

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR

ET DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITE CONSTANTINE 1

FACULTE DES SCIENCES DE LA TECHNOLOGIE

DEPARTEMENT D'ELECTRONIQUE

N° d'ordre : .....

Série : .....

## **MEMOIRE**

**Présenté pour obtenir le Diplôme de Magistère en**

**ELECTRONIQUE**

**Spécialité : Contrôle des Systèmes**

**THEME :**

***Sur La Commande Adaptative des Systèmes Non Linéaires :  
Approche Basée sur La Stabilité de Lyapunov***

**PAR**

**ZERARI NASSIRA**

**Soutenu le 26 / 11 / 2014**

**Devant le Jury :**

**Président : Chaabi Abdelhafid Prof. Université Constantine 1**

**Rapporteur : Chemachema Mohamed M.C. Université Constantine 1**

**Examineurs : Belarbi khaled Prof. Ecole National Polytechnique  
Constantine**

**Bououden Sofiane M.C. Université de Khenchela**

---

## **Remerciements**

---

*En tout premier lieu, notre profonde gratitude est envers notre créateur, le tout puissant, qui nous a spécifiés de l'intelligence pour bien croire à sa majesté. Nous tenons à remercier profondément notre promoteur **M. CHEMACHEMA** pour son encadrement de haut niveau, ses précieux conseils et ses critiques constructives et pertinentes.*

*J'exprime ma plus profonde gratitude à Monsieur le président de jury **A. Chaabi**, Professeur à l'université de Constantine 1 d'avoir bien voulu juger ce travail et d'avoir bien voulu accepter la présidence de ce jury.*

*J'adresse également mes sincères remerciements aux membres de jury d'avoir accepté la lourde tâche d'être examinateurs et qui me font l'honneur de jouer ce rôle important de ma mémoire. Je remercie particulièrement Monsieur **K. Belarbi** professeur à l'École Nationale Polytechnique de Constantine je n'oublierai jamais ses aides appréciables, ainsi que Monsieur **S. Bououden**, Maître de conférences à l'université de Khenchela qui viennent de loin et qui ont accepté immédiatement de faire partie de ce Jury*

*Que tous mes amis et collègues trouvent ici le témoignage de mon amitié et de ma reconnaissance.*

*Enfin, je ne saurais terminer ce remerciement sans remercier toute ma famille, qui m'a accompagné tout au long de mes études par son amour inconditionnel et son soutien constant.*

---

## *Table des matières*

---

Remerciements.....	i
Table des matières.....	ii
Résumé.....	iv
Abstract.....	v
ملخص.....	vi
<b><i>Introduction Générale.....</i></b>	<b>1</b>
<b><i>CHAPITRE 1. Les Réseaux de Neurones.....</i></b>	<b>4</b>
1.1. Introduction .....	5
1.2. Réseaux de Neurones Artificiels .....	5
1.3. L'approximation universelle par réseaux de neurones d'ordre supérieur HONN.....	6
1.3.1. Topologie du réseau HONN .....	6
1.3.2. Apprentissage des Réseaux de Neurone.....	7
1.3.2.1 Algorithmes d'ajustement.....	7
1.3.3. Théorème de l'approximation universelle.....	9
1.4. Conclusion.....	10
<b><i>CHAPITRE 2 .La Commande Adaptative Neuronale pour une Classe de Système Non Linéaire Monovariabiles Non-affine.....</i></b>	<b>11</b>
2.1. Introduction.....	12
2.2. La Transformation Du Système Non Linéaire.....	13
2.2.1. Position du problème .....	13
2.2.2. Représentation entrée-sortie .....	14
2.2.3. La commande Système de linéairisante .....	19
2.3. Commande adaptative indirecte neuronale .....	21
2.3.1. Conception de la commande neuronale.....	21
2.3.2. Analyse de stabilité .....	24
2.3.3. Résultats de simulation .....	28
2.4. Commande adaptative directe neuronale .....	31

2.4.1. Conception de la commande neuronale.....	31
2.4.2. Analyse de stabilité .....	32
2.4.3. Résultats de simulation .....	37
2.5. Conclusion.....	40
<b><i>Chapitre 3. La Commande Adaptative Neuronale pour une Classe de Système Non Linéaire Multivariables Non affine.....</i></b>	<b>41</b>
3.1. Introduction.....	42
3.2. La Transformation du système non Linéaire MIMO .....	43
3.2.1. Position du problème .....	43
3.2.2. Conception de la commande neuronale.....	48
3.2.3. Analyse de stabilité .....	50
3.3. Résultats de simulation .....	52
3.3.1. Exemple 1.....	52
3.3.2. Exemple 2 .....	54
3.4 Conclusion.....	57
<b><i>Conclusion générale.....</i></b>	<b>59</b>
<b><i>Bibliographie.....</i></b>	<b>61</b>

---

## *Résumé*

---

La commande adaptative neuronale peut être définie comme une méthode de contrôle robuste, dans laquelle les systèmes neuronaux sont utilisés pour approximer la dynamique du système où certains de ses paramètres sont adaptés en ligne afin de garantir la stabilité du système en boucle fermée. L'objectif principal de ce mémoire est de développer des lois de commande adaptatives neuronales stables pour réaliser la poursuite des certaines classes de systèmes non linéaires SISO et MIMO non affines en la commande. Pour ce faire, nous avons exploité en premier lieu le théorème de la fonction implicite pour montrer l'existence du contrôleur idéal inconnu. Ensuite, nous avons utilisé le développement en série de Taylor autour du contrôleur idéal pour transformer le modèle non affine en un modèle affine. A base de ce résultat nous avons proposé et développé des schémas de commande adaptative directe et indirecte pour une classe de systèmes non linéaires monovariabiles et multivariabiles, Le point commun entre les approches proposées est l'utilisation des systèmes neuronaux d'ordre supérieure HONN pour approcher en ligne les dynamiques incertaines des systèmes dans le cas des approches indirectes, et pour approcher des lois de commande stabilisatrices inconnues dans le cas des approches directes. De plus, l'analyse de la stabilité et de la robustesse des structures de commande proposées est effectuée par l'approche de Lyapunov. Pour chaque structure de commande proposée, des résultats de simulation sont présentés pour montrer ses performances.

### **Mots Clés :**

Réseau de neurones (HONN), commande adaptative neuronale, théorème de la fonction implicite, systèmes non linéaires non affines.

---

## *Abstract*

---

Adaptive neural networks control can be defined as a method of robust control, in which neural networks systems are used to approximate unknown functions, some of the system parameters are adapted online to ensure the stability of the closed loop system. The main objective In this thesis, An adaptive neural networks control approach is developed and proposed for a class of single-input single-output (SISO), and multi-input multi-output (MIMO) nonaffine nonlinear dynamic. Based on the implicit function theory, the existence of an ideal controller, that can achieve control objectives, is firstly shown. Then, Taylor series expansion is employed to transfer the normal nonlinear system into the standard affine form in the neighborhood of the ideal. Based on this elegant result , we have proposed adaptive neural controller direct and indirect for a class of single-input single-output (SISO) nonlinear systems, and multi-input multi-output (MIMO) nonlinear systems, The common feature between all developed adaptive controllers is the use of neural networks systems, that are updated on-line, with the purpose of producing approximations of the system's dynamics, in some cases (indirect methods), or of some unknown stabilizing controllers, in others (direct methods). In addition, stability and robustness analysis of the proposed control schemes are performed by using the Lyapunov synthesis method, and for each scheme, simulation results are given to highlight its performance.

**Key words:**

Neural network (HONN), adaptive control neuronal, implicit function theorem, nonlinear non affine Systems.

---

## ملخص

---

التحكم التلاؤمي العصبي يمكن تعريفه كوسيلة من وسائل التحكم القوية، والتي تستخدم الشبكة العصبية لتقريب دوال غير معروفة، أين يتم تكيف وسائط النظام لضمان استقرار النظام في حلقة مغلقة. الهدف الرئيسي لهذه الأطروحة هو وضع قوانين التحكم التلاؤمي العصبي لتحقيق استمرار مستقر من فئات معينة من الأنظمة غير الخطية أحادية المدخل و المخرج و الأنظمة غير الخطية متعددة المداخل و المخرج حيث التحكم يظهر بصورة غير خطية. من أجل ذلك، استخدمنا نظرية الدالة الضمنية لإظهار وجود الم تحكم المثالي الضمني. بعد ذلك، أدخلنا سلسلة تايلور حول التحكم المثالي الضمني حيث التحكم يظهر بصورة غير خطية من أجل تحويله إلى نموذج حيث التحكم يظهر بصورة خطية. بالاعتماد على هذه النتائج، قمنا بتطوير واقتراح مخططات التحكم التلاؤمي المباشر وغير المباشر للأنظمة الديناميكية اللاخطية أحادية ومتعددة المداخل والمخرج، القاسم المشترك بين طرق التحكم المدروسة يتمثل في استعمال الشبكة العصبية من نوع الدرجة المرتفعة كمثل عام للدوال المجهولة من أجل الحصول على تمثيل للمعادلة الديناميكية للنظام اللاخطي في بعض الحالات (الطريقة غير المباشرة) من أجل الحصول على تمثيل لقانون تحكم مثالي مجهول في الحالات الأخرى (الطريقة المباشرة)، بالإضافة إلى هذا، استخدمنا طريقة ليابونوف لدراسة استقرار ومتانة طرق التحكم التلاؤمي المقترحة وقدمنا بعض نتائج المحاكاة العددية لإبراز كفاءة أدائها.

### كلمات مفتاحية:

تحكم تلاؤمي، الشبكة العصبية ذات الدرجة المرتفعة، نظرية الدالة الضمنية، الأنظمة غير الخطية حيث التحكم يظهر في صورة غير خطية، استقرار ليابونوف.

---

## *Introduction Générale*

---

Depuis quelques années, les recherches sur la commande et l'automatisation industrielle se sont multipliées. Différentes structures de linéarisation ont été proposées pour résoudre le problème de la commande des systèmes non-linéaires. Contrairement à l'automatique linéaire, l'automatique non linéaire ne dispose pas de solutions universelles ni pour l'analyse des systèmes ni pour la conception de leurs contrôleurs. La plupart des approches de commande non linéaires exigent la connaissance d'un modèle mathématique du système. Les performances assurées seront directement liées à l'exactitude du modèle utilisé.

Les techniques de la linéarisation du premier ordre sont utilisées pour linéariser les systèmes non linéaires autour d'un point d'opération. Etant donné que c'est la première technique qui a été introduite pour la commande des systèmes non linéaire, elle a trouvé une large acceptation dans le monde industriel. Par ailleurs cette solution reste, cependant valable uniquement autour de ce point d'opération, car il est difficile de prédire l'effet de la non linéarités si on s'éloigne de ce point.

Sur le plan théorique, ce problème a été résolu par le développement de techniques de commande non linéaire basées sur la théorie de la géométrie différentielle. L'idée principale derrière cette théorie est de trouver une transformation non linéaire des états du système par lesquelles une classe des systèmes non linéaires peut être transformée en systèmes linéaires à travers le retour d'état, connu sous l'appellation anglophone **feedback linearization**. Cette approche a souvent été employée pour résoudre des problèmes pratiques de commande mais elle impose que le vecteur d'état soit mesuré et que le modèle du procédé à commander soit précis. De plus, les propriétés de robustesse ne sont pas garanties face aux incertitudes paramétriques du modèle. En effet, cette technique est basée sur l'annulation exacte des termes non linéaires. Par conséquent, la présence d'incertitudes de modélisation sur les termes non linéaires rend l'annulation inexacte et l'équation entrée-sortie résultante non linéaire.



A ce stade, et afin de pallier à ces problèmes décrits précédemment, la commande adaptative des systèmes non linéaires trouve un grand essor et elle est la voie de recherche de plusieurs travaux. Une des approches les plus intéressantes consiste à utiliser les réseaux de neurones artificiels, RNA, associés à un mécanisme d'adaptation. L'idée principale est d'utiliser ces réseaux pour approcher la dynamique du système ou pour approximer le contrôleur. En plus, Le système neuronale a la capacité d'approximer n'importe quelle dynamique non linéaire avec un degré de précision arbitraire. Les RNA à base de perceptrons multicouches, Multi Layer Perceptrons, MLP, et les fonctions radiales de base RBF, et les réseaux de neurones d'ordre supérieure HONN, ont trouvé une large acceptation dans le domaine de la commande adaptative des systèmes non linéaires.

Un grand nombre de travaux sont disponibles sur la commande adaptative neuronale pour les systèmes affines en la commande, contrairement aux systèmes non-affines c'est-à-dire des systèmes caractérisés par des entrées de commande qui n'apparaissent pas linéairement dans l'équation d'état du système. En raison du manque d'outils mathématiques pour les systèmes non-affines par rapport aux systèmes affines, la linéarisation par rétroaction utilisé pour de nombreux systèmes affines n'est pas directement applicable aux systèmes non affine. La conception de commande adaptative neuronale pour les systèmes non affines est basée sur le théorème de fonction implicite pour montrer l'existence d'une loi de commande idéale. Afin de simplifier le développement de loi de commande qui approxime la loi idéale, le théorème de la valeur moyenne ou la linéarisation par la méthode de Taylor sont exploités.

Le travail présenté dans ce mémoire est consacré au développement de lois de commande adaptative pour les systèmes non linéaires incertains dont le modèle est sous la forme générale non affines en la commande. Les deux types des systèmes monovariabiles et multivariabiles sont traités dans le développement de lois de commande. Les techniques développées dans ce mémoire assurent la stabilité et la robustesse des boucles de commande. Chaque technique proposée est testées par exemple de simulation pour confirmer ses performances. Le mémoire est organisé en trois chapitres :

Dans le premier chapitre, une description du réseau de neurone d'ordre supérieure utilisé est détaillée en donnant ses notions essentielles avec la formulation mathématique adéquate à son exploitation pour la commande adaptative.

Le deuxième chapitre présente le développement de la commande adaptative neuronale pour une classe de système non linéaire monovariabiles non affine en la commande sous la forme générale, où le développement en série de Taylor autour d'un contrôleur idéal inconnu est utilisé pour transformer le système non affine en un système affine. Et puis deux

approches distinctes se sont formulées pour la commande adaptative neuronale, après transformation, selon l'approche directe et l'approche indirecte. Dans l'approche directe, les systèmes neuronaux sont utilisés pour approximer un contrôleur idéal inconnu. En revanche la commande indirecte exploite les systèmes neuronaux pour estimer la dynamique inconnue du système non linéaire à commander afin de synthétiser des lois de commande basées sur ces estimations. Le problème de la singularité engendré par l'approche indirecte est résolu par l'utilisation d'un algorithme de projection.

Le troisième chapitre présente la commande adaptative neuronale directe pour une classe de systèmes non linéaires multivariables MIMO non affine en la commande. Pour ce faire, on exploite en premier lieu le théorème de la fonction implicite pour montrer l'existence du contrôleur idéal inconnu. Ensuite, on utilise le développement en série de Taylor autour du contrôleur idéal pour transformer le modèle non affine en un modèle affine. En fin, pour des raisons des incertitudes sur le modèle liées aux problèmes de transformations et modélisation, un réseau de neurone est utilisé pour approximer la loi de commande idéale. Les lois d'adaptation des paramètres ajustables sont déduites à partir de l'étude de stabilité au sens de Lyapunov. La robustesse étant assurée par l'ajout d'un terme de commande type mode glissant.

# *Chapitre 1*

## **Réseaux de Neurones**

---

**1. 1. Introduction**

**1. 2. Réseaux de neurones artificiels (ANN)**

**1. 3. L'approximation universelle par réseaux HONN**

**1. 4. Conclusions**

## 1.1. Introduction :

Les réseaux de neurones artificiels (ANN) jouent un rôle important dans l'ingénierie de contrôle, en particulier dans le contrôle du système non linéaire. En raison de leur excellente capacité d'apprentissage, les réseaux de neurones artificiels (ANN) ont la propriété d'approximation universelle, qui les rend capables d'approximer, avec un degré de précision arbitraire fixé, n'importe quelle fonction non linéaire.

La recherche active a été réalisée dans le contrôle du réseau de neurones en utilisant le fait que les réseaux de neurones peuvent se rapprocher d'une large gamme de fonctions non linéaires à un degré de précision désiré dans certaines conditions. Plusieurs approches de contrôle NN (neural networks) stables ont été proposées sur la base de la théorie de la stabilité de Lyapunov [Yesildirek-96] [Ge-01]. Un avantage principal de ces systèmes est que les lois adaptatives sont établies d'après la synthèse de Lyapunov et par conséquent, la stabilité du système est garantie sans l'exigence d'une formation en ligne. Il existe trois types de réseaux neuronaux sont largement utilisés, les réseaux de neurones de la fonction de base radiale (RBF) [Ge-01][Yesildirek-99], et les réseaux de neurones d'ordre supérieur (HONNS) [Kosmatopoulos-95], et les réseaux neuronaux multicouches (MNNS) [Ge-01][Yesildirek-99].

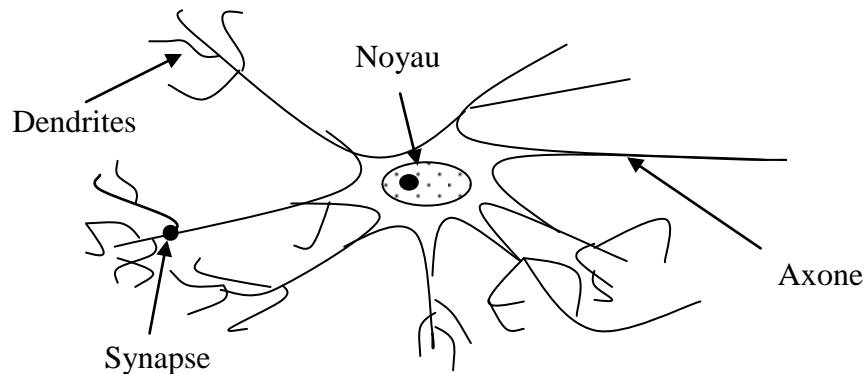
L'objectif de ce chapitre est présente la structure et les propriétés du réseau de neurone d'ordre supérieure HONN (high-order neural networks), et leur utilisations dans la conception de contrôleur.

## 1.2. Réseaux de Neurones Artificiels (ANN) :

Le cerveau humain est le meilleur modèle de machine polyvalente incroyablement rapide et surtout douée d'une parfaite capacité d'auto organisation. Son comportement est beaucoup plus mystérieux que le comportement de ses cellules de base. Il est constitué d'unités biologiques élémentaires (environ  $10^{12}$  neurones), chacune reçoit et envoie des informations (1000 à 10000 synapses par neurone). Les cellules nerveuses appelées "neurones" qui sont des éléments de base du système nerveux centrale. Elles sont constituées de trois parties essentielles, le corps cellulaire, les dendrites et l'axone comme indiqué dans la figure 1.1.

Le corps cellulaire contient le noyau de neurone et effectue les transformations biochimiques nécessaires à la synthèse des éléments assurant la vie du neurone. Chaque neurone possède des dendrites, qui entourent le corps cellulaire. Les dendrites sont les récepteurs principaux du neurone pour capter les signaux qui lui parviennent. L'axone qui

est à proprement parler la fibre nerveuse, sert de moyen de transport pour les signaux émis par le neurone. Les neurones sont connectés les uns aux autres suivant des répartitions spatiales complexes. Les connexions entre deux neurones sont réalisées au niveaux des synapses où ils sont séparés par un espace intercellulaire de quelques dizaines d'Angströms ( $10^{-9} m$ ) entre l'axone du neurone afférent et les dendrite du prochain neurone. La jonction entre deux neurones est appelée synapse.



**Fig. 1. 1:** Structure d'un neurone biologique

Un ANN traite l'information qu'il reçoit d'une manière analogue aux neurones du cerveau. Il est constitué de plusieurs éléments processeurs dits neurones artificiels reliés les uns aux autres par un réseau complexe. La sortie de chaque neurone peut être reliée en entrée à plusieurs autres neurones. Chaque neurone effectue une somme pondérée des signaux d'entrée modulés par une fonction dite activation (une fonction linéaire ou non linéaire) et génère une sortie qui sera appliqué aux autres neurones via des connexions pondérées. Avec cette simple structure et en choisissant un nombre approprié de neurones, ces réseaux sont capables d'approximer n'importe quelle fonction continue avec une certaine précision.

### 1.3. L'approximation universelle par réseaux de neurones HONN :

#### 1.3.1. Topologie du réseau HONN :

Le réseau de neurone d'ordre supérieure HONN (high-order neural networks) est un type de réseau de neurone linéaire paramétré. En effet, HONN ont des excellentes capacités d'approximation de n'importe quelle fonction continue non linéaire avec un degré de précision arbitraire fixe. La structure de HONN est exprimée comme suit :

$$\phi(W, z) = W^T S(z) \quad W, S(z) \in \mathfrak{R}^l \quad (1.1)$$

$$S(z) = [s_1(z), s_2(z), \dots, \dots, s_l(z)]^T \quad (1.2)$$

$$s_i(z) = \prod_{j \in I_i} [s(z_j)]^{d_{j,i}}, \quad i = 1, 2, \dots, l \quad (1.3)$$

où  $z \in \Omega_z \in \mathfrak{R}^m$  est l'entrée de HONN,  $l$  est le nombre de nœuds du réseau de neurone,  $\{I_1, I_2, \dots, I_l\}$  est un ensemble de  $l$  sous ensemble non ordonnée de  $\{1, 2, \dots, m\}$ , spécifiée par le concepteur,  $d_{j,i}$  sont des entiers positifs,  $W$  sont les vecteurs de paramètres ajustés, et  $s(z_j)$  est la fonction d'activation, linéaire ou de forme sigmoïde, elle est supposée dans ce travail comme une fonction de tangente hyperbolique et donnée par l'équation suivante:

$$s(z_j) = (e^{z_j} - e^{-z_j}) / (e^{z_j} + e^{-z_j}) \quad (1.4)$$

La structure de HONN est représentée sur la figure 1.2, où les lignes pointillées signifient qu'ils peuvent être connectés ou non connecté dépendants de  $I_i$ , et  $d_{j,i}$  la puissance de la fonction  $s(\cdot)$ .

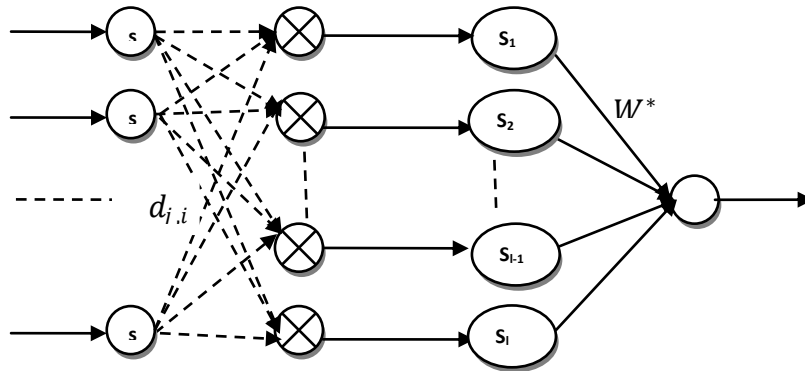


Fig. 1.2: La structure du neurone d'ordre supérieure (HONN)

### 1.3.2. L'apprentissage des réseaux de neurones:

L'information que peut acquérir un réseau de neurones est représentée dans les poids des connexions entre les neurones. L'apprentissage consiste donc à ajuster ces poids de telle façon que le réseau présente certains comportements désirés. En d'autres termes, l'apprentissage des réseaux de neurones consiste à ajuster les poids synaptiques de telle manière que les sorties du réseau soient aussi proches que possible des sorties désirées.

#### 1.3.2.1. Algorithmes d'ajustement :

Dans de nombreux problèmes pratiques, nous pouvons avoir une connaissance a priori de l'endroit où les paramètres idéales  $W^*$  sont situés dans  $\mathfrak{R}^n$ . Cette connaissance vient généralement en termes de limites supérieure et / ou inférieure pour les éléments de  $W^*$  ou en termes de localisation dans un sous-ensemble convexe de  $\mathfrak{R}^n$ . Si une telle information a priori est disponible, on veut limiter l'estimation en ligne pour être à l'intérieur de l'ensemble dans lequel se trouvent les paramètres inconnus. A cet effet, nous modifions les

algorithmes de gradient basé sur la minimisation sans contrainte de certains coûts en utilisant la méthode de projection du gradient [Ioannou-96].

L'algorithme du gradient avec projection est calculé en appliquant la méthode de gradient pour le problème de minimisation des contraintes suivantes:

$$\begin{aligned} & \text{Minimiser } J(W, e) \\ & \text{Soumise à } W \in \Omega_w \end{aligned}$$

En général, l'algorithme d'ajustement (de la loi d'adaptation) prend la forme :

$$\dot{W} = -\gamma \nabla J(W, e) = -\gamma \psi(z)e \quad (1.5)$$

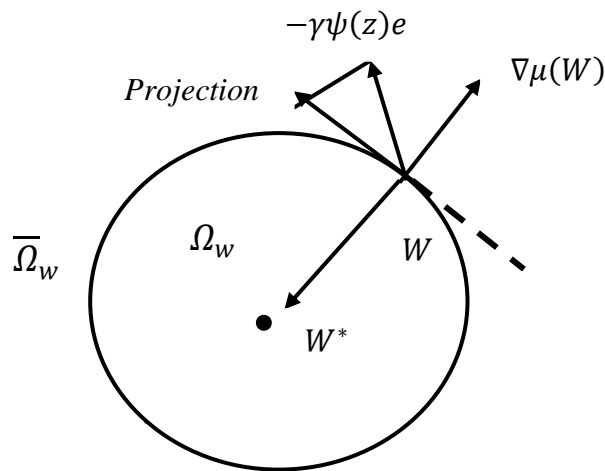
où :

$$\nabla J(W, e) = \frac{\partial J(W, e)}{\partial W} \text{ et } \gamma > 0$$

où  $\Omega_w$  c'est une région convexe définie par :

$$\Omega_w = \{W / \mu(W) \leq 0\}$$

avec  $\mu(w)$  est une fonction lisse.



**Fig. 1.3** : schématisation de l'algorithme de Projection

Un des moyens les plus simples et efficaces pour éviter la dérive des paramètres est de limiter les estimations des paramètres dans une région bornée et convexe  $\Omega_w$ , qui est choisie de telle sorte que  $W^* \in \Omega_w$  (figure 1.3). En outre, les conditions initiales  $W(0)$  sont choisies de telle sorte que  $W(0) \in \Omega_w$ .

La modification de projection met en œuvre cette idée comme suit : si l'estimation du paramètre  $W(t)$  est à l'intérieur de la région souhaitée  $\Omega_w$  ou sur le contour de cette région  $\Omega_w$ , avec sa dérivée pointant vers l'intérieur de la région, la loi d'adaptation (1.5) est mise en œuvre. Maintenant, si l'estimation du paramètre  $W(t)$  est sur le contour de cette région  $\Omega_w$  avec sa dérivée pointant vers l'extérieur de la région, alors sa dérivée est

projetée sur l'hyperplan tangent à  $\Omega_w$ . Par conséquent, la projection conserve le vecteur d'estimation de paramètres dans la région convexe  $\Omega_w$  tout le temps.

Enfin, l'algorithme de projection est défini comme suit :

$$\dot{W} = \begin{cases} -\gamma\psi(z)e & \text{si } |W| \in \Omega_w \quad \text{ou } (|W| \in \bar{\Omega}_w \quad \text{et } \nabla\mu(W)^T \gamma\psi(z)e \geq 0 \\ -\gamma\psi(z)e + \frac{\nabla\mu(W)\nabla\mu(W)^T}{\nabla\mu(W)^T\nabla\mu(W)} \gamma\psi(z)e & \text{si } |W| \in \bar{\Omega}_w \quad \text{et } \nabla\mu(W)^T \gamma\psi(z)e < 0 \end{cases}$$

$$\text{où } \nabla\mu(W) = \frac{\partial\mu(W)}{\partial W}$$

Si on choisit de contraindre les paramètres estimés tel que  $|W| \leq W_{max}$ , alors nous avons

$$\mu(W) = |W|^2 - W_{max}^2$$

$$\text{et } \nabla\mu(W) = 2W$$

L'algorithme de projection peut être écrit comme :

$$\dot{W} = \begin{cases} -\gamma\psi(z)e & \text{si } |W| \in \Omega_w \quad \text{ou } (|W| = W_{max} \quad \text{et } W^T \gamma\psi(z)e \geq 0 \\ -\gamma\psi(z)e + \frac{WW^T}{W^TW} \gamma\psi(z)e & \text{si } |W| = W_{max} \quad \text{et } W^T \gamma\psi(z)e < 0 \end{cases}$$

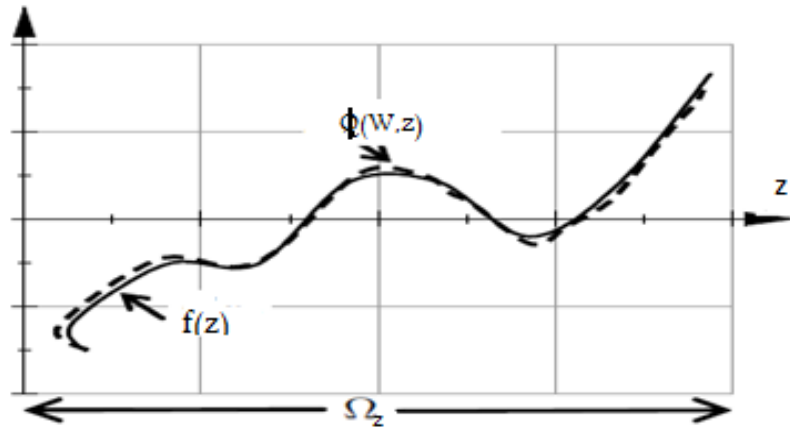
Cet algorithme garantit que  $|W| \leq W_{max}$ ,  $\forall t$  si  $|W(0)| \leq W_{max}$  [Ioannou-96].

### 1.3.3. Théorème de l'approximation universelle :

Parmi les propriétés des réseaux de neurones, l'approximation universelle est la plus importante du point de vue de l'identification et la commande. Cette propriété est la justification de l'utilisation des réseaux de neurones dans les systèmes de commande. L'approximation universelle a fait le sujet de beaucoup de travaux de recherche et plusieurs démonstrations pour des réseaux de neurones avec différentes fonctions d'activation, on dispose d'un nombre important de publications montrant que les systèmes neuronale sont des approximateurs universels [Funahashi-89] [Hornik-89], c'est-à-dire, pour toute fonction continue  $f(z)$  définie sur un ensemble compact  $\Omega_z$ , et pour toute constante positive  $\varepsilon$ , il existe un système neuronale d'ordre supérieur, avec  $q$  neurones dans la couche cachée de la forme (1.1) tel que :

$$\sup_{z \in \Omega_z} |f(z) - \phi(W, z)| < \varepsilon \quad (1.6)$$





**Fig. 1.4:** Fonction non linéaire  $f(z)$  sont réseaux approximateurs  $\phi(W, z)$  sur le domaine  $\Omega_z$ .

Cependant, la propriété de l'approximation universelle garantit l'existence d'un approximateur pour la fonction non linéaire continue  $f(z)$ , mais n'indique pas comment choisir la structure ou régler les paramètres du réseau de neurones pour atteindre une certaine précision désirée.

Dans ce travail on suppose que la structure du réseau de neurones et les paramètres des fonctions HONN sont correctement spécifiés par l'utilisateur a priori, alors la propriété d'approximation universelle peut être satisfaite en sélectionnant seulement les paramètres du réseau de neurones. Cela veut dire que la décision de l'utilisateur est nécessaire pour régler les paramètres ajustables et les paramètres de réseau HONN seront calculés en ligne par des algorithmes d'ajustement.

Notons qu'il existe d'autres types d'approximateurs universels tels que les réseaux de neurones multicouches et les systèmes flous qui peuvent faire usage des connaissances linguistiques d'une manière systématique [Wang-94].

#### 1.4. Conclusion

Dans ce chapitre, nous avons décrit une technique intelligente nouvellement introduite dans le monde de l'électronique de puissance. Il s'agit principalement des réseaux de neurones d'ordre supérieur HONN et les différentes structures qui leurs sont associées ainsi que nous abordons par la suite le contrôle de processus par les réseaux de neurones HONN pour la synthèse de lois de commandes.

## *Chapitre 2*

# *La Commande Adaptative Neuronale pour une Classe de Systèmes Non Linéaires SISO Non-affine*

---

- 2.1 Introduction
- 2.2 La transformation du système non-linéaire
- 2.3 Commande adaptative indirecte neuronale
- 2.4 Commande adaptative directe neuronale
- 2.5 Conclusion

## 2.1. Introduction :

La commande adaptative neuronale a reçu un grand succès pour le contrôle des systèmes non linéaires incertains et complexes en raison de l'excellente capacité d'approximations des réseaux de neurones.

De nombreux résultats remarquables ont été obtenus pour les systèmes non linéaires affines en la commande basés sur la linéarisation par retour d'état, et l'approximation des fonctions non linéaires à l'aide des réseaux de neurones et logique floue [Spooner-96] [Wang-94] [Yesildirek-95]. En pratique, de nombreux systèmes non linéaires ne peuvent pas être représentés sous la forme affine, et les problèmes deviennent beaucoup plus difficiles et compliqués où il ya moins de résultats disponibles dans la littérature.

Pour le cas de la commande adaptative neuronale indirecte le système non linéaire non affine SISO est d'abord transformé en une forme affine en considérant un développement en série de Taylor autour d'une trajectoire de fonctionnement, puis deux systèmes adaptatifs neuronaux pour approximer la dynamique du système affine [Ge-00]. Cependant, cette approche adaptative indirecte présente l'inconvénient du problème de la singularité du contrôleur, c'est à dire la division par zéro peut se produire dans la loi de commande. les auteurs utilisent un algorithme de projection [Ioannou-96][Spooner-96] pour maintenir les paramètres ajustables de l'approximateur du gain de commande à l'intérieur d'un ensemble de non singularité et de modifier les lois de commande par l'ajout de termes à structure variable pour garder l'amplitude du signal de commande borné [Yesildirek-95] [Labioud-05] .

L'approche adaptative neuronale directe pour une classe de systèmes non linéaires non affines est abordée par exemple dans les papiers [Ge-00] [Ge-98] [Ge-99] [Dai-14] [Zhang-98] [Ge-03], les réseaux neuronaux sont utilisés pour construire ce contrôleur implicite idéal et inconnu.

Les lois d'adaptation utilisées dans les travaux présentés précédemment sont basées sur l'approche de Lyapunov. Elles utilisent le signal d'erreur entre la sortie désirée et la sortie réelle du système afin d'ajuster en ligne les paramètres réglables. La robustesse des systèmes vis à vis des erreurs d'approximation et des perturbations externes est assurée par un terme de commande supplémentaire type  $H_\infty$  ou type mode glissant.

L'objectif de ce chapitre est de développer une commande adaptative neuronale pour une classe de systèmes non linéaires SISO non affines en la commande sous forme générale. À l'aide d'une technique basée sur la théorie de la géométrie différentielle, sous certaines hypothèses, le système possède un degré relatif complet. D'abord un difféomorphisme est défini pour transformer les systèmes non linéaires de formes générales en la forme normale. L'existence d'un contrôleur idéal inconnu est prouvée par le théorème de la fonction implicite. Le développement en série de Taylor autour du contrôleur idéal inconnu est utilisé pour transformer le système non affine en un système affine. La loi de commande est déduite par linéarisation par retour d'état du système affine approximé. Deux approches distinctes sont formulées pour la commande adaptative neuronale pour les systèmes monovariabiles SISO, l'approche directe et l'approche indirecte. Dans l'approche directe, les systèmes neuronaux sont utilisés pour approximer un contrôleur idéal inconnu. En revanche la commande indirecte exploite les systèmes neuronaux pour estimer la dynamique inconnue de système non linéaire à commander afin de synthétiser des lois de commande basées sur ces estimations. Dans cette approche, les lois de commande sont bien définies à tout moment, et le problème des erreurs de reconstruction est traité par l'introduction d'un terme de robustification de type mode glissant. Le problème de la singularité engendré par l'approche indirecte est résolu par l'utilisation d'un algorithme de projection.

Dans les deux approches adaptatives directe et indirecte, les algorithmes d'adaptations sont déduits à partir de la théorie de Lyapunov pour assurer la stabilité et la bornitude de tous les signaux de la boucle fermée du système sous certaines contraintes sur le gain de commande non linéaire et ses dérivés. Pour valider la structure de la commande développée et mettre en évidence son efficacité, des exemples de simulation, sont considérés.

## **2.2. La transformation du système non-linéaire :**

### **2.2.1. Position du problème :**

On considère une classe des systèmes non linéaires mono entrée mono sortie (SISO) non affines en la commande, dont la dynamique est décrite par les équations différentielles suivantes : [Ge-00]

$$\begin{cases} \dot{x} = f(x, u) \\ y = h(x) \end{cases} \quad (2.1)$$

où  $x \in \mathcal{R}^n$  est le vecteur d'état,  $u \in \mathcal{R}$  et  $y \in \mathcal{R}$  sont respectivement l'entrée et la sortie du système (2.1), et  $f(.,.)$  est un fonction continue non affine inconnue, et  $h(.)$  est une fonction scalaire. Sont supposés non linéaires de forme analytique inconnue.

Considérons la structure de la commande du système non linéaire non affine sous forme générale (2.1), cette structure ne permet pas d'avoir une relation directe entre l'entrée  $u$  et la sortie du système  $y$ , ce qui rend la mise en œuvre d'une commande en poursuite d'une trajectoire de référence difficile. Pour remédier à ce problème la techniques de la géométrie différentielle [Slotine-91] telles que la dérivée de Lie, crochet de Lie, le difféomorphisme, sont utilisés pour transformer le modèle du système vers la forme normale.

### 2.2.2. Représentation entrée-sortie :

L'objectif de la représentation entrée-sortie est de trouver une relation directe entre la sortie du système (2.1) et son entrée. Pour exprimer explicitement cette relation, il est nécessaire de dériver itérativement la sortie du système jusqu'à l'apparition de l'entrée  $u$ . On obtient alors [Isidori-89]:

$$\begin{aligned} y &= h(x) \\ \dot{y} &= L_f h(x) \\ \ddot{y} &= L_f^2 h(x) \\ &\vdots \\ y^{(\rho)} &= L_f^\rho h(x) \end{aligned} \quad (2.2)$$

avec :

$$L_f h(x) = \frac{\partial h(x)}{\partial t} = \frac{\partial h(x)}{\partial x} \frac{\partial x}{\partial t} = \frac{\partial [h(x)]}{\partial x} \dot{x} = \frac{\partial [h(x)]}{\partial x} f(x, u) \quad (2.3)$$

La relation entrée-sortie du système (2.1) est donc la suivante:

$$y^{(\rho)} = L_f^\rho h(x)$$

La notation  $L_f h(x)$  est une dérivée de Lie qui signifie la dérivée de  $h(x)$  dans la direction de  $f$  c'est-à-dire

$$L_f h(x) = \frac{\partial [h(x)]}{\partial x} f(x, u)$$

$\rho$ : le nombre de dérivées nécessaires pour qu'au moins une des entrées apparaisse, et est connu sous le nom du degré relatif correspondant à la sortie  $y$ .

Les dérivés de lie d'ordre supérieur peuvent être définis périodiquement comme suit :

$$L_f^k h = L_f \left( L_f^{(k-1)} \right), k > 0$$

Le degré relatif représente le retard subi par l'entrée  $u$  avant son influence sur la sortie  $y$ . Si le degré relatif  $\rho$  est égal à l'ordre  $n$  du système, le système (2.1) est dit dans sa forme normale et ne présente pas de dynamiques de zéros et est donc à minimum de phase. Si  $\rho$  est strictement inférieur à  $n$ , le système (2.1) est à minimum de phase [Slotine-91] [Isidori-89] si le sous-système d'ordre  $(n - \rho)$  est asymptotiquement stable (stabilité de la dynamique de zéros).

**Définition 2.1 :**

On dit que le système (2.1) possède un degré relatif  $\rho$  à  $(x_0, u_0)$ , s'il existe un nombre entier positif  $1 \leq \rho < \infty$  tels que

$$\forall x \in R: \frac{\partial [L_f^i h]}{\partial u} = 0, i = 0, 1, \dots, \rho - 1$$

$$\frac{\partial [L_f^\rho h]}{\partial u} \neq 0$$

avec  $\Omega_x \subset R^n$  et  $\Omega_u \subset R$  est un ensemble compacts contenant  $x_0$  et  $u_0$  respectivement.

Le Système (2.1) est possède un degré relatif complet  $\rho$  dans un ensemble compact  $U = \Omega_x \times \Omega_u$ . S'il a un degré relatif  $\rho$  de tout point  $(x_0, u_0) \in U$ .

**Hypothèse 2.1 :** Le système (2.1) possède un degré relatif complet  $\rho = n, \forall (x, u) \in U$ .

On peut donc trouver la forme normale du système en appliquant le changement de coordonnées (un difféomorphisme)  $\Phi(x) = [\phi_1, \phi_2, \dots, \phi_n]^T$  avec  $\phi_j(x) = L_f^j h(x), j = 1, 2, \dots, n$  est une matrice Jacobienne qui est non singulière pour tous  $x \in \Omega$  [Isidori-89] donc  $\Phi(x)$  est un difféomorphisme sur  $\Omega_x$  et  $\xi = \Phi(x)$  tel que :

$$\xi_i = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \dots \\ \xi_n \end{bmatrix} = \begin{bmatrix} \phi_1(x) \\ \phi_2(x) \\ \phi_3(x) \\ \dots \\ \phi_n(x) \end{bmatrix} = \begin{bmatrix} h(x) \\ L_f h(x) \\ L_f^2 h(x) \\ \dots \\ L_f^{\rho-1} h(x) \end{bmatrix} \quad (2.4)$$

Alors, (2.1) peut être transformé en une forme normale :

$$\begin{cases} \dot{\xi}_1 = \xi_2 \\ \dot{\xi}_2 = \xi_3 \\ \dots \dots \dots \\ \dot{\xi}_n = b(\xi, u) \\ y = \xi_1 \end{cases} \quad (2.5)$$

On peut alors facilement vérifier que:

$$\begin{cases} \dot{\xi}_i = \xi_{i+1}, i = 1 \dots n - 1 \\ \dot{\xi}_n = b(\xi, u) \\ y = \xi_1 \end{cases} \quad (2.6)$$

où :

$$b(\xi, u) = L_f^n h(\xi)$$

avec :

$$x = \Phi^{-1}(\xi)$$

Définir le domaine du système normal (2.6), comme suit :

$$\bar{U} = \{(\xi, u) | \xi \in \Phi(\Omega_x), u \in \Omega_u\} \quad (2.7)$$

Soit la fonction

$$b_u = \partial[b(\xi, u)]/\partial u$$

et la fonction continue :

$$\dot{b}_u = d(b_u)/dt$$

Selon l'hypothèse 2.1 et la définition 2.1, nous avons  $\partial[b(\xi, u)]/\partial u \neq 0, \forall (\xi, u) \in \bar{U}$ .

Cela implique que la fonction  $b_u$  est strictement positive ou négative pour tous  $(\xi, u) \in \bar{U}$ .

Supposons que le signe de  $b_u$  est connu, et sans perte de généralité, il est supposé que  $b_u > 0$  dans la discussion qui suit.

**Hypothèse 2.2 [Karimi-09]** : Il existe des constantes positives  $g_0$  et  $g_1$  de telle sorte que

$$g_0 \leq b_u \leq g_1 \quad (2.8)$$

avec

$$\left| \frac{d}{dt} \left[ \frac{b(\xi, u)}{\partial u} \right] \right| \leq H < \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)} g_0 \quad (2.9)$$

Pour tous  $(\xi, u) \in \bar{U}$ , avec  $\lambda_{\min}(Q)$  et  $\lambda_{\max}(P)$  sont les valeurs propres minimale et maximale des matrices Q et P respectivement données par l'équation (2.15)

**Hypothèse 2.3:** La trajectoire désirée  $\bar{\xi}_d(t)$  et ses dérivées jusqu'à l'ordre  $n$  sont connues et bornées.

Si l'on définit l'erreur de poursuite par

$$\tilde{\xi}_1(t) = y_d(t) - \xi(t) \quad (2.12)$$

et le vecteur d'erreur de poursuite de tous les états est défini par

$$\tilde{\xi} = \xi_d - \xi = [\tilde{\xi}, \dot{\tilde{\xi}}, \ddot{\tilde{\xi}}, \dots, \tilde{\xi}^{(n-1)}]^T$$

avec  $\xi_d(t) = [y_d, \dot{y}_d, \ddot{y}_d, \dots, y_d^{(n-1)}]^T \in R^n$  et  $\bar{\xi}_d = [\bar{\xi}_d^T, \bar{y}_d^n]^T \in R^{(n+1)}$

La dynamique des erreurs de poursuite sera alors donnée par:

$$\dot{\tilde{\xi}} = A\tilde{\xi} + b[-b(\xi, u) + y_d^{(n)}] \quad (2.13)$$

$$\text{avec } A = \begin{bmatrix} 0_{(n-1)} & I_{(n-1)} \\ 0 & 0_{(n-1)}^T \end{bmatrix} \in \mathfrak{R}^{(n) \times (n)}, b = \begin{bmatrix} 0_{(n-1)} \\ 1 \end{bmatrix} \in \mathfrak{R}^{(n) \times 1}$$

$$A = \begin{bmatrix} 0_{(n-1)} & I_{(n-1)} \\ 0 & 0_{(n-1)}^T \end{bmatrix}, b = \begin{bmatrix} 0_{(n-1)} \\ 1 \end{bmatrix}$$

où  $I_{(n-1)}$  et  $0_{(n-1)}$  indiquer la matrice d'identité et un vecteur de zéros de dimension  $(n - 1)$ , respectivement.

**Théorème de la fonction implicite [Park-04]:**

Soit  $h$  une fonction continue et dérivable en chaque point  $(a, b)$  de l'ensemble ouvert  $S \subset \mathfrak{R} \times \mathfrak{R}$ . Soit  $(a_0, b_0)$  un point dans  $S$ , pour le quel  $h(a_0, b_0) = 0$  et telle que la matrice Jacobienne  $[\delta h / \delta a](a_0, b_0)$  est non singulière. Alors, il existe sur un voisinage  $U \times V \subset \mathfrak{R} \times \mathfrak{R}$  du point  $(a_0, b_0)$ , solution unique de l'équation  $h(a, b) = 0$  pour chaque point  $b \in V$  et  $a \in U$ . De plus, la solution peut être donnée comme suit  $a = g(b)$ , où  $g$  est continue et dérivable en  $b = b_0$ .

**Lemme 2.1 [Ge-98] :** Considérons le système (2.6) satisfaisant les suppositions [2.1–2.3], alors il existe un contrôleur idéal  $u^*$  tel que

$$\dot{\tilde{\xi}} = (A - b\underline{k}^T)\tilde{\xi} \quad (2.14)$$

avec

$$A_k = (A - b\underline{k}^T)$$

Le vecteur de gain de retour d'état  $\underline{k}$  est calculé afin que la matrice  $(A - b\underline{k}^T)$  ait toutes ses valeurs propres à parties réelles négatives. Par conséquent le système bouclé sera exponentiellement stable. On lui associe une équation de Lyapunov, ayant pour solution unique une matrice symétrique définie positive, donnée par

$$A_k^T P + P A_k = -Q \quad (2.15)$$

**Démonstration :**

En ajoutant et en retranchant le terme  $b\underline{k}^T e$  sur le côté droit de (2.13), on obtient :

$$\dot{\tilde{\xi}} = (A - b\underline{k}^T)\tilde{\xi} - b[b(\xi, u) - \varphi] \quad (2.16)$$

où

$$\varphi = y_d^{(n)} + \underline{k}^T \tilde{\xi}$$

A partir de l'hypothèse 2.1 et le fait que  $\partial[b(\xi, u)]/\partial u \neq 0$  et  $\partial[\varphi]/\partial u = 0$ , ainsi



$$\partial[b(\xi, u) - \varphi]/\partial u \neq 0, \forall (\xi, u) \in \bar{U} \quad (2.17)$$

D'après le théorème de la fonction implicite **[Park-04]**, il existe un contrôleur idéal  $u^*(z)$ ,

avec  $z = [\xi^T, \xi_d^T, \varphi]$ , tel que

$$b(\xi, u^*) - \varphi = 0 \quad (2.18)$$

On considère la fonction candidate de Lyapunov suivante :

$$V = \frac{1}{2} \tilde{\xi}^T P \tilde{\xi} \quad (2.19)$$

En dérivant l'expression (2.19) par rapport au temps, on obtient

$$\begin{aligned} \dot{V} &= \frac{1}{2} (\dot{\tilde{\xi}}^T P \tilde{\xi} + \tilde{\xi}^T P \dot{\tilde{\xi}}) \\ &= \frac{1}{2} \tilde{\xi}^T [A_k^T P + P A_k] \tilde{\xi} - \tilde{\xi}^T P B [b(\xi, u) - \varphi] \end{aligned} \quad (2.20)$$

Si  $u = u^*$  (2.16), alors on obtient

$$\dot{V} = \frac{1}{2} \tilde{\xi}^T [A_k^T P + P A_k] \tilde{\xi} \quad (2.21)$$

$$\dot{V} \leq -\frac{1}{2} \tilde{\xi}^T Q \tilde{\xi} \leq 0 \quad (2.22)$$

Ce qui permet de déduire que  $\tilde{\xi}$  et ses dérivées  $\dot{\tilde{\xi}} \in L_\infty$

L'intégration de l'équation (2.22), donne

$$\int_0^\infty \|\tilde{\xi}\|^2 d\tau \leq \frac{1}{\lambda_{\min}(Q)} [|V(0)| - |V(t)|] \quad (2.23)$$

ce qui implique que  $\tilde{\xi}(t) \in L_2$  donc  $\dot{\tilde{\xi}}(t) \in L_\infty$ , et d'après le lemme de Barbalat **[Slotine-91]**, on conclut que  $\lim_{t \rightarrow \infty} \tilde{\xi}(t) = 0$ .

Cependant, le théorème de la fonction implicite garantit seulement l'existence du contrôleur idéal  $u^*(z)$  et ne fournit pas une méthode pour le concevoir. Dans la suite de travaille, on propose des systèmes adaptatifs neuronaux pour approximer le contrôleur idéal inconnu.

En se servant du lemme2.1 qui montre l'existence de l'entrée idéale  $u^*$ , un développement en série de Taylor autour de  $u^*(z)$  est effectué.

La fonction du système  $b(\xi, u)$  peut être écrite comme suit :

$$b(\xi, u) = b(\xi, u^*(z)) + \frac{\partial b(\xi, u^*(z))}{\partial u} (u - u^*) + O(u - u^*)^2 \quad (2.24)$$

L'équation (2.24) peut être réécrite sous la forme suivante :

$$b(\xi, u) = f(z) + g(z)u + O(u - u^*)^2 \quad (2.25)$$

avec

$$f(z) = b(\xi, u^*(z)) - \frac{\partial b(\xi, u^*(z))}{\partial u} u^* , \quad g(z) = \frac{\partial b(\xi, u^*(z))}{\partial u}$$

et  $O(u - u^*)^2$  désigne la somme des termes d'ordre supérieure du développement en série de Taylor noté par la suite par  $d$ . Ce dernier terme considéré petit et bornée comme suit :

$$|d| \leq \bar{d} \quad (2.26)$$

À partir de l'hypothèse 2.2 et l'équation (2.8), nous avons

$$g_0 \leq g(z) \leq g_1 \quad (2.27)$$

dans le voisinage de la commande idéal  $u^*(z)$ .

Ainsi, on obtient un système non linéaire affine autour d'une commande idéal comme suit:

$$\begin{cases} \dot{\xi}_i = \xi_{i+1}, i = 1 \dots n - 1 \\ \dot{\xi}_n = f(z) + g(z)u + d \\ y = \xi_1 \end{cases} \quad (2.28)$$

**Remarque 2.1:** À partir des dérivations successives, on voit que la commande du système (2.1) est équivalente à la commande du système (2.28) qui est sous la forme d'un système non linéaire affine.

### 2.2.3. La commande linéarisante :

Après avoir définie une structure permettant d'obtenir une relation directe entre l'entrée et la sortie du système, et une transformation qui permet d'avoir une forme affine d'un système non linéaire qui peut être appliquée simplement à une classe de systèmes non linéaires décrits par ce qu'on appelle la forme canonique si sa dynamique est représentée par:

$$\begin{cases} \dot{\xi}_i = \xi_{i+1}, i = 1 \dots n - 1 \\ \dot{\xi}_n = f(z) + g(z)u + d \\ y = \xi_1 \end{cases} \quad (2.29)$$

où  $\xi = [\xi_1, \xi_2, \xi_3, \dots, \xi_n]^T \in R^n$  est le vecteur d'état,  $u \in \mathcal{R}$  et  $y \in \mathcal{R}$  sont respectivement l'entrée et la sortie du système,  $f(z)$  et  $g(z)$  sont des fonctions non linéaires continues incertaines.

L'objectif de la commande est de concevoir un contrôleur adaptatif neuronale capable de forcer la sortie du système  $\xi(t)$  à suivre une trajectoire de référence  $y_d(t)$  borné bien déterminé, sous la contrainte que tous les signaux impliqués soient bornés.

Pour cela on détermine une loi d'adaptation pour ajuster le vecteur de paramètres tel que les conditions suivantes soient satisfaites :

(i) le système en boucle fermée doit d'être stable et robuste dans le sens où toutes les variables estimées soient uniformément bornées.

(ii) l'erreur de poursuite, doit être la plus petite possible.

Le système (2.28) peut être réécrit sous la représentation d'état suivante :

$$\begin{aligned}\dot{\xi} &= A\xi + b[f(z) + g(z)u + d] \\ y &= C^T \xi\end{aligned}\quad (2.30)$$

où

$$A = \begin{bmatrix} 0_{(n-1)} & I_{(n-1)} \\ 0 & 0_{(n-1)}^T \end{bmatrix} \in \mathfrak{R}^{(n) \times (n)} \quad , \quad b = \begin{bmatrix} 0_{(n-1)} \\ 1 \end{bmatrix} \in \mathfrak{R}^{(n) \times 1}$$

$$C = \begin{bmatrix} 1 \\ 0_{(n-1)} \end{bmatrix} \in \mathfrak{R}^{(n) \times 1}$$
(2.31)

$I_{(n-1)}$  et  $0_{(n-1)}$  indiquent la matrice d'identité et un vecteur de zéros de dimension  $(n - 1)$ , respectivement.

En remplaçant la fonction  $b(\xi, u)$  par leur transformation équivalente dans l'équation dynamique de l'erreur (2.13), on obtient :

$$\dot{\xi} = A\xi + b[-f(z) - g(z)u + y_d^{(n)} - d] \quad (2.32)$$

Si les fonctions non linéaires  $f(z)$  et  $g(z)$  sont connues, les objectifs de commande peuvent être atteints en choisissant la loi de commande idéale suivante :

$$u^* = \frac{1}{g(z)} [-f(z) + y_d^{(n)}(t) + \underline{k}^T \tilde{\xi}(t)] \quad (2.33)$$

où  $\underline{k}^T = [k_0, k_1, \dots, \dots, k_{(n-1)}]$  est le vecteur de gain de retour d'état, par substitution l'équation (2.33) par l'équation (2.28) l'équation suivante est obtenue en considérant  $d=0$  :

$$\xi^n + k_1 \xi^{(n-1)} + \dots + k_{(n)} \xi = 0 \quad (2.34)$$

Les coefficients  $k_j, j = 0, \dots, n - 1$ , sont choisis de façon à ce que le polynôme:  $S^{(n)} + k_{(n-1)}S^{(n-1)} + \dots + k_0 = 0$ . Soit Hurwitzien (racines à partie réelle négative), qui prouve la convergence de l'erreur de suivi de trajectoire  $\tilde{\xi}(t)$  vers zéro quand  $t \rightarrow \infty$ , c'est-à-dire :

$$\lim_{t \rightarrow \infty} \|\tilde{\xi}\|^2 = 0 \quad (2.35)$$

Cependant, les fonctions  $f(z)$  et  $g(z)$  sont inconnues et la perturbation extérieure  $d$  existe, donc il est difficile d'appliquer la loi de commande (2.33) pour le système non linéaire (2.29). La solution la plus courante pour résoudre ce problème est d'utiliser des approximateurs universels pour approximer cette commande dans un schéma de

commande directe et indirect. Dans la suite de ce chapitre, nous développons une commande adaptative directe et indirecte du système (2.29) avec une approximation par un réseau de neurone d'ordre supérieur HONN.

### 2.3. Commande adaptative indirecte neuronale :

La commande adaptative indirecte consiste à approximer les fonctions  $f(z)$  et  $g(z)$  par deux systèmes neuronales, pour en déduire la loi de commande par rétroaction. En effet, on considère que le modèle du processus à commander est inconnu. Afin d'utiliser la même structure de commande, deux systèmes neuronales de la forme (1.1) ont été introduits. Les approximateurs ainsi construits sont injectés directement dans la loi de commande pour atteindre les performances de poursuite désirées. Les lois d'adaptation avec algorithme de projection sont déduites de l'étude de la stabilité.

#### 2.3.1. Conception de la commande neuronale :

Les systèmes neuronales sont des approximateurs universels du point de vue des experts humains et ils peuvent être uniformément approchés des fonctions continues non-linéaires avec l'exactitude arbitraire, afin d'atteindre les objectifs de contrôle proposés. Les fonctions non linéaires  $f(z)$  et  $g(z)$  seront rapprochées en réglant les paramètres des systèmes neuronaux correspondant. Alors, suivant les résultats de l'approximation exposés au chapitre précédent il existe un réseau de neurone d'ordre supérieur capable d'approcher les fonctions non linéaires incertaines  $f(z)$  et  $g(z)$  telle que :

$$\hat{f}(z/W_f) = W_f^T S_f(z) \quad (2.36)$$

$$\hat{g}(z/W_g) = W_g^T S_g(z) \quad (2.37)$$

avec  $S_f(z)$  et  $S_g(z)$  sont des vecteurs de fonctions neuronales sont supposés convenablement fixés en avance par l'utilisateur,  $W_f$  et  $W_g$  sont les vecteurs de paramètres ajustés.

Les fonctions réelles  $f(z)$  et  $g(z)$  peuvent être exprimées en fonction de l'approximation neuronale de la manière suivante :

$$f(z) = W_f^{*T} S_f(z) + \varepsilon_f(z) \quad , \quad z \in \Omega_z \subset R^{(2n+1)} \quad (2.38)$$

$$g(z) = W_g^{*T} S_g(z) + \varepsilon_g(z) \quad , \quad z \in \Omega_z \subset R^{(2n+1)} \quad (2.39)$$

avec  $\varepsilon_f$  et  $\varepsilon_g$  représentent les erreurs d'approximation neuronale,  $W_f^*$  et  $W_g^*$  sont respectivement les paramètres optimaux de  $W_f$  et  $W_g$  , c'est-à-dire, les valeurs des

paramètres  $W_f$  et  $W_g$  minimisant respectivement les erreurs d'approximation  $|\varepsilon_f|$  et  $|\varepsilon_g|$ . Ces paramètres optimaux satisfont :

$$W_f^* = \operatorname{argmin}_{W_f \in \Omega_f} \{ \sup_{z \in \Omega_z} |\hat{f}(z/W_f) - f(z)| \} \quad (2.40)$$

$$W_g^* = \operatorname{argmin}_{W_g \in \Omega_g} \{ \sup_{z \in \Omega_z} |\hat{g}(z/W_g) - g(z)| \} \quad (2.41)$$

Notons que les paramètres optimaux  $W_f^*$  et  $W_g^*$  sont des constantes artificielles inconnues introduites uniquement pour faire l'étude théorique de la stabilité de l'algorithme de commande. En fait, la connaissance de leurs valeurs n'est pas nécessaire pour l'implantation des lois de commande adaptatives.

$$f(z) - \hat{f}(z/W_f) = \tilde{W}_f^T S_f(z) + \varepsilon_f(z) \quad (2.42)$$

$$g(z) - \hat{g}(z/W_g) = \tilde{W}_g^T S_g(z) + \varepsilon_g(z) \quad (2.43)$$

avec

$$\tilde{W}_i = W_i^* - W_i \quad \text{avec} \quad \dot{\tilde{W}}_i = -\dot{W}_i \quad \text{et} \quad i = f, g$$

**Hypothèse 2.4:** Les erreurs d'approximation sont bornées comme suit

$$|\varepsilon_f(z)| \leq \bar{\varepsilon}_f \quad \text{et} \quad |\varepsilon_g(z)| \leq \bar{\varepsilon}_g$$

où  $\bar{\varepsilon}_f$  et  $\bar{\varepsilon}_g$  sont des constantes positives inconnues.

Cette hypothèse est raisonnable puisque nous supposons que les systèmes neuronaux utilisés pour l'approximation des fonctions inconnues possèdent la propriété d'approximateur universel.

Considérons maintenant la loi de commande suivante :

$$u = \frac{1}{\hat{g}(z/W_g)} \left[ -\hat{f}(z/W_f) + y_d^{(n)}(t) + \underline{k}^T \tilde{\xi}(t) \right] \quad (2.44)$$

cette loi de commande résulte de (2.33) en remplaçant les fonctions  $f(z)$  et  $g(z)$  par leurs approximations neuronaux  $\hat{f}(z/W_f)$  et  $\hat{g}(z/W_g)$ , et elle ne peut garantir à elle seule la stabilité du système bouclé. Ceci est dû, d'une part, à l'existence des erreurs d'approximation et, d'autre part, au fait que cette loi de commande n'est pas définie lorsque  $\hat{g}(z/W_g) = 0$ . Pour traiter le problème des erreurs d'approximation, on ajoute un terme de robustification à la loi de commande équivalente, et pour éviter la division par zéro lors du processus d'adaptation des paramètres  $W_g$ , nous proposons l'utilisation des lois d'adaptation avec projection.

Maintenant considérons la loi de commande suivante :

$$u(t) = u_c(t) + u_r(t) \quad (2.45)$$

La loi de commande (2.45) est la somme de deux termes : un terme de commande adaptative,  $u_c(t)$  utilisé pour compenser les non linéarités du système, et un terme de robustification  $u_r(t)$ , introduit pour pallier le problème des erreurs d'approximation.

Le terme adaptatif est défini comme suit

$$u_c = \frac{1}{\hat{g}(z/W_g)} \left[ -\hat{f}(z/W_f) + y_d^{(n)}(t) + \underline{k}^T \tilde{\xi}(t) \right] \quad (2.28)$$

et le terme de robustification  $u_r$  par

$$u_r = \frac{1}{g_0} (\hat{\varepsilon} + \hat{\varepsilon}_g |u_c(t)|) \text{sgn}(\tilde{\xi}^T P b) \quad (2.47)$$

où  $\hat{\varepsilon}$  et  $\hat{\varepsilon}_g$  sont des estimations des paramètres inconnus,  $\varepsilon^* = (\bar{\varepsilon}_f + \bar{d})$  et  $\varepsilon_g^* = \bar{\varepsilon}_g$ ,  $\text{sgn}(\tilde{\xi}^T P b)$  est la fonction signe donnée par :

$$\text{sgn}(\tilde{\xi}^T P b) = \begin{cases} +1, & \tilde{\xi}^T P b > 0 \\ 0, & \tilde{\xi}^T P b = 0 \\ -1, & \tilde{\xi}^T P b < 0 \end{cases} \quad (2.48)$$

La bornitude des paramètres ajustables comme nous le verra par la suite, une condition nécessaire pour l'établissement de la stabilité de la boucle de commande. Comme déjà introduit au chapitre précédant, on peut utiliser les algorithmes du gradient avec projection pour assurer la bornitude des paramètres ajustés et aussi pour éviter le problème de n'est pas définie lorsque  $\hat{g}(z/W_g) = 0$ .

Pour cela l'hypothèse suivante est employée :

**Hypothèse 2.5:** Les paramètres du réseau de neurones sont bornés par les ensembles de contraintes  $\Omega_f$  et  $\Omega_g$  telles que :

$$\Omega_f = \{W_f : \|W_f\| \leq M_f\}$$

$$\Omega_g = \{W_g : \|W_g\| \leq M_g\}$$

Les bornes employées dans l'hypothèse 2.6 sont introduites pour assurer la bornitude des sorties des réseaux de neurones.

A cet effet, les paramètres du contrôleur neuronal sont ajustés en utilisant l'algorithme suivant [**Wang-93**] :

$$\dot{W}_f = \Gamma_f \tilde{\xi}^T P b \left[ -S_f(z) + I_f \rho_f \frac{W_f W_f^T S_f(z)}{\|W_f\|^2} \right] \quad (2.49)$$

$$\dot{W}_g = \Gamma_g \tilde{\xi}^T P b \left[ -S_g(z) u_c(t) + I_g \rho_g \frac{W_g W_g^T S_g(z) u_c(t)}{\|W_g\|^2} \right] \quad (2.50)$$

où

$$I_f = \begin{cases} 0, & \text{si } \|W_f\| < M_f \text{ ou } (\|W_f\| = M_f \text{ et } \tilde{\xi}^T P b W_f^T S_f(z) \geq 0) \\ 1, & \text{si } \|W_f\| = M_f \text{ et } \tilde{\xi}^T P b W_f^T S_f(z) < 0 \end{cases} \quad (2.51)$$

$$I_g = \begin{cases} 0, & \text{si } \|W_g\| < M_g \text{ ou } (\|W_g\| = M_g \text{ et } \tilde{\xi}^T P b W_g^T S_g(z) u_c(t) \geq 0) \\ 1, & \text{si } \|W_g\| = M_g \text{ et } \tilde{\xi}^T P b W_g^T S_g(z) u_c(t) < 0 \end{cases} \quad (2.52)$$

avec  $\rho_f \geq 1$  et  $\rho_g \geq 1$

et  $\Gamma_g, \Gamma_f > 0$  sont les pas d'ajustement sélectionnés par le concepteur, et  $P > 0$  est une matrice définie positive qu'on déterminera par la suite.

$$\dot{\hat{\epsilon}} = \gamma_0 |\tilde{\xi}^T P b| \quad (2.53)$$

$$\dot{\hat{\epsilon}}_g = \gamma_0 |\tilde{\xi}^T P b u_c| \quad (2.54)$$

avec  $\gamma_0 > 0$ .

### 2.3.2. Analyse de stabilité :

Dans cette section nous allons analyser la stabilité de la commande adaptative neuronale indirecte proposée. Pour démontrer la stabilité de la commande adaptative, nous allons procéder en deux étapes : En premier lieu nous démontrerons que les lois d'adaptation (2.49)-(2.50) gardent les paramètres ajustables sont bornés, et en deuxième lieu, nous démontrerons la stabilité de la dynamique de l'erreur en boucle fermée.

Considérons le système SISO non linéaire affiné dans la commande (2.30), la loi de commande (2.47) avec les lois d'adaptation (2.49)-(2.50), si les hypothèses [2.1-2.5] sont satisfaites, alors le schéma global garantit que :

- (i) Les lois d'adaptation (2.49)-(2.50) assurent que :  $\|W_f\| < M_f$  ,  $\|W_g\| < M_g \forall t \geq 0$ , si les valeurs initiales des paramètres  $W_f(0)$  et  $W_g(0)$  sont sélectionnées correctement.
- (ii) les signaux du système en boucle fermée sont bornés et l'erreur de poursuite converge asymptotiquement vers zéro.

#### Démonstration :

Pour démontrer que :  $\|W_f\| < M_f$ , considérons la fonction de Lyapunov :

$$V_f = \frac{1}{2} W_f^T \Gamma_f^{-1} W_f \quad (2.55)$$

d'où

$$\dot{V}_f = W_f^T \Gamma_f^{-1} \dot{W}_f \quad (2.56)$$

Si  $I_f = 0$ , nous avons  $\|W_f\| < M_f$  ou  $\|W_f\| = M_f$  et

$$\dot{V}_f = - \underbrace{\Gamma_f \tilde{\xi}^T P b W_f^T S_f(z)}_{\geq 0} \leq 0 \quad (2.27)$$

c'est-à-dire, que nous avons toujours  $\|W_f\| \leq M_f$

Si  $I_f = 1$ , nous avons  $\|W_f\| = M_f$  et

$$\dot{V}_f = \Gamma_f \tilde{\xi}^T P b \left[ W_f^T \left( -S_f(z) + \rho_f \frac{W_f W_f^T S_f(z)}{\|W_f\|^2} \right) \right] \quad (2.58)$$

$$\dot{V}_f = \Gamma_f \tilde{\xi}^T P b \left[ -W_f^T S_f(z) + \rho_f \frac{\|W_f\|^2 W_f^T S_f(z)}{\|W_f\|^2} \right] \quad (2.59)$$

$$\dot{V}_f = \underbrace{\Gamma_f \tilde{\xi}^T P b W_f^T S_f(z)}_{< 0} (\rho_f - 1) < 0 \quad (2.60)$$

Puisque  $\tilde{\xi}^T P b W_f^T S_f(z) < 0$  et  $\|W_f\| = M_f$  et  $\rho_f \geq 1$  nous obtenons  $\dot{V}_f \leq 0$ , ce que implique que  $\|W_f\| \leq M_f$ , alors nous obtenons  $\|W_f\| \leq M_f \forall t \geq 0$

Une analyse similaire peut être utilisée pour montrer que  $\|W_g\| \leq M_g$ .

Pour la démonstration de la stabilité de la dynamique de l'erreur en boucle fermée, nous commençons par remplacer la loi de commande effective (2.43) dans le système d'erreur (2.32) ce qui donne :

$$\begin{aligned} \dot{\tilde{\xi}} &= A \tilde{\xi} + b \left[ -f(z) - g(z)u_c - g(z)u_r + y_d^{(n)} - d \right] \\ &= A \tilde{\xi} + b \left[ -f(z) - \left( g(z) - \hat{g} \left( \frac{z}{W_g} \right) \right) u_c - \hat{g} \left( \frac{z}{W_g} \right) u_c - g(z)u_r + y_d^{(n)} - d \right] \end{aligned} \quad (2.61)$$

Remplaçant (2.46) dans (2.61), on aura

$$\dot{\tilde{\xi}} = A \tilde{\xi} + b \left[ - \left( f(z) - \hat{f} \left( \frac{z}{W_f} \right) \right) - \left( g(z) - \hat{g} \left( \frac{z}{W_g} \right) \right) u_c - \underline{k}^T \tilde{\xi} - g(z)u_r - d \right] \quad (2.62)$$

et avec (2.42) et (2.43) on a :

$$\begin{aligned} \dot{\tilde{\xi}} &= (A - b \underline{k}^T) \tilde{\xi} + b \left[ -\tilde{W}_f^T S_f(z) - \tilde{W}_g^T S_g(z) u_c - \varepsilon_f(z) - \varepsilon_g(z)u_c - g(z)u_r - d \right] \\ &= A_k \tilde{\xi} + b \left[ -\tilde{W}_f^T S_f(z) - \tilde{W}_g^T S_g(z) u_c - \varepsilon_f(z) - \varepsilon_g(z)u_c - g(z)u_r - d \right] \end{aligned} \quad (2.63)$$

où  $A_k = (A - b \underline{k}^T)$

La matrice  $A_k$  étant stable (valeurs propres à partie réelle négative), pour une matrice  $Q$  symétrique définie positive. On lui associe une équation de Lyapunov, ayant pour solution unique une matrice symétrique définie positive, donnée par



$$A_k^T P + P A_k = -Q \quad (2.64)$$

On considère la fonction candidate suivante

$$V = \frac{1}{2} \left( \tilde{\xi}^T P \tilde{\xi} + \tilde{W}_f^T \Gamma_f^{-1} \tilde{W}_f + \tilde{W}_g^T \Gamma_g^{-1} \tilde{W}_g + \frac{1}{\gamma_{f0}} \tilde{\varepsilon}^2 + \frac{1}{\gamma_0} \tilde{\varepsilon}_g^2 \right) \quad (2.65)$$

et  $\Gamma_f = \Gamma_f^T > 0, \Gamma_g = \Gamma_g^T > 0, \tilde{W}_i = W_i^* - W_i$  avec  $i = f, g$  et  $\gamma_0 > 0, \gamma_{f0} > 0$

où  $\tilde{\varepsilon} = \varepsilon^* - \hat{\varepsilon}, \tilde{\varepsilon}_g = \varepsilon_g^* - \hat{\varepsilon}_g$ , et  $\dot{\tilde{\varepsilon}} = -\dot{\hat{\varepsilon}}, \dot{\tilde{\varepsilon}}_g = -\dot{\hat{\varepsilon}}_g, \dot{\tilde{W}}_i = -\dot{W}_i, i = f, g$ .

En dérivant  $V$  par rapport au temps, on obtient :

$$\begin{aligned} \dot{V} &= \frac{1}{2} \left( \dot{\tilde{\xi}}^T P \tilde{\xi} + \tilde{\xi}^T P \dot{\tilde{\xi}} \right) - \tilde{W}_f^T \Gamma_f^{-1} \dot{W}_f - \tilde{W}_g^T \Gamma_g^{-1} \dot{W}_g - \frac{1}{\gamma_0} \tilde{\varepsilon} \dot{\tilde{\varepsilon}} - \frac{1}{\gamma_0} \tilde{\varepsilon}_g \dot{\tilde{\varepsilon}}_g \\ &= \frac{1}{2} \tilde{\xi}^T [A_k^T P + P A_k] \tilde{\xi} + \tilde{\xi}^T P b [-\tilde{W}_f^T S_f(z) - \tilde{W}_g^T S_g(z) u_c - \varepsilon_f(x) - \varepsilon_g(x) u_c \\ &\quad - g(x) u_r - d] - \tilde{W}_f^T \Gamma_f^{-1} \dot{W}_f - \tilde{W}_g^T \Gamma_g^{-1} \dot{W}_g - \frac{1}{\gamma_0} \tilde{\varepsilon} \dot{\tilde{\varepsilon}} - \frac{1}{\gamma_0} \tilde{\varepsilon}_g \dot{\tilde{\varepsilon}}_g \end{aligned} \quad (2.66)$$

En utilisant (2.64), (2.66) peut être arrangée comme suit:

$$\begin{aligned} \dot{V} &= -\frac{1}{2} \tilde{\xi}^T Q \tilde{\xi} + \tilde{\xi}^T P b [-\varepsilon_f(z) - \varepsilon_g(z) u_c - g(z) u_r - d] \\ &\quad - \tilde{W}_f^T (\tilde{\xi}^T P b S_f(z) + \Gamma_f^{-1} \dot{W}_f) - \tilde{W}_g^T (\tilde{\xi}^T P b S_g(z) u_c + \Gamma_g^{-1} \dot{W}_g) \\ &\quad - \frac{1}{\gamma_0} \tilde{\varepsilon} \dot{\tilde{\varepsilon}} - \frac{1}{\gamma_0} \tilde{\varepsilon}_g \dot{\tilde{\varepsilon}}_g \end{aligned} \quad (2.67)$$

Peut être défini  $\dot{V}$  comme suit :

$$\dot{V} = -\frac{1}{2} e^T Q e + \dot{V}_1 + \dot{V}_2 \quad (2.68)$$

avec

$$\dot{V}_1 = -\tilde{W}_f^T (\tilde{\xi}^T P b S_f(z) + \Gamma_f^{-1} \dot{W}_f) - \tilde{W}_g^T (\tilde{\xi}^T P b S_g(z) u_c + \Gamma_g^{-1} \dot{W}_g) \quad (2.69)$$

$$\dot{V}_2 = \tilde{\xi}^T P b [-\varepsilon_f(z) - \varepsilon_g(z) u_c - g(z) u_r - d] - \frac{1}{\gamma_0} \tilde{\varepsilon} \dot{\tilde{\varepsilon}} - \frac{1}{\gamma_0} \tilde{\varepsilon}_g \dot{\tilde{\varepsilon}}_g \quad (2.70)$$

A partir des lois d'adaptation (2.49)-(2.52),  $\dot{V}_1$  se réduit à

$$\dot{V}_1 = -\tilde{W}_f^T \left( \underbrace{I_f \rho_f \tilde{\xi}^T P b \frac{W_f W_f^T S_f(z)}{\|W_f\|^2}}_{\varphi_f} \right) - \tilde{W}_g^T \left( \underbrace{I_g \rho_g \tilde{\xi}^T P b \frac{W_g W_g^T S_g(z) u_c(t)}{\|W_g\|^2}}_{\varphi_g} \right) \quad (2.71)$$

Alors,  $\dot{V}_1$  devient

$$\dot{V}_1 = -I_f \tilde{W}_f^T \varphi_f - I_g \tilde{W}_g^T \varphi_g \quad (2.72)$$

Démontrons que  $\tilde{W}_f^T \varphi_f \geq 0$ .

Si  $I_f = I_g = 0$ , alors  $\|W_f\| < M_f, \varphi_f = 0$  la conclusion est évidente.

Si  $I_f = I_g = 1$ , alors  $\|W_f\| = M_f$  et  $\xi^T P b W_f^T S_f(z) < 0$ . D'autre part, nous avons :

$$\tilde{W}_f W_f = \frac{1}{2} \|W_f^*\|^2 - \frac{1}{2} \|W_f\|^2 - \frac{1}{2} \|W_f - \tilde{W}_f\|^2 \leq 0 \quad (2.73)$$

Puisque  $\frac{1}{2} \|W_f^*\|^2 \leq M_f^2$  et  $\frac{1}{2} \|W_f\|^2 = M_f^2$  et  $\frac{1}{2} \|W_f - \tilde{W}_f\|^2 \geq 0$  alors  $\tilde{W}_f W_f \leq 0$   
d'où :

$$\tilde{W}_f^T \varphi_f = \rho_f \xi^T P b \tilde{W}_f^T \frac{W_f W_f^T S_f(z)}{\|W_f\|^2} \geq 0 \quad (2.74)$$

De la même manière, on peut démontrer que

$$\tilde{W}_g^T \varphi_g \geq 0 \quad (2.75)$$

En utilisant l'hypothèse 2.5 et l'équation (2.26),  $\dot{V}_2$  peut être bornée comme suit

$$\begin{aligned} \dot{V}_2 &= -\xi^T P b \varepsilon_f(z) - \xi^T P b \varepsilon_g(z) u_c - \xi^T P b d - \xi^T P b g(z) u_r - \frac{1}{\gamma_0} \dot{\varepsilon} \dot{\varepsilon} - \frac{1}{\gamma_0} \dot{\varepsilon}_g \dot{\varepsilon}_g \\ &\leq |\xi^T P b| \bar{\varepsilon}_f + |\xi^T P b u_c| \bar{\varepsilon}_g + |\xi^T P b| \bar{d} - \xi^T P b \left( \frac{g(z)}{g_0} \right) (\hat{\varepsilon} + \hat{\varepsilon}_g |u_c(t)|) \text{sgn}(\xi^T P b) \\ &\quad - \frac{1}{\gamma_0} (\varepsilon^* - \hat{\varepsilon}) \dot{\varepsilon} - \frac{1}{\gamma_0} (\varepsilon_g^* - \hat{\varepsilon}_g) \dot{\varepsilon}_g \end{aligned} \quad (2.76)$$

A partir des (2.53) et (2.54), et nous avons  $\varepsilon^* = (\bar{\varepsilon}_f + \bar{d})$  et  $\varepsilon_g^* = \bar{\varepsilon}_g$ ,  $\dot{V}_2$  devient

$$\dot{V}_2 = -\xi^T P b \left( \frac{g(z)}{g_0} \right) (\hat{\varepsilon} + \hat{\varepsilon}_g |u_c(t)|) \text{sgn}(\xi^T P b) + |\xi^T P b| \hat{\varepsilon} + |\xi^T P b u_c| \hat{\varepsilon}_g \leq 0 \quad (2.77)$$

A partir de l'analyse ci-dessus,  $\dot{V}$  devient :

$$\dot{V} \leq -\frac{1}{2} \xi^T Q \xi \leq 0 \quad (2.78)$$

donc  $\dot{V}(t) \leq 0$  et  $V \in L_\infty$  ce qui implique la bornitude des signaux :  $\xi(t), \tilde{W}_f(t), \tilde{W}_g(t), y_d(t), \dot{y}_d(t), \ddot{y}_d(t), \dots, y_d^{(n-1)}(t), u(t) \in L_\infty$ . La limite  $\lim_{t \rightarrow \infty} V(t) = V(\infty)$  existe.

En intégrant (2.78) de 0 à  $\infty$ , on obtient :

$$\int_0^\infty \dot{V} dt = V(\infty) - V(0) \leq -\frac{1}{2} \int_0^\infty \xi^T(t) Q \xi(t) dt \quad (2.79)$$

ou encore

$$\frac{1}{2} \int_0^\infty \xi^T(t) Q \xi(t) dt \leq V(0) - V(\infty) \quad (2.80)$$

Sachant que  $V$  est une fonction de Lyapunov, on a:

$$V(0) \geq V(\infty) \quad (2.81)$$

Des équations (2.80) et (2.81), on peut alors déduire que:

$$\tilde{\xi}^T(t)Q\tilde{\xi}(t) \in L_2 \quad (2.82)$$

ce qui implique que  $\tilde{\xi}(t) \in L_2$  Donc  $\dot{\tilde{\xi}}(t) \in L_\infty$ , et d'après le lemme de Barbalat[Slotine-91], on conclut que  $\lim_{t \rightarrow \infty} \tilde{\xi}(t) = 0$ , alors (2.49), (2.50), et (2.45) nous avons tous sont bornée. Maintenant, en utilisant la bornitude de  $u, W_f, W_g$  et la convergence de  $\tilde{\xi}(t)$ , nous avons  $\lim_{t \rightarrow \infty} \dot{W}_f(t) = 0, \lim_{t \rightarrow \infty} \dot{W}_g(t) = 0$ .

**Remarque 2.2 :** Pour éviter le phénomène de broutement provoqué par le terme signe dans le terme de commande  $u_r$  (2.47), la fonction signe (2.48) est remplacée par la fonction tangente Hyperbolique  $\varepsilon \tanh(x/\varepsilon)$ , Où  $\varepsilon$  est un petit constant positif ou par la fonction de saturation «sat» de la forme suivante

$$sat(x) = \begin{cases} \varepsilon \operatorname{sign}(x) & \text{si } |x| \geq \varepsilon \\ x & \text{si } |x| < \varepsilon \end{cases}$$

**Remarque 2.3 :** Pour avoir une loi de commande bien définie même lorsque  $\hat{g}(z/W_g) = 0$ , nous avons remplacé  $\hat{g}^{-1}(z/W_g)$  par  $\hat{g}(z/W_g)/(\varepsilon_0 + \hat{g}^2(z/W_g))$ .

### 2.3.3. Résultats de simulation :

Un exemple de simulation est développé, est utilisé pour illustrer l'efficacité de la commande adaptative neuronale proposée pour un système non linéaire non affine en la commande.

On considère le système non linéaire non affiné décrit par l'équation différentielle suivante [Ge-00]:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_1^2 + 0.15u^3 + 0.1(1 + x_2^2)u + \sin(0.1u) \\ y = x_1 \end{cases} \quad (2.83)$$

Comme  $\partial[b(x,u)]/\partial u = 0.45u^2 + 0.1(1 + x_2^2) + 0.1 \cos(0.1u) > 0$  pour tout  $(x,u) \in \mathbb{R}^{n+1}$  cela implique que l'hypothèse 2.1 est satisfaite.

L'objective de la commande est d'assurer une bonne poursuite de signale de référence  $y_d(t) = \sin(t) + \cos(0.5t)$ , par la sortie  $y(t) = x_1$ , pour montre la robustesse du contrôleur proposée, une perturbation externe de la forme  $d(t) = 0.5 \sin(10t)$  est considérée.

Le système (2.30) possède deux réseaux de neurone sous la forme (1.1) pour approximer en ligne les deux fonctions inconnus  $f(z)$  et  $g(z)$ , le nombre des réseaux a été choisi avec

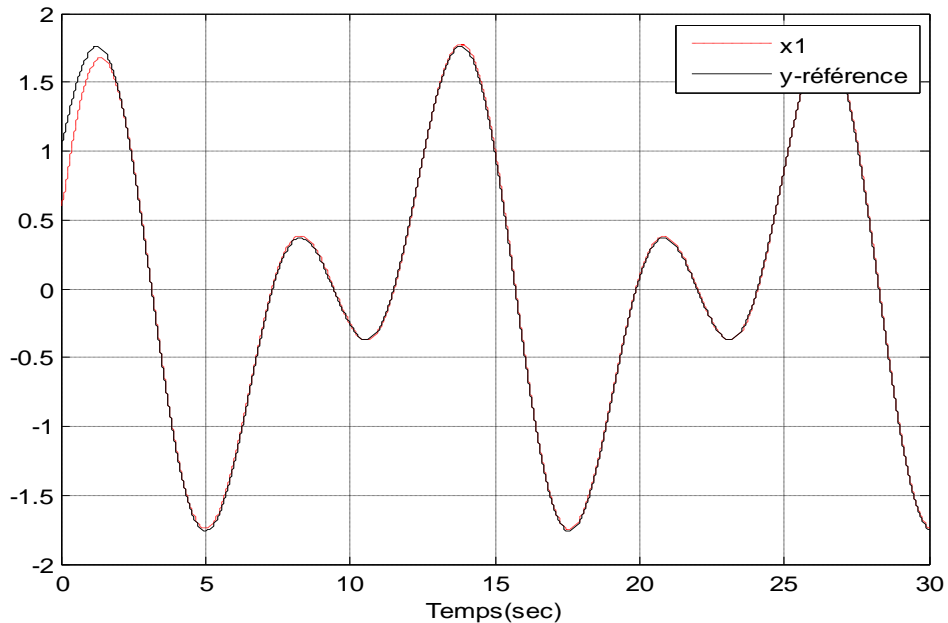
$l = 25$ , et les variables d'entrée sont choisis comme  $z = [z_1, z_2, z_3, z_4, z_5]^T$  avec  $z_1 = x_1$ ,  $z_2 = x_2$ ,  $z_3 = y_d$ ,  $z_4 = \dot{y}_d$  et  $z_5 = \ddot{y}_d + k^T e$  pour chaque fonction  $f(z)$  et  $g(z)$ .

Les paramètres de synthèses et les conditions initiales sont donnés dans le tableau (2.1) suivant :

Les paramètres de synthèses	Les valeurs numériques	Les conditions initiales	Les valeurs numériques
$\Gamma_f$	0.01	$W_f(\mathbf{0})$	0
$\Gamma_g$	1000000	$W_g(\mathbf{0})$	$[-1, 1]$
$\rho_f$	1.1	$\hat{\epsilon}(\mathbf{0})$	0
$\rho_g$	1.1	$\hat{\epsilon}_g(\mathbf{0})$	0
$M_f$	16	$x_1(\mathbf{0})$	0.6
$M_g$	1.6	$x_2(\mathbf{0})$	0.5
$k$	$[1; 2]$	$g_0$	0.1
$Q$	$Diag[10, 10]$	$g_1$	0.6
$P$	$[15 \ 5; 5 \ 5]$		

**Tab.2.1** : les paramètres de synthèses & les conditions initiales

Les figures 2.1 et 2.2 montrent une convergence rapide des états du système  $x_1$  et  $x_2$  vers leurs références. La figure 2.3 représente le signal de commande borné.



**Fig. 2.1** : La variation de l'état du système  $y = x_1$  et sa référence  $y_d$ .

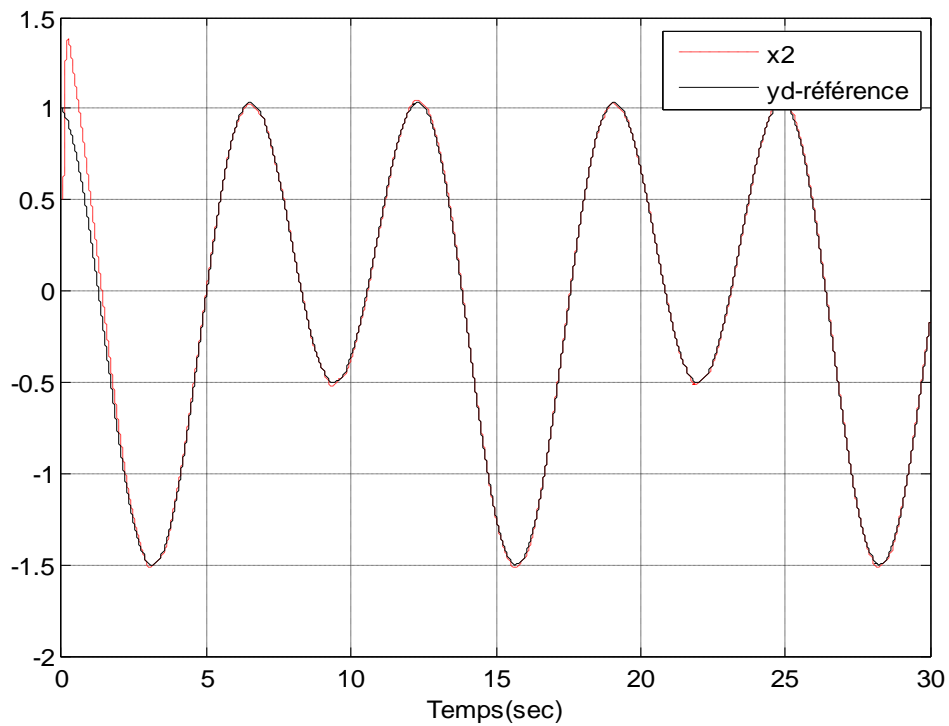


Fig. 2.2 : La variation de l'état du système du système  $\dot{y} = x_2$  et sa référence  $\dot{y}_d$

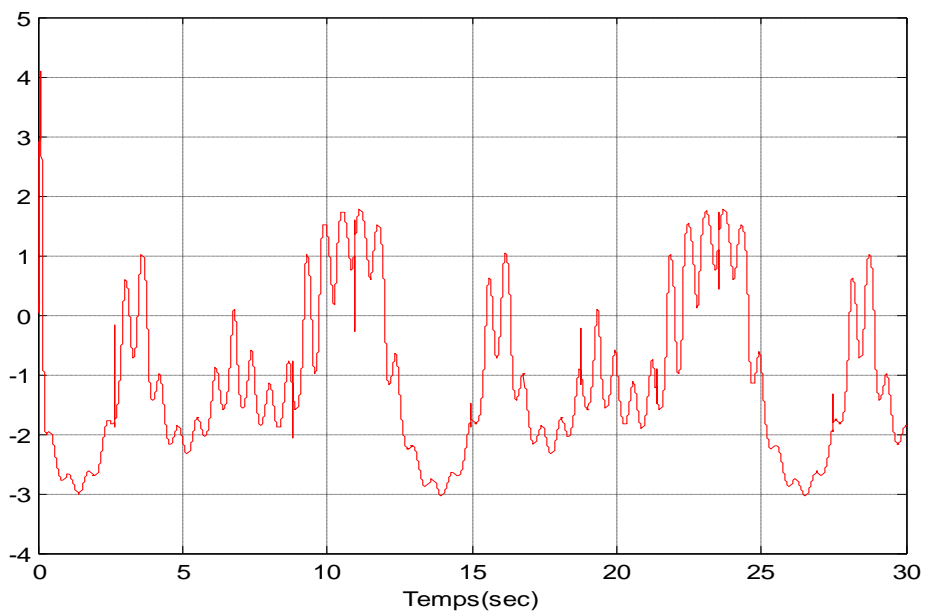


Fig. 2.3 : La variation de la commande.

## 2.4 Commande adaptative directe neuronale :

Dans la commande adaptative directe, un réseau de neurone est utilisé pour approximer la loi de commande idéale  $u^*$  de (2.33). Les paramètres du contrôleur neuronal sont adaptés en utilisant l'algorithme d'adaptation. La stabilité asymptotique du système résultant en boucle fermée est garantie en utilisant la théorie de Lyapunov.

### 2.4.1. Conception de la commande neuronale :

Le lemme 2.1 garantit seulement l'existence d'une loi de commande idéale  $u^*$  qui assure la convergence de l'erreur de poursuite vers zéro mais ne fournit pas de méthode pour sa détermination.

Du fait, l'approximation neuronale adaptative de la commande idéale  $u^*$  (2.33) est définie par

$$\hat{u}(z/W) = W^T S(z) \quad (2.84)$$

avec  $S(z)$  sont de vecteur de fonction neuronale supposés convenablement fixés en avance par l'utilisateur, avec  $W = [w_1, \dots, w_l]^T \in \mathfrak{R}^l$  sont les vecteurs des paramètres ajustables, et  $l$  est le nombre de nœud, et  $z \in \Omega_z \subset \mathfrak{R}^m$  est le vecteur d'entrée, étant  $\Omega_z \subset \mathfrak{R}^m$  un ensemble compact.

Soit :

$$W^* = \underset{w}{\operatorname{argmin}} \{ \sup_{z \in \Omega_z} |\hat{u}(z/W) - u(z)| \} \quad (2.85)$$

Soit  $W$  l'estimer de  $W^*$  et qu'il sera calculé à partir d'un algorithme d'adaptation, puisque les paramètres optimaux  $W^*$  sont des constantes artificielles inconnues introduites uniquement pour faire l'étude théorique de la stabilité de l'algorithme de commande, En fait, la connaissance de leurs valeurs n'est pas nécessaire pour l'implantation des lois de commande adaptatives. Notons que :

$$\tilde{W} = W^* - W \quad (2.86)$$

L'erreur d'estimation paramétrique, et par

$$\varepsilon(z) = u(z) - \hat{u}(z/W^*) \quad (2.87)$$

**Hypothèse 2. 6:** Les erreurs d'approximation sont bornées, c'est-à-dire

$$|\varepsilon(z)| \leq \bar{\varepsilon}$$

où  $\bar{\varepsilon}$  est de constante positive inconnue.

Cette hypothèse est raisonnable puisque nous supposons que les systèmes neuronales utilisés pour l'approximation des fonctions inconnues possèdent la propriété d'approximateur universel.

Pour atteindre les objectifs de commande, nous définissons les lois d'adaptation des paramètres comme suit :

$$\dot{W} = \Gamma \tilde{\xi}^T P b S^T(z) \quad (2.88)$$

et  $\Gamma > 0$  est le pas d'ajustement sélectionnés par le concepteur, et  $P > 0$  est une matrice définie positive qu'on déterminera par la suite.

#### 2.4.2. Analyse de stabilité :

Dans cette section nous allons démontrer que la commande adaptative neuronale directe proposée est stable. Pour démontrer la stabilité de la commande adaptative, nous allons procéder en deux étapes :(i) En premier lieu nous démontrerons la convergence asymptotique de l'erreur de poursuite vers zéros dans le cas ou la perturbation extérieure  $d$  n'existe pas, (ii) et en deuxième lieu, nous démontrerons que la convergence de l'erreur de poursuite vers un ensemble borné dans le cas ou la perturbation extérieure  $d$  existe.

Considérons le système SISO non linéaire affiné dans la commande (2.30), la loi de commande (2.84) avec sa loi d'adaptation (2.88), si les hypothèses [2.1, 2.2, 2.3, 2.6] sont satisfaites, alors tous les signaux de la boucle fermée sont bornés, et le schéma global garantit que :

(i) l'erreur de poursuite converge asymptotiquement vers zéro.

(ii) l'erreur de poursuite  $\tilde{\xi}(t)$  converge à l'ensemble compact  $\Omega_{\tilde{\xi}}$  défini par :

$$\Omega_{\tilde{\xi}} = \left\{ \tilde{\xi} : |\tilde{\xi}| > \frac{\omega_1}{\omega_2} \right\}$$

#### Démonstration :

Pour analyser la stabilité du système bouclé, nous commençons par remplacer la loi de commande effective (2.84) dans le système d'erreur (2.32) ce qui donne :

$$\dot{\tilde{\xi}} = A \tilde{\xi} + b \left[ -f(z) - g(z) \hat{u}(z/W) + y_d^{(n)} - d \right] \quad (2.89)$$

En substituant la valeur idéale de  $f(z)$  à partir de (2.33) dans (2.89), (2.89) peut être réécrite comme suit :

$$\begin{aligned} \dot{\tilde{\xi}} &= A \tilde{\xi} + b \left[ g(z)u - g(z) \hat{u}(z/W) - \underline{k}^T \tilde{\xi}(t) - d \right] \\ &= A_k \tilde{\xi} + b \left[ g(z)(u - \hat{u}(z/W)) - d \right] \end{aligned} \quad (2.90)$$

où  $A_k = (A - b \underline{k}^T)$

et

$$A_k = \begin{bmatrix} 0 & \cdots & I_{(n-1)} \\ \vdots & \ddots & \vdots \\ -k_1 & \cdots & -k_n \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix} \quad (2.91)$$

La matrice  $A_k$  étant stable (valeurs propres à partie réelle négative), pour une matrice  $Q$  symétrique définie positive. On lui associe une équation de Lyapunov  $P$ , ayant pour solution unique une matrice symétrique définie positive, donnée par

$$A_k^T P + P A_k = -Q \quad (2.92)$$

En ajoutant et en soustrayant  $g(z)\hat{u}(z/W^*)$  à (2.90), on obtient :

$$\dot{\xi} = A_k \xi + b[g(z)(u + \hat{u}(z/W^*) - \hat{u}(z/W) - \hat{u}(z/W^*)) - d] \quad (2.93)$$

En utilisant maintenant (2.84), alors on peut écrire :

$$\begin{aligned} \hat{u}(z/W) &= W^T S(z) \\ \hat{u}(z/W^*) &= W^{*T} S(z) \end{aligned} \quad (2.94)$$

A partir de (2.86) et (2.94), alors (2.93) devient :

$$\dot{\xi} = A_k \xi + b[g(z)(u - \hat{u}(z/W^*)) + g(z)\tilde{W}^T S(z) - d] \quad (2.95)$$

(i) on considère que la perturbation extérieure n'existe pas, c'est-à-dire  $d = 0$  pour démontre la partie (i). Alors

Réécrivons le système d'erreur (2.95) comme suit :

$$\dot{\xi} = A_k \xi + b[g(z)(u - \hat{u}(z/W^*)) + g(z)\tilde{W}^T S(z)] \quad (2.96)$$

Considérons maintenant la fonction de Lyapunov candidate suivante :

$$V = \frac{1}{2} \frac{\xi^T P \xi}{b_u} + \frac{1}{2} \tilde{W}^T \Gamma^{-1} \tilde{W} \quad (2.97)$$

où  $\Gamma = \Gamma^T > 0$ , ou  $\tilde{W} = W^* - W$  et  $\dot{\tilde{W}} = -\dot{W}$

La dérivée temporelle de  $V$  est

$$\dot{V} = \frac{1}{2} \frac{\dot{\xi}^T P \xi}{b_u} + \frac{1}{2} \frac{\xi^T P \dot{\xi}}{b_u} - \frac{\xi^T P \xi \dot{b}_u}{2 b_u^2} - \tilde{W}^T \Gamma^{-1} \dot{W} \quad (2.98)$$

En remplaçant par (2.92) et (2.96) dans (2.98), on aura :

$$\begin{aligned} \dot{V} &= \frac{1}{2} \frac{\xi^T Q \xi}{b_u} + \frac{1}{b_u} \xi^T P b [g(z)(u - \hat{u}(z/W^*)) + g(z)\tilde{W}^T S(z)] - \frac{\xi^T P \xi \dot{b}_u}{2 b_u^2} \\ &\quad - \tilde{W}^T \Gamma^{-1} \dot{W} \end{aligned} \quad (2.99)$$

Selon l'hypothèse2.2 et l'équation (2.8), on peut arrangée (2.99) comme suit:



$$\dot{V} = -\frac{\tilde{\xi}^T Q \tilde{\xi}}{2 b_u} + \frac{1}{b_u} \tilde{\xi}^T P b \{g(z)(u - \hat{u}(z/W^*))\} - \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{2 b_u^2} + \tilde{W}^T \Gamma^{-1} \{ \Gamma \tilde{\xi}^T P b S(z) - \dot{W} \} \quad (2.100)$$

En utilisant la loi d'adaptation (2.88) et après simplification, on trouve

$$\dot{V} \leq -\frac{\tilde{\xi}^T Q \tilde{\xi}}{2 b_u} - \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{2 b_u^2} + \tilde{\xi}^T P b |u - \hat{u}(z/W^*)| \quad (2.101)$$

En utilisant le fait que le système neuronale HONN utilisé est un approximateur universel, on a  $|u - \hat{u}(z/W^*)| \leq \varepsilon$ . Si l'on impose  $\varepsilon \rightarrow 0$ ,

$$\dot{V} \leq -\frac{1}{2} \left( \frac{\tilde{\xi}^T Q \tilde{\xi}}{b_u} + \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{b_u^2} \right) \quad (2.102)$$

Pour compléter la démonstration, nous devons montrer que  $\left( \frac{\tilde{\xi}^T Q \tilde{\xi}}{b_u} + \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{b_u^2} \right)$  est positif.

A partir de l'hypothèse 2.2 et l'équation (2.8), nous avons :

$$0 < g_0 \leq b_u \leq g_1 \implies \frac{1}{b_u} \leq \frac{1}{g_0}, \frac{-1}{b_u} \leq \frac{-1}{g_1}$$

On peut écrire

$$-H \tilde{\xi}^T P \tilde{\xi} \leq \tilde{\xi}^T P \tilde{\xi} \dot{b}_u \quad (2.103)$$

du fait que

$$\tilde{\xi}^T P \tilde{\xi} \leq \lambda_{\max}(P) \tilde{\xi}^T \tilde{\xi} \quad (2.104)$$

En utilisant (2.102), (2.103), nous pouvons avoir

$$-H \lambda_{\max}(P) \tilde{\xi}^T \tilde{\xi} \leq \tilde{\xi}^T P \tilde{\xi} \dot{b}_u \quad (2.105)$$

Étant donné  $-H \lambda_{\max}(P) \tilde{\xi}^T \tilde{\xi} < 0$  et  $\frac{b_u}{g_0} \geq 1$ , l'inégalité (2.105) peut être simplifiée comme suit :

$$-H \lambda_{\max}(P) \tilde{\xi}^T \tilde{\xi} \frac{b_u}{g_0} \leq \tilde{\xi}^T P \tilde{\xi} \dot{b}_u \quad (2.106)$$

On peut écrire aussi

$$\lambda_{\min}(Q) \tilde{\xi}^T \tilde{\xi} b_u \leq \tilde{\xi}^T Q \tilde{\xi} b_u \quad (2.107)$$

Maintenant en utilisant (2.106) et (2.107) pour obtenir

$$\lambda_{\min}(Q) \tilde{\xi}^T \tilde{\xi} b_u - H \lambda_{\max}(P) \tilde{\xi}^T \tilde{\xi} \frac{b_u}{g_0} \leq \tilde{\xi}^T Q \tilde{\xi} b_u + \tilde{\xi}^T P \tilde{\xi} \dot{b}_u \quad (2.108)$$

On multiplie l'inégalité (2.108) par  $\frac{g_0}{\lambda_{\max}(P)}$  on peut écrire

$$\tilde{\xi}^T Q \tilde{\xi} \frac{b_u g_0}{\lambda_{\max}(P)} + \tilde{\xi}^T P \tilde{\xi} \frac{\dot{b}_u g_0}{\lambda_{\max}(P)} \geq \frac{\lambda_{\min}(Q) \tilde{\xi}^T \tilde{\xi} b_u g_0}{\lambda_{\max}(P)} - H \tilde{\xi}^T \tilde{\xi} b_u \quad (2.109)$$

Étant donné l'hypothèse 2.2 et l'équation (2.9) nous avons

$$\frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)} g_0 > H$$

On peut réécrire l'équation (2.109) comme suit

$$\tilde{\xi}^T Q \tilde{\xi} \frac{b_u g_0}{\lambda_{\max}(P)} + \tilde{\xi}^T P \tilde{\xi} \frac{\dot{b}_u g_0}{\lambda_{\max}(P)} \geq H \tilde{\xi}^T \tilde{\xi} b_u - H \tilde{\xi}^T \tilde{\xi} \dot{b}_u = 0 \quad (2.110)$$

À partir (2.109), peut être facilement montré que

$$\frac{\tilde{\xi}^T Q \tilde{\xi}}{b_u} + \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{b_u^2} \geq 0 \quad (2.111)$$

alors :

$$\dot{V} \leq -\frac{1}{2} \left( \frac{\tilde{\xi}^T Q \tilde{\xi}}{b_u} + \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{b_u^2} \right) \leq 0 \quad (2.112)$$

ce qui prouve la stabilité de la structure de commande.

D'après le lemme de Barbalat [**Slotine-91**], nous sommes garantis que l'erreur de poursuite converge asymptotiquement à un voisinage d'origine. En plus, la convergence de la loi d'adaptation (2.88) à zéro est établie par sa définition.

(ii) on considère que la perturbation extérieure n'existe pas, c'est-à-dire  $d \neq 0$  pour démontre la partie (ii).

Alors, l'équation d'erreur (2.95) peut être écrite comme :

$$\dot{\tilde{\xi}} = A_k \tilde{\xi} + b[g(z)(u - \hat{u}(z/W^*)) + g(z)\tilde{W}^T S(z) - d] \quad (2.113)$$

En utilisant l'équation d'erreur d'approximation minimale (2.87), (2.113) devient :

$$\dot{\tilde{\xi}} = A_k \tilde{\xi} + b[g(z)\tilde{W}^T S(z) + g(z)\varepsilon(z) - d] \quad (2.114)$$

Considérons maintenant la fonction de Lyapunov candidate suivante :

$$V = \frac{1}{2} \frac{\tilde{\xi}^T P \tilde{\xi}}{b_u} + \frac{1}{2} \tilde{W}^T \Gamma^{-1} \tilde{W} \quad (2.115)$$

où  $\Gamma = \Gamma^T > 0$ , ou  $\tilde{W} = W^* - W$  et  $\dot{\tilde{W}} = -\dot{W}$

La dérivée temporelle de  $V$  est

$$\dot{V} = \frac{1}{2} \frac{\dot{\tilde{\xi}}^T P \tilde{\xi}}{b_u} + \frac{1}{2} \frac{\tilde{\xi}^T P \dot{\tilde{\xi}}}{b_u} - \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{2 b_u^2} - \tilde{W}^T \Gamma^{-1} \dot{\tilde{W}} \quad (2.116)$$

En remplaçant par (2.92) et (2.114) dans (2.116), on aura :

$$\dot{V} = \frac{1}{2} \frac{\tilde{\xi}^T Q \tilde{\xi}}{b_u} + \frac{1}{b_u} \tilde{\xi}^T P b (g(z) \tilde{W}^T S(z) + g(z) \varepsilon(z) - d) - \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{2 b_u^2} - \tilde{W}^T \Gamma^{-1} \dot{W} \quad (2.117)$$

Selon l'hypothèse 2.2 et l'équation (2.8) et (2.27), peut être arrangée (2.117) comme suit :

$$\dot{V} = -\frac{\tilde{\xi}^T Q \tilde{\xi}}{2 b_u} + \frac{1}{b_u} \tilde{\xi}^T P b (g(z) \varepsilon(z) - d) - \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{2 b_u^2} + \tilde{W}^T \Gamma^{-1} (\Gamma \tilde{\xi}^T P b S(z) - \dot{W}) \quad (2.118)$$

En utilisant la loi d'adaptation (2.88), on trouve

$$\dot{V} \leq -\frac{\tilde{\xi}^T Q \tilde{\xi}}{2 b_u} - \frac{\tilde{\xi}^T P \tilde{\xi} \dot{b}_u}{2 b_u^2} + \frac{1}{b_u} \tilde{\xi}^T P b (g(z) \varepsilon(z) - d) \quad (2.119)$$

de plus, (2.119) peut être bornée par

$$\dot{V} \leq -\frac{1}{2b_u} \lambda_{\min}(Q) |\tilde{\xi}|^2 + \frac{1}{2b_u} \frac{|\dot{b}_u|}{b_u} \lambda_{\min}(P) |\tilde{\xi}|^2 + \frac{1}{b_u} |\tilde{\xi}| |Pb| (g(z) |\varepsilon(z)| + |d|) \quad (2.120)$$

ou encore, a partir de l'hypothèse 2.6, l'hypothèse 2.2 et l'équation (2.28), et nous avons  $\frac{1}{b_u} \leq \frac{1}{g_0}$ , on aura :

$$\dot{V} \leq -\frac{|\tilde{\xi}|}{2g_0} \left( |\tilde{\xi}| \lambda_{\min}(Q) + \frac{H}{g_0} |\tilde{\xi}| \lambda_{\min}(P) - 2|Pb|(g(z) \bar{\varepsilon} + \bar{d}) \right) \quad (2.121)$$

Étant donné de l'hypothèse 2.2 et l'équation (2.9) nous avons

$$\frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)} g_0 > H$$

$$\dot{V} \leq -\frac{|\tilde{\xi}|}{2g_0} \left( |\tilde{\xi}| \lambda_{\min}(Q) + \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)} \lambda_{\min}(P) |\tilde{\xi}| - 2|Pb|(g(z) \bar{\varepsilon} + \bar{d}) \right) \quad (2.122)$$

donc :

$$\dot{V} \leq -\frac{|\tilde{\xi}|}{2g_0} (|\tilde{\xi}| \omega_2 - \omega_1) \quad (2.223)$$

avec :

$$\omega_1 = 2|Pb|(g(z) \bar{\varepsilon} + \bar{d})$$

$$\omega_2 = \frac{\lambda_{\min}(Q)(\lambda_{\max}(P) - \lambda_{\min}(P))}{\lambda_{\max}(P)}$$

$\dot{V}$  deviendra négatif tant que  $|\tilde{\xi}| > \omega_1/\omega_2$ . Par conséquent, les erreurs de poursuite convergent vers une région bornée donnée par:

$$\Omega_{\tilde{\xi}} = \left\{ \tilde{\xi} : |\tilde{\xi}| \leq \frac{\omega_1}{\omega_2} \right\} \quad (2.224)$$

ce qui implique la bornitude de tous les signaux dans le système bouclé, c'est-à-dire la perturbation étant supposé bornée et l'erreur d'approximation minimal peut être construite arbitrairement petite par l'utilisateur de la propriété d'approximation neuronal, alors la quantité  $\omega_1$  est borné, ce qui implique que l'ensemble  $\Omega_{\tilde{\xi}}$  dans (2.224) est borné.

Maintenant  $\dot{V}$  deviendra négatif tant que  $\tilde{\xi}(t)$  est en dehors de l'ensemble  $\Omega_{\tilde{\xi}}$ , selon la théorie de la stabilité de Lyapunov, nous concluons que l'erreur  $\tilde{\xi}(t)$  est bornée et va converger vers  $\Omega_{\tilde{\xi}}$ . Alors et la fonction candidate  $V$  est bornée. Par conséquent, on peut démontrer que les erreurs  $\tilde{\xi}(t) \in L_2$  sont carrés intégrables et que  $\dot{\tilde{\xi}}(t) \in L_\infty$  sont bornées, d'où, par application du lemme de Barbalat [**Slotine-91**], on déduit la convergence vers zéro des erreurs de poursuite  $\lim_{t \rightarrow \infty} \|\tilde{\xi}(t)\| = 0$  de tous les systèmes.

### 2.4.3 Résultats de simulation :

Pour montrer les performances de la méthode de commande neuronale adaptative directe proposée, cette partie, présente une simulation numérique effectuée sur le modèle dynamique non affine donné en section 2.3.3. On peut démontrer que le modèle dynamique satisfait les hypothèses [2.1- 2.3].

L'objectif de la commande consiste à forcer la sortie du système  $y(t)$  à suivre une consigne désirée  $y_d(t) = \sin(t)$ .

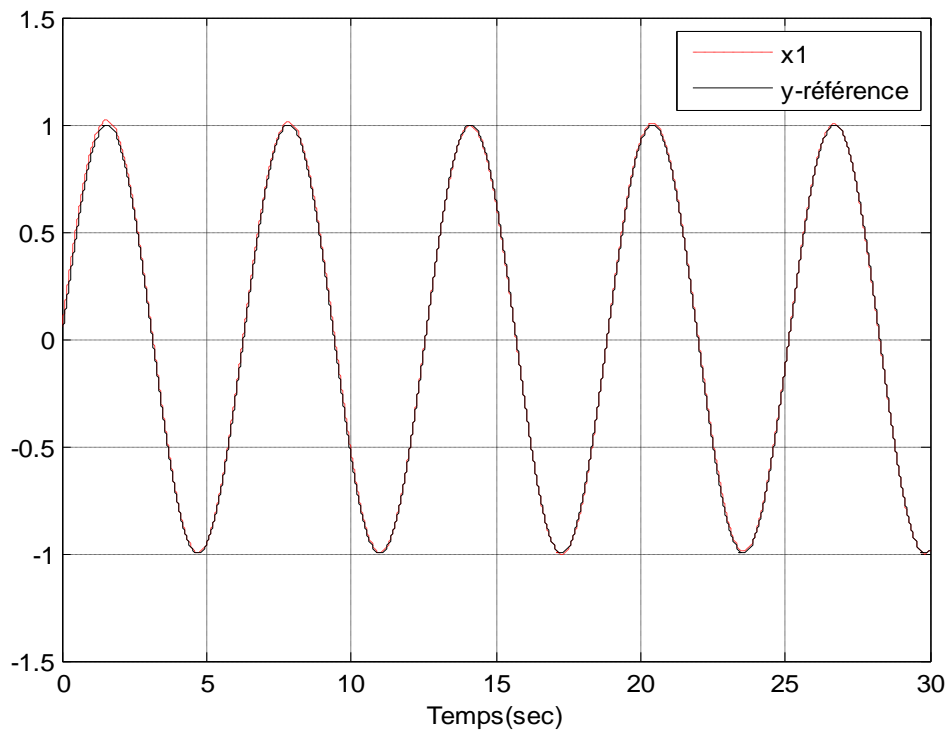
Dans cet exemple de simulation, les réseaux de neurone HONN sous la forme (1.1) sont utilisée pour approximer le contrôleur implicite idéale inconnu, le nombre des réseaux est choisi avec  $l = 25$ , et les variables d'entrée sont choisis comme  $z = [z_1, z_2, z_3, z_4, z_5]^T$  avec  $z_1 = x_1$ ,  $z_2 = x_2$ ,  $z_3 = y_d$ ,  $z_4 = \dot{y}_d$  et  $z_5 = \ddot{y}_d + k^T e$ .

Les paramètres de synthèses et les conditions initiales sont donnés dans le tableau (2.2) suivant :

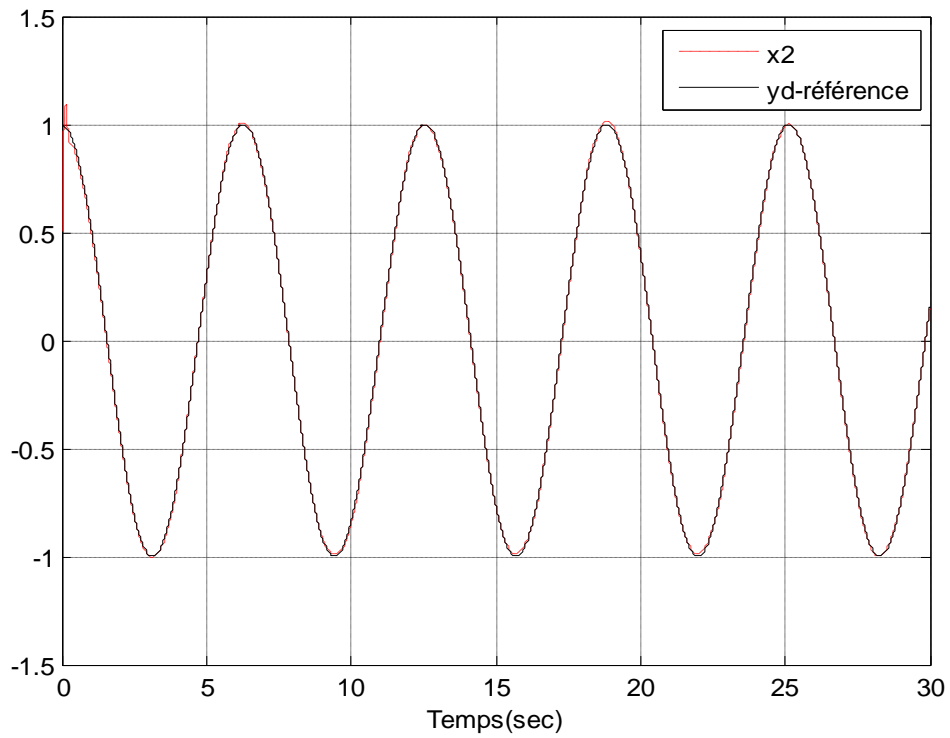
Les paramètres de synthèses	Les valeurs numériques	Les conditions initiales	Les valeurs numériques
$\Gamma$	120	$W(0)$	0
$k$	[10 ; 5]	$x_1(0)$	0.06
$Q$	$Diag[10, 10]$	$x_2(0)$	0.5
$P$	[13.50 0.5 ; 0.5 1.10]		

**Tab.2.2** : les paramètres de synthèses et les conditions initiales.

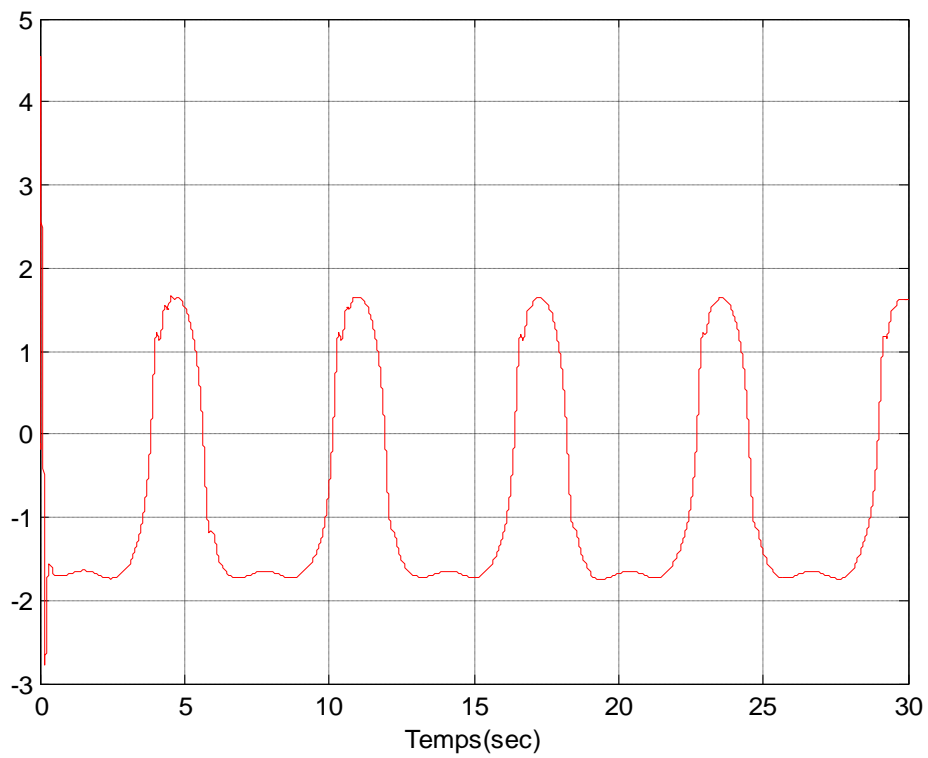
Les figures 2.4 - 2.5 montrent une convergence rapide des états du système  $x_1$  et  $x_2$  vers leurs références. La figure 2.6 représente le signal de commande.



**Fig. 2.4** : la variation de l'état du système  $y = x_1$  et sa référence  $y_d$ .



**Fig. 2.5 :** Evolution de l'état du système du système  $\dot{y} = x_2$  et sa référence  $\dot{y}_d$ .



**Fig. 2.6 :** Signal de commande.

## **2.5. Conclusion :**

Dans ce chapitre, nous avons proposé des approches adaptatives neuronales, pour la commande d'une classe de systèmes non linéaires SISO non affines en la commande sous la forme générale, nous avons utilisé le développement en série de Taylor autour d'un contrôleur idéal inconnu pour transformer le système non affine en un système affine. Et puis deux approches distinctes formulent pour la commande adaptative neuronale de système transformé, l'approche directe et l'approche indirecte. Dans l'approche directe, les systèmes neuronaux sont utilisés pour approximer un contrôleur idéal inconnu. En revanche la commande indirecte exploite deux systèmes adaptatifs neuronale pour approximer la dynamique du système affine, et la loi de commande est synthétisée en se basant sur ces estimées. Dans cette approche, les lois de commande sont bien définies à tout moment, et le problème des erreurs de reconstruction et la perturbation externe sont traité par l'introduction d'un terme de robustification de type mode glissant. Le problème de la singularité engendré par cette approche est résolu, par l'utilisation d'un algorithme de projection.

Les lois d'adaptation sont déduites via l'étude de stabilité, au sens de Lyapunov, du système en boucle fermée. Deux exemples de simulation sont mis en œuvre pour valider l'approche proposée.

Dans le chapitre suivant, une approche basée sur la commande adaptative neuronale directe, qui exploite le théorème de la fonction implicite globale, sera développée pour prendre en charge les systèmes non linéaires MIMO non affines en la commande.

---

## *Chapitre 3*

### *La Commande Adaptative Neuronale pour une Classe de Systèmes Non Linéaire MIMO Non-affine*

---

3.1 Introduction

3.2 La transformation du système non-linéaire MIMO

3.3 Commande adaptative directe neuronale

3.4 Conclusion



### 3.1. Introduction :

La conception de contrôleurs adaptatifs robustes pour les systèmes multivariables non-linéaires non affines en la commande reste une tâche difficile dans le domaine des systèmes de commande. La principale difficulté pour la commande des systèmes non affines en la commande réside dans le fait que les signaux de commande ne peuvent pas être explicitement obtenus même si la dynamique du système est bien connue.

En pratique, il existe plusieurs systèmes non linéaires avec une structure non affine en la commande, c'est-à-dire des systèmes caractérisés par des entrées de commande qui n'apparaissent pas linéairement dans l'équation d'état du système, tels que les processus biochimiques, certains réacteurs chimiques, neutralisation du pH, les colonnes de distillation, la lévitation magnétique, etc.

Dans la littérature, la commande des systèmes MIMO sous la forme normale est très peu documentée mis à part la commande par logique floue [Liu-07] [Wang-07]. Peu de résultats existent pour les systèmes de forme strict [Ge-04]. Dans [Wang-07], les auteurs se basent sur le développement en série de Taylor autour d'un point  $(x_0, u_0)$  pour obtenir un modèle affine en la commande. Le modèle affine obtenu est approximé par un modèle flou de type T-S afin de déduire le contrôleur qui garanti les objectifs de la commande. Un vecteur des termes de commande type mode glissant est ajouté au contrôleur pour compenser l'effet des erreurs d'approximation et des perturbations externes. Dans [Liu-07], les auteurs utilisent le théorème de la valeur moyenne pour construire un modèle affine. Cependant les travaux précités présentent des contrôleurs adaptatifs indirects où le problème de la singularité est résolu par l'algorithme de projection.

L'objectif de ce chapitre est la synthèse d'un contrôleur neuronal adaptatif direct d'une classe de système MIMO non linéaire non affine en la commande. Premièrement, le théorème de la fonction implicite globale [Zhang-06] est utilisé pour démontrer l'existence d'un contrôleur idéal inconnu qui est capable d'accomplir l'objective de la commande. Deuxièmement, on transforme le modèle non affine en un modèle affine par un développement en série de Taylor autour du contrôleur idéal, puis on synthétise la loi de commande par feedback linearization. En fin, pour des raisons des incertitudes sur le modèle liées aux problèmes de transformations et modélisation, un réseau de neurone est utilisé pour approximer la loi de commande idéale.

La robustesse des systèmes vis à vis des erreurs d'approximation est assurée par un terme de commande supplémentaire de type mode glissant. L'analyse de stabilité des systèmes en boucle fermée est étudiée en utilisant la synthèse de Lyapunov. Deux exemples de simulation, sont présentés pour mettre en évidence l'efficacité de l'approche proposée.

### 3.2. La transformation du système MIMO non-linéaire :

#### 3.2.1. Position du problème :

Considérons la classe des systèmes non linéaires multivariables non affines en la commande décrite par les équations différentielles suivantes :

$$\begin{cases} \dot{y}_1^{(n_1)} = f_1(x, u) \\ \vdots \\ \dot{y}_p^{(n_p)} = f_p(x, u) \end{cases} \quad (3.1)$$

avec  $x = [x_1^T, x_2^T \dots, x_p^T]^T \in \mathfrak{R}^n$  représente le vecteur d'état mesurable, où  $x_i^T = [y_i, \dot{y}_i \dots y_i^{(n_i-1)}] \in \mathfrak{R}^{n_i} \quad \forall i = 1, \dots, p$ , et  $n = \sum_{i=1}^p n_i$ .  $u = [u_1, \dots, u_p]^T \in \mathfrak{R}^p$  et  $y = [y_1, \dots, y_p]^T \in \mathfrak{R}^p$  sont respectivement l'entrée et la sortie du système (3.1), et  $f_i(\dots), i = 1, \dots, p$  sont des fonctions continues non affines inconnues.

On dénote

$$y^n = [y_1^{n_1}, y_2^{n_2} \dots, y_p^{n_p}]^T \in \mathfrak{R}^p, \quad F(x, u) = [f_1(x, u), \dots, f_p(x, u)]^T \in \mathfrak{R}^p$$

Le système (3.1) peut être réécrit sous la représentation d'état suivante :

$$\begin{cases} \dot{x} = \underline{A}x + \underline{B}[F(\xi, u)] \\ y = \underline{C}^T x \end{cases} \quad (3.2)$$

$$\underline{A} \triangleq \text{diag}[A_1, A_2, \dots, A_p], \quad \underline{B} \triangleq [b_1, b_2, \dots, b_p]^T, \quad \underline{C} \triangleq [c_1, c_2, \dots, c_p]^T$$

où

$$A_i = \begin{bmatrix} 0_{(n-1)} & I_{(n-1)} \\ 0 & 0_{(n-1)}^T \end{bmatrix} \in \mathfrak{R}^{n_i \times n_i}, \quad b_i = \begin{bmatrix} 0_{(n-1)} \\ 1 \end{bmatrix} \in \mathfrak{R}^{n_i \times 1}, \quad c_i = \begin{bmatrix} 1 \\ 0_{(n-1)} \end{bmatrix} \in \mathfrak{R}^{n_i \times 1}$$

et  $I_{(n-1)}$  et  $0_{(n-1)}$  indiquent la matrice d'identité et un vecteur de zéros de dimension  $(n - 1)$ , respectivement.

L'objectif est de concevoir une loi commande capable de forcer les sorties du système  $y_1(t) \dots \dots y_p(t)$  à suivre de trajectoires de référence  $y_{d1}(t) \dots \dots y_{dp}(t)$  prédéterminées, sous la contrainte que tous les signaux impliqués soient bornés.

Pour aboutir aux objectifs précités, on considère les suppositions et théorèmes suivants :

**Hypothèse 3.1 :** Le vecteur de fonctions  $F(x, u)$  est de classe  $C^1$  pour tout  $(x, u) \in \mathfrak{R}^{n \times p}$

**Hypothèse 3.2 :** Le système (3.2) satisfait l'inégalité suivante :

$\left| \left[ \frac{\partial}{\partial u} F(x, u) \right]_{ii} - \sum_{j \neq i} \left| \left[ \frac{\partial}{\partial u} F(x, u) \right]_{ij} \right| \geq d, \forall (x, u) \in \mathfrak{R}^{n \times p}, i = 1, \dots, p,$  où  $d$  est une constante positive.

**Hypothèse 3.3 :** La matrice  $[\partial F(x, u)/\partial u]$  est symétrique et définie positive.

**Hypothèse 3.4 [Doudou-13] :** Il existe une fonction non linéaire positive inconnue  $\beta(\underline{z})$  telle que

$$\frac{1}{2} \left\| G(\underline{z})^{-1} \right\| \leq \alpha \beta(\underline{z})$$

où  $\alpha$  est la valeur propre minimale de la matrice  $G(\underline{z})^{-1}$ .

**Hypothèse 3.5 :** la matrice  $G(\underline{z})$  est une matrice inversible.

**Hypothèse 3.6:** Nous supposons que nous avons  $p$  trajectoires de référence  $y_{di}, i = 1, \dots, p$  bornées, avec des dérivées  $y_{di}^{(j)}, i = 1, \dots, p, j = 1, \dots, n_i$  bornées. Si on définit le vecteur de référence comme :

$$y_d^T = \left[ y_{d1} \quad \dot{y}_{d1} \quad \dots \quad y_{d1}^{(n_1-1)} \quad y_{d2} \quad \dot{y}_{d2} \quad \dots \quad y_{d2}^{(n_2-1)} \quad \dots \quad y_{dp} \quad \dot{y}_{dp} \quad \dots \quad y_{dp}^{(n_p-1)} \right]^T$$

Alors, la dynamique de la trajectoire de référence peut être décrite par l'équation d'état :

$$\dot{y}_d = \underline{A}y_d + \underline{B}m \quad (3.3)$$

$$m^T = \left[ y_{d1}^{(n_1)} \quad y_{d2}^{(n_2)} \quad \dots \quad y_{dp}^{(n_p)} \right] \in \mathfrak{R}^{1 \times p}$$

On définit les erreurs de poursuite par :

$$\begin{aligned} e_1(t) &= y_{d1}(t) - y_1(t) \\ &\vdots \\ e_p(t) &= y_{dp}(t) - y_p(t) \end{aligned} \quad (3.4)$$

La dynamique des erreurs de poursuite sera alors donnée par:

$$\dot{e} = \underline{A}e + \underline{B}[-F(x, u) + m] \quad (3.5)$$

$$\underline{A} \triangleq \text{diag}[A_1, A_2, \dots, A_p], \quad \underline{B} \triangleq [b_1, b_2, \dots, b_p]^T$$

où  $A_i = \begin{bmatrix} 0 & I^{(n_i-1)} \\ 0 & \dots & 0 \end{bmatrix} \in \mathfrak{R}^{n_i \times n_i}, b_i = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix} \in \mathfrak{R}^{n_i \times 1}.$

**Théorème de la fonction implicite globale [Zhang-06]:** Soit  $f: \mathfrak{R}^n \times \mathfrak{R}^m \rightarrow \mathfrak{R}^k (k \leq m)$  un vecteur de fonctions continues et dérivables pour la deuxième variable  $u \in \mathfrak{R}^m$  et

$\left[ \frac{\partial}{\partial u} f(x, u) \right]_{ij}$ ,  $i = 1, \dots, k$  et  $j = 1, \dots, m$  est le terme d'ordre  $ij$  de la matrice Jacobienne  $\frac{\partial}{\partial u} f(x, u)$ .

Si l'inégalité suivante est vérifiée

$$\left| \left[ \frac{\partial}{\partial u_i} f(x, u) \right]_{ii} \right| - \sum_{j \neq i} \left| \left[ \frac{\partial}{\partial u_j} f(x, u) \right]_{ij} \right| \geq d, \forall (x, u) \in \mathfrak{R}^{n \times m}, i = 1, \dots, p$$

où  $d$  est une constante positive, alors il existe une fonction continue unique  $g: \mathfrak{R}^n \rightarrow \mathfrak{R}^m$  telle que  $f(x, g(x)) = 0$ .

**Lemme 3.1 :** Considérons le système (3.2) satisfaisant les suppositions 3.1–3.6, alors il existe un contrôleur idéal  $u^*$  tel que

$$\dot{e} = (\underline{A} - \underline{Bk})e \quad (3.6)$$

avec  $\underline{A}_k = (\underline{A} - \underline{Bk})$

Le vecteur de gain de retour d'état  $\underline{k} \in \mathfrak{R}^{p \times n}$ , est choisie tel que la matrice  $\underline{A}_k$  est Hurwitz (toutes ses valeurs propres à parties réelles négatives), par conséquent le système bouclé sera exponentiellement stable. On lui associe une équation de Lyapunov, ayant pour solution unique une matrice symétrique définie positive, donnée par

$$\underline{A}_k^T P + P \underline{A}_k = -Q \quad (3.7)$$

**Démonstration :** En ajoutant et en retranchant le terme  $\underline{Bk}e$  sur le côté droit de (3.10), on obtient :

$$\dot{e} = (\underline{A} - \underline{Bk})e - \underline{B}[F(x, u) - \varphi] \quad (3.8)$$

avec :

$$\begin{aligned} \varphi_1 &= y_{d_1}^{(n_1)} + k_{1,n_1} e_1^{(n_1-1)} + \dots + k_{1,1} e_1 \\ &\quad \vdots \\ \varphi_p &= y_{d_p}^{(n_p)} + k_{p,n_p} e_1^{(n_p-1)} + \dots + k_{p,1} e_p \end{aligned} \quad (3.9)$$

et  $\underline{\varphi} = [\varphi_1, \dots, \varphi_p]^T$ .

A partir de l'hypothèse 3.3, et le fait que  $\partial[\varphi]/\partial u = 0$  ainsi

$$\left| \left[ \frac{\partial}{\partial u} (F(x, u) - \underline{\varphi}) \right]_{ii} \right| - \sum_{j \neq i} \left| \left[ \frac{\partial}{\partial u} (F(x, u) - \underline{\varphi}) \right]_{ij} \right| \geq d, \forall (x, u) \in \mathfrak{R}^{n \times p},$$

d'après le théorème de la fonction implicite globale, il existe un contrôleur idéal  $u^*(z)$ ,

avec  $\underline{z} = [y^T, y_d^T, \underline{\varphi}]^T$ , tel que

$$F(x, u^*(\underline{z})) - \underline{\varphi} = 0 \quad (3.10)$$

On considère la fonction candidate de Lyapunov suivante :

$$V = \frac{1}{2} e^T P e \quad (3.11)$$

La dérivée de  $V$  est donnée par :

$$\begin{aligned} \dot{V} &= \frac{1}{2} ((\dot{e}^T P e + e^T P \dot{e})) \\ &= \frac{1}{2} e^T [\underline{A}_k^T P + P \underline{A}_k] e - e^T P \underline{B} [F(x, u) - \underline{\varphi}] \end{aligned}$$

Si  $u = u^*$  (3.8), alors on obtient

$$\dot{V} = \frac{1}{2} e^T [\underline{A}_k^T P + P \underline{A}_k] e \quad (3.12)$$

$$\dot{V} \leq -\frac{1}{2} e^T Q e \leq 0 \quad (3.13)$$

Ce qui permet de déduire que  $e$  et  $\dot{e} \in L_\infty$

L'intégration de l'équation (3.13), donne

$$\int_0^\infty \|e\|^2 d\tau \leq \frac{1}{\lambda_{\min}(Q)} [|V(0)| - |V(t)|] \quad (3.14)$$

ce qui implique que  $e(t) \in L_2$  donc  $\dot{e}(t) \in L_\infty$ , et d'après le lemme de Barbalat [**Slotine-91**], on conclut que  $\lim_{t \rightarrow \infty} e(t) = 0$  .

Pour faciliter la conception du contrôleur, on transforme le système non affine (3.2) à un système affine en effectuant un développement en série de Taylor autour d'une commande idéale inconnue  $u^*(z)$  comme suit :

$$F(x, u) = F(\underline{z}) + G(\underline{z})u + H(\underline{z}, u) \quad (3.15)$$

avec

$$\begin{aligned} F(\underline{z}) &= [F_1(\underline{z}), F_2(\underline{z}), \dots, F_p(\underline{z})]^T \\ &= F(x, u^*(\underline{z})) - [\partial F(x, u) / \partial u]_{u=u^*} u^*(\underline{z}) \end{aligned} \quad (3.16)$$

$$G(\underline{z}) = [g_{ij}(\underline{z})] = [\partial F(x, u) / \partial u]_{u=u^*}$$

et

$$[\partial F(x, u) / \partial u]_{u=u^*} = \begin{bmatrix} \frac{\partial f_1(x, u)}{\partial u_1} & \dots & \frac{\partial f_1(x, u)}{\partial u_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_p(x, u)}{\partial u_1} & \dots & \frac{\partial f_p(x, u)}{\partial u_p} \end{bmatrix}$$

où  $H(\underline{z}, u)$  sont les termes d'ordre supérieur du développement, on suppose que la somme des termes d'ordre supérieur est négligeable alors on considère que  $H(\underline{z}, u) = 0$ .

Par substitution (3.15) dans (3.2), un terme affine par rapport à  $u$  apparaît comme suit :

$$\begin{cases} \dot{x} = \underline{A}x + \underline{B}[F(\underline{z}) + G(\underline{z})u] \\ y = \underline{C}^T x \end{cases} \quad (3.17)$$

avec  $\underline{A} \triangleq \text{diag}[A_1, A_2, \dots, A_p]$ ,  $\underline{B} \triangleq [b_1, b_2, \dots, b_p]^T$ ,  $\underline{C} \triangleq [c_1, c_2, \dots, c_p]^T$

où

$$A_i = \begin{bmatrix} 0_{(n-1)} & I_{(n-1)} \\ 0 & 0_{(n-1)}^T \end{bmatrix} \in \mathfrak{R}^{n_i \times n_i}, b_i = \begin{bmatrix} 0_{(n-1)} \\ 1 \end{bmatrix} \in \mathfrak{R}^{n_i \times 1}, c_i = \begin{bmatrix} 1 \\ 0_{(n-1)} \end{bmatrix} \in \mathfrak{R}^{n_i \times 1}$$

et  $I_{(n-1)}$  et  $0_{(n-1)}$  indiquent la matrice d'identité et un vecteur de zéros de dimension  $(n - 1)$ , respectivement.

À partir (3.5) et (3.15), nous pouvons obtenir l'équation d'erreur comme suit :

$$\dot{e} = \underline{A}e + \underline{B}[-F(\underline{z}) - G(\underline{z})u + m] \quad (3.18)$$

Si les fonctions non linéaires  $F(\underline{z})$  et  $G(\underline{z})$  sont connues, les objectifs de commande peuvent être atteints en choisissant la loi de commande idéale suivante :

$$u = u^* = [G(\underline{z})]^{-1} \left[ -F(\underline{z}) + m + \underline{k}^T e + \frac{e^T P e}{e^T P \underline{B}} \beta(\underline{z}) \right] \quad (3.19)$$

où  $\underline{k}^T = [k_{1c}^T, k_{2c}^T, \dots, k_{pc}^T]$  avec  $k_{ic}^T = [k_{i1}, k_{i2}, \dots, k_{in_i}]$  est le vecteur de gain de retour d'état, et  $e \triangleq [e_1^T, e_2^T, \dots, e_p^T]^T$  avec  $e_i \triangleq [e_i, \dot{e}_i, \dots, e_i^{(n_i-1)}]^T$ .

Par substitution l'équation (3.19) par l'équation (3.17) l'équation suivante est obtenue :

$$\begin{bmatrix} e_1^{(n_1)} + k_{1n_1} e_1^{(n_1-1)} + \dots + k_{11} e_1 \\ e_2^{(n_2)} + k_{2n_2} e_2^{(n_2-1)} + \dots + k_{21} e_2 \\ \vdots \\ e_p^{(n_p)} + k_{pn_p} e_p^{(n_p-1)} + \dots + k_{p1} e_p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.20)$$

Les coefficients  $k_{ij}$  sont choisis de façon à ce que le polynôme de l'équation (3.20) Soit Hurwitzien (racines à partie réelle négative), qui prouve la convergence de l'erreur de suivi de trajectoire  $e(t)$  vers zéro quand  $t \rightarrow \infty$ , c'est-à-dire :

$$\lim_{t \rightarrow \infty} \|e\|^2 = 0 \quad (3.21)$$

Puisque les fonctions non linéaires sont incertaines, l'implantation de loi de commande idéale (3.19) est impossible. Dans cette situation, notre but est d'approcher cette commande idéale en utilisant les systèmes neuronaux.

### 3.2.2. Conception de la commande adaptative directe neuronale :

En se basant sur le fait que le système neuronal, de la forme (1.1), est un approximateur universel, chaque composante de la loi de commande idéale (3.19)  $u^* = [u_1^*, \dots, u_p^*]^T$  peut être approximée comme suit :

$$u_i^*(z) = W_i^{*T} S_i(z) + \varepsilon_i(z) \quad i = 1, \dots, p \quad (3.22)$$

avec  $z \in \Omega_z$  est le vecteur d'entrée, étant  $\Omega_z$  un ensemble compact,  $\varepsilon_i(z)$  est une erreur d'approximation considérée arbitrairement petite et bornée à partir de la théorie de l'approximation, et  $W_i^*$  sont les paramètres optimaux de  $W_i$  c'est-à-dire, le vecteur des paramètres ajustés  $W_i$  minimisant l'erreur  $|\varepsilon_i(z)|$ . Ces paramètres optimaux satisfont

$$W_i^* = \underset{W_i \in \Omega_w}{\operatorname{argmin}} \{ \sup_{z \in \Omega_z} |u_i^*(z) - \hat{u}_i(z)| \} \quad (3.23)$$

Par conséquent, le contrôleur idéal  $u^*$  peut être défini comme suit :

$$u^*(z) = W^{*T} \underline{S}(z) + \varepsilon(z) \quad (3.24)$$

où  $W^* = [W_1^{*T}, \dots, W_p^{*T}]^T$ ,  $\underline{S}(z) = \operatorname{diag}[S_1(z), \dots, S_p(z)]^T$  et

$$\varepsilon(z) = [\varepsilon_1(z), \dots, \varepsilon_p(z)]^T$$

où  $\tilde{W} = W^* - W$  est le vecteur d'erreur des paramètres estimés.

Cependant le vecteur des paramètres optimaux  $W^*$  dans (3.23) est inconnu, alors il doit être ajusté par une loi d'adaptation convenable. Soit  $W$  l'estimation du vecteur des paramètres optimaux  $W^*$  et  $u$  la loi de commande qui est l'approximation neuronale du contrôleur idéal (3.19).

Dans ces conditions, la loi de commande proposée pour le system (3.2) est donnée par :

$$\hat{u}(z) = W^T \underline{S}(z) + u_s \quad (3.25)$$

où  $u_s = [u_{s1}, \dots, u_{sp}]^T$  est un vecteur de signaux supplémentaires, capable de compenser ou d'atténuer l'influence de l'erreur d'approximation neuronale sur les performances de commande et ainsi d'assurer la robustesse de la structure de commande.

D'où, on peut écrire

$$u^*(z) - \hat{u}(z) = \tilde{W} \underline{S}(z) + \varepsilon(z) - u_s \quad (3.26)$$

avec la loi de commande supplémentaire donnée comme suit :

$$u_s = \psi \tanh(e^T P \underline{B} / \epsilon) \quad (3.27)$$

avec  $\tanh(\cdot)$  est la fonction Hyperbolique,  $\psi = [\psi_1, \dots, \psi_p]^T$ , et  $\epsilon$  est une constante positive.

**Hypothèse 3.7:** L'erreur d'approximation  $\epsilon(\underline{z})$  est bornée, c'est-à-dire

$$\|\epsilon(\underline{z})\| \leq \bar{\epsilon}$$

où  $\bar{\epsilon}$  est une constante positive inconnue.

**Hypothèse 3.8 :** Les paramètres du réseau de neurones sont bornés par les ensembles de contraintes  $\Omega_w$  telles que :

$$\Omega_w = \{W: \|W\| \leq M\}$$

Les bornes employées dans l'hypothèse 3.8 sont introduites pour assurer la bornitude des sorties des réseaux de neurones.

A cet effet, les paramètres du contrôleur neuronal sont ajustés en utilisant l'algorithme de projection suivant :

$$\dot{W} = \Gamma e^T P \underline{B} \underline{S}(\underline{z}) - I_d \Gamma e^T P \underline{B} W W^T \underline{S}(\underline{z}) \left( \frac{1 + \|W\|}{M} \right)^2 \quad (3.28)$$

où

$$I_d = \begin{cases} 0, & \text{si } \|W\| < M \text{ ou } (\|W\| = M \text{ et } e^T P \underline{B} W^T \underline{S}(\underline{z}) \leq 0) \\ 1, & \text{si } \|W\| = M \text{ et } e^T P \underline{B} W^T \underline{S}(\underline{z}) > 0 \end{cases} \quad (3.29)$$

et  $\Gamma > 0$  sont les paramètres de conception, et  $P = P^T > 0$  est la solution, pour une matrice donnée  $Q = Q^T > 0$ , de l'équation de Lyapunov (3.7).

Alors, l'introduction de la loi de commande (3.25) dans (3.18) produit :

$$\dot{e} = \underline{A}e + \underline{B}[m - F(\underline{z}) - G(\underline{z})(W^T \underline{S}(\underline{z}) + u_s)] \quad (3.30)$$

à partir de (3.19), (3.30) peut être réécrite comme suit :

$$\dot{e} = (\underline{A} - \underline{B}k)e + \underline{B}G(\underline{z}) \left[ W^{*T} \underline{S}(\underline{z}) + \epsilon(\underline{z}) - W^T \underline{S}(\underline{z}) - u_s - G^{-1}(\underline{z}) \frac{e^T P e}{e^T P \underline{B}} \beta(x) \right] \quad (3.31)$$

et

$$\dot{e} = \underline{A}_k e + \underline{B}G(\underline{z}) \left[ \tilde{W} \underline{S}(\underline{z}) + \epsilon(\underline{z}) - u_s - G^{-1}(\underline{z}) \frac{e^T P e}{e^T P \underline{B}} \beta(x) \right] \quad (3.32)$$

avec  $\underline{A}_k = (\underline{A} - \underline{B}k)$

Le vecteur de gain de retour d'état  $\underline{k}$  est choisie tel que la matrice  $\underline{A}_k$  est Hurwitz (toutes ses valeurs propres à parties réelles négatives), par conséquent le système bouclé sera exponentiellement stable. On lui associe une équation de Lyapunov, ayant pour solution unique une matrice symétrique définie positive, donnée par



$$\underline{A}_k^T P + P \underline{A}_k = -Q \quad (3.33)$$

### 3.2.3. Analyse de stabilité :

Dans cette section nous allons démontrer que la commande adaptative neuronale directe du système non linéaire multi variable (3.2) proposée est stable, la loi de commande (3.25) et (3.27) avec sa loi d'adaptation (3.25), si les hypothèses 3.1 jusqu'au 3.8 sont satisfaites, alors tous les signaux de la boucle fermée sont bornés, et le schéma global garantit que :

(i) l'erreur de poursuite  $e(t)$  converge à l'ensemble compact  $\Omega_e$  défini par :

$$\Omega_e = \left\{ e : \|e\| \geq \sqrt{\frac{w_2}{w_1}} \right\}$$

**Démonstration:** On considère la fonction candidate de Lyapunov suivante :

$$V = \frac{1}{2} (e^T G^{-1}(\underline{z}) P e + \tilde{W}^T \Gamma^{-1} \tilde{W}) \quad (3.34)$$

En utilisant (3.32) et (3.33) et le fait que  $\dot{\tilde{W}} = -\dot{W}$ , la dérivée par rapport au temps de la fonction  $V$  peut être réécrite comme suit:

$$\dot{V} = \dot{V} = \frac{1}{2} \dot{e}^T G^{-1}(\underline{z}) P e + \frac{1}{2} e^T \dot{G}^{-1}(\underline{z}) P e + \frac{1}{2} e^T G^{-1}(\underline{z}) P \dot{e} + \frac{1}{2} e^T \dot{G}^{-1}(\underline{z}) P e - \tilde{W}^T \Gamma^{-1} \dot{W} \quad (3.35)$$

$$\begin{aligned} \dot{V} = & -\frac{1}{2} e^T G^{-1}(\underline{z}) Q e + e^T P \underline{B} G^{-1}(\underline{z}) \left[ G(\underline{z}) \left( \varepsilon(\underline{z}) - u_s - G^{-1}(\underline{z}) \frac{e^T P e}{e^T P \underline{B}} \beta(x) \right) \right] \\ & + \frac{1}{2} e^T \dot{G}^{-1}(\underline{z}) P e + \tilde{W}^T \Gamma^{-1} (\Gamma e^T P \underline{B} \underline{S}(\underline{z}) - \dot{W}) \end{aligned} \quad (3.36)$$

En substituant (3.28) et (3.29) dans (3.36), nous avons :

$$\begin{aligned} \dot{V} \leq & -\frac{1}{2} e^T G^{-1}(\underline{z}) Q e + |e^T P \underline{B}| \bar{\varepsilon} - e^T P \underline{B} \psi \tanh(e^T P \underline{B} / \epsilon) - G^{-1}(\underline{z}) e^T P e \beta(x) \\ & + \frac{1}{2} e^T \dot{G}^{-1}(\underline{z}) P e + \\ & \tilde{W}^T \Gamma^{-1} \left( \Gamma e^T P \underline{B} \underline{S}(\underline{z}) - \Gamma e^T P \underline{B} \underline{S}(\underline{z}) + I_d \Gamma e^T P \underline{B} W W^T \underline{S}(\underline{z}) \left( \frac{1 + \|W\|}{M} \right)^2 \right) \end{aligned} \quad (3.37)$$

En supposant que le paramètre de conception  $\psi$  est choisi grand tels que  $\psi \geq \bar{\varepsilon}$  et en utilisant l'inégalité suivante [Polycarpous-95],  $-x \tanh(x/\epsilon) + |x| \leq \kappa \epsilon$  avec  $\kappa = 0.2785$ . D'après l'hypothèse 3.5 et 3.8 on obtient :

$$\begin{aligned} \dot{V} \leq & -\frac{1}{2} e^T G^{-1}(\underline{z}) Q e + \bar{\varepsilon} \kappa \varepsilon - \|e\|^2 \lambda_{\min}(P) \alpha \beta(x) + \frac{1}{2} \|e\|^2 \|\dot{G}^{-1}(\underline{z})\| \lambda_{\min}(P) \\ & + \tilde{W}^T \Gamma^{-1} \left( I_d \Gamma e^T P \underline{B} W W^T \underline{S}(\underline{z}) \left( \frac{1 + \|W\|}{M} \right)^2 \right) \end{aligned} \quad (3.38)$$

où  $I_d = 1$  si la seconde de (3.29) est vrai,  $I_d = 1$  si la première ligne de (3.33) est vrai.

Nous prouvons maintenant que  $I_d \Gamma e^T P \underline{B} W W^T \underline{S}(\underline{z}) \left( \frac{1 + \|W\|}{M} \right)^2 \tilde{W}^T W \leq 0$

Si  $I_d = 0$ , la conclusion est évidente.

Si  $I_d = 1$ , alors  $\|W\| = M$  et  $e^T P \underline{B} W W^T \underline{S}(\underline{z}) > 0$ . D'autre part, nous avons :

$$\tilde{W}^T W = \frac{1}{2} \|W^*\|^2 - \frac{1}{2} \|W\|^2 - \frac{1}{2} \|W - \tilde{W}\|^2 \leq 0 \quad (3.39)$$

Puisque  $\frac{1}{2} \|W^*\|^2 \leq M^2$  et  $\frac{1}{2} \|W\|^2 = M^2$  et  $\frac{1}{2} \|W - \tilde{W}\|^2 \geq 0$  alors  $\tilde{W}^T W \leq 0$

et nous avons

$$I_d \Gamma e^T P \underline{B} W W^T \underline{S}(\underline{z}) \left( \frac{1 + \|W\|}{M} \right)^2 \tilde{W}^T W \leq 0 \quad (3.40)$$

A partir de (3.44), et (3.42) nous avons

$$\dot{V} \leq -\frac{1}{2} \lambda_{\min}(G^{-1}(\underline{z})Q) \|e\|^2 + \bar{\varepsilon} \kappa \varepsilon \quad (3.41)$$

où  $\lambda_{\min}(G^{-1}(\underline{z})Q)$  est la valeur propre minimale de la matrice  $G^{-1}(\underline{z})Q$ .

Alors

$$\dot{V} \leq -\underline{w}_1 \|\tilde{\xi}\|^2 + \underline{w}_2 \quad (3.42)$$

avec  $\underline{w}_1 = \frac{1}{2} \lambda_{\min}(G^{-1}(\underline{z})Q)$  et  $\underline{w}_2 = \bar{\varepsilon} \kappa \varepsilon$ .

où  $\lambda_{\min}(G^{-1}(\underline{z})Q)$  est la valeur propre minimale de la matrice  $G^{-1}(\underline{z})Q$ .

De (3.42), on peut conclure que  $\dot{V} \leq 0$  si  $\|e\| \geq \sqrt{\frac{\underline{w}_2}{\underline{w}_1}}$ . Par conséquent, les erreurs de poursuite convergent vers une région bornée donnée par:

$$\Omega_e = \left\{ e : \|e\| \leq \sqrt{\frac{\underline{w}_2}{\underline{w}_1}} \right\} \quad (3.43)$$

ce qui implique la bornitude de tous les signaux dans le système bouclé, c'est-à-dire l'erreur d'approximation minimal peut être construite arbitrairement petite par l'utilisateur de la propriété d'approximation neuronal, alors la quantité  $\underline{w}_2$  est borné, ce qui implique que l'ensemble  $\Omega_e$  dans (3.43) est borné. Maintenant  $\dot{V}$  deviendra négatif tant que  $e(t)$  est

en dehors de l'ensemble  $\Omega_e$ , selon la théorie de la stabilité de Lyapunov, nous concluons que l'erreur  $e(t)$  est bornée et va converger vers  $\Omega_e$ .

Alors et la fonction candidate  $V$  est bornée. Par conséquent, on peut démontrer que les erreurs  $e(t) \in L_2$  sont carrés intégrables et que  $\dot{e}(t) \in L_\infty$  sont bornées, d'où, par application du lemme de Barbalat [Slotine-91], on déduit la convergence vers zéro des erreurs de poursuite  $\lim_{t \rightarrow \infty} \|e(t)\| = 0$  de tous les systèmes.

### 3.3. Résultats de simulation :

Études de simulation sont réalisées pour démontrer l'efficacité de la commande proposée. Deux problèmes de commande sont considérés. La première concerne un système académique MIMO non-affine, tandis que la seconde concerne un modèle dynamique de manipulateur à deux degrés de liberté.

#### 3.3.1. Exemple 1 :

Le problème de la commande d'un système académique non-linéaire non affine MIMO est considéré là pour montrer l'efficacité de ce contrôleur proposé. Ses équations dynamiques sont donnés par :

$$\begin{cases} \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = x_{11}^2 + x_{12}^2 + 0.15u_1^3 + (2 + \cos(x_{11}))u_1 - u_2 \\ \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = x_{22}^2 + x_{11} + x_{12}^2 - 0.5u_1 + (1 + x_{21}^2)u_2^3 + (2 + \sin(x_{21}))u_2 \\ y_1 = x_{11} \\ y_2 = x_{21} \end{cases} \quad (3.44)$$

où  $x = [x_{11}, x_{12}, x_{21}, x_{22}]^T$  est le vecteur d'état du système,  $u_1$  et  $u_2$  sont les entrées de commande,  $y_1$  et  $y_2$  sont les sorties du système.

L'objectif de la commande est de forcer les sorties du système  $y_1$  et  $y_2$  suivre les trajectoires désirées  $y_{d1}(t) = \sin(t)$  et  $y_{d2}(t) = \sin(t)$ , respectivement.

Deux systèmes neuronaux de type réseaux de neurone d'ordre supérieur de la forme (1.1) sont utilisés pour générer les commandes  $u_1$  et  $u_2$ . Chaque réseau de neurone possède le

vecteur d'entrée  $z = [z_1^T, z_2^T, z_3^T]^T$   $\underline{z} = [\xi^T, y_d^T, \underline{\varphi}]^T$  avec  $z_1^T = [x_{11}, x_{12}, x_{21}, x_{22}]$ ,  $z_2^T = [y_{d1}, \dot{y}_{d1}, y_{d2}, \dot{y}_{d2}]$ ,  $z_3^T = [\ddot{y}_{d1} + k_{12}\dot{e}_1 + k_{11}e_1, \ddot{y}_{d2} + k_{21}\dot{e}_2 + k_{22}e_2]$

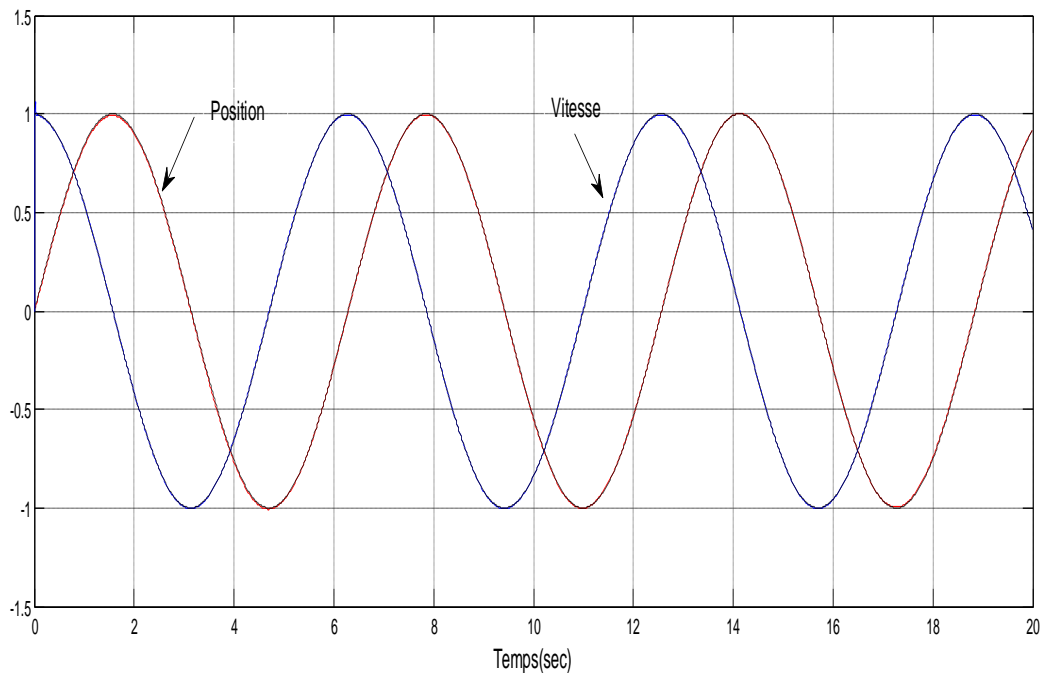
Les paramètres de synthèses et les conditions initiales sont donnés dans le tableau (3.1) suivant :

Les paramètres de synthèses	Les valeurs numériques	Les conditions initiales	Les valeurs numériques
$\Gamma$	0.01	$W(0)$	0
$M$	16	$x_{11}(0)$	0.001
$\epsilon$	30	$x_{12}(0)$	0
$k$	$\begin{bmatrix} 1 & 0 & -2 & 0; \\ 0 & -1 & 0 & -2 \end{bmatrix}$	$x_{21}(0)$	0.001
$Q$	$Diag[5.5 \ 5.5 \ 5 \ 5]$	$x_{22}(0)$	0
$P$	$\begin{bmatrix} 8.125 & 0 & 2.75 & 0; \\ 0 & 8.125 & 0 & 2.75; \\ 2.75 & 0 & 2.625 & 0; \\ 0 & 2.75 & 0 & 2.625 \end{bmatrix}$		

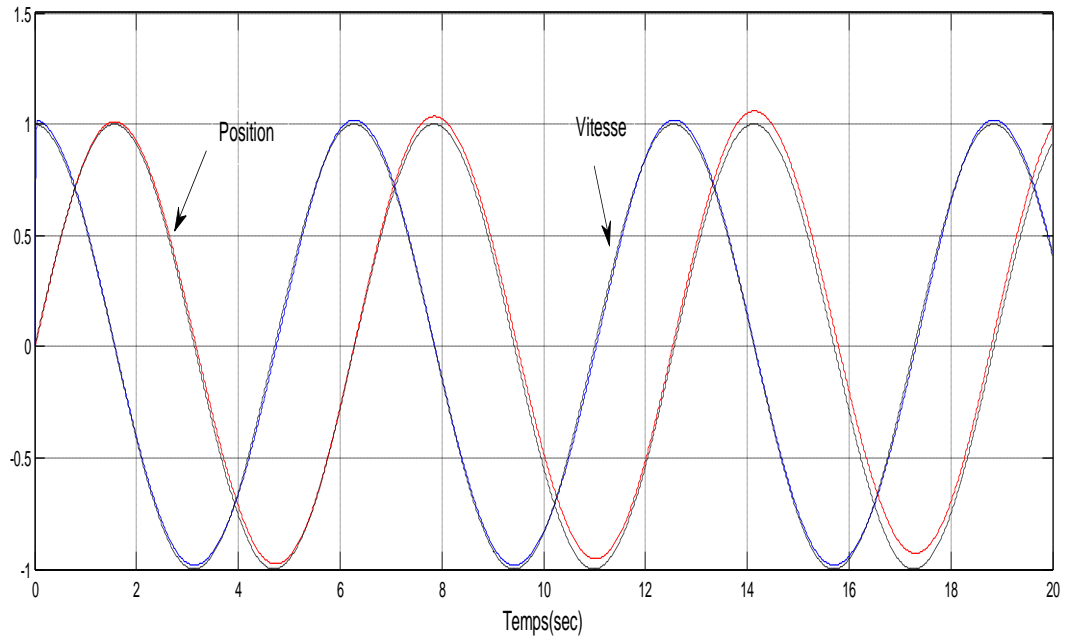
**Tab.3.1** : les paramètres de synthèses & les conditions initiales.

Les résultats de simulation sont montrés sur les figures (3.1)-(3.4), la sortie  $y_1$  et sa référence  $y_{d1}$  ainsi que la sortie  $y_2$  et sa référence  $y_{d2}$  sont représentés sur les figures (3.1) et (3.2) respectivement. L'allure des signaux de commande  $u_1$  et  $u_2$  est donnée par les figures (3.3) et (3.4).

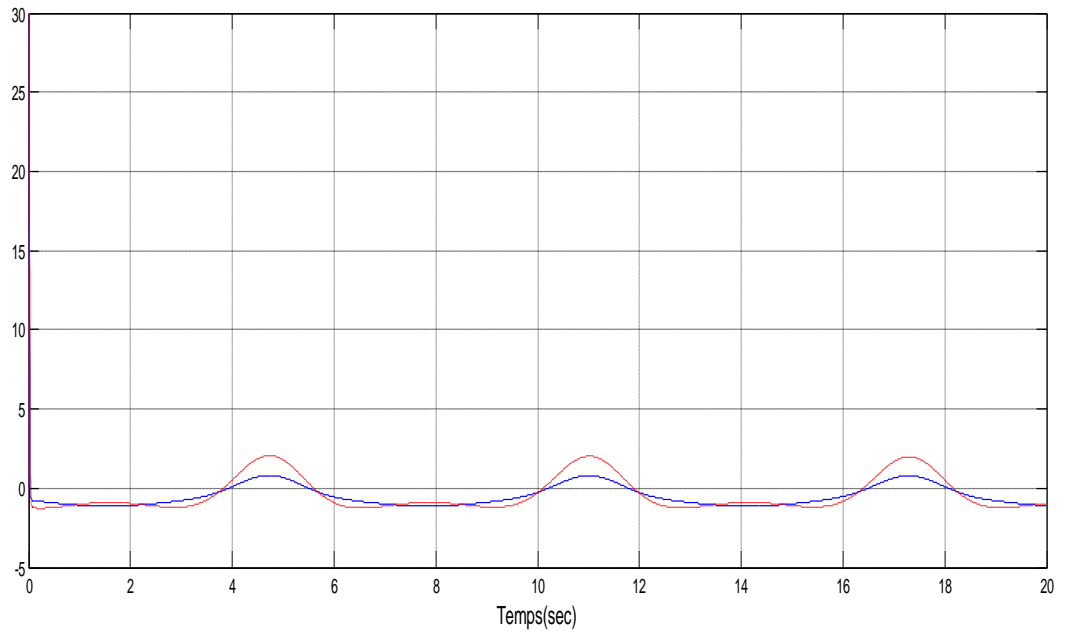
A partir des résultats de simulations obtenues, nous pouvons noter des bonnes performances de poursuite réalisées grâce aux signaux de commande  $u_1$  et  $u_2$  qui sont bornés.



**Fig. 3.1** : Courbes de poursuite de sous système1 ; sorties (traits continus), consignes (traits discontinus).



**Fig. 3. 2:** Courbes de poursuite de sous système 2 ; sorties (traits continus), consignes (traits discontinus).



**Fig. 3.3 :** Signaux de commande;  $u_1$  (trait continu),  $u_2$  (trait discontinu).

### 3.3.2. Exemple 2 :

Dans cette section, nous présentons les résultats de simulation montrant les performances de suivi de la commande proposée appliqué à un deux-lien rigide robot manipulateur. La dynamique de ce système MIMO sont donnés par [Slotine-91] :

$$\begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{pmatrix} = \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix}^{-1} \left\{ \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} - \begin{pmatrix} -h\dot{q}_2 & -h(\dot{q}_1 + \dot{q}_2) \\ h\dot{q}_1 & 0 \end{pmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix} \right\} \quad (3.45)$$

avec

$$H_{11} = a_1 + 2a_3 \cos q_2 + 2a_4 \sin q_2$$

$$H_{12} = H_{21} = a_2 + a_3 \cos q_2 + a_4 \sin q_2$$

$$H_{22} = a_2$$

$$h = a_3 \sin q_2 - a_4 \cos q_2$$

et

$$a_1 = I_1 + m_1 l_{c_1}^2 + I_e + m_e l_{c_e}^2 + m_e l_1^2$$

$$a_2 = I_e + m_e l_{c_e}^2$$

$$a_3 = m_e l_1 l_{c_e} \cos(\delta_e)$$

$$a_4 = m_e l_1 l_{c_e} \sin(\delta_e)$$

où  $u_1$  et  $u_2$  sont les couples de commande appliqués aux articulations, et les sorties sont les positions articulaires  $q_1$  et  $q_2$ . Dans cette simulation, le tableau (3.2) suivant contient les valeurs des paramètres utilisées

Les paramètres de	Les valeurs numériques	Les paramètres de	Les valeurs numériques
$m_1$	1kg	$l_{c_e}$	0.6m
$m_e$	2kg	$I_1$	0.12kg m <sup>2</sup>
$l_1$	1m	$I_e$	0.25 kg m <sup>2</sup>
$l_{c_1}$	0.5m	$\delta_e$	30°

**Tab.3.2 :** Paramètres du modèle

Posons  $y = [q_1, q_2]^T$ ,  $u = [u_1, u_2]^T$ ,  $x = [q_1, \dot{q}_1, q_2, \dot{q}_2]^T$  et

$$F(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix} = -H^{-1} \begin{pmatrix} -h\dot{q}_2 & -h(\dot{q}_1 + \dot{q}_2) \\ h\dot{q}_1 & 0 \end{pmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix}$$

$$G(x) = \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} = H^{-1} = \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix}^{-1}$$

Il est bien connu que la matrice  $H$  est définie positive [Slotine-91] et, de ce fait, le modèle du robot satisfait l'hypothèse 3.6.

L'objectif de commande est de forcer les sorties du système  $q_1$  et  $q_2$  de suivre respectivement les trajectoires sinusoïdales  $y_{d_1}(t) = \sin(t)$  et  $y_{d_2}(t) = \sin(t)$ .

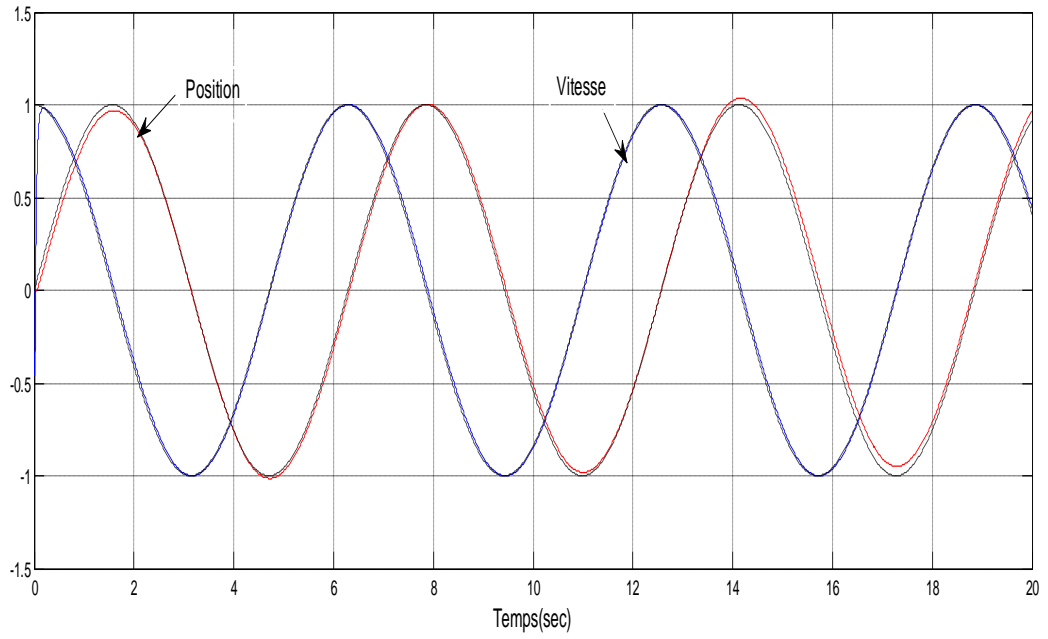
Deux systèmes neuronaux de type réseaux de neurone d'ordre supérieure de la forme (1.1) sont utilisées pour générer les commandes  $u_1$  et  $u_2$ . Chaque réseau de neurone possède le vecteur d'entrée  $z = [z_1^T, z_2^T, z_3^T]^T$  avec  $z_1^T = [x_{11}, x_{12}, x_{21}, x_{22}]$ ,  $z_2^T = [y_{d_1}, \dot{y}_{d_1}, y_{d_2}, \dot{y}_{d_2}]$ ,  $z_3^T = [\ddot{y}_{d_1} + k_{12}\dot{e}_1 + k_{11}e_1, \ddot{y}_{d_2} + k_{21}\dot{e}_2 + k_{22}e_2]$ .

Les paramètres de synthèses et les conditions initiales sont donnés dans le tableau (3.3) suivant :

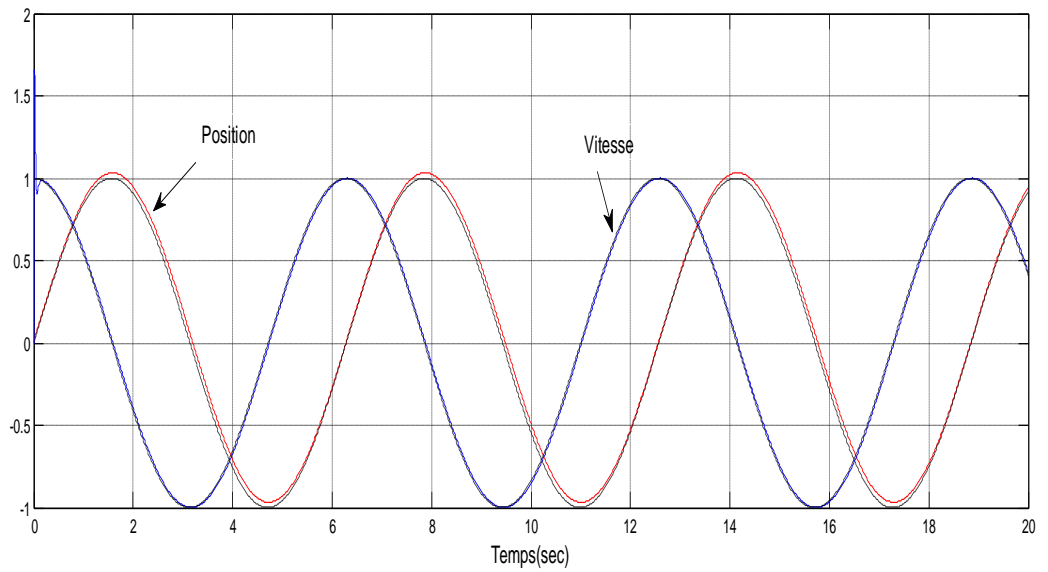
Les paramètres de synthèses	Les valeurs numériques	Les conditions initiales	Les valeurs numériques
$\Gamma$	0.001	$W(0)$	0
$M$	16	$x_{11}(0)$	0.00005
$\epsilon$	100	$x_{12}(0)$	0
$k$	$\begin{bmatrix} 1 & 0 & -2 & 0; \\ 0 & -1 & 0 & -2 \end{bmatrix}$	$x_{21}(0)$	0.00005
$Q$	$Diag[5.5 \ 5.5 \ 5 \ 5]$	$x_{22}(0)$	0
$P$	$\begin{bmatrix} 8.125 & 0 & 2.75 & 0; \\ 0 & 8.125 & 0 & 2.75; \\ 2.75 & 0 & 2.625 & 0; \\ 0 & 2.75 & 0 & 2.625 \end{bmatrix}$		

**Tab.3.3** : les paramètres de synthèses & les conditions initiales.

Les Figures 3.4 à 3.6 présentent les résultats de simulation. La Figure 3.4 donne le suivi de trajectoire pour la première liaison, la Figure 3.5 celui de la deuxième, et la Figure 3.6 montre les signaux de commande. On peut constater que les trajectoires réelles convergent vers les trajectoires désirées, et que les signaux de commande  $u_1$  et  $u_2$  qui sont bornés.

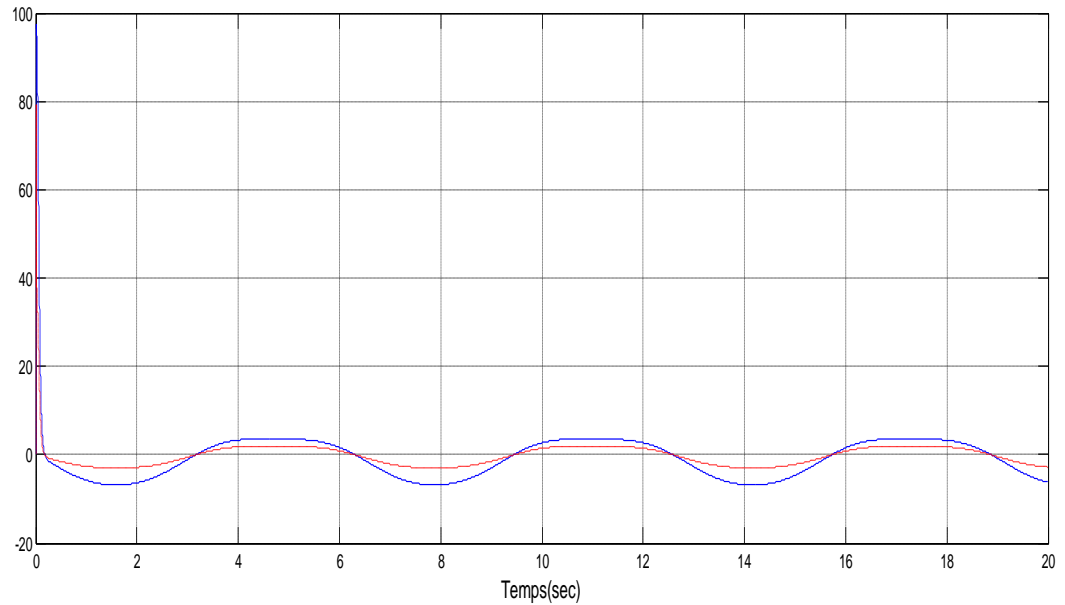


**Fig. 3.4 :** Courbes de poursuite pour la liaison 1 ; sorties (traits continus), consignes (traits discontinus).



**Fig. 3.5 :** Courbes de poursuite pour la liaison 2 ; sorties (traits continus), consignes (traits discontinus).





**Fig. 3.6 :** Signaux de commande;  $u_1$  (trait continu),  $u_2$  (trait discontinu).

### 3.4. Conclusion :

Dans ce chapitre, nous avons proposé et développé une commande adaptative neuronale directe pour une classe de systèmes non linéaires MIMO non affines en la commande. En premier lieu Nous avons utilise le théorème de la fonction implicite globale pour démontrer l'existence d'un contrôleur idéal inconnu qui est capable d'accomplir l'objective de la commande, autour de ce dernier nous avons transformée le modèle non affine en un modèle affine par un développement en série de Taylor en deuxième lieu. Puis nous avons synthétisé la loi de commande par feedback linearization. En fin, pour des raisons des incertitudes sur le modèle liées aux problèmes de transformations et modélisation, un réseau de neurone est utilisé pour approximer la loi de commande idéale. Les lois d'adaptation ont déduites à travers l'analyse de la stabilité, au sens de Lyapunov, du système en boucle fermée. Les résultats de simulation obtenus pour les exemples proposés montrent des bonnes performances de poursuite et de robustesse, et mettent en évidence la capacité et l'efficacité du contrôleur neuronale développé à prendre en charge une certaine classe de systèmes non linéaires MIMO non affines en la commande.

---

## *Conclusion générale*

---

D'une manière générale, l'analyse et la commande non linéaires sont des problèmes difficiles et la majorité des approches de la commande non linéaire exigent la disponibilité d'un modèle mathématique. Cette commande doit être robuste dans le sens où elle devra assurer une faible sensibilité aux incertitudes sur les paramètres, à leurs variations et aux perturbations.

L'objectif principal de ce mémoire est de développer des lois de commande adaptatives neuronales stables et robustes pour réaliser la poursuite des certaines classes de systèmes non linéaires SISO et MIMO non affines en la commande.

Dans la première partie de ce travail, nous avons commencé par présenter l'ensemble des définitions et des outils nécessaires à la mise au point de la stratégie adoptée pour le développement des lois de commande garantissant les performances souhaitées. En effet, un rappel sur les principes de réseaux de neurones d'ordre supérieur HONN, et l'utilisation des systèmes neuronaux comme éléments de base pour la modélisation et la commande des systèmes non linéaires est présenté.

Dans la deuxième partie du travail, nous avons démontré que la commande du système non linéaire non affine est équivalente à la commande du système non linéaire affine autour d'un contrôleur idéal inconnu qui nous avons démonté leur existence par le théorème de la fonction implicite. A base de ce résultat nous avons proposé et développé des schémas de commande adaptative directe et indirecte pour une classe de systèmes non linéaires monovariables. Dans le schéma de commande adaptative directe neuronale, le but était l'approximation d'une loi de commande stabilisatrice inconnue. En revanche le schéma de commande adaptative indirecte exploite deux systèmes adaptatifs neuronaux pour approximer la dynamique du système affine, et aussi évite le problème de singularité de la loi de commande par l'utilisation d'un algorithme de projection.

Dans la dernière partie de ce travail, nous avons développé un contrôleur adaptatif direct neuronal pour une classe de systèmes non linéaires multivariables non affines en la commande. Pour ce faire, nous avons exploité en premier lieu le théorème de la fonction implicite globale pour montrer l'existence du contrôleur idéal inconnu. Ensuite, nous avons

utilisé le développement en série de Taylor autour du contrôleur idéal pour transformer le modèle non affine en un modèle affine. En fin, pour des raisons des incertitudes sur le modèle liées aux problèmes de transformations et modélisation, un réseau de neurone est utilisé pour approximer la loi de commande idéale.

Les lois d'adaptation des paramètres ajustables sont déduites à partir de l'étude de stabilité au sens de Lyapunov des systèmes en boucle fermée. La robustesse étant assurée par l'ajout d'un terme de commande type mode glissant. La simulation, des systèmes non linéaires SISO et MIMO non affines en la commande, a permis de mettre en évidence l'apport des approches proposées.

---

## ***Bibliographie***

---

**[Dai-14]**

Shi-Lu Dai, Cong Wang, and Min Wang, "Dynamic Learning From Adaptive Neural Network Control of a Class of Nonaffine Nonlinear Systems", *IEEE Transactions on Neural Networks and learning systems*, vol. 25, no. 1, pp.111-123, january 2014.

**[Doudou-13]**

Doudou, S. "Contribution à la Commande Moderne des Systèmes Non Linéaires Multivaraibles Non affines", *Thèse de Doctorat, Université de Setif, Alger*, 2013.

**[Funahashi-89]**

Funahashi, K. L., "On the approximate realization of continuous mapping by neural networks", *Neural Networks*, no.2, pp.183-192, 1989.

**[Ge-99]**

S. S. Ge, C. C. Hang, and T. Zhang, "Adaptive Neural Network Control of Nonlinear Systems by State and Output Feedback," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 29, no. 6, pp. 818–828, Dec. 1999.

**[Ge-04]**

S. S. Ge, and C. Wang, "Adaptive Neural Control of Uncertain MIMO Nonlinear Systems," *IEEE Trans. Neural Netw*, vol. 15, no. 3, pp.674–692, May 2004.

**[Ge-03]**

S. S. Ge, and Zhang J., "Neural-Network Control of Nonaffine Nonlinear System With Zero Dynamics by State and Output Feedback" *IEEE Trans. Neural Networks*, Vol. 14, No. 4, pp. 900-918, July 2003.

**[Ge-01]**

S. S. Ge, C. C. Hang, T. H. Lee, and T. Zhang, *Stable Adaptive Neural Network Control*. Norwell, MA: Kluwer, 2001.

**[Ge-00]**

S. S. Ge, T. H. Lee, and J. Wang, "Adaptive control of non-affine nonlinear systems using neural networks," in *Proceedings of the 15th IEEE International Symposium on Intelligent Control (ISIC 2000)*, pp. 13–18, Rio, Patras, Greece, 2000.

**[Ge-98]**

S. S. Ge, Hang C. C. and Zhang T, "Nonlinear Adaptive Control Using Neural Networks and its Application to CSTR Systems", *Journal of Process Control* Vol. 9, pp. 313-323, 1998.

**[Hornik-89]**

Hornik, K. Stinchcombe M. & White H., "Multilayer feedforward networks are universal approximators", *IEEE Trans. Neural Networks*, , 2, 1083-1112, 1989.

**[Isidori-89]**

Isidori A ., "Nonlinear Control Systems" , 2<sup>nd</sup>, *Springer-Verleg* :Berlin,1989.

**[Ioannou-96]**

Ioannou P A and Sun J, "Adaptive control", Englewood Cliffs, NJ: Prentice-hall, 1996.

**[Karimi-09]**

R. Karimi, M. B. Menhaj, M. Karimi-Ghartemani, and I. Saboori, "Decentralized Adaptive Control of Large-scale Affine and Non-affine Nonlinear Systems", *IEEE Transaction on Instrumentation and Measurement*, Vol.58, No.8, pp. 2459-2467, 2009.

**[Kosmatopoulos-95]**

E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, and P.A. Ioannou, "High-Order Neural Network Structures for Identification of Dynamical Systems," *IEEE Trans. Neural Networks*, vol. 6, pp. 422–431, Mar. 1995.

**[Labiou-05]**

Labiou, S. "Contribution à la Commande Adaptative Floue des Systèmes Non Linéaires" *Thèse préparée au sein du Laboratoire de Commande des Processus de l'ENP2005.*

**[Liu-07]**

Liu, Y.J., Wang, W.: ‘Adaptive Fuzzy Control for a Class of Uncertain Non-affine Nonlinear Systems’, *Inf. Sci.* **177**, pp. 3901– 3917, 2007.

**[Park-04]**

Park, J. H., and Kim, S.H.,” Direct Adaptive Output Feedback Fuzzy Controller for Non-affine Nonlinear System”, *Proc. of IEE, Control Theory Applications*, Vol. 151, No.1, pp. 65–72, 2004.

**[Polycarpous-95]**

M. M. Polycarpous and P. A. Ioannou,” A Robust Adaptive Nonlinear Control Design”, *Automatica*, Vol. 32, No. 3, pp. 423-427, 1995.

**[Slotine-91]**

J.E. Slotine and W. Li, *Applied nonlinear control*, Prentice Hall, Englewood, Cliffs, NJ, 1991.

**[Spooner-96]**

Spooner J.T., Passino K.M., ‘Stable Adaptive Control Using Fuzzy Systems and Neural Networks’, *IEEE Trans. on Fuzzy Systems*, Vol. 4, No. 3, pp. 339-359, August 1996.

**[Wang-93]**

L.-X. Wang, ‘Stable Adaptive Fuzzy Control of Nonlinear Systems ‘’, *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, pp. 146–155, 1993.

**[Wang-94]**

L.X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Prentice-Hall, Englewood Clifs, NJ, 1994.

**[Wang-07]**

Wang, W.Y., Chien, L.C., Li, I.H., and Su, S.F., ‘MIMO Robust Control via T-S Fuzzy Models for Non-affine Nonlinear Systems’, *Proc of IEEE Inter Conf. on Fuzzy Systems*, pp. 1–6, 2007.

**[Yesildirek-99]**

A. Yesildirek, F. L. Lewis, and S. Jagannathan , *Neural Network Control of Robot Manipulators and Nonlinear Systems*. New York: Taylor & Francis, 1999.

**[Yesildirek-95]**

Yesildirek, A. and Lewis Frank, F. L. "Feedback Linearization Using Neural Networks", *Automatica*, 31, 1659-1664, 1995.

**[Yesildirek-96]**

F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer Neural-Net Robot Controller With Guaranteed Tracking Performance," *IEEE Trans. Neural Networks*, vol. 7, pp. 388–398, Mar. 1996.

**[Zhang-06]**

Zhang, W., and Ge, S.S., "A Global Implicit Function Theorem without Initial Point and its Applications to Control of Non-affine Systems of High Dimensions", *Jour of Mathematical Analysis and Applications*, Vol. 313, pp. 251–261, 2006.

**[Zhang-98]**

T. Zhang, S.S. Ge, C.C. Hang, "Direct Adaptive Control of Non-affine Nonlinear System Using Multilayer Neural Networks", in: *Proc. ACC*, pp. 515–519, 1998.

