qui n'est pas disponible et qui va apparaître dans la loi d'adaptation des paramètres de conséquences du contrôleur flou TS.

IV.1 L'approche indirecte

L'architecture est basée sur deux systèmes adaptés en ligne : un système d'inférence flou (FIS) du type Takagi-Sugeno (TS) et un réseaux à fonctions de base radiales (RBF). Le système FIS avec comme sortie $u(\underline{x}, \underline{\theta})$ sert pour la construction du contrôleur adaptatif en approximant la loi de commande idéale u^* de (IV.2). En se basant sur cette approximation on construit le système RBF avec comme sortie $b(\underline{x}, \underline{\theta}_b)$ qui sera une estimation du gain de la commande $b(\underline{x})$. L'adaptation des paramètres (les poids de connections $\underline{\theta}_b$ du réseau RBF et les paramètres des conséquences $\underline{\theta}$ du contrôleur TS) est basée sur la théorie de stabilité de Lyapunov. Les centres du réseau RBF sont adaptés en ligne en utilisant l'algorithme des *k-means* [38, 39]. La stabilité asymptotique est établie telle que l'erreur de poursuite converge autour de l'origine sans l'utilisation du terme de supervision dans la loi de commande. Le contrôleur adaptatif TS-RBF est appliqué pour le contrôle de deux systèmes non linéaires : le pendule inversé et le système masse- ressort- amortisseur. Les résultats de simulation sont présentés pour montrer les performances de la méthode proposée.

IV.1.1 Analyse de la stabilité et lois d'adaptation des paramètres

Dans ce qui suit, on dérive les lois d'adaptation des paramètres des conséquences $\underline{\theta}$ du contrôleur TS et des poids de connections $\underline{\theta}_b$ du réseau RBF. En remplaçant l'entrée de

commande u dans (IV.1) par la commande $u(\underline{x}, \underline{\theta})$ produite par le FIS du type TS on obtient :

$$x^{(n)} = f(\underline{x}) + b(\underline{x}).u(\underline{x}, \underline{\theta}) + d \quad (IV.3)$$

On ajoute maintenant et soustrait $b.u^*$ à (IV.3), on obtient :

$$x^{(n)} = f(\underline{x}) + b(\underline{x}).u(\underline{x}, \underline{\theta}) + d + b(\underline{x}).u^* - b(\underline{x}).u^* \quad (IV.4)$$

Remplaçant (IV.2) dans (IV.4), on obtient :

$$x^{(n)} = f(\underline{x}) + b(\underline{x}).u(\underline{x}, \underline{\theta}) + d - b(\underline{x}).u^* + x_d^{(n)} - K^T . \underline{e} - f(\underline{x}) \quad (IV.5)$$

Alors :

$$x^{(n)} - x_d^{(n)} = -K^T . \underline{e} + b(\underline{x}).(u(\underline{x}, \underline{\theta}) - u^*) + d \quad (IV.6)$$

Sachant que $e(t) = x(t) - x_d(t)$ et $\underline{e} = (e, \dot{e}, \dots, e^{(n-1)})^T \in \mathbb{R}^n$

l'équation (IV.6) mène au système d'erreur :

$$\dot{\underline{e}} = A_c . \underline{e} + b_c [b(\underline{x}).(u(\underline{x}, \underline{\theta}) - u^*) + d] \quad (IV.7)$$

avec

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ & & & & & \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -k_0 & -k_1 & -k_2 & \dots & -k_{n-2} & -k_{n-1} \end{bmatrix}, \quad b_c = \begin{bmatrix} 0 \\ 0 \\ . \\ 0 \\ 1 \end{bmatrix} \quad (IV.8)$$

A partir de (IV.7), nous allons étudier la stabilité du système pour développer une loi adaptative pour ajuster le vecteur de paramètres $\underline{\theta}$ du contrôleur TS.

Nous commençons par définir le vecteur optimal de paramètres $\underline{\theta}^*$ comme le vecteur de paramètres qui correspond au meilleur approximateur (optimal) $u(\underline{x}, \underline{\theta}^*)$ du signal de commande u^* dans (IV.2). On peut définir alors l'erreur d'approximation minimale :

$$w = u(\underline{x}, \underline{\theta}^*) - u^*(\underline{x}) \quad (IV.9)$$

L'équation d'erreur (IV.7) peut être écrite ainsi :

$$\dot{\underline{e}} = A_c . \underline{e} + b_c [b(\underline{x}).(u(\underline{x}, \underline{\theta}) - u(\underline{x}, \underline{\theta}^*)) + b(\underline{x}).w + d] \quad (IV.10)$$

De (III.11) dans la section III.3.1 du chapitre III, la sortie d'un système flou TS peut être mise sous la forme :

$$u(\underline{x}) = \underline{\theta}^T . \underline{\xi}(\underline{x}) \quad (IV.11)$$

Où le vecteur $\underline{\theta}^T = [\theta_1^T \ \theta_2^T \ \dots \ \theta_n^T]$ contient tous les paramètres ajustables et $\underline{\xi}(x)$ est le vecteur des fonctions de base floues. Utilisant (IV.11), on peut écrire :

$$u(x, \underline{\theta}) = \underline{\theta}^T \cdot \underline{\xi}(x), \text{ et } u(x, \underline{\theta}^*) = \underline{\theta}^{*T} \cdot \underline{\xi}(x) \quad (\text{IV.12})$$

Soit $\varphi = \underline{\theta} - \underline{\theta}^*$ et utilisant (IV.12), alors (IV.10) devient :

$$\dot{e} = A_c \cdot e + b_c \cdot b(x) \cdot \varphi^T \cdot \underline{\xi}(x) + b_c \cdot (b(x) \cdot w + d) \quad (\text{IV.13})$$

Définissant la fonction de Lyapunov candidate :

$$V = \frac{1}{2} \underline{e}^T \cdot P \cdot \underline{e} + \frac{1}{2\gamma} \varphi^T \cdot \varphi \quad (\text{IV.14})$$

Où γ est une constante positive et P est la solution de l'équation de Lyapunov :

$$A_c^T \cdot P + P \cdot A_c = -Q \text{ avec } Q > 0. \quad (\text{IV.15})$$

Dérivant V par rapport au temps :

$$\dot{V} = \frac{1}{2} \underline{e}^T \cdot P \cdot \dot{e} + \frac{1}{2} \underline{e}^T \cdot P \cdot \dot{e} + \frac{1}{2\gamma} \dot{\varphi}^T \cdot \varphi + \frac{1}{2\gamma} \varphi^T \cdot \dot{\varphi} \quad (\text{IV.16})$$

Utilisant (IV.13) et (IV.15), l'équation (IV.16) devient

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \underline{e}^T P b_c b(x) \varphi^T \underline{\xi}(x) + \underline{e}^T P b_c (b(x) \cdot w + d) + \frac{1}{\gamma} \varphi^T \cdot \dot{\varphi} \quad (\text{IV.17})$$

Soit P_n la dernière colonne de la matrice P et utilisant (IV.8) on obtient :

$$\underline{e}^T P b_c = \underline{e}^T P_n \quad (\text{IV.18})$$

Remplaçant (IV.18) dans (IV.17), on aura :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \frac{1}{\gamma} \varphi^T [\gamma b(x) \underline{e}^T P_n \underline{\xi}(x) + \dot{\varphi}] + \underline{e}^T P b_c (b(x) \cdot w + d) \quad (\text{IV.19})$$

Si on choisit la loi d'adaptation :

$$\dot{\underline{\theta}} = -\gamma b(x) \underline{e}^T P_n \underline{\xi}(x) \quad (\text{IV.20})$$

On a $\dot{\varphi} = \dot{\underline{\theta}} - \dot{\underline{\theta}}^* = \dot{\underline{\theta}}$, parce que le vecteur de paramètres optimal $\underline{\theta}^*$ étant constant, sa dérivée est nulle $\dot{\underline{\theta}}^* = 0$, alors, dans l'équation (IV.19), on aura :

$$\frac{1}{\gamma} \varphi^T (\gamma b(x) \underline{e}^T P_n \underline{\xi}(x) + \dot{\varphi}) = 0 \quad (\text{IV.21})$$

Et donc (IV.19) devient :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \underline{e}^T P b_c (b(x) \cdot w + d) \quad (\text{IV.22})$$

Ou encore :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} - \underline{e}^T P \cdot b_c \cdot w_1 \quad (\text{IV.23})$$

Avec :

$$w_1 = -(b(\underline{x}) \cdot w + d) \quad (\text{IV.24})$$

Comme $\dot{\underline{\theta}} = -\gamma \cdot b(\underline{x}) \underline{e}^T P_n \xi_1(\underline{x})$ est en fonction du terme inconnu $b(\underline{x})$, il reste maintenant à développer une estimation utilisant un second système (réseau) RBF pour $b(\underline{x})$ afin de pouvoir adapter le vecteur de paramètres $\underline{\theta}$ du premier système contrôleur flou TS. Sur la base d'une première initialisation du terme $b(\underline{x})$ à une valeur différente de zéro, on peut donc utiliser la loi d'adaptation (IV.20) pour ajuster le vecteur de paramètres $\underline{\theta}$ du contrôleur TS comme première étape, et donc le signal de commande optimale \underline{u}^* (ou l'entrée de commande \underline{u} dans le cas général) possède une première valeur en utilisant le premier système contrôleur TS avec comme sortie $u(\underline{x}, \underline{\theta}) = \underline{\theta}^T \cdot \xi_1(\underline{x})$, c'est à dire, la première valeur de $u(\underline{x}, \underline{\theta})$ existe. On va utiliser maintenant ce fait pour calculer une première valeur de $b(\underline{x}, \underline{\theta}_b)$ et en général pour aboutir à une loi d'adaptation et pouvoir donc ajuster le vecteur de paramètres $\underline{\theta}_b$ du second système RBF possédant la sortie $b(\underline{x}, \underline{\theta}_b) = \underline{\theta}_b^T \cdot \xi_2(\underline{x})$.

Alors, de (IV.3) on a :

$$x^{(n)} = f(\underline{x}) + b(\underline{x}) \cdot u(\underline{x}, \underline{\theta}) + d \quad (\text{IV.25})$$

Si la perturbation extérieure d dans (IV.25) n'existe pas, alors l'expression de $b(\underline{x})$ est

$$b^* = \frac{1}{u(\underline{x}, \underline{\theta})} \cdot [v_b - f(\underline{x})] \quad (\text{IV.26})$$

avec $u(\underline{x}, \underline{\theta}) \neq 0$. Le cas où $u(\underline{x}, \underline{\theta}) = 0$ sera discuté après comme remarque 1.

On choisit l'entrée artificielle v_b (l'entrée équivalente) comme un simple contrôleur à placement de pôles :

$$v_b = x_d^{(n)} - K_b^T \cdot \underline{e} \quad (\text{IV.27})$$

avec $K_b \in R^n$ choisi tel que le polynôme : $s^n + k_{b_{n-1}} \cdot s^{n-1} + \dots + k_{b_0} = 0$ possède toutes ses racines à partie réelle négative. Alors l'équation (IV.26) peut être écrite ainsi :

$$b^* = \frac{1}{u(\underline{x}, \underline{\theta})} \cdot [x_d^{(n)} - K_b^T \cdot \underline{e} - f(\underline{x})] \quad (\text{IV.28})$$

Remplaçant (IV.28) dans (IV.25) et sachant que $e(t) = x(t) - x_d(t)$ et $\underline{e} = (e, \dot{e}, \dots, e^{(n-1)})^T \in R^n$, on obtient :

$$e^{(n)} + K_b^T \underline{e} = 0 \quad (\text{IV.29})$$

$$\text{Ou directement } e^{(n)} + k_{b_{n-1}} \cdot e^{(n-1)} + \dots + k_{b_0} \cdot e = 0 \quad (\text{IV.30})$$

Donc $\lim_{t \rightarrow \infty} e(t) \rightarrow 0$. De (IV.28), $u(\underline{x}, \underline{\theta})$ est connue (c'est la sortie du premier système contrôleur TS donnée par (IV.12)), mais en réalité, $f(\underline{x})$ n'est pas connue et la perturbation extérieure d existe, alors b^* dans (IV.28) ne peut pas être implémenté. Notre second objectif est de construire un second système (réseau) RBF avec comme sortie $b(\underline{x}, \underline{\theta}_b)$ pour approximer l'expression idéale b^* de $b(\underline{x})$. Dans l'équation (IV.25), on peut donc remplacer $b(\underline{x})$ par son estimée $b(\underline{x}, \underline{\theta}_b)$, donc (IV.25) devient :

$$x^{(n)} = f(\underline{x}) + b(\underline{x}, \underline{\theta}_b) \cdot u(\underline{x}, \underline{\theta}) + d \quad (\text{IV.31})$$

On ajoute et on soustrait maintenant $b^* \cdot u(\underline{x}, \underline{\theta})$ à (IV.31), alors (IV.31) devient :

$$x^{(n)} = f(\underline{x}) + b(\underline{x}, \underline{\theta}_b) \cdot u(\underline{x}, \underline{\theta}) + b^* \cdot u(\underline{x}, \underline{\theta}) - b^* \cdot u(\underline{x}, \underline{\theta}) + d \quad (\text{IV.32})$$

Ou ce qui est équivalent à :

$$x^{(n)} = f(\underline{x}) + u(\underline{x}, \underline{\theta}) \cdot (b(\underline{x}, \underline{\theta}_b) - b^*) + b^* \cdot u(\underline{x}, \underline{\theta}) + d \quad (\text{IV.33})$$

Remplaçant (IV.28) dans (IV.33), on peut obtenir :

$$x^{(n)} = x_d^{(n)} - K_b^T \underline{e} + u(\underline{x}, \underline{\theta}) \cdot (b(\underline{x}, \underline{\theta}_b) - b^*) + d \quad (\text{IV.34})$$

Avec :

$$x^{(n)} - x_d^{(n)} = -K_b^T \underline{e} + u(\underline{x}, \underline{\theta}) \cdot (b(\underline{x}, \underline{\theta}_b) - b^*) + d \quad (\text{IV.35})$$

Sachant que $e(t) = x(t) - x_d(t)$ et $\underline{e} = (e, \dot{e}, \dots, e^{(n-1)})^T \in R^n$, l'équation (IV.35) nous mène au système d'erreur :

$$\dot{\underline{e}} = A_b \underline{e} + b_b [u(\underline{x}, \underline{\theta}) \cdot (b(\underline{x}, \underline{\theta}_b) - b^*) + d] \quad (\text{IV.36})$$

Avec :

$$A_b = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -k_{b_0} & -k_{b_1} & -k_{b_2} & \dots & -k_{b_{n-2}} & -k_{b_{n-1}} \end{bmatrix}, \quad b_b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (\text{IV.37})$$

Etudions maintenant la stabilité du système pour développer une loi adaptative afin de pouvoir ajuster le vecteur de paramètres $\underline{\theta}_b$. Soit $\underline{\theta}_b^*$ le vecteur de paramètres qui correspond

au meilleur approximateur (optimal) $b(\underline{x}, \underline{\theta}_b^*)$ du terme b^* dans (IV.28), on définit alors l'erreur d'approximation minimale :

$$w_b = b(\underline{x}, \underline{\theta}_b^*) - b^*. \quad (\text{IV.38})$$

Alors, l'équation d'erreur (IV.36) peut être écrite comme :

$$\dot{\underline{e}} = A_b \underline{e} + b_b [u(\underline{x}, \underline{\theta}) \cdot (b(\underline{x}, \underline{\theta}_b) - b(\underline{x}, \underline{\theta}_b^*)) + u(\underline{x}, \underline{\theta}) \cdot w_b + d] \quad (\text{IV.39})$$

Selon l'équation (IV.11) on peut écrire :

$$b(\underline{x}, \underline{\theta}_b) = \underline{\theta}_b^T \cdot \xi_2(\underline{x}) \text{ et } b(\underline{x}, \underline{\theta}_b^*) = \underline{\theta}_b^{*T} \cdot \xi_2(\underline{x}) \quad (\text{IV.40})$$

Soit $\varphi_b = \underline{\theta}_b - \underline{\theta}_b^*$ et utilisant (IV.40), donc (IV.39) devient :

$$\dot{\underline{e}} = A_b \underline{e} + b_b \cdot u(\underline{x}, \underline{\theta}) \cdot \varphi_b^T \cdot \xi_2(\underline{x}) + b_b \cdot (u(\underline{x}, \underline{\theta}) \cdot w_b + d) \quad (\text{IV.41})$$

Définissons la fonction de Lyapunov candidate :

$$V_b = \frac{1}{2} \underline{e}^T \cdot P_b \underline{e} + \frac{1}{2 \cdot \alpha} \varphi_b^T \varphi_b \quad (\text{IV.42})$$

où α est une constante positive et P_b est la solution de l'équation de Lyapunov ci-dessous :

$$A_b^T \cdot P_b + P_b \cdot A_b = -Q_b \text{ avec } Q_b > 0. \quad (\text{IV.43})$$

Dérivons V_b par rapport au temps :

$$\dot{V}_b = \frac{1}{2} \dot{\underline{e}}^T \cdot P_b \underline{e} + \frac{1}{2} \underline{e}^T \cdot P_b \dot{\underline{e}} + \frac{1}{2\alpha} \dot{\varphi}_b^T \varphi_b + \frac{1}{2\alpha} \varphi_b^T \dot{\varphi}_b \quad (\text{IV.44})$$

En remplaçant par (IV.41) et (IV.43) dans (IV.44), on aura :

$$\dot{V}_b = -\frac{1}{2} \underline{e}^T Q_b \underline{e} + \underline{e}^T P_b b_b u(\underline{x}, \underline{\theta}) \varphi_b^T \xi_2(\underline{x}) + \underline{e}^T P_b b_b (u(\underline{x}, \underline{\theta}) \cdot w_b + d) + \frac{1}{\alpha} \varphi_b^T \dot{\varphi}_b \quad (\text{IV.45})$$

Soit P_{bn} la dernière colonne de la matrice P_b et utilisant (IV.37) on aura :

$$\underline{e}^T P_b b_b = \underline{e}^T P_{bn} \quad (\text{IV.46})$$

De plus en insérant (IV.46) dans (IV.45), on obtient :

$$\dot{V}_b = -\frac{1}{2} \underline{e}^T Q_b \underline{e} + \frac{1}{\alpha} \varphi_b^T [\alpha u(\underline{x}, \underline{\theta}) \underline{e}^T P_{bn} \xi_2(\underline{x}) + \dot{\varphi}_b] + \underline{e}^T P_b b_b (u(\underline{x}, \underline{\theta}) \cdot w_b + d) \quad (\text{IV.47})$$

Si on choisit la loi d'adaptation :

$$\dot{\underline{\theta}}_b = -\alpha u(\underline{x}, \underline{\theta}) \underline{e}^T P_{bn} \xi_2(\underline{x}) \quad (\text{IV.48})$$

Comme $\dot{\varphi}_b = \dot{\underline{\theta}}_b - \dot{\underline{\theta}}_b^* = \dot{\underline{\theta}}_b$, parce que $\underline{\theta}_b^*$ est constant et donc sa dérivée $\dot{\underline{\theta}}_b^* = 0$, on aura :

$$\frac{1}{\alpha} \varphi_b^T (\alpha u(\underline{x}, \underline{\theta}) \underline{e}^T P_{bn} \xi_2(\underline{x}) + \dot{\varphi}_b) = 0 \quad (\text{IV.49})$$

Et donc, (IV.47) devient :

$$\dot{V}_b = -\frac{1}{2} \underline{e}^T Q_b \underline{e} + \underline{e}^T P_b b_b (u(\underline{x}, \underline{\theta}) \cdot w_b + d) \quad (\text{IV.50})$$

Ou encore :

$$\dot{V}_b = -\frac{1}{2} \underline{e}^T Q_b \underline{e} - \underline{e}^T P_b b_b \cdot w_2 \quad (\text{IV.51})$$

Avec :

$$w_2 = -(u(\underline{x}, \underline{\theta}) \cdot w_b + d) \quad (\text{IV.52})$$

En résumé, les lois adaptatives obtenues à partir du développement ci-dessus sont alors pour le système contrôleur TS:

$$\dot{\underline{\theta}} = -\gamma b(\underline{x}, \underline{\theta}_b) \underline{e}^T P_n \xi_1(\underline{x}) \quad (\text{IV.53})$$

et pour le système RBF :

$$\dot{\underline{\theta}}_b = -\alpha u(\underline{x}, \underline{\theta}) \underline{e}^T P_{bn} \xi_2(\underline{x}) \quad (\text{IV.54})$$

Le théorème suivant montre les propriétés du contrôleur adaptatif indirect TS-RBF.

Théorème

Considérons le système non linéaire (IV.1), avec la loi de commande $u(\underline{x}, \underline{\theta}) = \underline{\theta}^T \cdot \xi_1(\underline{x})$ donnée par (IV.12) et l'estimation du gain de commande $b(\underline{x}, \underline{\theta}_b) = \underline{\theta}_b^T \cdot \xi_2(\underline{x})$ donnée par (IV.40) avec les lois d'adaptation données par (IV.53) et (IV.54) pour les vecteurs de paramètres $\underline{\theta}$ et $\underline{\theta}_b$ respectivement, alors le schéma global garantit que :

- i) L'erreur de poursuite $\underline{e}(t)$ converge à l'ensemble compact Ω_e défini par :

$$\Omega_e = \left\{ \underline{e} : |\underline{e}| \leq \sqrt{\frac{\delta}{\mu}} \right\} \quad (\text{IV.55})$$

Où δ et μ sont deux constantes positives.

- ii) Si w_1 dans (IV.23) est sommable carrée (squared integrable), c'est à dire

$$\int_0^{\infty} \|w_1(t)\|^2 dt < \infty, \text{ alors } \lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$$

- iii) L'erreur de poursuite $\underline{e}(t)$ converge à l'ensemble compact Ω_{be} défini par :

$$\Omega_{be} = \left\{ \underline{e} : |\underline{e}| \leq \sqrt{\frac{\delta_b}{\mu_b}} \right\} \quad (\text{IV.56})$$

Où δ_b et μ_b sont deux constantes positives.

iv) Si w_2 dans (IV.51) est sommable carrée, c'est à dire $\int_0^\infty \|w_2(t)\|^2 dt < \infty$, alors

$$\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$$

Démonstration

On commence maintenant par démontrer la première partie du théorème

i) Soit $\lambda_{Q_{\min}}$ la valeur propre minimale de la matrice Q , alors de (IV.23) on a :

$$\dot{V} \leq -\frac{1}{2} \lambda_{Q_{\min}} \|\underline{e}\|^2 - \underline{e}^T P b_c w_1 \quad (\text{IV.57})$$

Qui peut être écrite ainsi :

$$\begin{aligned} \dot{V} &\leq -\frac{1}{2} \lambda_{Q_{\min}} \|\underline{e}\|^2 - \underline{e}^T P b_c w_1 + \frac{1}{2} \|\underline{e}\|^2 - \frac{1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P b_c w_1\|^2 - \frac{1}{2} \|P b_c w_1\|^2 \\ &= -\frac{\lambda_{Q_{\min}} - 1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P b_c w_1\|^2 - \frac{1}{2} \|\underline{e} + P b_c w_1\|^2 \end{aligned} \quad (\text{IV.58})$$

Et qui peut être simplifiée à :

$$\dot{V} \leq -\frac{\lambda_{Q_{\min}} - 1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P b_c w_1\|^2 \quad (\text{IV.59})$$

Parce que le terme $\frac{1}{2} \|\underline{e} + P b_c w_1\|^2$ est supérieur ou égale à 0

On peut choisir Q telle que $\lambda_{Q_{\min}} > 1$, ce qui donne :

$$\dot{V} \leq -\mu \|\underline{e}\|^2 + \delta \quad (\text{IV.60})$$

$$\text{où } \mu = \frac{\lambda_{Q_{\min}} - 1}{2}, \text{ et } \delta = \frac{1}{2} \|P b_c w_1\|^2 \quad (\text{IV.61})$$

De (IV.60), on peut conclure que $\dot{V} < 0$ si $\|\underline{e}\| > \sqrt{\frac{\delta}{\mu}}$. On définit l'ensemble compact :

$$\Omega_e = \left\{ \underline{e} : \|\underline{e}\| \leq \sqrt{\frac{\delta}{\mu}} \right\} \quad (\text{IV.62})$$

De (IV.24), on a : $w_1 = -(b(x).w + d)$, où l'erreur d'approximation minimale w dans (IV.9) peut être arbitrairement limitée à une valeur petite par utilisation d'un choix approprié de

nombre d'approximateurs flous (fonctions de base floues et leurs paramètres) [20, 21, 22], [41]. D'autre part, le gain virtuel $b(\underline{x})$ et la perturbation d étant supposés bornés, alors la quantité w_1 est bornée. Sur cette base, à partir de (IV.61), δ est bornée et donc l'ensemble Ω_e dans (IV.62) est borné. Maintenant \dot{V} est négative tant que $\underline{e}(t)$ est en dehors de l'ensemble compact Ω_e . Conformément à la théorie de la stabilité de Lyapunov, on conclut que l'erreur $\underline{e}(t)$ est bornée et va converger vers l'ensemble Ω_e .

ii) On intègre les deux côtés de (IV.59), on obtient :

$$[V(t) - V(0)] \leq -\frac{\lambda_{Q_{\min}} - 1}{2} \int_0^t \|\underline{e}(\tau)\|^2 d\tau + \frac{1}{2} \|Pb_c\|^2 \cdot \int_0^t \|w_1(\tau)\|^2 d\tau \quad (\text{IV.63})$$

Alors :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{2}{\lambda_{Q_{\min}} - 1} [V(0) - V(t)] + \frac{1}{\lambda_{Q_{\min}} - 1} \|Pb_c\|^2 \cdot \int_0^t \|w_1(\tau)\|^2 d\tau \quad (\text{IV.64})$$

Ce qui donne :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{1}{\lambda_{Q_{\min}} - 1} [2(\|V(0)\| + \|V(t)\|) + \|Pb_c\|^2 \cdot \int_0^t \|w_1(\tau)\|^2 d\tau] \quad (\text{IV.65})$$

Comme montré par Wang [20], cela implique que si $w_1 \in L_2$ (sommable carrée), alors de (IV.65), on a $\underline{e}(t) \in L_2$, et sur la base de la conclusion ci-dessus (conclusion après preuve de la première partie du théorème), conformément à la théorie de stabilité de Lyapunov, $\underline{e}(t)$ est bornée. D'un autre côté de (IV.13), $\dot{\underline{e}}(t) \in L_\infty$ (bornée) parce que tous les membres de son côté droit sont bornés. Selon le lemme de Barbalat, on conclut que $\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$.

iii) Soit $\lambda_{Q_b \min}$ la valeur propre minimale de la matrice Q_b , alors de (IV.51) on a :

$$\dot{V}_b \leq -\frac{1}{2} \lambda_{Q_b \min} \|\underline{e}\|^2 - \underline{e}^T P_b b_b \cdot w_2 \quad (\text{IV.66})$$

Qui peut être écrite ainsi :

$$\begin{aligned} \dot{V}_b &\leq -\frac{1}{2} \lambda_{Q_b \min} \|\underline{e}\|^2 - \underline{e}^T P_b b_b \cdot w_2 + \frac{1}{2} \|\underline{e}\|^2 - \frac{1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P_b b_b w_2\|^2 - \frac{1}{2} \|P_b b_b w_2\|^2 \\ &= -\frac{\lambda_{Q_b \min} - 1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P_b b_b w_2\|^2 - \frac{1}{2} \|\underline{e} + P_b b_b w_2\|^2 \quad (\text{IV.67}) \end{aligned}$$

Et qui peut être simplifiée à :

$$\dot{V}_b \leq -\frac{\lambda_{Q_b \min} - 1}{2} \|\underline{e}\|^2 + \frac{1}{2} \|P_b b_b w_2\|^2 \quad (\text{IV.68})$$

Parce que le terme $\frac{1}{2} \|\underline{e} + P_b b_b w_2\|^2$ est supérieur ou égale à 0.

On peut choisir Q_b telle que $\lambda_{Q_b \min} > 1$, alors :

$$\dot{V}_b \leq -\mu_b \|\underline{e}\|^2 + \delta_b \quad (\text{IV.69})$$

$$\text{où } \mu_b = \frac{\lambda_{Q_b \min} - 1}{2}, \text{ et } \delta_b = \frac{1}{2} \|P_b b_b w_2\|^2 \quad (\text{IV.70})$$

De (IV.69), on peut conclure que $\dot{V}_b < 0$ si $\|\underline{e}\| > \sqrt{\frac{\delta_b}{\mu_b}}$. On définit l'ensemble compact :

$$\Omega_{be} = \left\{ \underline{e} : \|\underline{e}\| \leq \sqrt{\frac{\delta_b}{\mu_b}} \right\} \quad (\text{IV.71})$$

De (IV.52), on a : $w_2 = -(u(\underline{x}, \underline{\theta}) \cdot w_b + d)$, où l'erreur d'approximation minimale w_b dans (IV.38) peut être limitée à une valeur suffisamment petite par un choix approprié d'un nombre de fonctions de base radiales approximatrices (nombre suffisant de neurones) [43]. D'autre part, la perturbation d est supposée bornée, l'élément clef dans w_2 , et donc dans δ_b , ou directement dans Ω_{be} est l'entrée de commande $u(\underline{x}, \underline{\theta})$.

Nous avons démontré dans la première et dans la deuxième partie du théorème que \dot{V} est négative et que si $w_1 \in L_2$ (i.e., sommable carrée), alors $\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$. Cela implique que l'entrée de commande $u(\underline{x}, \underline{\theta})$ estimée reste bornée et donc la quantité w_2 est bornée. Sur cette base, à partir de (IV.70), δ_b est borné et donc l'ensemble Ω_{be} dans (IV.71) est borné. Maintenant \dot{V}_b est négative tant que $\underline{e}(t)$ est en dehors de l'ensemble compact Ω_{be} . Conformément à la théorie de la stabilité de Lyapunov, on conclut que l'erreur $\underline{e}(t)$ est bornée et va converger vers l'ensemble Ω_{be} .

iv) On intègre les deux côtés de (IV.68), on obtient:

$$[V_b(t) - V_b(0)] \leq -\frac{\lambda_{Q_b \min} - 1}{2} \int_0^t \|\underline{e}(\tau)\|^2 d\tau + \frac{1}{2} \|P_b b_b\|^2 \cdot \int_0^t \|w_2(\tau)\|^2 d\tau \quad (\text{IV.72})$$

Alors :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{2}{\lambda_{Q_b \min} - 1} [V_b(0) - V_b(t)] + \frac{1}{\lambda_{Q_b \min} - 1} \|P_b b_b\|^2 \cdot \int_0^t \|w_2(\tau)\|^2 d\tau \quad (\text{IV.73})$$

Ce qui donne :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{1}{\lambda_{Qb \min} - 1} [2(\|V_b(0)\| + \|V_b(t)\|) + \|P_b b_b\|^2 \cdot \int_0^t \|w_2(\tau)\|^2 d\tau] \quad (\text{IV.74})$$

Comme montré par Wang [20], cela implique que si $w_2 \in L_2$ (sommable carrée), alors de (IV.74) $\underline{e}(t) \in L_2$, et en se basant sur la conclusion ci-dessus (conclusion après preuve de la troisième partie du théorème), conformément à la théorie de stabilité de Lyapunov, $\underline{e}(t)$ est bornée. D'un autre côté de (IV.41) $\dot{\underline{e}}(t) \in L_\infty$ (bornée) parce que tous les membres de son côté droit sont bornés. Selon le lemme de Barbalat, nous concluons que $\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$.

Remarque

Comme on peut voir de (IV.54), la loi d'adaptation $\underline{\theta}_b$ est en fonction du terme $u(\underline{x}, \underline{\theta})$, alors quand $u(\underline{x}, \underline{\theta}) = 0$ (le cas mentionné juste après l'équation (IV.26)), le vecteur de paramètres $\underline{\theta}_b$ ne sera pas adapté en utilisant l'équation (IV.54) et va seulement garder sa valeur précédente (sa dernière valeur) avant que le terme $u(\underline{x}, \underline{\theta})$ soit nul. De plus, le cas $u(\underline{x}, \underline{\theta}) = 0$ dans l'équation (IV.28) ne va causer aucun problème, parce que la valeur optimale b^* ne va pas être calculée en utilisant (IV.28), mais va être estimée par un réseau RBF, c'est-à-dire : que son calcul sera effectué en utilisant l'alternative du second système approximateur (réseau RBF) avec la sortie $b(\underline{x}, \underline{\theta}_b) = \underline{\theta}_b^T \cdot \xi_2(\underline{x})$ donnée par (IV.40) comme mentionné précédemment.

IV.1.2 Conception du contrôleur adaptatif indirect TS-RBF

A partir de l'analyse ci-dessus, la conception du contrôleur adaptatif TS-RBF indirect peut être résumée par les étapes suivantes :

Etape1: Calculs hors ligne

- Spécifier pour chaque entrée, le nombre des fonctions d'appartenance ainsi que leurs formes, et leurs paramètres (paramètres des prémisses du contrôleur flou (TS)).
- Définir le nombre des fonctions de base avec des centres uniformément repartis dans le domaine de variation des données pour le réseau RBF.
- Spécifier les paramètres k_0, \dots, k_{n-1} pour le premier système TS, et k_{b0}, \dots, k_{bn-1} pour le deuxième système (réseau RBF) telles que toutes les racines des deux polynômes :

$s^n + k_{n-1}.s^{n-1} + \dots + k_1.s^1 + k_0 = 0$ et $s^n + k_{bn-1}.s^{n-1} + \dots + k_{b1}.s^1 + k_{b0} = 0$ sont dans le demi-plan gauche.

- Spécifier les matrices $n \times n$ définies positives Q et Q_b , où n est le degré du système.
- Résoudre l'équation de Lyapunov (IV.15) pour obtenir la matrice $P > 0$, et l'équation de Lyapunov (IV.43) pour obtenir la matrice $P_b > 0$.
- Sélectionner les valeurs scalaires positives γ et α .
- Donner des valeurs initiales au vecteur des paramètres des conséquences $\underline{\theta}$ pour le premier système flou TS (contrôleur) $u(\underline{x}, \underline{\theta}) = \underline{\theta}^T \cdot \xi_1(\underline{x})$.
- Donner des valeurs initiales au vecteur de paramètres (poids de connections) $\underline{\theta}_b$ pour le deuxième système ou réseau RBF (approximateur du gain de commande virtuel) $b(\underline{x}, \underline{\theta}_b) = \underline{\theta}_b^T \cdot \xi_2(\underline{x})$.

Etape2: Adaptation en ligne

- Appliquer la loi de commande (IV.12) : $u(\underline{x}, \underline{\theta}) = \underline{\theta}^T \cdot \xi_1(\underline{x})$ (la sortie du premier système flou ou contrôleur TS) au système (IV.1).
- Utiliser la loi d'adaptation (IV.53) pour ajuster le vecteur des paramètres des conséquence $\underline{\theta}$ du contrôleur flou TS, où le terme $b(\underline{x}, \underline{\theta}_b)$ (la sortie du deuxième système ou réseau RBF) est donné par (IV.40).
- Utiliser la loi d'adaptation (IV.54) pour ajuster le vecteur de paramètres $\underline{\theta}_b$ (les poids de connections) de l'estimateur RBF du gain virtuel de commande, où le terme $u(\underline{x}, \underline{\theta})$ (la sortie du premier système contrôleur TS) est donné par (IV.12).
- Utiliser l'algorithme des *k-means* décrit dans la section II.4.1.b.i du chapitre II pour ajuster les centres des fonctions de base radiales pour le réseau RBF.

IV.1.3 Résultats de simulation

Dans cette partie, on teste en simulation les performances de l'approche de commande adaptative TS-RBF indirecte proposée sur deux systèmes non linéaires: le pendule inversé et le système masse- ressort- amortisseur.

IV.1.3.1 Exemple 1

Le modèle du système du pendule inversé représenté par la figure IV.1 est décrit par les équations dynamiques suivantes :

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(\underline{x}) + b(\underline{x})u(t) + d \\ y &= x_1 \end{aligned} \quad (\text{IV.75})$$

Où

$$f(\underline{x}) = \frac{g \sin(x_1) - \frac{mlx_2^2 \cos(x_1) \sin(x_1)}{M+m}}{l\left(\frac{4}{3} - \frac{m \cos^2(x_1)}{M+m}\right)}, \quad \text{et} \quad b(\underline{x}) = \frac{\cos(x_1)}{l\left(\frac{4}{3} - \frac{m \cos^2(x_1)}{M+m}\right)} \quad (\text{IV.76})$$

$x_1 = \theta$ est la position angulaire et $x_2 = \dot{\theta}$ est la vitesse angulaire du pendule inversé, $g = 9.8 \text{ m/s}^2$, $M = 1 \text{ kg}$ est la masse de la voiture (chariot), $m = 0.1 \text{ kg}$ est la masse du bras et $l = 0.5 \text{ m}$ est la moitié de la longueur du bras.

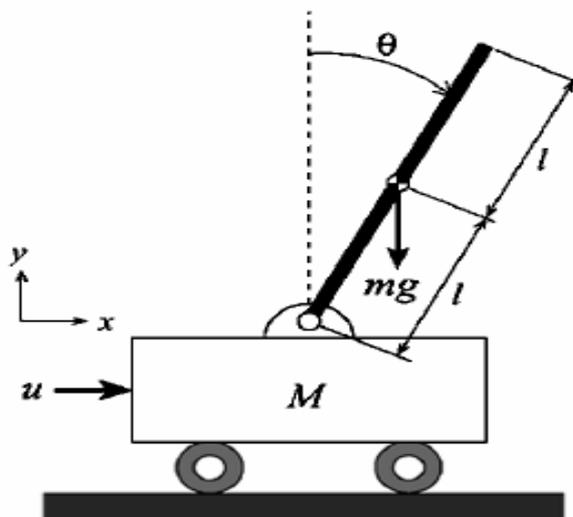
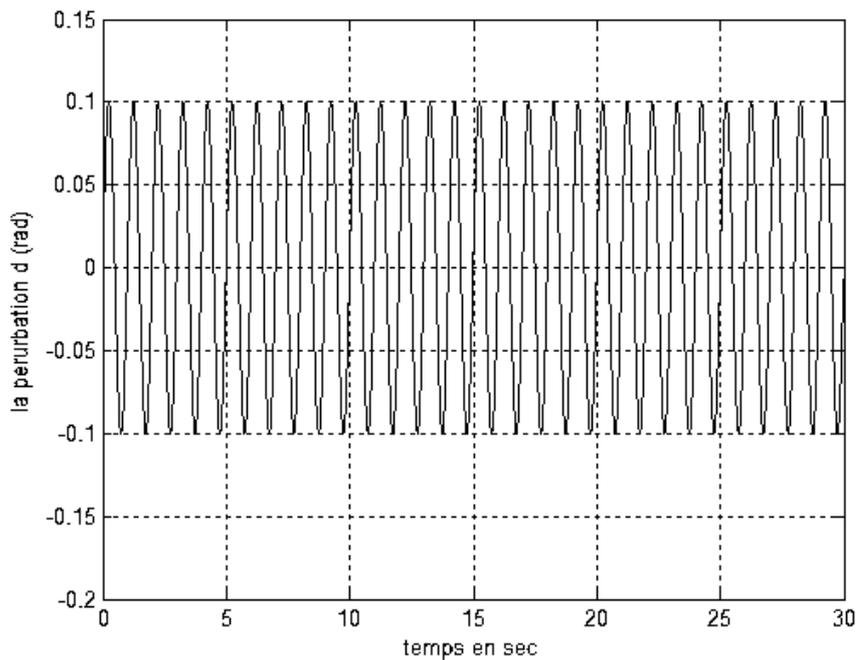


Figure IV.1. Le système du pendule inversé

L'objectif de la commande est de faire suivre au bras du pendule inversé une trajectoire $\theta_d = y_d = AM.\sin(t)$ pour différentes amplitudes AM avec à la fin de la simulation $AM=0$. Clairement, les dérivées du signal de référence $y_d = x_{1d}$ existent et sont bornées. La perturbation extérieure d à la sortie du système est une onde sinusoïdale, avec une amplitude ± 0.1 et une période 2π comme montré dans la figure IV.2.



1) Les paramètres du premier système contrôleur TS sont choisis comme suit :

$\gamma = 35$, et $K = [k_0 \ k_1]^T = [36 \ 5]^T$ pour avoir toutes les racines du polynôme $s^2 + k_1.s + k_0 = 0$ dans le demi plan gauche. De (IV.15), $Q = \text{diag}(300,300) > 0$, donc $\lambda_{Q_{\min}} = 300 > 0$, et par résolution de (IV.15) on obtient :

$$P = \begin{bmatrix} 1130.8 & 4.2 \\ 4.2 & 30.8 \end{bmatrix} \quad (\text{IV.77})$$

Le contrôleur flou TS possède deux entrées $\underline{x} = \underline{z} = [z_1 \ z_2] = [\theta_d \ (\theta_d^{(2)} - K^T \underline{e})]$, avec $\underline{e} = [e \ \dot{e}] = [\theta - \theta_d \ \dot{\theta} - \dot{\theta}_d]$. Chaque entrée possède trois sous ensembles flous avec des fonctions d'appartenance similaires données par :

$$\mu_N(z_i) = \exp(-(z_i - c_N)^2 / 2.(\sigma_N)^2) \quad (\text{IV.78})$$

$$\mu_Z(z_i) = \exp(-(z_i - c_Z)^2 / 2.(\sigma_Z)^2) \quad (\text{IV.79})$$

$$\mu_P(z_i) = \exp(-(z_i - c_P)^2 / 2.(\sigma_P)^2) \quad (\text{IV.80})$$

Les largeurs des fonctions d'appartenance sont : $\sigma_N = \sigma_Z = \sigma_P = 0.25$ pour la première et la deuxième entrée z_1 et z_2 . Les centres ont les valeurs : $c_N = -0.05$, $c_Z = 0$ et $c_P = 0.05$ pour z_1 et z_2 . Cela donne neuf règles floues de la forme :

$$R^i : \text{if } z_1 \text{ is } A_1^i \text{ and } z_2 \text{ is } A_2^i \text{ then } u_i = a_1^i z_1 + a_2^i z_2 \quad (\text{IV.81})$$

On a donc 18 paramètres dans le vecteur θ à ajuster qui sont tous initialisés à zéro.

2) Les paramètres du deuxième système estimateur RBF sont choisis comme suit : $\alpha = 0.0039$ et $K_b = [k_{b0} \ k_{b1}]^T = [0.9 \ 5]^T$ pour avoir toutes les racines du polynôme $s^2 + k_{b1}.s + k_{b0} = 0$ dans le demi plan gauche. De (IV.43), $Q_b = \text{diag}(127,127) > 0$, donc $\lambda_{Q_b \min} = 127 > 0$ et par résolution de (IV.43) on obtient:

$$P_b = \begin{bmatrix} 376.90 & 70.55 \\ 70.55 & 26.81. \end{bmatrix} \quad (\text{IV.82})$$

Le réseaux RBF estimateur est composé de cinq fonctions de base, il possède deux entrées : l'erreur $e = \theta - \theta_d = y - y_d$ et la variation de l'erreur $de = \dot{e} = \dot{\theta} - \dot{\theta}_d$. Les paramètres $\underline{\theta}_b$ sont tous initialisés à 0.5. Les centres des fonctions de base sont uniformément distribués dans l'intervalle $[-0.66 \ 0.66]$. Les largeurs σ pour chaque fonction de base dans le réseau RBF sont données la valeur 0.47.

Les conditions initiales sont $(x_1(0), x_2(0)) = (-0.2 \text{ rad}, 0 \text{ rad/sec})$. La période d'échantillonnage est fixée à 0.01 s. Les résultats pour différentes amplitudes du signal de référence sont montrés dans les figures IV.3 à IV.7. La position du pendule $x_1 = \theta$ (ou sortie du système $y(t)$) est en continu, par contre le signal de référence $y_d(t)$ est en pointillé.

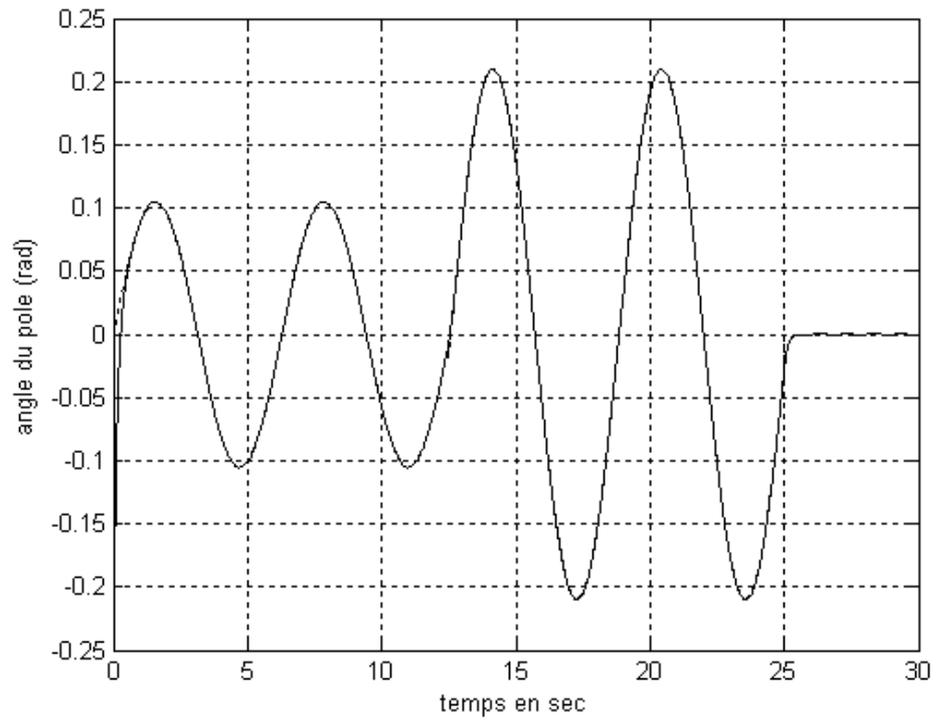


Figure IV.3 : L'évolution de l'angle du pendule

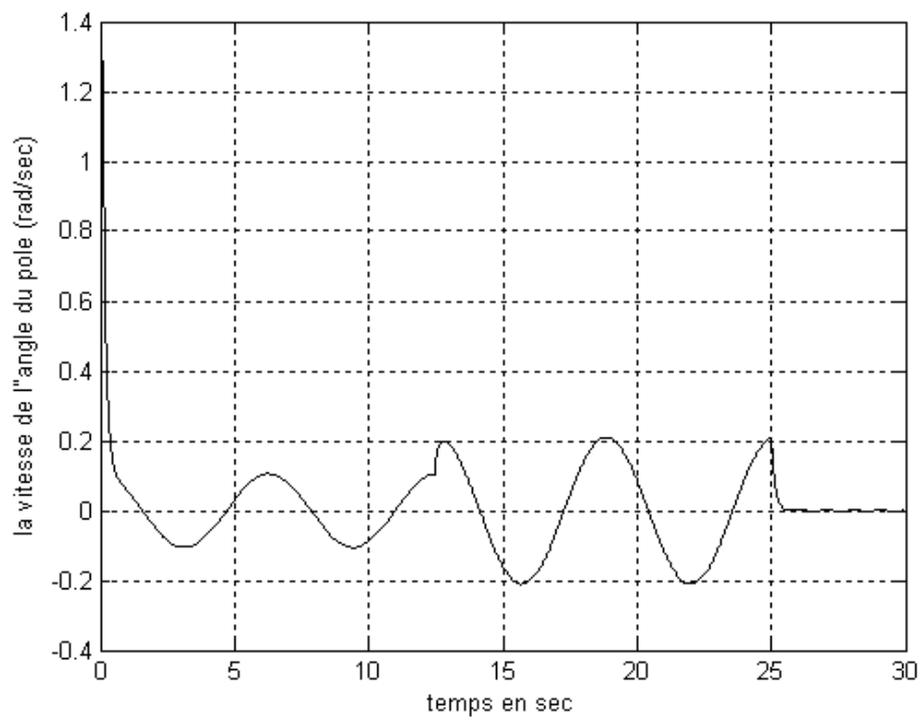


Figure IV.4 : La vitesse correspondante du pendule

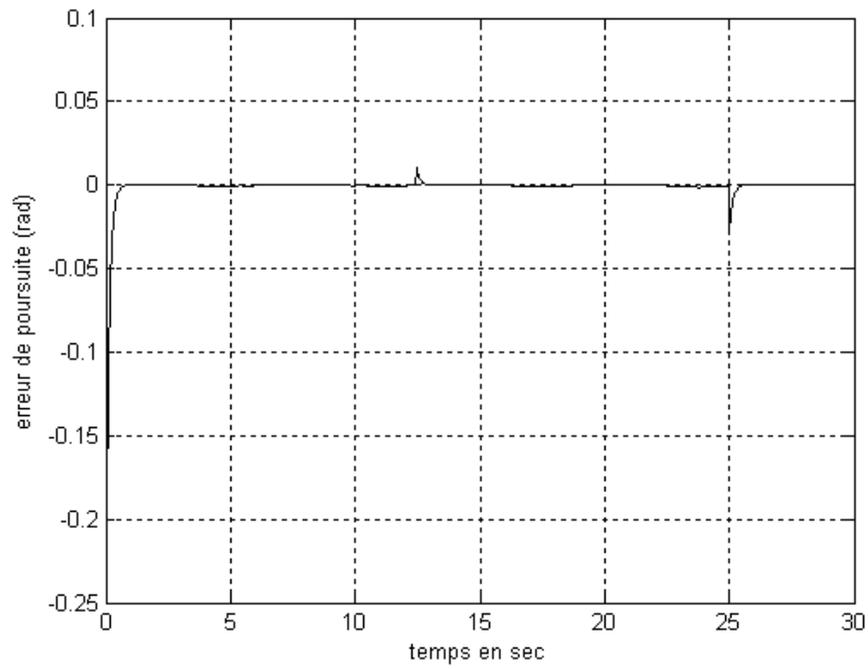


Figure IV.5 : L'erreur de poursuite

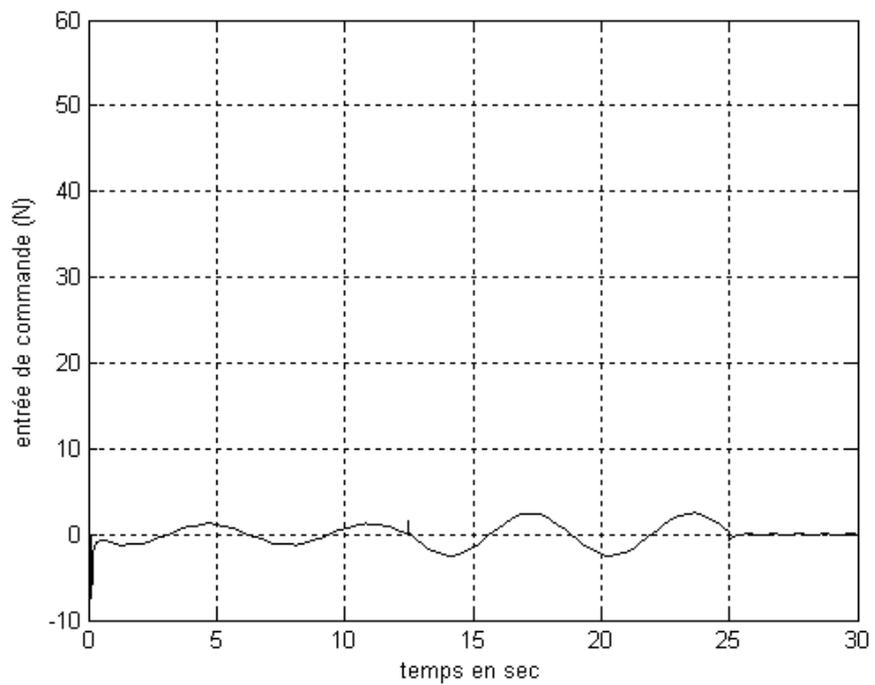


Figure IV.6 : L'entrée de commande

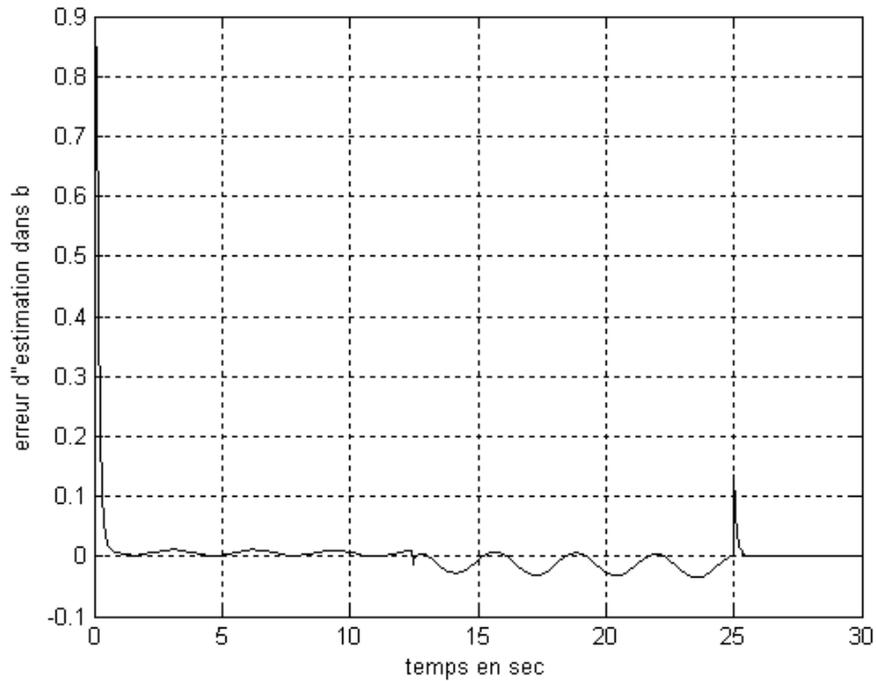


Figure IV.7 : L'erreur d'estimation dans $b(\underline{x})$

La figure IV.3 montre la courbe de réponse de l'angle à partir de la position initiale $(x_1(0), x_2(0)) = (-0.2, 0)$. L'amplitude de la référence est telle que $AM = \pi/30$ pendant l'intervalle de temps $t \in [0 \ 12.5] s$, $AM = \pi/15$ pendant l'intervalle de temps $t \in [12.5 \ 25] s$ et finalement $AM = 0$ à $t \in [25 \ 30] s$. On peut voir de la figure IV.3 que l'état du système $x_1(t) = \theta$ suit la trajectoire désirée $x_{1d} = y_d = \theta_d$. La figure IV.4 montre la vitesse du pendule. La figure IV.5 montre que l'erreur de poursuite converge rapidement à une valeur au voisinage de zéro. La figure IV.6 représente l'entrée de commande correspondante qui fait des pics à $t = 12.5 s$ (le premier changement de l'amplitude de la référence : de $AM = \pi/30$ à $AM = \pi/15$), et à $t = 25 s$ (le deuxième changement : de $AM = \pi/15$ à $AM = 0$). La figure IV.7 montre que l'estimation de l'erreur (entre la valeur réelle du gain de commande virtuel $b(\underline{x})$ dans l'équation du pendule inversé et la valeur fournie par le réseau RBF estimateur $b(\underline{x}, \underline{\theta}_b)$) est lisse et se stabilise rapidement autour de zéro, ce qui confirme les bonnes performances des systèmes RBF. A partir de ces figures, on

peut bien voir que le système contrôlé se comporte bien dans toutes les situations: la poursuite et ainsi que le cas de la régulation.

IV.1.3.2 Exemple 2

Dans ce deuxième exemple, nous considérons un système mécanique non linéaire : masse- ressort- amortisseur. Ce processus est décrit par la figure IV.8 et modélisé par les équations suivantes :

$$M\ddot{x}(t) + g(x(t), \dot{x}(t)) + f(x(t)) = \phi(\dot{x}(t))u(t) \quad (\text{IV.83})$$

Avec :

$$g(x(t), \dot{x}(t)) = c_1x(t) + c_2\dot{x}(t) \quad (\text{IV.84})$$

$$f(x(t)) = c_3x(t) + c_4x(t)^3 \quad (\text{IV.85})$$

$$\phi(\dot{x}(t)) = 1 + c_5\dot{x}(t)^2 \quad (\text{IV.86})$$

Où M , $x(t)$ et u sont respectivement : la masse, le déplacement de la masse et la force. $f(x(t))$, $g(x(t), \dot{x}(t))$, $\phi(\dot{x}(t))$ sont les non linéarités dû respectivement à l'élasticité du ressort, l'amortisseur et l'entrée. Les paramètres sont donnés par $M = 1.0$, $c_1 = 0$, $c_2 = 1$, $c_3 = 0.01$, $c_4 = 0.1$, et $c_5 = 0.13$. Le modèle est alors donné par :

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_2 - 0.01x_1 - 0.1x_1^3 + (1 + 0.13x_2^2)u \end{aligned} \quad (\text{IV.87})$$

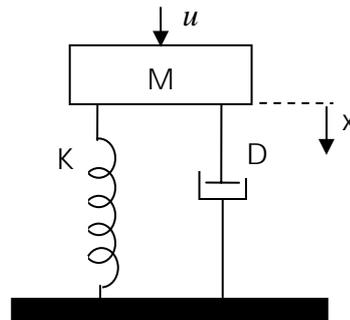


Figure IV.8 Le système: mass- ressort- amortisseur

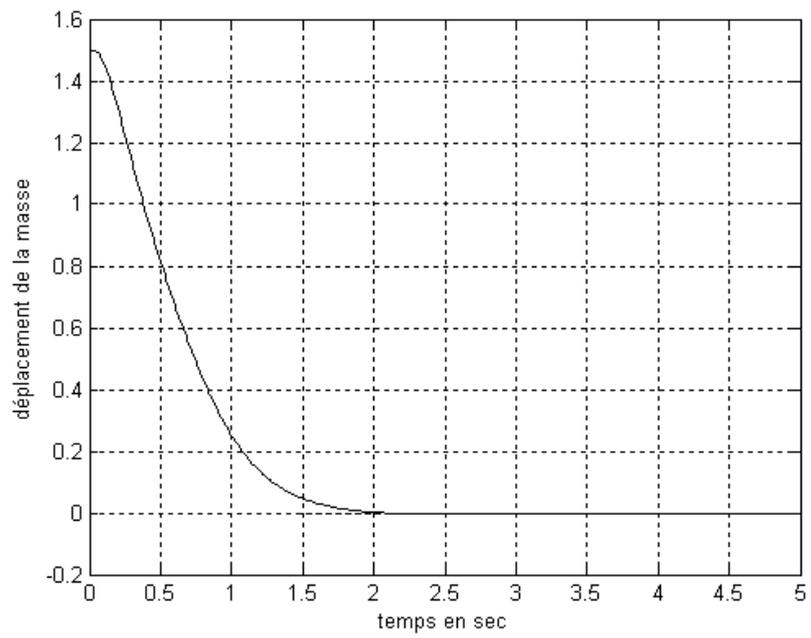
L'objectif de cette commande est de maintenir ou ramener la masse à l'origine de déplacement $x_{1d}(t) = 0$. Les états initiaux sont $(x_1(0), x_2(0)) = (1.5, 0)$. La période d'échantillonnage est fixée à 0.01 seconde et la plage de variation du déplacement $x_1(t)$ de la masse est $[-1.5, 1.5]$.

1) Les paramètres du premier système contrôleur TS sont choisis comme suit : $\gamma = 35$, et $k = [k_0 \ k_1]^T = [5 \ 3]^T$ pour avoir toutes les racines du polynôme $s^2 + k_1.s + k_0 = 0$ dans le demi plan gauche. De (IV.15), $Q = \text{diag}(5,5) > 0$, donc $\lambda_{Q_{\min}} = 5 > 0$ et par résolution de (IV.15) on obtient :

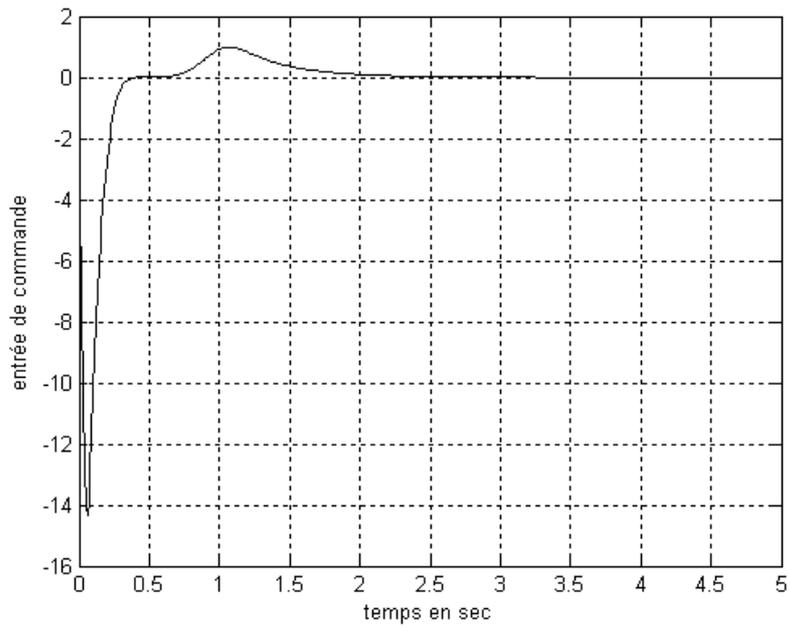
$$P = \begin{bmatrix} 6.5 & 0.5 \\ 0.5 & 1 \end{bmatrix} \quad (\text{IV.88})$$

La même structure du contrôleur flou TS que dans l'exemple 1 est utilisée avec les mêmes entrées : $\underline{x} = \underline{z} = [z_1 \ z_2] = [\theta_d \ (\theta_d^{(2)} - K^T \underline{e})]$ possédant les mêmes sous ensembles flous caractérisés par les fonctions d'appartenance données par (IV.78), (IV.79) et (IV.80). Les largeurs des fonctions d'appartenance pour ce deuxième exemple sont : $\sigma_N = \sigma_Z = \sigma_P = 1$ pour la première entrée z_1 et sont $\sigma_N = \sigma_Z = \sigma_P = 0.5$ pour la deuxième entrée z_2 . Les centres ont les valeurs : $c_N = -1$, $c_Z = 0$ et $c_P = 1$ pour z_1 et $c_N = -0.75$, $c_Z = 0$ et $c_P = 0.75$ pour z_2

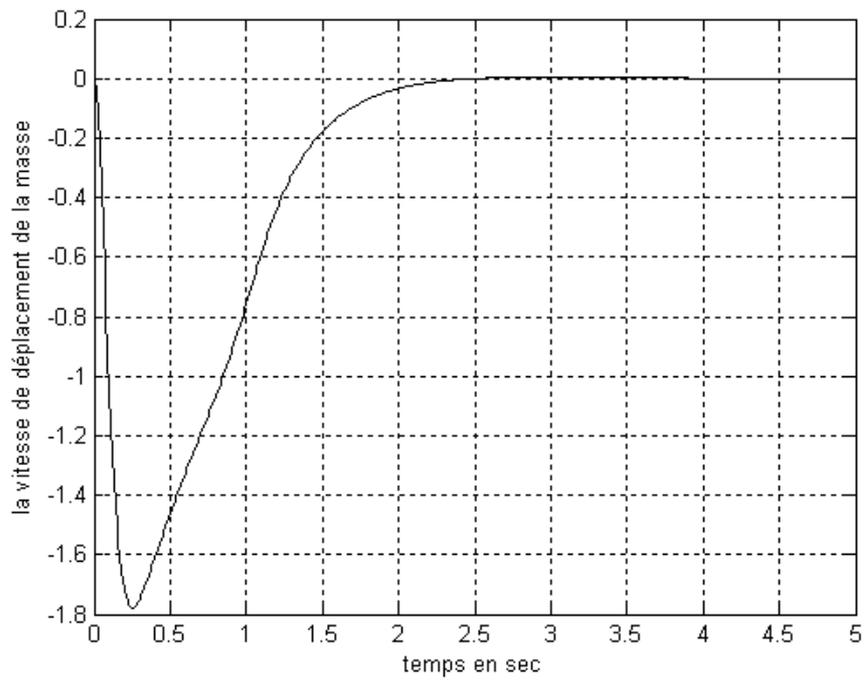
2) Les paramètres du deuxième système estimateur RBF (avec la même structure que dans l'exemple 1) sont choisis comme suit : $\alpha = 0.005$, $K_b = K$ et $Q_b = Q$, où K et Q sont les paramètres du premier système contrôleur TS et donc $P_b = P$ (après résolution de l'équation (IV.15)). Les centres des fonctions de base du réseau RBF sont uniformément distribués dans l'intervalle $[-0.83 \ 0.83]$. Les largeurs σ pour chaque fonction de base du réseau RBF sont données la valeur 0.5. Les résultats de simulation sont montrés par les figures IV.9 aux IV.13.



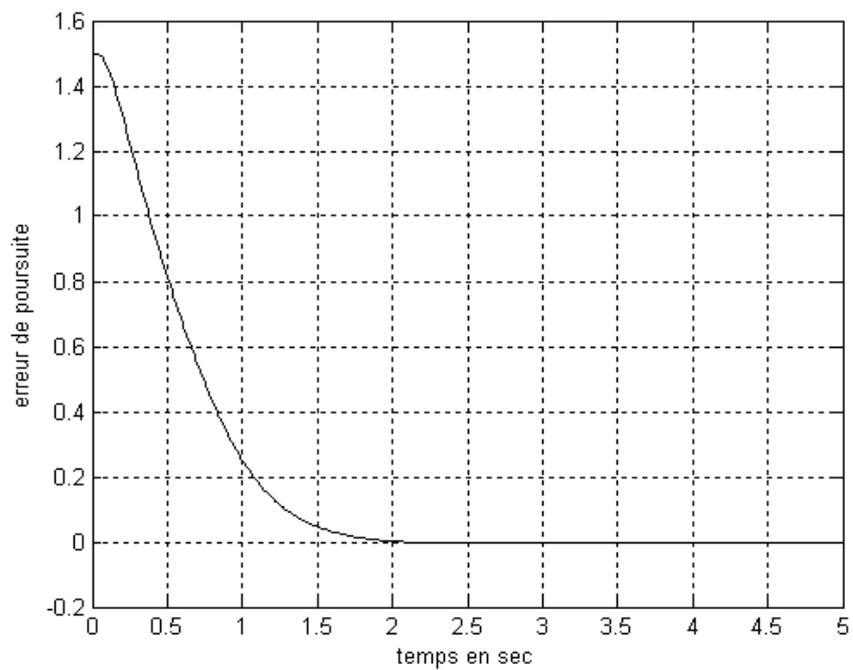
Figures IV.9 : Déplacement de la masse



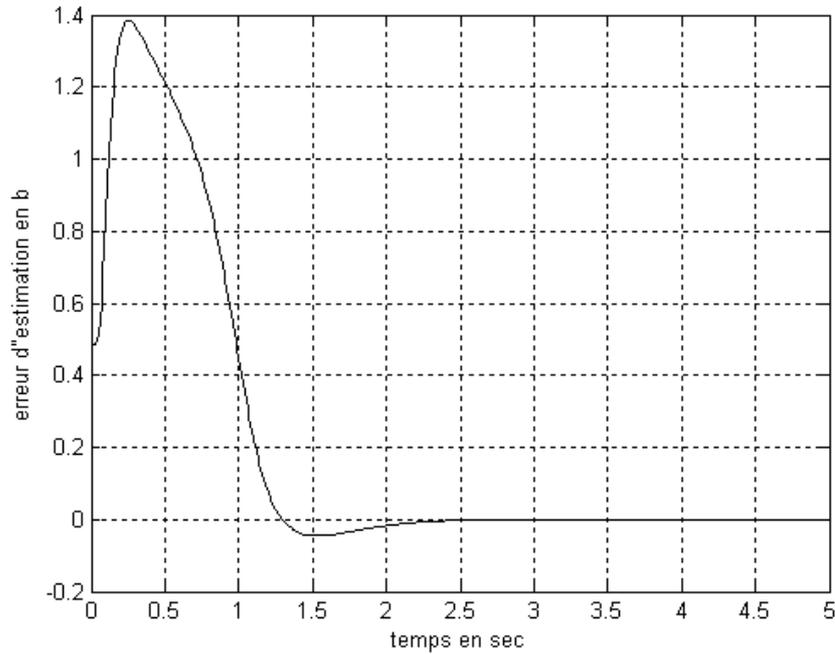
Figures IV.10 : signal de commande



Figures IV.11 : Vitesse du déplacement de la masse



Figures IV.12 : Erreur de poursuite



Figures IV.13 : Erreur d'estimation du gain de commande $b(\underline{x})$

D'après la figure IV.9, on peut voir que la position du déplacement $x_1(t)$ se stabilise à la position d'équilibre désirée $x_{1d}(t) = 0$ à l'instant $t = 2$ s. La figure IV.10 montre aussi que le signal de commande est borné et lisse. La figure IV.11 montre l'évolution de la vitesse du déplacement $x_2(t)$. D'après la figure IV.12 qui montre l'erreur entre la sortie du système $x_1(t)$ et la valeur désirée $x_{1d}(t) = 0$, on peut confirmer les bonnes performances de la poursuite sans addition du terme superviseur à la loi de commande. La figure IV.13 montre que l'erreur entre le gain de commande virtuel réel $b(\underline{x})$ et celui estimé par le réseau RBF estimateur $b(\underline{x}, \underline{\theta}_b)$ converge rapidement vers la valeur zéro, ce qui confirme les bonnes propriétés d'approximation des systèmes RBF.

IV.1.4 Conclusion

Nous avons proposé un schéma de commande adaptative indirecte pour une classe des systèmes non linéaires. Nous avons utilisé pour cela un système d'inférence flou du type TS pour approximer le signal de commande idéal et un réseau RBF pour estimer le gain de commande virtuel qui est une partie des paramètres du système. Les centres des fonctions de base du réseau RBF sont ajustés en ligne en utilisant l'algorithme des *k-means*. Les poids de connections entre la couche cachée et la couche de sortie ainsi que les paramètres du contrôleur TS sont ajustés en ligne selon des lois dérivées en utilisant la théorie de stabilité de Lyapunov. La méthode proposée TS-RBF a pu garantir la stabilité du système résultant en boucle fermée dans le sens que tous les signaux fournis étaient uniformément bornés sans addition du terme superviseur à la loi de commande. Les résultats de simulation sur deux exemples ont montré les bonnes performances et les propriétés lisses du schéma de commande proposé.

IV.2 L'approche directe

Dans cette deuxième approche directe, un système FIS du type TS est utilisé pour approximer la loi de commande idéale u^* de (IV.2). Les paramètres du contrôleur FIS sont adaptés en utilisant la loi de la descente du gradient basée sur l'erreur de commande estimée par un système flou du type Mamdani. La base de règle de l'estimateur Mamdani est construite à partir de connaissances d'un expert. La stabilité asymptotique du système résultant en boucle fermée est garantie en utilisant la théorie de Lyapunov.

IV.2.1 Le contrôleur adaptatif flou TS

Nous avons vu au chapitre précédent que la sortie du système flou TS peut être mise sous la forme :

$$u(\underline{x}) = \underline{\theta}^T \cdot \underline{\xi}(\underline{x}) \quad (\text{IV.89})$$

Où $\underline{\xi}(\underline{x})$ est le vecteur des fonctions de base floues. Il a été prouvé que la loi (III.11) ou (IV.11) peut approximer autour d'un ensemble compact Ω_x , n'importe qu'elle fonction lisse à un degré de précision donné [20, 21, 22]. Cela peut être utilisé pour approximer la loi de commande u^* donnée dans (IV.2) si bien que :

$$u^* = \underline{\theta}^{*T} \underline{\xi}(x) + \varepsilon = u_{Fopt} + \varepsilon \quad (IV.90)$$

où $\underline{\theta}^*$ sont les paramètres optimaux, $\underline{\xi}(x)$ est le vecteur des fonctions de base floues, ε est l'erreur d'approximation minimale dans Ω_x et $u_{Fopt} = \underline{\theta}^{*T} \underline{\xi}(x)$ est l'approximateur optimal.

À chaque instant, la loi de commande implémentée sera :

$$u_c(x, \underline{\theta}) = \underline{\theta}^T \underline{\xi}(x) \quad (IV.91)$$

Les paramètres inconnus de $\underline{\theta}^T$ seront estimés en ligne. L'objectif de l'adaptation des paramètres est de minimiser l'erreur entre l'approximateur optimal u_{Fopt} et la sortie

$u_c(x, \underline{\theta})$ du contrôleur TS :

$$e_u = u_{Fopt} - u_c(x, \underline{\theta}) \quad (IV.92)$$

Cela mène à la fonction coût :

$$\min J = \frac{(u_{Fopt} - u_c(x, \underline{\theta}))^2}{2} = \frac{e_u^2}{2} \quad (IV.93)$$

Afin d'obtenir les lois d'adaptation résultantes de cet objectif, on utilise l'algorithme de la descente du gradient :

$$\dot{\underline{\theta}} = -\gamma \frac{\partial J}{\partial \underline{\theta}} \quad (IV.94)$$

Avec $\gamma > 0$ est le tau d'apprentissage et de (IV.93) on a :

$$\frac{\partial J}{\partial \underline{\theta}} = -e_u \frac{\partial u_c}{\partial \underline{\theta}} \quad (IV.95)$$

De (IV.92), on peut voir que l'erreur de commande e_u ne peut pas être calculée tant que u_{Fopt} est inconnu. Cependant, si on peut obtenir une estimée \hat{e}_u de e_u avec le signe correct :

$$e_u = \alpha \hat{e}_u \quad (IV.96)$$

où $\alpha > 0$ est un coefficient positif inconnu, la direction du gradient ne sera pas changé.

En insérant (IV.96) dans (IV.95) à la place de e_u , utilisant (IV.91), (IV.95) et (IV.96), (IV.94) peut être écrit comme :

$$\dot{\underline{\theta}} = \gamma \alpha \hat{e}_u \frac{\partial u_c}{\partial \underline{\theta}} = \gamma' \hat{e}_u \underline{\xi}(x) \quad (IV.97)$$

Tant que α et γ sont positifs, ils peuvent être rassemblés pour former le nouveau pas d'adaptation $\gamma' > 0$. Sous la condition que ce pas soit suffisamment petit, la fonction coût va converger dans le pire cas à un minimum local de (IV.93) [50], e_u va donc rester bornée

pendant la procédure de recherche. Le fait que e_u soit une quantité bornée est le résultat clef de cette approche. Dans le paragraphe suivant, on décrit le système d'inférence flou (FIS) du type Mamdani pour obtenir une estimée \hat{e}_u du terme e_u avec le signe correct. Pour assurer que les paramètres restent bornés, on peut utiliser comme suggéré dans [1] la e-modification donnée par :

$$\dot{\underline{\theta}} = \gamma' \hat{e}_u \cdot \xi(x) + \gamma' |\hat{e}_u| v_0 \underline{\theta} \quad (\text{IV.98})$$

$v_0 > 0$ est une constante.

IV.2.2 L'estimateur flou de l'erreur de commande

Dans ce paragraphe, on développe l'analyse qui permet d'obtenir l'estimateur flou de l'erreur de commande. On identifie en premier lieu les situations où le signal de commande est "correct" dans le sens qu'il conduise la sortie envers la référence. Sans aucune connaissance particulière sur le système, il est clair que cela peut avoir lieu lorsque l'erreur de poursuite est maintenue à zéro ou lorsqu'elle diminue. D'un autre côté, le signal de commande est clairement non correct lorsque la sortie s'éloigne de la référence.

Introduisant les variables floues: *ERREUR*(e), *VARIATION DE L'ERREUR*(de) et *ERREUR DE COMMANDE*(\hat{e}_u), où chacune prend trois valeurs floues: *ZERO*(ou *Z*), *NEGATIVE*(ou *N*) et *POSITIVE*(ou *P*). On obtient la base de règle, RB1 pour les systèmes dont le gain entrée sortie est négatif (le signal de commande et la sortie ont une direction opposée) [37]:

Cas1: (le signal de commande est correct lorsque l'erreur de poursuite est zéro ou décroissante) : cela implique les règles suivantes :

if e is N and de is P then \hat{e}_u is Z

if e is Z and de is Z then \hat{e}_u is Z

if e is P and de is N then \hat{e}_u is Z

Cas2 : (la sortie s'éloigne du côté haut) : cela implique les règles suivantes :

if e is N and de is N then \hat{e}_u is N

if e is N and de is Z then \hat{e}_u is N

if e is Z and de is N then \hat{e}_u is N

Cas3 : (la sortie s'éloigne du côté bas) : cela implique les règles suivantes :

if e is Z and de is P then \hat{e}_u is P

if e is P and de is Z then \hat{e}_u is P

if e is P and de is P then \hat{e}_u is P

La base de règles RB2 pour les systèmes dont le gain entrée sortie est positif (le signal de commande et la sortie ont la même direction) peut être obtenue de la même manière, où le cas1 reste le même, par contre les conséquences du cas2 et cas3 sont interchangées. Selon l'analyse ci-dessus, il est clair que cette procédure fournit une estimation de l'erreur de commande \hat{e}_u avec le signe correct. Afin de définir complètement l'estimateur flou, on choisit premièrement la forme de toutes les fonctions d'appartenance et leur distribution dans l'univers de discours comme montré dans la figure IV.14, et c_N, c_Z, c_P sont les points où les fonctions d'appartenance atteignent leur maximum. Après la fuzzification et la stratégie d'inférence (*prod, max*), l'erreur de commande estimée crisper (numérique) \hat{e}_{u_crisp} est calculée selon la formule de defuzzification du centre de gravité :

$$\hat{e}_{u_crisp} = \frac{c_{Neu} \times \mu_N(\hat{e}_u) + c_{Zeu} \times \mu_Z(\hat{e}_u) + c_{Peu} \times \mu_P(\hat{e}_u)}{\mu_N(\hat{e}_u) + \mu_Z(\hat{e}_u) + \mu_P(\hat{e}_u)} \quad (IV.99)$$

Où $\mu_N(\hat{e}_u)$, $\mu_Z(\hat{e}_u)$ et $\mu_P(\hat{e}_u)$ sont les degrés d'appartenance dans les ensembles flous NEGATIVE, ZERO et POSITIVE. Tout cela est calculé à partir des degrés d'appartenance de l'erreur $\mu_N(e)$, $\mu_Z(e)$ et $\mu_P(e)$ et sa variation $\mu_N(de)$, $\mu_Z(de)$ et $\mu_P(de)$ en utilisant le mécanisme d'inférence (*prod, max*).

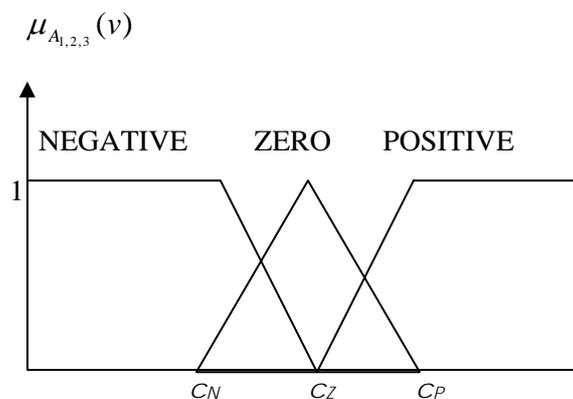


Figure IV.14 : La distribution des fonctions d'appartenance dans l'univers de discours

Les degrés d'appartenance de l'erreur et sa variation sont calculés en utilisant les formules suivantes :

$$\mu_N(v) = \begin{cases} 1 & \text{si } v \leq c_{Nv} \\ \frac{c_{Zv} - v}{c_{Zv} - c_{Nv}} & \text{si } c_{Nv} < v \leq c_{Zv} \\ 0 & \text{si } v > c_{Zv} \end{cases} \quad (\text{IV.100})$$

$$\mu_Z(v) = \begin{cases} 0 & \text{si } v \leq c_{Nv} \\ \frac{v - c_{Nv}}{c_{Zv} - c_{Nv}} & \text{si } c_{Nv} < v \leq c_{Zv} \\ \frac{c_{Pv} - v}{c_{Pv} - c_{Zv}} & \text{si } c_{Zv} < v < c_{Pv} \\ 0 & \text{si } v \geq c_{Pv} \end{cases} \quad (\text{IV.101})$$

$$\mu_P(v) = \begin{cases} 0 & \text{si } v \leq c_{Zv} \\ \frac{v - c_{Zv}}{c_{Pv} - c_{Zv}} & \text{si } c_{Zv} < v < c_{Pv} \\ 1 & \text{si } v \geq c_{Pv} \end{cases} \quad (\text{IV.102})$$

Où v désigne l'erreur $e(t)$ ou sa variation $de(t)$.

IV.2.3 Analyse de la stabilité

Afin d'analyser la stabilité du contrôleur adaptatif flou, nous dérivons premièrement l'équation de l'erreur. Insérant le terme $u_c(x, \underline{\theta})$ dans l'équation du système (IV.1) et se basant sur les équations (IV.90) et (IV.92), nous le remplaçons par:

$$u_c(\underline{x}, \underline{\theta}) = u_{\text{Fopt}} - e_u = u^* - \varepsilon - e_u \quad (\text{IV.103})$$

En ajoutant et soustrayant $b(\underline{x})u^*$ de (IV.1), utilisant (IV.2) et après quelques manipulations simples, l'équation de l'erreur gouvernant le système en boucle fermée est :

$$\dot{\underline{e}} = A_c \underline{e} + B(d - b(\underline{x})(e_u + \varepsilon)) \quad (\text{IV.104})$$

Où d est une perturbation bornée et :

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -k_0 & -k_1 & -k_2 & \dots & -k_{n-2} & -k_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (\text{IV.105})$$

En prenant la fonction de Lyapunov suivante :

$$V = \frac{1}{2} \underline{e}^T . P . \underline{e} \quad (\text{IV.106})$$

Où P est la solution de l'équation de Lyapunov :

$$A_c^T . P + P . A_c = -Q \quad \text{avec} \quad Q > 0. \quad (\text{IV.107})$$

Dérivant V par rapport au temps :

$$\dot{V} = \frac{1}{2} \dot{\underline{e}}^T . P . \underline{e} + \frac{1}{2} \underline{e}^T . P . \dot{\underline{e}} \quad (\text{IV.108})$$

Utilisant (IV.104) et (IV.107), on obtient :

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + \underline{e}^T P . B . (d - b(x)(e_u + \varepsilon)) \quad (\text{IV.109})$$

Ou encore :

$$\dot{V} = -\frac{1}{2} \underline{e}^T . Q . \underline{e} - \underline{e}^T P . B . d_1 \quad \text{avec} \quad d_1 = -(d - b(x)(e_u + \varepsilon)) \quad (\text{IV.110})$$

Théorème

Considérons le système non linéaire (IV.1) avec la loi de commande donnée par (IV.89) et la loi d'adaptation donnée par (IV.98) pour le vecteur de paramètres $\underline{\theta}$, alors le système global est tel que :

i) L'erreur de poursuite $\underline{e}(t)$ converge vers un ensemble Ω_e défini par :

$$\Omega_e = \left\{ \underline{e} : |\underline{e}| \leq \sqrt{\frac{\delta}{\mu}} \right\} \quad (\text{IV.111})$$

Où δ et μ sont deux constantes positives.

ii) si d_1 dans (IV.110) est sommable carrée, c'est-à-dire $\int_0^\infty \|d_1(t)\|^2 dt < \infty$, alors

$$\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0.$$

Démonstration

i) Soit $\lambda_{Q_{\min}}$ la valeur propre minimale de Q alors, de (IV.110) on a :

$$\dot{V} \leq -\frac{1}{2}\lambda_{Q_{\min}}\|\underline{e}\|^2 - \underline{e}^T P b_c d_1 \quad (\text{IV.112})$$

Qui peut être écrite ainsi :

$$\begin{aligned} \dot{V} &\leq -\frac{1}{2}\lambda_{Q_{\min}}\|\underline{e}\|^2 - \underline{e}^T P b_c d_1 + \frac{1}{2}\|\underline{e}\|^2 - \frac{1}{2}\|\underline{e}\|^2 + \frac{1}{2}\|P b_c d_1\|^2 - \frac{1}{2}\|P b_c d_1\|^2 \\ &= -\frac{\lambda_{Q_{\min}} - 1}{2}\|\underline{e}\|^2 + \frac{1}{2}\|P b_c d_1\|^2 - \frac{1}{2}\|\underline{e} + P b_c d_1\|^2 \end{aligned} \quad (\text{IV.113})$$

Le terme $\frac{1}{2}\|\underline{e} + P b_c d_1\|^2$ étant supérieur ou égale à 0, nous obtenons :

$$\dot{V} \leq -\frac{\lambda_{Q_{\min}} - 1}{2}\|\underline{e}\|^2 + \frac{1}{2}\|P b_c d_1\|^2 \quad (\text{IV.114})$$

Si Q est choisie telle que $\lambda_{Q_{\min}} > 1$, parce qu'elle est déterminée par le concepteur, alors :

$$\dot{V} \leq -\mu\|\underline{e}\|^2 + \delta \quad (\text{IV.115})$$

$$\text{Où } \mu = \frac{\lambda_{Q_{\min}} - 1}{2}, \text{ et } \delta = \frac{1}{2}\|P b_c d_1\|^2 \quad (\text{IV.116})$$

de (IV.115) on peut conclure que : $\dot{V} < 0$ si $\|\underline{e}\| > \sqrt{\frac{\delta}{\mu}}$. On définit l'ensemble :

$$\Omega_e = \left\{ \underline{e} : \|\underline{e}\| \leq \sqrt{\frac{\delta}{\mu}} \right\} \quad (\text{IV.117})$$

De (IV.110), on a : $d_1 = -(d - b(\underline{x})(e_u + \varepsilon))$ où ε peut être construite arbitrairement petite par utilisation d'un nombre suffisant d'approximateurs flous [20, 21, 22], le terme $b(\underline{x})$ et la perturbation d sont supposées bornés, l'élément clef dans Ω_e est l'erreur de commande e_u . De la section IV.2.1, on sait que si la taille du pas γ' est convenablement choisie, la fonction coût va converger dans le pire cas à un minimum local de (IV.93) [50], l'erreur de commande e_u reste donc bornée, la quantité d_1 est bornée, et donc, de (IV.116), δ est bornée ce qui implique que l'ensemble Ω_e dans (IV.117) est bornée.

Maintenant, \dot{V} est négative tant que $\underline{e}(t)$ est en dehors de l'ensemble Ω_e , selon la théorie de stabilité de Lyapunov, on conclut que l'erreur $\underline{e}(t)$ est bornée et va converger vers Ω_e

ii) En intégrant les deux côtés de (IV.114), on obtient :

$$[V(t) - V(0)] \leq -\frac{\lambda_{Q_{min}} - 1}{2} \int_0^t \|\underline{e}(\tau)\|^2 d\tau + \frac{1}{2} \|PB\|^2 \int_0^t \|d_1(\tau)\|^2 d\tau \quad (IV.118)$$

Donc :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{2}{\lambda_{Q_{min}} - 1} [V(0) - V(t)] + \frac{1}{\lambda_{Q_{min}} - 1} \|PB\|^2 \int_0^t \|d_1(\tau)\|^2 d\tau \quad (IV.119)$$

Cela nous donne :

$$\int_0^t \|\underline{e}(\tau)\|^2 d\tau \leq \frac{1}{\lambda_{Q_{min}} - 1} [2(\|V(0)\| + \|V(t)\|) + \|PB\|^2 \int_0^t \|d_1(\tau)\|^2 d\tau] \quad (IV.120)$$

Comme montré par Wang [20], cela implique que si $d_1 \in L_2$ (c'est à dire, sommable carrée), alors de ((IV.120)) $\underline{e}(t) \in L_2$, et se basant sur la conclusion ci-dessus (conclusion après preuve de la première partie du théorème), selon la théorie de stabilité de Lyapunov, $\underline{e}(t)$ est bornée. D'un autre côté : de (IV.104) $\dot{\underline{e}}(t) \in L_\infty$ (bornée) parce que tous les éléments de son côté droit sont bornés. Utilisant le lemme de Barbalat on peut conclure que $\lim_{t \rightarrow \infty} \|\underline{e}(t)\| = 0$.

IV.2.4 Conception du contrôleur adaptatif direct TS-Mamdani

A partir de l'analyse ci-dessus, la conception du contrôleur adaptatif TS-Mamdani direct peut être résumée aux étapes suivantes

Etape1: Calculs hors ligne

- Spécifier pour chaque entrée du contrôleur flou TS, le nombre des fonctions de base floues (fonctions d'appartenance) ainsi que leurs formes, et leurs paramètres (paramètres des prémisses du contrôleur flou TS), et qui sont les centres et les largeurs.
- Spécifier pour chaque entrée et pour la sortie de l'estimateur flou Mamdani, le nombre des fonctions de base floues (fonctions d'appartenance) ainsi que leurs formes, et leurs centres et largeurs.

- Spécifier les paramètres k_0, \dots, k_{n-1} pour le premier système TS, telles que toutes les racines du polynôme $s^n + k_{n-1}.s^{n-1} + \dots + k_1.s^1 + k_0 = 0$ sont dans le demi-plan gauche.
- Spécifier la matrice $n \times n$ définie positive Q , où n est le degré du système.
- Résoudre l'équation de Lyapunov (IV.107) pour obtenir la matrice $P > 0$.
- Sélectionner les valeurs scalaires positives γ et α ou directement le nouveau pas d'adaptation γ' parce que $\gamma' = \gamma \cdot \alpha$.
- Donner des valeurs initiales au vecteur de paramètres (paramètres des conséquences) $\underline{\theta}$ pour le premier système flou TS (contrôleur) $u(x, \underline{\theta}) = \underline{\theta}^T \cdot \xi(x)$.

Etape2: Calcul en ligne

- Appliquer la loi de commande (IV.89), (la sortie du premier système flou ou contrôleur TS) au système (IV.1).
- Utiliser la loi d'adaptation (IV.98) pour ajuster le vecteur de paramètres de conséquence $\underline{\theta}$ du contrôleur flou TS.

IV.2.5 Résultats de simulation

Dans cette section, nous testons les performances du contrôleur adaptatif flou direct TS-Mamdani proposé sur les deux mêmes systèmes étudiés dans la première approche.

IV.2.5.1 Exemple 1

Dans cet exemple, on applique en simulation le contrôleur TS-Mamdani sur le système du pendule inversé (figure IV.1) étudié dans la première approche avec le même problème de poursuite et la même perturbation externe (figure IV.2.).

1) Les paramètres du premier système contrôleur TS sont choisis comme suit : $\gamma' = 3.4$, $v_0 = 0.01$, et $k = [k_0 \quad k_1]^T = [35 \quad 5]^T$. $Q = \text{diag}(2, 2) > 0$, donc $\lambda_{Q_{\min}} = 2 > 0$. Par résolution de (IV.107) on obtient :

$$P = \begin{bmatrix} 7.342 & 0.028 \\ 0.028 & 0.205 \end{bmatrix} \quad (\text{IV.121})$$

Le vecteur de données \underline{x} du contrôleur flou TS est composé de trois entrées :

$\underline{x} = \underline{z} = [z_1 \ z_2 \ z_3] = [x_1 \ x_{1d} \ x_d^{(n)} - k^T \underline{e}] = [\theta \ \theta_d \ (\theta_d^{(2)} - K^T \underline{e})]$, avec :

$\underline{e} = [e \ \dot{e}] = [\theta - \theta_d \ \dot{\theta} - \dot{\theta}_d]$. Chaque entrée (variable linguistique) prend trois valeurs linguistiques (*ZERO*, *NEGATIVE* et *POSITIVE*) associées à des sous ensembles flous. Ces derniers sont caractérisés par des fonctions d'appartenance représentées par la figure IV.14 et qui sont données par les formules : (IV.100) à (IV.102).

Les valeurs de c_N, c_Z, c_P sont fixées à $c_N = -3$, $c_Z = 0$ et $c_P = 3$ pour chacune des trois entrées. Cela donne 27 règles de la forme:

$$R^i : \text{if } z_1 \text{ is } A_1^i \text{ and } z_2 \text{ is } A_2^i \text{ and } z_3 \text{ is } A_3^i \text{ then } u_i = a_1^i z_1 + a_2^i z_2 + a_3^i z_3 \quad (\text{IV.122})$$

On a donc 81 paramètres dans le vecteur θ à ajuster et qui sont tous initialisés à zéro.

2) Le système estimateur flou est tel que décrit dans la section IV.2.2, avec la base de règles RB1 (le système est du premier type décrit dans la section IV.2.2). Les paramètres définissant les fonctions d'appartenance sont choisis comme suit : $c_N = -2$, $c_Z = 0$, $c_P = 2$ pour l'erreur e , la variation de l'erreur de (ou \dot{e}) et pour l'erreur de commande estimée \hat{e}_u .

Les conditions initiales suivantes pour le pendule inversé sont utilisées dans la simulation :

$$(x_1(0), x_2(0)) = (-0.2 \text{ rad}, 0 \text{ rad/sec}) \text{ et la période d'échantillonnage est fixée à } dt = 0.01 \text{ s.}$$

Les résultats de simulations pour différentes amplitudes du signal de référence sont montrés dans les figures IV.15 à IV.19. La sortie du système $y(t)$ ou $x_1 = \theta$ (angle du bras du pendule) est en trillé, par contre le signal de référence $y_d(t)$ ou $x_{1d} = \theta_d$ est en continu.

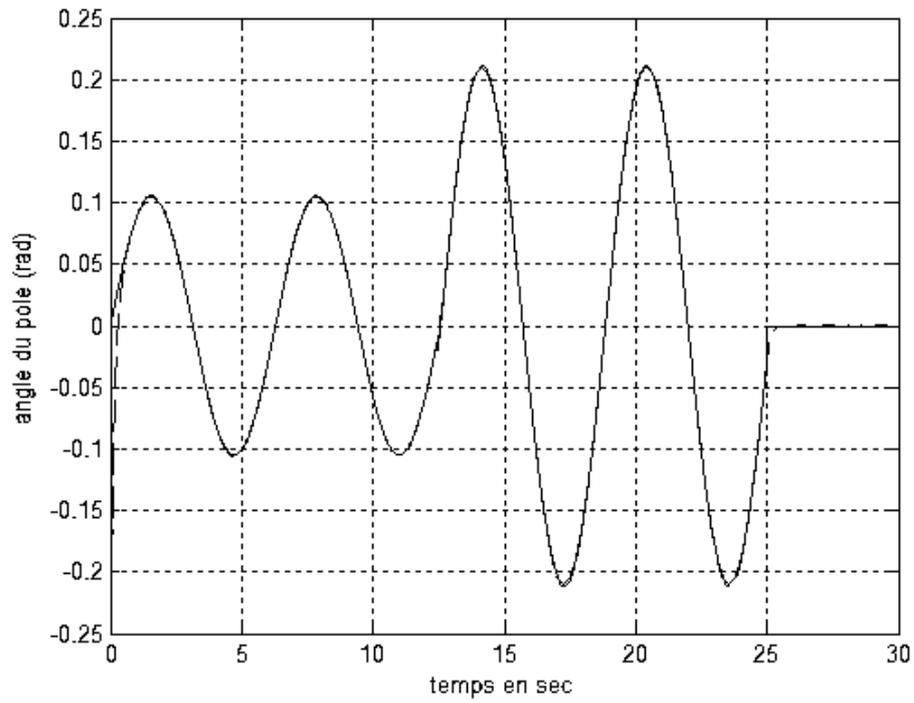


Figure IV.15 : L'évolution de l'angle du pendule

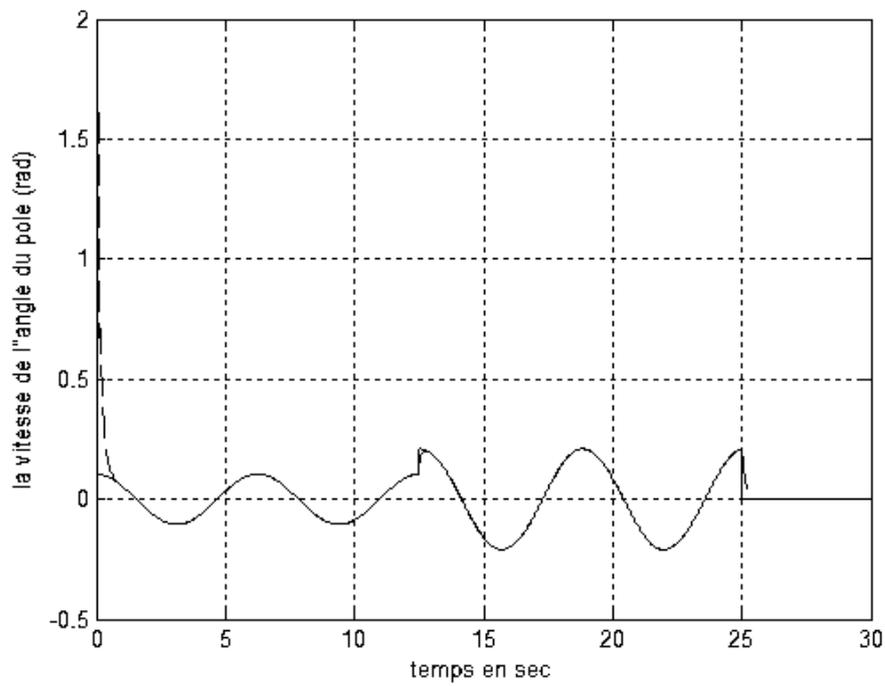


Figure IV.16 : La vitesse correspondante du pendule

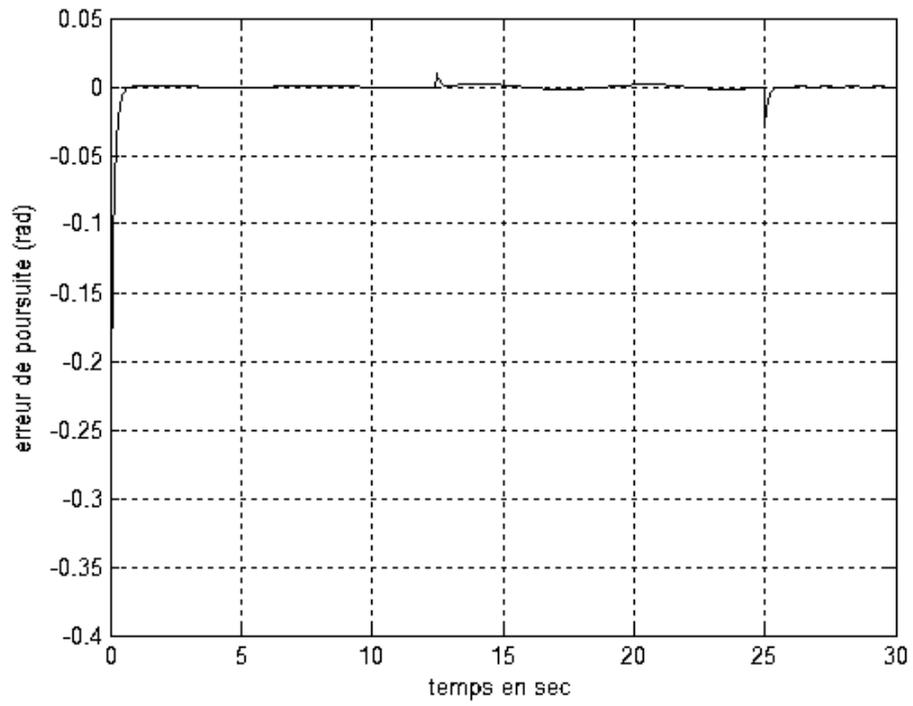


Figure IV.17 : L'erreur de poursuite

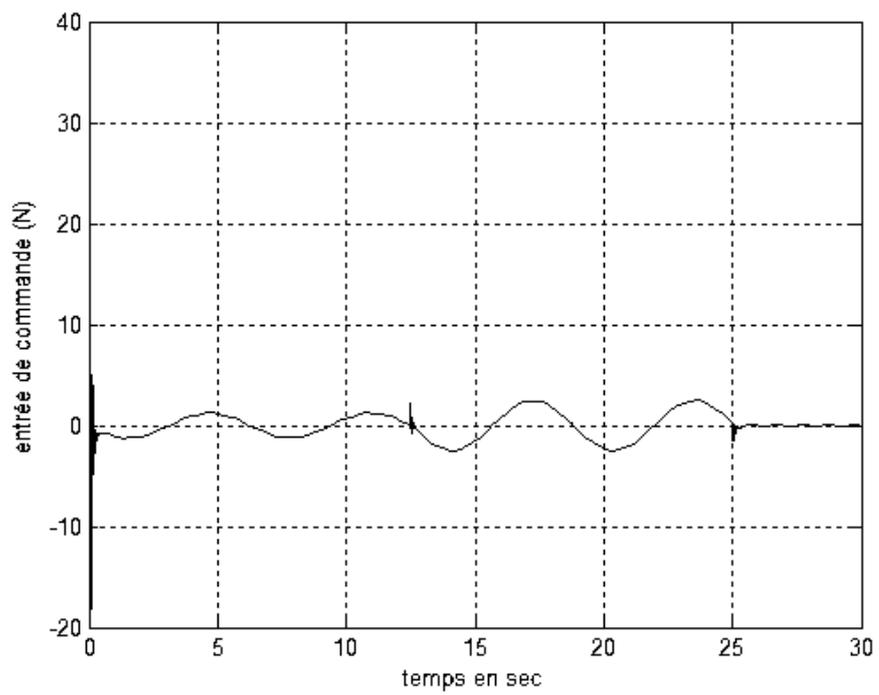


Figure IV.18 : L'entrée de commande

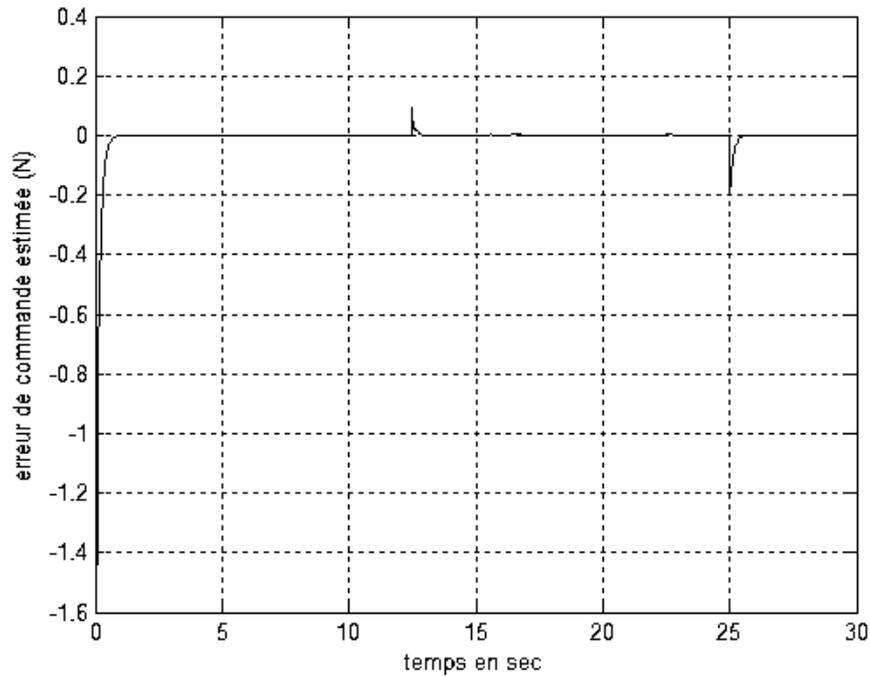


Figure IV.19 : Erreur de commande estimée \hat{e}_u

La figure IV.15 montre la courbe de réponse de l'angle du bras du pendule inversé à partir de la position initiale $(x_1(0), x_2(0)) = (-0.2, 0)$. On peut voir de cette figure que l'état du système $x_1(t) = \theta$ suit la trajectoire désirée $x_{1d} = \theta_d$ parfaitement. La figure IV.16 montre la vitesse de l'angle du bras du pendule inversé. La figure IV.17 montre que l'erreur de poursuite converge rapidement à une valeur au voisinage de zéro. La figure IV.18 représente l'entrée de commande correspondante qui fait des pics à $t = 12.5 s$ (le premier changement de l'amplitude de la référence : de $AM = \pi/30$ à $AM = \pi/15$), et à $t = 25 s$ (le deuxième changement : de $AM = \pi/15$ à $AM = 0$). La figure IV.19 montre que l'erreur de commande estimée fournie par l'estimateur flou de Mamdani est lisse et reste bornée et converge rapidement vers une valeur au voisinage de zéro, ce qui confirme les bonnes propriétés et performances d'approximation des systèmes à logique floue. A partir de ces figures, on peut bien voir que le système contrôlé se comporte bien dans toutes les situations (le cas de la poursuite et le cas de la régulation). En plus il est capable d'éliminer les perturbations introduites pendant tout le temps de test sans addition du terme superviseur à la loi de commande.

IV.2.5.2 Exemple 2

Dans cet exemple, on applique en simulation le contrôleur TS-Mamdani sur le système mécanique : masse- ressort- amortisseur étudié dans la première approche comme deuxième exemple, et qui est montré par la figure IV.8, et décrit par les équations (IV.83-86) ou directement par l'équation (IV.87).

L'objectif de cette commande est toujours de maintenir ou ramener la masse à l'origine de déplacement $x_1(t) = 0$. Les états initiaux sont toujours $(x_1(0), x_2(0)) = (1.5, 0)$ avec la même période d'échantillonnage qui est fixée à 0.01 s, et la plage de variation du déplacement $x_1(t)$ de la masse est $[-1.5, 1.5]$.

1) Les paramètres du premier système contrôleur TS sont choisis comme suit : $\gamma' = 0.5$, $v_0 = 0.065$, $k = [k_0 \ k_1]^T = [3.5 \ 1]^T$, $Q = \text{diag}(2, 2) > 0$, donc $\lambda_{Q_{\min}} = 2 > 0$ et par résolution de (IV.107) on obtient :

$$P = \begin{bmatrix} 4.78 & 0.28 \\ 0.28 & 1.28 \end{bmatrix} \quad (\text{IV.123})$$

La structure du contrôleur flou TS est la même que dans l'exemple 1 avec le même vecteur d'entrées $\underline{x} = \underline{z} = [z_1 \ z_2 \ z_3] = [\theta \ \theta_d \ (\theta_d^{(2)} - K^T \underline{e})]$ possédant les mêmes sous ensembles flous caractérisés par les fonctions d'appartenance données par (IV.100) à (IV.102). Les valeurs de c_N , c_Z et c_P sont fixées à : $c_N = -0.75$, $c_Z = 0$ et $c_P = 0.75$ pour chacune des trois entrées.

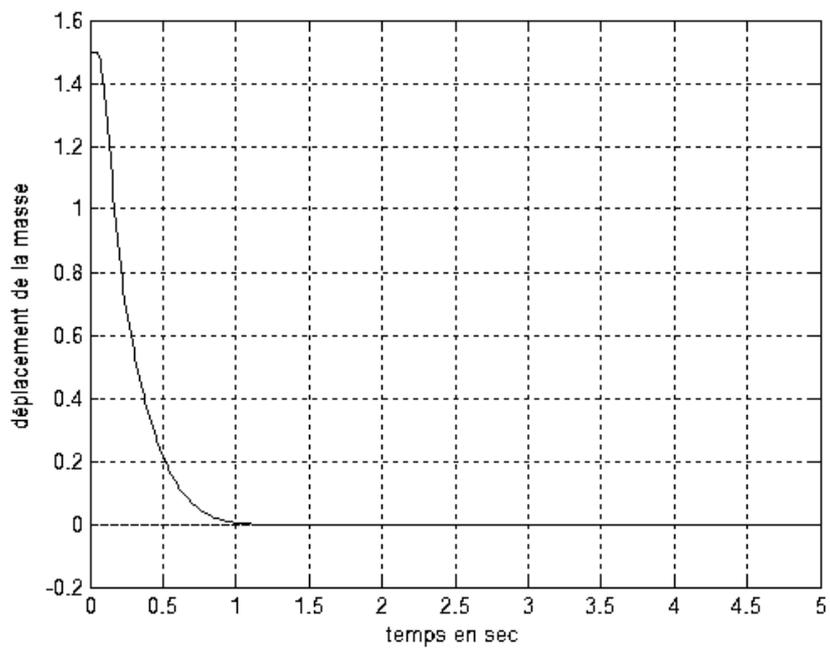
2) Le deuxième système estimateur flou Mamdani est implanté avec la base de règles RB1. Les paramètres définissant les fonctions d'appartenance comme montré dans la figure IV.14 sont choisis comme suit :

$c_N = -2$, $c_Z = 0$, $c_P = 2$ pour l'erreur e .

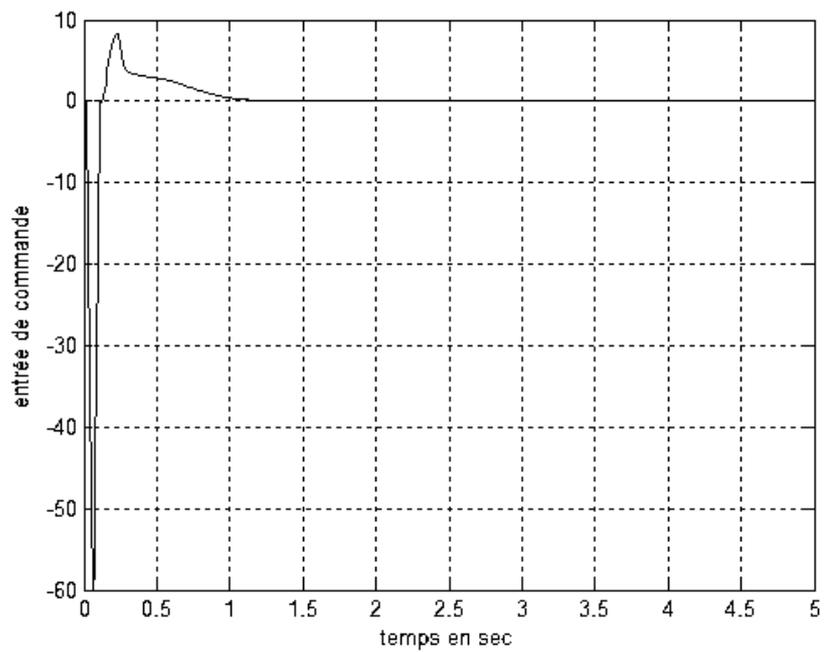
$c_N = -1$, $c_Z = 0$, $c_P = 1$ pour la variation de l'erreur de .

$c_N = -1.5$, $c_Z = 0$, $c_P = 1.5$ pour l'erreur de commande estimée \hat{e}_u .

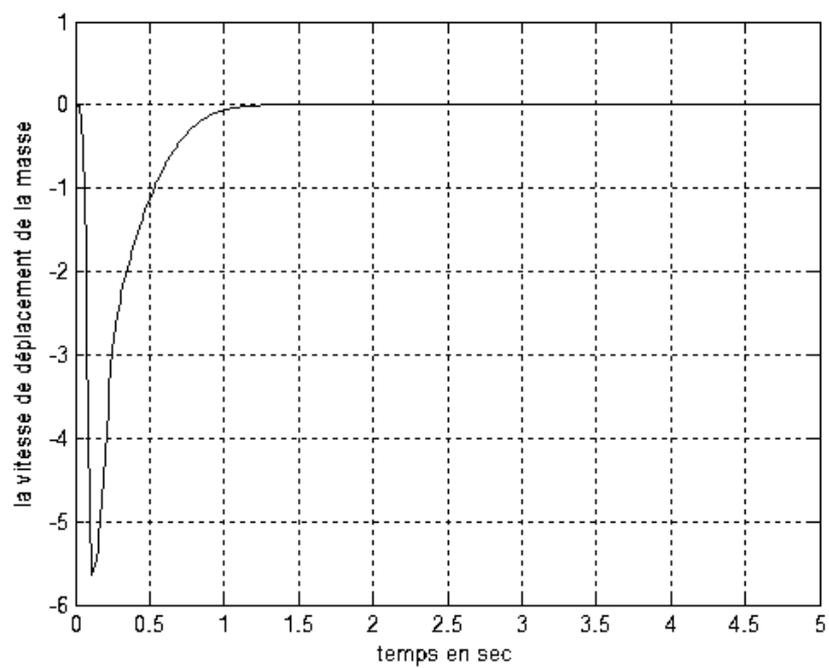
Les résultats de simulation sont montrés par les figures IV.20 aux IV.24.



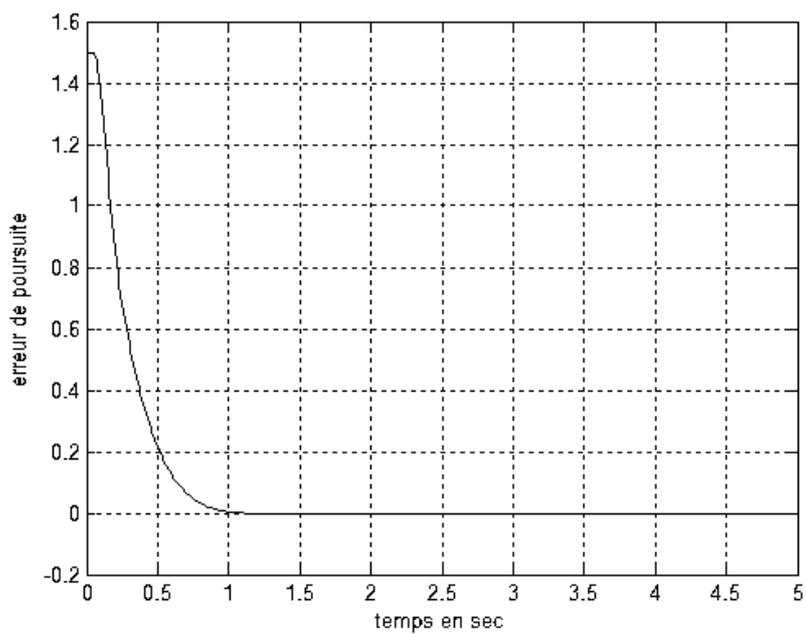
Figures IV.20 : Déplacement de la masse



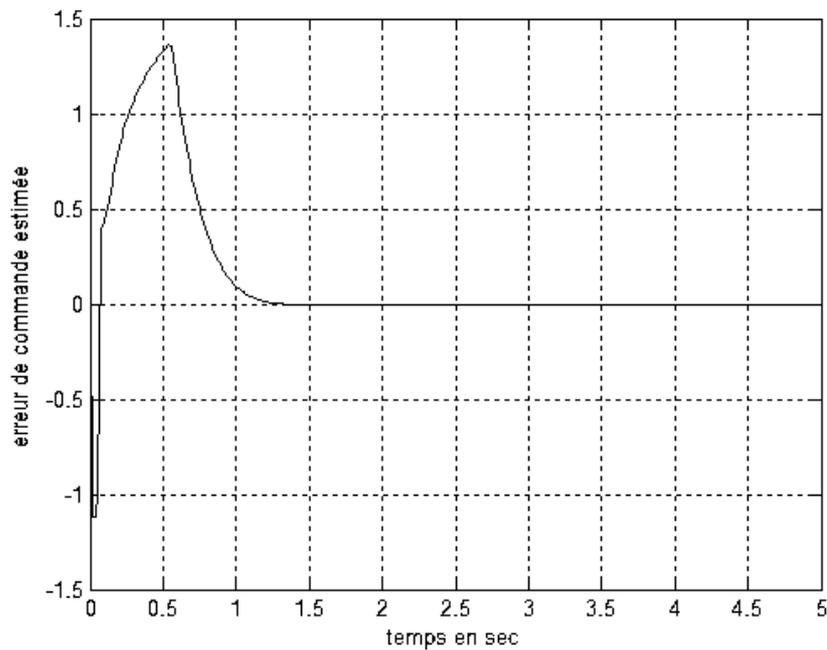
Figures IV.21 : signal de commande



Figures IV.22 : Vitesse du déplacement de la masse



Figures IV.23 : Erreur de poursuite



Figures IV.24 : Erreur de commande estimée \hat{e}_u

D'après la figure IV.20, on peut bien voir que la position du déplacement $x_1(t)$ se stabilise à la position d'équilibre $x_{1d}(t) = 0$ à l'instant $t = 1$ s, ce qui montre l'effet rapide du contrôleur. La figure IV.21 montre aussi que le contrôleur est borné en plus de son effet rapide et qu'il ne présente pas de pics importants pour ramener la sortie du système à la position d'équilibre. La figure IV.22 montre l'évolution de la vitesse du déplacement $x_2(t)$. D'après la figure IV.23 qui montre l'erreur entre la sortie du système $x_1(t)$ et la valeur désirée $x_{1d}(t) = 0$, on peut confirmer les bonnes performances de la poursuite. La figure IV.24 montre que l'erreur de commande estimée fournie par l'estimateur flou Mamdani est lisse et reste bornée et converge rapidement vers une valeur au voisinage de zéro, ce qui confirme les bonnes propriétés et performances d'approximations des systèmes à logique floue.

IV.2.6 Conclusion

Une commande adaptative floue directe pour une classe des systèmes non linéaires a été proposée. Un système d'inférence flou du type TS est utilisé pour approximer la loi de commande à linéarisation par retour. L'adaptation des paramètres de conséquence du système TS est dérivée à partir de la minimisation de l'erreur de commande et est basée sur l'algorithme de la descente du gradient. L'erreur de commande inconnue qui apparaît dans la loi d'adaptation est remplacée par une estimée fournie par un système FIS du type Mamdani. L'analyse de la stabilité est établie en se basant sur la théorie de Lyapunov. L'algorithme de commande TS-Mamdani est appliqué en simulation pour contrôler deux systèmes non linéaires. Un premier exemple est présenté pour traiter un problème de commande pour le système du pendule inversé pour différentes situations (poursuite et régulation). Un deuxième exemple est présenté pour traiter un problème de régulation ou de maintien à l'origine de la masse du système mécanique masse- ressort- amortisseur. Les résultats de simulation ont montré les propriétés lisses et les bonnes performances d'approximation des systèmes à logique floue.

Conclusion générale

Dans cette thèse, nous avons proposé deux approches de commande adaptative des systèmes non linéaires basées sur la linéarisation par retour d'état. Nous avons considéré le cas des systèmes non linéaires à une entrée, une sortie (SISO) et affines en l'entrée de commande. Pour ce développement, nous avons exploité la propriété d'approximation universelle des systèmes flous et des réseaux de neurones pour approximer d'une façon directe et indirecte la loi de commande idéale.

Dans la première partie de ce travail, deux solutions ont été proposées pour traiter le cas particulier où le gain de commande virtuel est constant. Dans la première approche, nous avons introduit comme contrôleur adaptatif direct un système d'inférence flou du type TS pour approximer le signal de commande idéal. Ce dernier est calculé et exprimé en premier lieu en se basant sur la théorie de linéarisation par retour d'état. Les paramètres de conséquence du contrôleur flou utilisé sont adaptés selon une loi dérivée en utilisant la théorie de stabilité de Lyapunov. La méthode proposée a pu garantir la stabilité asymptotique du système en boucle fermée dans le sens que tous les signaux sont uniformément bornés sans l'utilisation du terme superviseur dans la loi de commande. Nous avons testé en simulation l'algorithme flou TS adaptatif direct pour contrôler le niveau dans le système à trois bacs et pour la régulation d'un système instable. Les résultats de simulation ont montré les bonnes performances et la robustesse de la commande proposée. Dans la deuxième approche, de la même manière et comme alternative du contrôleur flou TS introduit dans la première approche, nous avons utilisé un système de réseau de neurone RBF en ligne pour approximer le signal de commande idéal. L'algorithme des *k-means* été utilisé pour l'ajustement des centres des fonctions de base radiales et les poids de connections sont adaptés selon une loi dérivée en utilisant la théorie de Lyapunov. La stabilité asymptotique du système global résultant en boucle fermée est garantie sans l'utilisation du terme superviseur dans la loi de commande. Les résultats de simulation pour le contrôle du niveau dans le système à trois bacs

et pour la régulation d'un système instable ont montré l'efficacité et la propriété de robustesse du système RBF proposé.

La deuxième partie de cette thèse concerne la commande adaptative des systèmes non linéaires avec un gain de commande virtuel dépendant des états. Deux approches ont été considérées: l'approche indirecte et l'approche directe. Dans l'approche indirecte TS-RBF, nous avons développé un schéma de commande adaptative indirecte avec un système d'inférence flou du type TS pour approximer le signal de commande idéal et un réseau RBF pour estimer le gain de commande virtuel qui est une partie inconnue des paramètres du système. Dans le réseau RBF, l'algorithme des *k-means* a été utilisé pour l'ajustement des centres des fonctions de base. Les paramètres du contrôleur TS et les poids de connections du réseau RBF sont ajustés en ligne selon des lois dérivées en utilisant la théorie de stabilité de Lyapunov. La méthode proposée TS-RBF a pu garantir la stabilité du système résultant en boucle fermée dans le sens que tous les signaux fournis étaient uniformément bornés sans addition du terme superviseur à la loi de commande. Les résultats de simulation sur deux systèmes : le pendule inversé et le système mécanique masse- ressort- amortisseur ont montré les bonnes performances et les propriétés lisses du schéma de commande TS-RBF proposé. Dans l'approche directe TS-Mamdani, nous avons considéré l'introduction d'une commande floue adaptative directe. La méthode utilise un système d'inférence flou du type TS pour approximer la loi linéarisante par retour d'état et un deuxième système d'inférence flou du type Mamdani pour estimer l'erreur du signal de commande qui n'est pas disponible et qui apparaît dans la loi d'adaptation des paramètres du contrôleur TS. Ces derniers ont été adaptés en utilisant la loi de la descente du gradient en se basant sur l'erreur de commande estimée. La base de règles de l'estimateur flou Mamdani a été construite à partir de connaissances d'un expert. L'algorithme de commande TS-Mamdani a été appliqué en simulation pour contrôler les deux systèmes non linéaires: Le pendule inversé pour différentes situations de poursuite et régulation et le système mécanique masse- ressort- amortisseur. Les résultats de simulation ont montré les propriétés et les bonnes performances d'approximation des systèmes à logique floue.

Comme perspective de ce travail, notre idée concerne la commande des systèmes non linéaires non affines en l'entrée de commande avec des états non accessibles à la mesure.

Bibliographie

- [1] P.A. Ioannou and J Sun, *Robust Adaptive Control*, Prentice Hall, New Jersey, 1996.
- [2] I.D. Landau, *Adaptive control: the model reference approach*, Marcel Dekker, 1979.
- [3] J.J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, New Jersey, 1991.
- [4] K.J. Astrorn and B. Wittenmark, *Adaptive control*, Addison-wesley, 2nd ed, 1995.
- [5] A. Isidori, *Nonlinear Control Systems*, Springer, Berlin, 1995.
- [6] R. Marino and P. Tomei, *Nonlinear Control Design: Geometric, Adaptive, and Robust*, Prentice Hall, New Jersey, 1995.
- [7] A.J. Krener, "Approximate linearization of state feedback and coordinate change," *Syst. Contr. Lett.*, vol. 5, pp. 181-185, 1984.
- [8] W.T. Baumann and W.J. Rugh, "Feedback control of nonlinear systems by extended linearization," *IEEE Trans. Autom. Contr.*, vol. 31, no. 1, pp. 40-46, 1986.
- [9] R. Ghanadan and G.L. Blankenship, "Adaptive control of nonlinear systems via approximation linearization," *IEEE Trans. Autom. Contr.*, vol. 41, pp. 618-625, 1996.
- [10] Y. Kim and F.L. Lewis, *High Level Feedback Control with Neural Networks*, World Scientific, 1998.
- [11] S.S. Sastry and A. Isidori, "Adaptive Control of Linearizable Systems," *IEEE Trans. Autom. Contr.*, vol. 34, no. 11, pp. 1123-1131, 1989.
- [12] D.G. Taylor, P.V. Kokotovic, R. Marino, and I. Kanellakopoulos, "Adaptive Regulation of Nonlinear Systems with Unmodeled Dynamics," *IEEE Trans. Autom. Contr.*, vol. 34, no. 4, pp. 405-412, 1989.
- [13] I. Kanellakopoulos, P.V. Kokotovic, and A.S. Morse, "Systematic Design of Adaptive Controllers for Feedback Linearizable Systems," *IEEE Trans. Autom. Contr.*, vol. 36, no. 8, pp. 1241-1253, 1991.
- [14] J.B. Pomet and L. Praly, "Adaptive Nonlinear Regulation: Estimation from the Lyapunov Equation," *IEEE Trans. Autom. Contr.*, vol. 37, no. 6, pp. 729-740, 1992.

- [15] M.M. Gupta and D.H. Rao, *Neuro-control systems: theory and applications*, IEEE Press, 1994.
- [16] G. Cybenko, "Approximation by Superpositions of Sigmoidal Function," *Math. Contr., Signals, Syst.*, vol. 2, no. 4, pp. 303-314, 1989.
- [17] F. L. Funahashi, "On the approximate realization of continuous mapping by neural networks," *Neural Networks*, vol. 2, no. 3, pp. 183-192, 1989.
- [18] K.S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on Neural Networks*, vol. 1, no. 1, pp. 4-27, 1990.
- [19] R. M. Sanner and J.J. Slotine, "Gaussian networks for direct adaptive control," *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 837-863, 1992.
- [20] L. X. Wang, "Stable adaptive fuzzy control of nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 2, pp. 146-155, 1993.
- [21] L.-X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least squares learning," *IEEE Trans. Neural Networks*, vol. 3, no. 5, pp. 807-814, 1992.
- [22] J. L. Castro, "Fuzzy logic controllers are universal approximators," *IEEE Trans. Syst. Man, Cybern.*, vol. 25, pp. 629-635, Apr. 1995.
- [23] Y. Diao and K. M. Passino, "Adaptive neural/fuzzy control for interpolated nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 5, pp. 583-595, 2002.
- [24] H. Lee and M. Tomizuka "Robust adaptive control using a universal approximator for SISO nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 1, pp. 95-106, 2000.
- [25] Y. Lee and H. Zak, "Uniformly ultimately bounded fuzzy adaptive tracking controllers for uncertain systems," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 6, pp. 797-811, 2004.
- [26] J.T. Spooner and K. M. Passino, "Stable adaptive control using fuzzy systems and neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 3, pp. 339-359, 1996.
- [27] D. Vélez-Díaz and Y. Tang, "Adaptive robust fuzzy control of nonlinear systems," *IEEE Trans. Systems man Cybernetics, SMC-B*, vol. 34, no. 3, pp. 1596 -601, 2004.
- [28] A. Wu and P. K. S. Tam, "Stable fuzzy neural tracking control of a class of unknown nonlinear systems based on fuzzy hierarchy error approach," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 6, pp. 779-789, 2002.

- [29] H. N. Nounou, and K. M. Passino, “Stable auto-tuning of adaptive fuzzy/neural controllers for nonlinear discrete-time systems” *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 1, pp. 70-83, 2004.
- [30] R. Boukezzoula, S. Galichet, and L. Foulloy, “Fuzzy adaptive control for non-affine systems,” in: *Proc. IEEE Internat. Conf. Fuzzy Systems*, pp. 543-548, 2003.
- [31] K. Fischle and D. Shroder, “An improved stable adaptive fuzzy control method,” *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 1, pp. 27–40, Feb.1999.
- [32] S. Labiod and T. M. Guerra, “Adaptive fuzzy control of a class of SISO non-affine nonlinear Systems,” *Fuzzy Sets and Systems*, vol. 158, pp.1126– 1137, 2007.
- [33] J.H. Park, G.T. Park, S.H. Kim, C.-J. Moon, “Direct adaptive self-structuring fuzzy controller for non-affine nonlinear system,” *Fuzzy Sets and Systems*, vol. 153, pp. 429–445, 2005.
- [34] C. H. Wang, H. Liu and T. Lin, “Direct adaptive fuzzy-neural control with state observer and supervisory controller for unknown nonlinear dynamical systems,” *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 1, pp. 39-49, 2002.
- [35] M. Wang, B. Chen and S. Daib, “Direct adaptive fuzzy tracking control for a class of perturbed strict-feedback nonlinear systems,” *Fuzzy Sets and Systems*, vol. 158, pp. 2655– 2670, 2007.
- [36] H. C. Anderson, A. Lotfi and A.C. Tsoi, “A new approach to adaptive fuzzy control: The controller output error method,” *IEEE Trans. Syst. , Man, Cybern.*, vol. 27, no. 4, pp. 686-691, 1997.
- [37] K. Belarbi and M. Chemachema, “Stable direct adaptive neural network controller with a fuzzy estimator of the control error for a class of perturbed nonlinear systems,” *IET Control Theory Appl.*, vol. 1, no. 5, pp. 1398–1404, 2007.
- [38] C. Darken, and J. Moody, “Fast Adaptive k-means Clustering: Some empirical Results,” *International Joint conf on Neural Networks.*, vol. 2, pp. 233-238, 1990.
- [39] Zheru Chi, and Hong. Yan, “Image segmentation using fuzzy rules derived from k-means Clusters,” *Journal of Electronic imaging*, vol. 4, no. 2, pp. 199-206, 1995.
- [40] S. S. Ge, C. C. Hang, and T. Zhang, “Adaptive Neural Network Control of Nonlinear Systems by State and Output Feedback,” *IEEE Trans. on Systems, Man, and Cybernetics part B, Cybernetics*, vol. 29, no. 6. pp. 818 – 828, 1999.

- [41] G. Feng, senior member IEEE, "A Survey on analysis and Design on Model Based-Fuzzy Control Systems," IEEE Transactions on Fuzzy systems, vol. 14, no. 5, pp. 676-697, 2006.
- [42] T. P. Chen and H. Chen, "Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis function neural networks," IEEE. Trans. Neural Networks, vol. 6, no. 4, pp. 904-910, 1995.
- [43] N. B. Karyanis, "Reformulated radial basis function neural network trained by gradient descent," IEEE transaction on neural network, vol. 10, no. 3, pp. 657- 669, May 1999.
- [44] S. Haykin, *Neural Networks, A Comprehensive Foundation*, Prentice-Hall, 1994.
- [45] Hugang Han, Chun-Yi Su, and Yury Stepanenko. "Adaptive Control of a Class of Nonlinear Systems with Nonlinearly Parameterized Fuzzy Approximators," IEEE Transactions on Fuzzy Systems, vol. 9, no. 2, pp. 315-323, 2001.
- [46] S. Chen, S. A. Billings, and P. M. Grant, "Recursive hybrid algorithm for nonlinear systems identification using Radial Basis Function Networks," International journal of Control, vol. 55, pp. 1051-1070, 1992.
- [47] Kevin. M. Passino and Stephen. Yurkovich., *Fuzzy Control* , Addison-Wesley, July 1997.
- [48] Krstic. M., Kanellakopoulos. L., and Kokotovic. P., *Nonlinear and Adaptive Control Design*, John Wiley and Sons, 1995.
- [49] Tsay, D.-L., Chung, H. Y., and Lee, C. J., "The Adaptive Control of Nonlinear Systems Using the Sugeno-Type of Fuzzy Logic," IEEE Transactions on Fuzzy Systems, vol.2, no. 7, pp. pp. 225-229, 1999.
- [50] M. Minoux, *Mathematical programming: theory and algorithms*, John Wiley and Sons, 1986.

Annexe:

Théorie de Stabilité de Lyapunov

A. Introduction à la théorie de stabilité de Lyapunov

A.1 Quelques définitions de la stabilité

A.1.1 Type de systèmes considérés et point d'équilibre

Nous considérons des systèmes non linéaires présentant un point d'équilibre en ($x=0$).

$$\dot{x} = f(x), \quad x \in \mathfrak{R}^n, \quad f(0) = 0 \quad (\text{A.1})$$

L'état $x = 0 \in \mathfrak{R}^n$ est un point (état) d'équilibre du système (A.1), si une fois $x(t)$ est égale à 0, il reste égale à 0 pour tout le temps future. Mathématiquement, cela signifie que le vecteur constant $x = 0 \in \mathfrak{R}^n$ satisfait la condition suivante $f(0) = 0$.

Donc, le point d'équilibre peut être trouvé par résolution de l'équation algébrique non linéaire $f(0) = 0$.

Notons que par changement de variables, il est toujours possible de ramener un point d'équilibre ($\tilde{x} / f(\tilde{x}) = 0$), à l'origine. Il suffit de poser ($x' = x - \tilde{x}$)

A.1.2 Stabilité au sens de Lyapunov

Le point d'équilibre ($x=0$) est stable au sens de Lyapunov, si et seulement si :

$$\forall \varepsilon > 0 \quad \exists \delta(\varepsilon) > 0 \quad / \quad \text{si } |x(0)| < \varepsilon \quad \text{alors } |x(t)| < \delta(\varepsilon) \quad (\text{A.2})$$

La stabilité au sens de Lyapunov traduit le fait que si nous plaçons un système dans un état proche d'un point d'équilibre stable au sens de Lyapunov, la trajectoire issue de cet état reste toute entière dans un voisinage de ce point d'équilibre.

A.1.3 Stabilité asymptotique

La stabilité asymptotique est nécessaire parce que dans beaucoup d'applications, la stabilité ordinaire (appelée stabilité de Lyapunov) n'est pas suffisante. Par exemple lorsque l'altitude d'un satellite est perturbée de sa position nominale, on ne veut pas seulement que le satellite maintienne son altitude dans le rang déterminé par la grandeur de la perturbation, i.e., la stabilité de Lyapunov, mais aussi exige que l'altitude revienne graduellement à sa valeur originale. Ce type d'exigence est capturé par le concept de la stabilité asymptotique.

Le point d'équilibre ($x=0$) est asymptotiquement stable, si et seulement si :

1) il est stable au sens de Lyapunov

$$2) \exists r > 0 \quad / \quad \text{si } |x(0)| < r \quad \text{alors } \lim_{t \rightarrow \infty} x(t) = 0. \quad (\text{A.3})$$

La stabilité asymptotique traduit le fait qu'il existe un voisinage du point d'équilibre du système, pour lequel toute trajectoire issue d'un état situé dans ce voisinage non seulement ne s'écarte jamais trop de ce point d'équilibre, mais finit toujours par le rejoindre.

A.1.4 Stabilité exponentielle

Le point d'équilibre $x=0$ est exponentiellement stable, si et seulement si :

1) il est asymptotiquement stable

$$2) \exists M > 0 \text{ et } \lambda > 0 \quad / \quad |x(t)| < M |x(0)| e^{-\lambda t} \quad \forall t > 0.$$

La stabilité exponentielle exprimée par la condition 2) signifie que le vecteur d'état d'un système exponentiellement stable converge à l'origine plus rapide qu'une fonction exponentielle, ou en d'autre terme, estime la rapidité de la trajectoire du système pour approcher le point d'équilibre ($x=0$). Le nombre positif λ est souvent appelé le taux de convergence exponentielle.

A.1.5 Stabilité globale

Si les différentes propositions définies ci-devant tiennent quel que soit l'état initial ($x(0) \in \mathfrak{R}^n$), l'on parle de stabilités globales. Un système ne peut posséder qu'un seul état d'équilibre globalement stable. Le domaine d'attraction est tout l'espace \mathfrak{R}^n . La notion de stabilité globale peut alors s'étendre au système dans son entier.

A.2 définition de positivité et de monotonie de fonctions

A.2.1 Fonctions définies positives négatives: au sens strict, au sens large

Considérons le vecteur x à n dimensions prenant ses valeurs dans un sous ensemble X du corps des réels \mathfrak{R}^n , et contenant le vecteur nul :

$$x \in X \subset \mathfrak{R}^n \quad / \quad x=0 \in X$$

Considérons une application $V(x)$ de X dans \mathfrak{R}^1

$$V : X \rightarrow \mathfrak{R}^1 \quad (\text{A.4})$$

fonction scalaire du vecteur x .

La fonction $V(x)$ est qualifiée de fonction définie strictement positive sur X , si et seulement si :

- 1) les dérivées partielles de $V(x)$, ou $\frac{\partial V(x)}{\partial x}$ sont continues $\forall x \in X$;
- 2) $V(0)=0$;
- 3) $V(x)>0, \quad \forall x \neq 0 \in X$.

Notons que $\frac{\partial V(x)}{\partial x}$ est un vecteur ligne, appelé gradient de $V(x)$ par rapport à x .

Dans le cas des fonctions définies positives au sens large (on dit aussi : semi- définies positives), la condition 3) devient ;

- 3') $V(x) \geq 0, \quad \forall x \neq 0 \in X$.

Pour les fonctions définies strictement négatives, nous avons semblablement :

$$3'') V(x) < 0, \quad \forall x \neq 0 \in X.$$

Et pour les négatives au sens large, (on dit aussi : semi- définies négatives) :

$$3''') V(x) \leq 0, \quad \forall x \neq 0 \in X.$$

A.2.2 Fonctions monotones croissantes, décroissantes : au sens strict, au sens large

Exprimons un vecteur quelconque ($x \in X \subset \mathfrak{R}^n$) suivant son module ($|x| = \sqrt{x^T x}$).

Etant donné que $|x|$ est toujours strictement positif sauf en $x=0$, suivant le signe de

$$\frac{\partial V(x)}{\partial x} x.$$

Ainsi :

Une fonction scalaire $V(x)$ d'un vecteur ($x \in X \subset \mathfrak{R}^n$) est une fonction qualifiée de monotone croissante au sens strict, si et seulement si :

1) si $\frac{\partial V(x)}{\partial x} x > 0 \quad \forall x \neq 0 \in X$, la fonction est monotone croissante au sens strict ;

1') si $\frac{\partial V(x)}{\partial x} x \geq 0 \quad \forall x \in X$, la fonction est monotone croissante au sens large ;

1'') si $\frac{\partial V(x)}{\partial x} x < 0 \quad \forall x \neq 0 \in X$, la fonction est monotone décroissante au sens strict ;

1''') si $\frac{\partial V(x)}{\partial x} x \leq 0 \quad \forall x \in X$, la fonction est monotone décroissante au sens large ;

A.2.3 Fonction de Lyapunov

Nous appellerons fonction de Lyapunov définie sur un sous ensemble X de \mathfrak{R}^n , contenant l'origine, toute fonction scalaire $V(x)$ telle que :

1) $V(x)$ est une fonction définie strictement positive sur X ;

2) $V(x)$ est une fonction monotone croissante au sens strict sur X ;

3) $\dot{V}(x) = \frac{dV(x)}{dt} = \frac{\partial V(x)}{\partial x} \dot{x}$ est une fonction définie négative au sens large sur X .

Soit donc, formulé en termes mathématiques :

$$1) V(x) > 0 \quad \forall x \neq 0 \in X \quad \text{et} \quad V(0) = 0 ;$$

$$2) \frac{\partial V(x)}{\partial x} x > 0 \quad \forall x \neq 0 \in X ;$$

$$3) \frac{\partial V(x)}{\partial x} \dot{x} \leq 0 \quad \forall x \in X ;$$

A.3 Théorème de stabilité de Lyapunov (seconde méthode ou méthode directe)

Soit un système présentant un point d'équilibre en $(x=0)$, défini par (A.1), c'est-à-dire:

$$\dot{x} = f(x), \quad x \in \mathfrak{R}^n, \quad f(0) = 0$$

Si sur un sous espace X de \mathfrak{R}^n contenant l'origine, il est possible de trouver une fonction de Lyapunov, alors l'origine $(x=0)$ est un point d'équilibre stable, et toutes les trajectoires issues d'un point quelconque du sous espace X tendent asymptotiquement vers l'origine.

ملخص

في هذه الأطروحة، نظرنا في السيطرة على أنظمة غير خطية داة إدخال واحد، وإخراج واحد (SISO) وأفيني (Affine) في مدخلات السيطرة. كمساهمة أولى في هذا العمل، تم تقديمها وفق نهجين مباشرين. في الأولى، كنظام تحكم استدلال غامض مباشر من نوع Takagi–Sugeno (TS)، وفي النهج الثاني، الشبكة العصبية من نوع دالة القاعدة الشعاعية (RBF) كوحدة تحكم بديلة لنظام التحكم الغامض المباشر TS. وكمساهمة رئيسية في هذه الرسالة، استخدمنا نهجين: النهج غير المباشر والنهج المباشر. في النهج غير المباشر، أدخلنا نظاما غامضا لتقريب إشارة التحكم المثالي وشبكة RBF لتقدير السيطرة الظاهرية. في النهج المباشر، استخدمنا نظام الاستدلال الغامض من نوع TS لتقريب قانون التحكم المثالي و نظام Mamdani الغامض لتقدير إشارة خطأ التحكم التي تظهر في قانون التحكم. تم بناء القواعد الأساسية في غامض Mamdani من معارف الخبراء. مقارب الاستقرار دائما مضمونة على أساس نظرية Lyapunov.

كلمات البحث: الخطية بإرجاع المدخلات، التحكم المتكيف للأنظمة غير الخطية، المنطق الغامض (Sugeno-Takagi)، الشبكة العصبية (RBF)، خوارزمية k-means، استقرار Lyapunov.

Abstract

In this thesis, we considered the control of nonlinear systems with single input, single output (SISO) and affine in the control input. As a first contribution in this work, two direct approaches are introduced. In the first, we used as a direct controller a fuzzy inference system of Takagi-Sugeno (TS) type. In the second approach, a neural network of Radial Basis Function (RBF) type is introduced as an alternative to the TS controller. In these two first approaches, the virtual control gain is constant.

As a main contribution in this thesis, two approaches are addressed: The indirect approach and the direct approach. In the indirect approach, we introduced a fuzzy system of TS type to approximate the ideal control signal and an RBF network to estimate the virtual control gain. In the direct approach, we used a TS fuzzy inference system to approximate the feedback linearization law and a second fuzzy system of Mamdani type to estimate the control error signal that appears in the adaptation law of the TS controller parameters. The rule base of the Mamdani fuzzy estimator is constructed from expert knowledge. Asymptotic stability based on Lyapunov theory is always guaranteed.

Keywords: Linearization by state feedback, Adaptive control of nonlinear systems, Fuzzy logic (Takagi-Sugeno, Mamdani), Neural network (RBF), k-means algorithm, Lyapunov Stability.

Résumé

Dans cette thèse, nous avons considéré la commande des systèmes non linéaires à une seule entrée, une seule sortie (SISO) et affines en l'entrée de commande. Comme premier apport dans ce travail, deux approches directes sont introduites. Dans la première, nous avons utilisé comme contrôleur direct un système d'inférence flou du type Takagi-Sugeno (TS). Dans la deuxième approche, un réseau de neurones à fonctions de base radiales (Radial basis Function : (RBF)) est introduit comme alternative du contrôleur TS.

Comme apport principale dans cette thèse, deux approches sont adressées : L'approche indirecte et l'approche directe. Dans l'approche indirecte, nous avons introduit un système flou du type TS pour approximer le signal de commande idéal et un réseau RBF pour estimer le gain de commande virtuel. Dans l'approche directe, nous avons utilisé un système d'inférence flou du type TS pour approximer la loi linéarisante par retour d'état et un deuxième système flou du type Mamdani pour estimer l'erreur du signal de commande qui apparaît dans la loi d'adaptation des paramètres du contrôleur TS. La base de règles de l'estimateur flou Mamdani est construite à partir de connaissances d'un expert. La stabilité asymptotique basée sur la théorie de Lyapunov est toujours garantie.

Mots clefs: Linéarisation par retour d'état, Commande adaptative des systèmes non linéaires, Logique floue (Takagi-Sugeno, Mamdani), Réseau de neurones (RBF), Algorithme des k-means, Stabilité de Lyapunov.

Résumé

Dans cette thèse, nous avons considéré la commande des systèmes non linéaires à une seule entrée, une seule sortie (SISO) et affines en l'entrée de commande. Comme premier apport dans ce travail, deux approches directes sont introduites. Dans la première, nous avons utilisé comme contrôleur direct un système d'inférence flou du type Takagi-Sugeno (TS). Dans la deuxième approche, un réseau de neurones à fonctions de base radiales (Radial basis Function : (RBF)) est introduit comme alternative du contrôleur TS.

Comme apport principale dans cette thèse, deux approches sont adressées : L'approche indirecte et l'approche directe. Dans l'approche indirecte, nous avons introduit un système flou du type TS pour approximer le signal de commande idéal et un réseau RBF pour estimer le gain de commande virtuel. Dans l'approche directe, nous avons utilisé un système d'inférence flou du type TS pour approximer la loi linéarisante par retour d'état et un deuxième système flou du type Mamdani pour estimer l'erreur du signal de commande qui apparaît dans la loi d'adaptation des paramètres du contrôleur TS. La base de règles de l'estimateur flou Mamdani est construite à partir de connaissances d'un expert. La stabilité asymptotique basée sur la théorie de Lyapunov est toujours garantie.

Mots clefs: Linéarisation par retour d'état, Commande adaptative des systèmes non linéaires, Logique floue (Takagi-Sugeno, Mamdani), Réseau de neurones (RBF), Algorithme des k-means, Stabilité de Lyapunov.