

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Bejaia
Faculté de technologie
Département de génie électrique

MÉMOIRE DE MASTER

En automatique

Présenté par

Mr. Belarbi Sidali

THÈME

Commande et identification d'un bras de robot à deux degrés de liberté par les réseaux de neurones

Soutenu le : 20/06/2013 devant le jury composé de :

M ^r MEBARKI Youssef	Université de Bejaia	Président
M ^r HEDDAR Hocine	Université de Bejaia	Examineur
M ^{me} MEZZAH Samia	Université de Bejaia	Promotrice

2012/2013

Remerciements

Je tiens d'abord à remercier très sincèrement Notre Dieu de nous avoir donné la force physique et morale pour accomplir ce travail.

Je tiens à remercier ma promotrice Mme. Mezzah Samia pour ces précieux conseils et orientations.

Au terme de ce travail, je suis bienheureux de pouvoir exprimer toute ma gratitude sincère envers les personnes qui m'ont aidées.

Je tiens à exprimer ma sincère gratitude aux membres du jury pour avoir pris le temps de lire et d'évaluer ce travail.

Liste des abréviations

ddl : Degrés de liberté

OT : Organe terminal (effecteur)

NN1 : Neural network 1 (réseau de neurones)

NN2 : Neural network 2 (réseau de neurones)

RNA : Les réseaux de neurones artificiels

MLP : Les réseaux de neurones multicouches

PID : Proportional integral derivative

LE : Lagrange-Euler

Liste des figures

Numéro	Titre	Page
Figure I.1	Schéma d'un neurone biologique	3
Figure I.2	Le modèle d'un neurone formel	4
Figure I.3	Différentes fonctions d'activation.	5
Figure I.4	Un réseau multi-couches comportant 2 neurones d'entrée, 4 neurones cachés et un neurone de sortie	6
Figure II.1	Manipulateurs à liaisons sérielles et parallèles	9
Figure II.2	Manipulateur plan à deux liaisons	10
Figure II.3	Exemple d'un bras manipulateur (KUKA KR6 ARC) à 7ddl	11
Figure III.1	schéma général d'Identification.	18
Figure III.2	Identification de processus par la méthode série-parallèle	19
Figure III.3	Identification de processus par la méthode parallèle	19
Figure III.4	Définition d'un processus	20
Figure III.5	Schéma d'apprentissage pour reproduire le comportement d'un processus	20
Figure III.6	Identification d'un système à l'aide de lignes à retard	21
Figure III.7	Différents schémas neuronaux d'identification d'un système	22
Figure IV.1	Pelican : bras de robot expérimental de CICESE, laboratoire de robotique	24
Figure IV.2	Diagramme du bras pelican (2 degrés de liberté)	25
Figure IV.3	La trajectoire angulaire désiré (q_{1d}) pour la 1 ^{er} liaisons du bras.	30
Figure IV.4	La trajectoire angulaire désiré (q_{2d}) pour la 2 ^{ème} liaisons du bras.	30
Figure IV.5	Schéma d'identification du modèle bras robotique	31
Figure IV.6	Schéma du réseau d'identification NN1	31
Figure IV.7	Contrôle adaptatif direct	32
Figure IV.8	Contrôle adaptatif indirect	33
Figure IV.9	Schéma de la commande adaptative neuronale directe du modèle Identifié du bras manipulateur.	33
Figure IV.10	schéma du réseau contrôleur NN2	34
Figure IV.11	Résultats d'identification (1 ^{er} sortie)	35
Figure IV.12	L'erreur d'identification pour la 1 ^{er} sortie	35
Figure IV.13	Résultats d'identification (2 ^{ème} sortie)	36
Figure IV.14	L'erreur d'identification pour la 2 ^{ème} sortie	36
Figure IV.15	Résultat de Commande sur la 1 ^{er} sortie	37
Figure IV.16	Résultat de Commande sur la 2 ^{er} sortie	37

Liste des tableaux

Numéro	Titre	Page
Tableau I	Paramètres du robot (pelican)	25

Table des matières

Introduction générale.....	1
Chapitre I : Réseaux de neurones artificiels	
I.1 Introduction.....	2
I.2 Le neurone biologique.....	2
I.2.1 Fonctionnement.....	3
I.3 Le neurone formel.....	4
I.3.1 La fonction d'activation	5
I.4 Les réseaux multicouches.....	5
I.5 Apprentissage des réseaux de neurones.....	6
I.5.1 L'apprentissage supervisé.....	7
I.5.2 Apprentissage non supervisé.....	7
I.5.3 Apprentissage par renforcement.....	7
I.6 L'algorithme de rétropropagation du gradient.....	7
Chapitre II : Les bras manipulateurs	
II.1 Introduction.....	9
II.2 Commande des Robots manipulateur.....	12
II.2.1 Introduction.....	12
II.2.2 Commande dans l'espace libre.....	13
II.2.2.1 Commande par articulation.....	13
II.2.2.2 Commande jacobéenne.....	13
II.2.2.3 Commande par découplage non linéaire.....	14
II.2.2.4 Commande passive.....	14
II.2.2.5 Commande adaptative.....	15
II.2.2.6 Commande prédictive.....	15
II.2.2.7 Commande robuste.....	15
II.2.2.8 Commande optimale.....	15
II.2.3 Commande dans l'espace contraint.....	16
Chapitre III : Identification des systèmes non linéaires par les réseaux de neurones	
III.1 Introduction.....	17
III.2 Identification des systèmes.....	17
III.3 Les structures d'identification.....	18
III.4 Identification par réseaux de neurones.....	19
Chapitre IV : Identification et commande d'un bras manipulateur à 2 ddl avec les réseaux de neurones artificiels	
IV.1 Introduction.....	23
IV.2 Le robot manipulateur.....	24
IV.3 Le modèle dynamique.....	26
IV.3.1 Introduction.....	26
IV.3.2 Formulation LE.....	26
IV.4 La trajectoire désirée.....	29
IV.5 Identification du système bras manipulateur.....	31
IV.6 La commande adaptative neuronale.....	32
IV.7 Application de la commande adaptative neuronale directe au modèle identifié du bras manipulateur.....	33
IV.8 les résultats de la simulation.....	35
IV.9 Conclusion.....	38
Conclusion général	39
Références bibliographiques	40

Introduction générale

La théorie de contrôle fournit des outils d'analyse et de synthèse parfaitement adaptée aux systèmes linéaires. Cependant en pratique, ces méthodes ne s'avèrent pas toujours applicables à cause de non linéarité des systèmes réels tel que les bras manipulateurs et parce qu'il n'est pas toujours possible de linéariser le système à commander, de ce fait des modèles non linéaires ont été considérés.

Le terme « identification » couvre à la fois une démarche et un ensemble de techniques dont l'objet est la détermination de modèle de comportement d'un procédé physique à partir de mesures caractéristiques de son fonctionnement dynamique. Ce modèle ne cherche qu'à reproduire " au mieux " un fonctionnement dynamique dans un contexte donné, sans se préoccuper de la signification physique éventuelle des paramètres dont il dépend.

L'utilisation des réseaux de neurones pour l'identification et la commande des systèmes non linéaires peut fournir des solutions pour ce problème, nous parlons alors d'identification neuronale et de commande neuronale.

Dans ce travail, nous nous sommes intéressés par l'identification et la commande d'un bras manipulateur de deux degrés de liberté avec les réseaux de neurones.

Le premier chapitre donne un aperçu général sur les réseaux de neurones, définitions et concepts de base.

Le deuxième chapitre parle brièvement sur les bras manipulateur et leur commandes.

Le chapitre trois donne un aperçu sur l'identification, d'abord en général ensuite par les réseaux de neurones.

Dans le quatrième chapitre nous présenterons l'identification d'un bras manipulateur à deux degrés de liberté et la commande neuronale adaptative direct de son modèle identifié et les résultats de la simulation sous l'environnement Matlab.

Chapitre I :

Réseaux

de neurones

artificiels

Chapitre I :

Réseaux de neurones artificiels

I.1 Introduction

Les réseaux de neurones sont des ensembles d'éléments de base appelés neurones. La philosophie derrière ces réseaux de neurones est d'imiter le cerveau humain, mais l'écart entre les réseaux de neurones et le cerveau est toujours grand, dû à la complexité de ce dernier. Cette complexité et une connaissance toujours améliorée du cerveau ont amené une multitude de solutions pour la conception des réseaux de neurones. L'absence de normalisation ajoute aussi à la difficulté de présenter clairement une théorie sur les réseaux de neurones. Nous allons présenter certains concepts qui sont normalement respectés. Les réseaux de neurones se modifient pour tenir compte de leur environnement. Cette capacité d'apprendre à partir d'exemples est d'un grand intérêt et elle s'apparente à celle du cerveau.

Les études des réseaux de neurones artificiels (RNA) datent depuis les années 1940. Grâce aux développements des recherches sur le cerveau et la disponibilité des outils de simulation, les chercheurs étudièrent des ensembles de neurones formels interconnectés. Ces réseaux, déjà développés à l'époque, permettaient d'effectuer quelques opérations logiques simples. Jusqu'aux années 1980, la recherche était freinée par la limitation théorique du perceptron. Peu après cette époque, Hopfield lança de nouveau en 1982 la recherche dans ce domaine après avoir montré l'analogie entre les RNA et les systèmes physiques. Après les années 1990, quelques travaux scientifiques ont vu le jour dans le domaine de la robotique, parmi ces applications, on trouve la commande des systèmes d'entraînement et des systèmes de positionnement de haute performance.[5]

I.2 Le neurone biologique

les réseaux neurones artificiels sont inspirés des neurones biologiques. Le cerveau se compose d'un grand nombre de neurones (approximativement 10^{11}) fortement connectés. Les neurones ont trois composants principal : Le corps cellulaire , les dendrites , l'axone. [3]

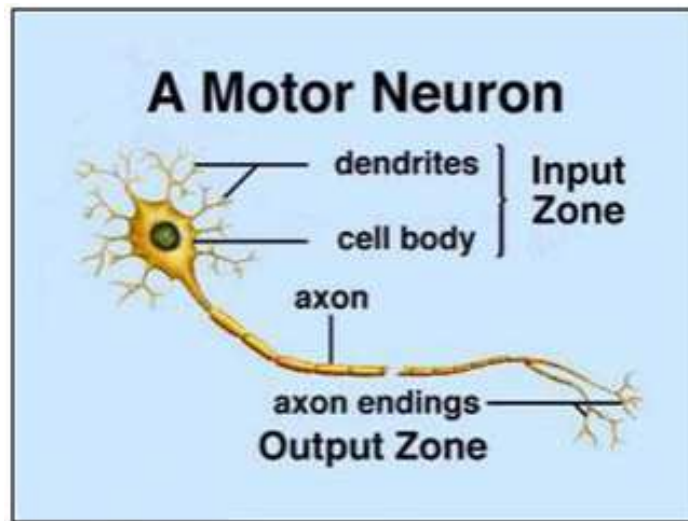


Figure I.1 Schéma d'un neurone biologique.

1-Le corps cellulaire :

Il contient le noyau du neurone ainsi que la machine biochimique nécessaire à la synthèse des enzymes. Ce corps cellulaire de forme sphérique ou pyramidale contient aussi les autres molécules essentielles à la vie de la cellule. Sa taille est de quelques microns de diamètre.[2]

2-Les dendrites :

Ce sont de fines extensions tubulaires qui se ramifient autour du neurone et forment une sorte de vaste arborescence. Les signaux envoyés au neurone sont captés par les dendrites. Leur taille est de quelques dizaines de microns de longueur.[2]

3-L'axone :

C'est le long de l'axone que les signaux partent du neurone. Contrairement aux dendrites qui se ramifient autour du neurone, l'axone est plus long et se ramifie à son extrémité ou il se connecte aux dendrites des autres neurones.[2]

I.2.1 Fonctionnement

Au point de vu fonctionnel, il faut considérer le neurone comme une entité polarisée, c'est-à-dire que l'information ne se transmet que dans un seul sens : des dendrites vers l'axone.

Le neurone va donc recevoir des informations, venant d'autres neurones, grâce à ses dendrites. Il va ensuite y avoir sommation, au niveau du corps cellulaire, de toutes ces

informations et via un potentiel d'action (un signal électrique) le résultat de l'analyse va transiter le long de l'axone jusqu'aux terminaisons synaptiques. A cet endroit, lors de l'arrivée du signal, des vésicules synaptiques vont venir fusionner avec la membrane cellulaire, ce qui va permettre la libération des neurotransmetteurs (médiateurs chimiques) dans la fente synaptique. Le signal électrique ne pouvant pas passer la synapse (dans le cas d'une synapse chimique), les neurotransmetteurs permettent donc le passage des informations, d'un neurone à un autre.

Les neurotransmetteurs excitent (neurotransmetteurs excitateurs) ou inhibent (neurotransmetteurs inhibiteurs) le neurone suivant et peuvent ainsi générer ou interdire la propagation d'un nouvel influx nerveux.

En effet, au niveau post-synaptique, sur la membrane dendritique, se trouvent des récepteurs pour les neurotransmetteurs. Suivant le type de neurotransmetteur et le type des récepteurs, l'excitabilité du neurone suivant va augmenter ou diminuer, ce qui fera se propager ou non l'information. [2]

I.3 Le neurone formel

Le premier modèle du neurone formel date des années quarante. Il été présenté par Mc Culloch et Pitts. S'inspirant de leurs travaux sur les neurones biologiques ils ont proposés le modèle suivant:

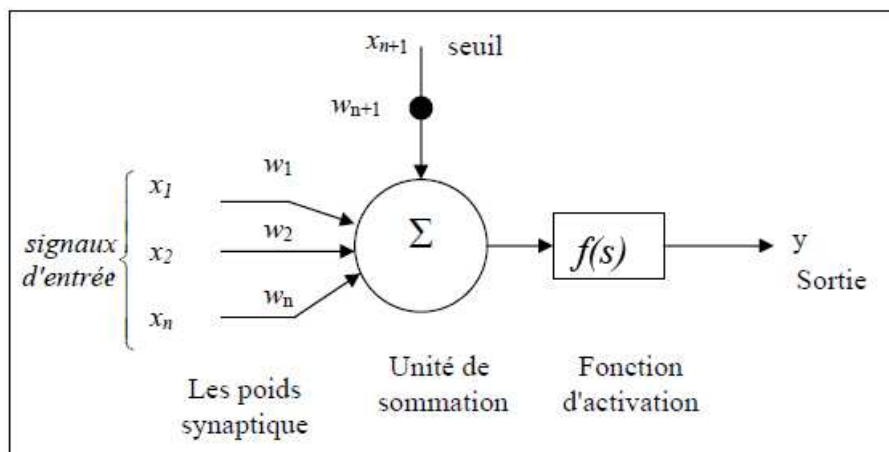


Figure I.2 Le modèle d'un neurone formel.

Un neurone formel est une fonction algébrique non linéaire et bornée, dont la valeur dépend des paramètres appelés poids synaptiques ou poids des connexions.

D'une façon plus générale, un neurone formel est un élément de traitement (opérateur mathématique) possédant n entées (qui sont les neurones externes ou les sorties des autres neurones), et une seule sortie. Ce modèle est décrit mathématiquement par les

équations suivantes:

$$s = \sum_{i=1}^{n+1} w_i x_i$$

$$y = f(s) \dots (I.1)$$

Où w_i, x_i, f, y sont respectivement, les poids synaptiques (paramètres), les entrées, la fonction d'activation et la sortie du neurone.[4]

I.3.1 La fonction d'activation:

Les fonctions d'activations représentent généralement certaines formes de non linéarité. L'une des formes de non linéarité la plus simple, et qui est appropriée aux réseaux discret, est la fonction signe (Figure I.3.a). Une autre variante de ce type des non linéarités est la fonction de Heaviside (Figure I.3.b). Pour la majorité des algorithmes d'apprentissage il est nécessaire d'utiliser des fonctions sigmoïdes différentiables, telles que la fonction sigmoïde unipolaire, Figure I.3.c et la fonction sigmoïde bipolaire (Figure I.3.d). La classe, la plus utilisée des fonctions d'activation, dans le domaine de la modélisation et de la commande des systèmes non linéaires est la fonction sigmoïde bipolaire.[4]

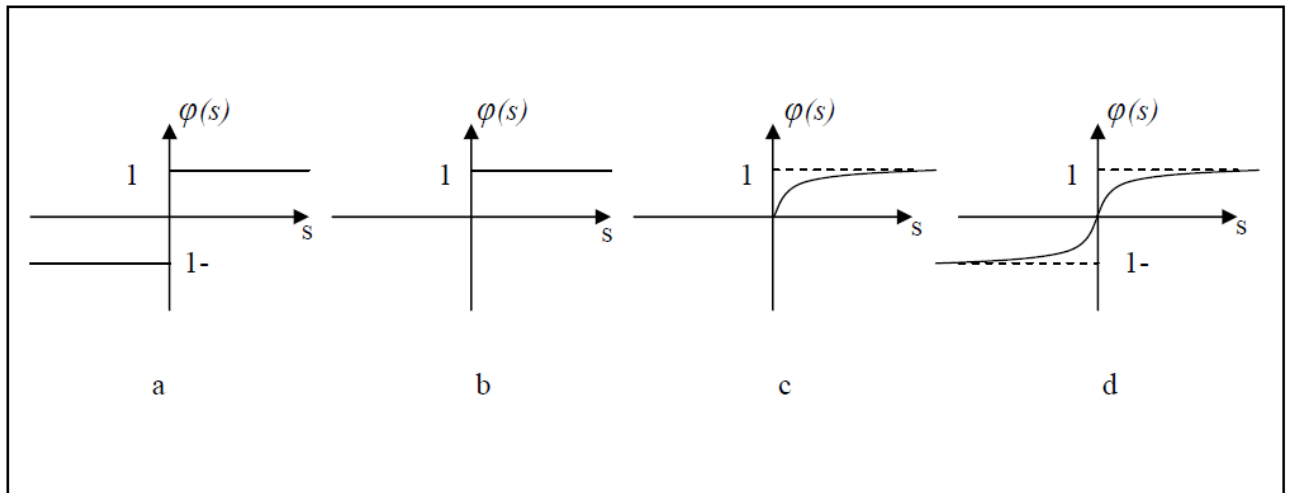


Figure I.3 Différentes fonctions d'activation.

I.4 Les réseaux multicouches

Nous présentons ici une des architectures de réseaux les plus utilisées. Elle correspond à une organisation des neurones en n couches successives ($n \geq 3$). La première couche, dont les neurones voient leur activation forcée à la valeur des données d'entrée, est appelée couche d'entrée. La dernière est appelée couche de sortie. Les seules connexions présentes dans ce type de réseau relie chaque neurone avec l'ensemble de

ceux de la couche suivante (voir la figure suivante). La propagation de l'information se déroule ainsi en sens unique depuis la couche d'entrée vers la couche de sortie.

La fonction d'activation utilisée pour les neurones peut être n'importe quelle fonction croissante et dérivable. On utilise souvent une fonction sigmoïde telle que :

$$f(x) = \frac{1}{1 + \exp(-x)} \dots (I.2)$$

Elle prend pour paramètre la somme pondérée des entrées du neurone :

$$s_i = \sum_j w_{ij}x_j + b_i \dots (I.3)$$

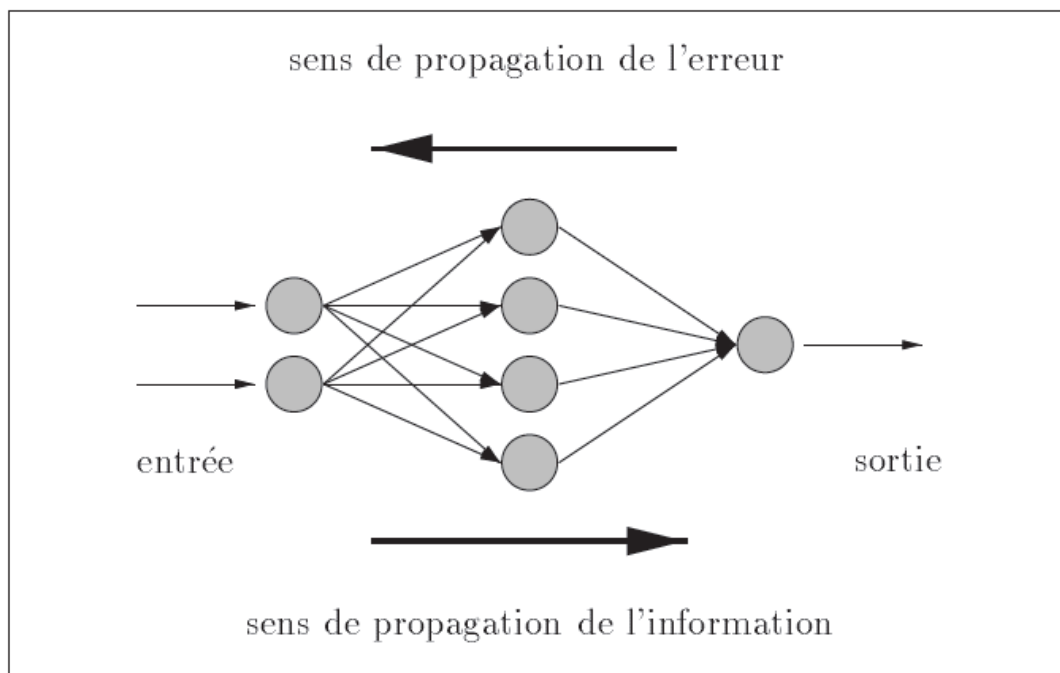


Figure I.4 Un réseau multi-couches comportant 2 neurones d'entrée, 4 neurones cachés et un neurone de sortie.

Où j parcourt l'ensemble des neurones envoyant une connexion vers le neurone i , w_{ij} est le poids de la connexion entre le neurone j et le neurone i et, b_i est un paramètre optionnel appelé biais .[31]

I.5 Apprentissage des réseaux de neurones

Le besoin d'apprentissage se manifeste lorsque l'information à priori est incomplète, et son type dépend du degré de plénitude de cette information, Comme 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. Mathématiquement l'apprentissage est défini par :

$$\frac{\partial w}{\partial t} \neq 0 \dots (I. 4)$$

Où W est la matrice des poids.

On peut distinguer trois types d'apprentissage, un apprentissage supervisé, apprentissage non supervisé et apprentissage par renforcement. [19]

I.5.1 L'apprentissage supervisé

Ce type d'apprentissage nécessite que la réponse désirée du système à entraîner soit connue à priori (présence d'un maître qui fournit la réponse désirée), et il est effectué de la façon suivante : on présente au réseau les valeurs d'entrée et on calcule sa sortie actuelle correspondante ensuite les poids sont ajustés de façon à réduire l'écart entre la réponse désirée et celle du réseau en utilisant l'erreur de sortie (la différence entre la réponse du réseau et celle désirée). Cette procédure est répétée jusqu'à ce qu'un critère de performance soit satisfait. Une fois la procédure d'apprentissage est achevée, les coefficients synaptiques prennent des valeurs optimales au regard des configurations mémorisées et le réseau peut être opérationnel.[19]

I.5.2 Apprentissage non supervisé

Dans ce cas, la connaissance à priori de la sortie désirée n'est pas nécessaire, et la procédure d'apprentissage est basée uniquement sur les valeurs d'entrées. Le réseau s'auto-organise de façon à optimiser une certaine fonction de coût, sans qu'on lui fournisse la réponse désirée. Cette propriété est appelée auto-organisation.[19]

I.5.3 Apprentissage par renforcement

L'idée de base de l'apprentissage par renforcement est inspirée des mécanismes d'apprentissage chez les animaux. Dans ce type d'apprentissage on suppose qu'il n'existe pas de maître (superviseur) qui peut fournir la réponse correcte, mais le système à entraîner est informé, d'une manière indirecte, sur l'effet de son action choisie. Cette action est renforcée si elle conduit à une amélioration des performances du système entraîné, et les éléments qui contribuent dans la génération de cette action sont soit récompensés ou punis [19].

I.6 L'algorithme de rétropropagation du gradient

L'apprentissage supervisé pour les réseaux de neurones, consiste à adapter les poids synaptiques de telle manière que l'erreur entre la sortie du réseau et la sortie désirée soit aussi petite que possible. La plupart des algorithmes d'apprentissage des

réseaux de neurones est basée sur les méthodes du gradient: ils cherchent à minimiser un critère de la forme suivante:

$$J = \frac{1}{2} \sum_{q=1}^m (y_{rq}(Xp) - y_q^d(Xp))^2 \dots (I.5)$$

Où y_{rq} et y_q^d sont la sortie du réseau et la sortie désirée pour le vecteur d'entrée Xp .

Cette optimisation se fait d'une manière itérative, en modifiant les poids en fonction du gradient de la fonction de coût selon la loi d'adaptation suivante:

$$w_{ij}^l(k+1) = w_{ij}^l - \mu \frac{\partial J}{\partial w_{ij}^l} \dots (I.6)$$

Où

w_{ij}^l est le poids de la connexion entre le $i^{\text{ème}}$ neurone de la couche l et le $j^{\text{ème}}$ neurone de la couche $l-1$

$j = 1, \dots, n+1$ la $j^{\text{ème}}$ composante du vecteur d'entrée,

$i = 1, \dots, m+1$ la $i^{\text{ème}}$ composante du vecteur de sortie,

$l = 1, \dots, L$ l'ordre d'une couche dans le réseau de neurone,

μ est une constante positive appelée taux d'apprentissage.

Le gradient est calculé par une méthode spécifique aux réseaux de neurones, dites méthode de rétro propagation. Dans cette méthode il est possible de calculer le gradient de la fonction coût en retropropageant l'erreur commise en sortie vers les couches cachées.[4]

Chapitre II :

Les bras

manipulateurs

Chapitre II :

Les bras manipulateurs

II.1 Introduction

Depuis quelques décennies, la recherche dans la robotique s'est concentrée presque entièrement sur la commande des robots manipulateurs.

Récemment le besoin pour des manipulations complexes et l'évolution des dispositifs tels que les effecteurs à plusieurs doigts et les plates-formes à plusieurs pieds a engendré un large domaine de recherche dans l'étude des systèmes robotiques. Cependant la pièce maîtresse de ces systèmes est le robot manipulateur.

Un robot manipulateur se compose de plusieurs liaisons connectées par des articulations pour former un bras. On peut retrouver deux formes de manipulateurs en fonction de la manière dont les liaisons sont connectées : une forme sérielle et une autre parallèle [6].

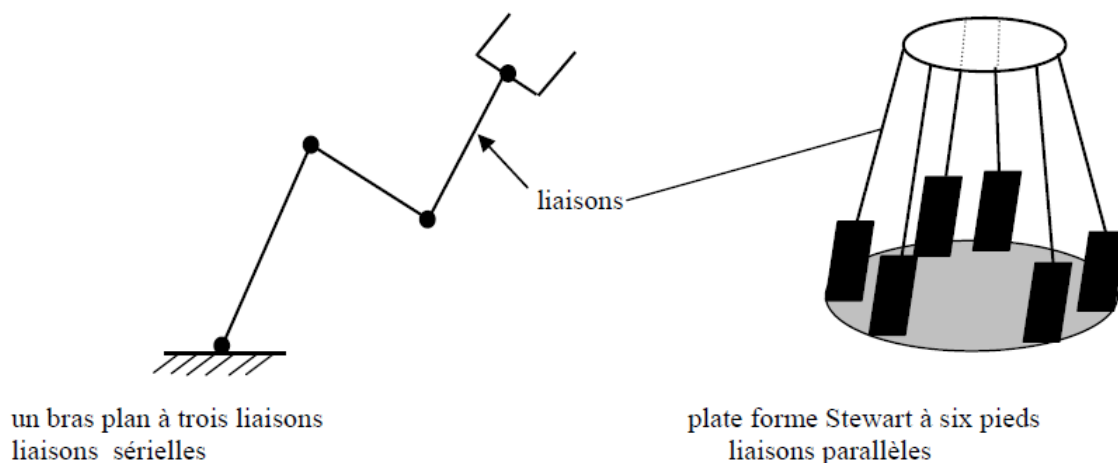


Figure II.1 Manipulateurs à liaisons sérielles et parallèles.

La plupart des robots utilisés actuellement sont des robots ayant des liaisons connectées sériellement.

Les robots manipulateurs sériels, dans leur formes de base, sont des chaînes cinématiques ouvertes composées de liaisons rigides ou flexibles connectées par des articulations : ces robots peuvent se déplacer circulairement dans l'espace par des mouvements générés par des articulations commandées par les actionneurs.

Typiquement chaque articulation possède un degré de liberté et elle est de type rotoïde ou prismatique dépendant du mouvement permis entre les deux articulations et qui peut être un mouvement de rotation ou un mouvement de translation.

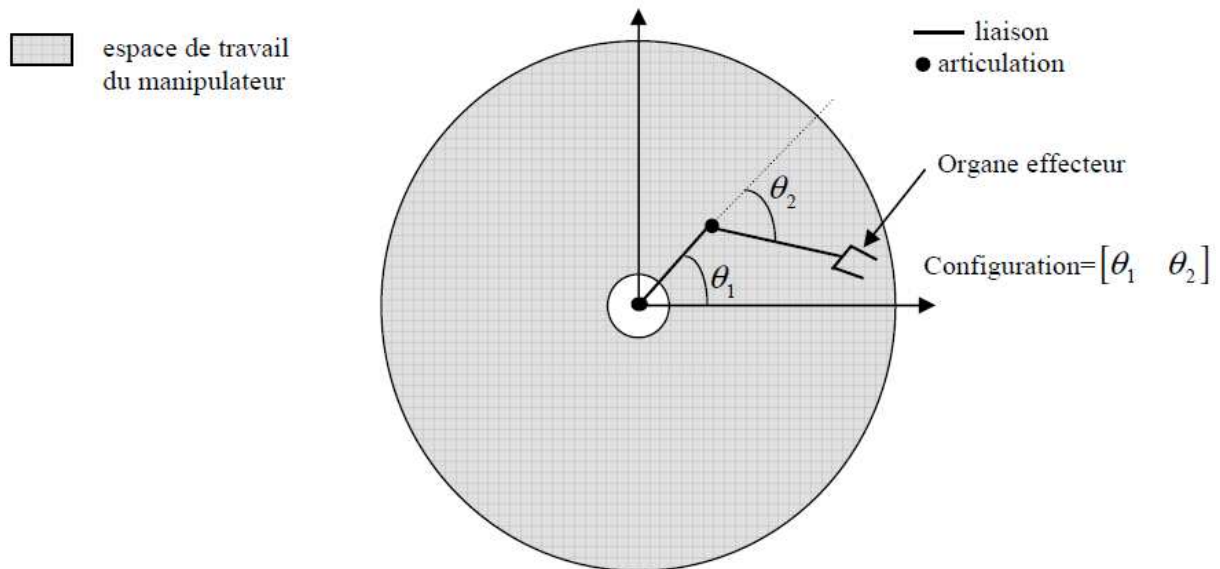


Figure II.2 Manipulateur plan à deux liaisons.

Un organe effecteur ou une pince ayant la forme d'une main est reliée au bras au moyen du poignet.

Les positions articulaires $\{\theta_i\}$ déterminent la configuration du bras à qui correspond une position et une orientation unique de l'organe effecteur. L'espace de travail du manipulateur appelé aussi espace de la tâche est composé de tout les points qui peuvent être atteints par l'extrémité du bras ou un point quelconque de son poignet (pas nécessairement l'organe effecteur puisque ce dernier peut être un outil changeable).

Un paramètre important d'un manipulateur est le nombre de degrés de libertés (nombre d'articulations) qu'il possède pour déplacer et orienter un objet dans un espace à trois dimensions. Un mécanisme doit avoir au moins six degrés de libertés dont trois pour positionner l'organe effecteur vers un point prescrit dans l'espace de la tâche du manipulateur et les trois autres pour l'orienter proprement à cette position, ainsi un bras ayant six articulations (six degrés de libertés).

La configuration articulaire correspondant à une position et une orientation données de l'organe effecteur peut être unique à quelques variations finies .

Les bras robotiques peuvent être cinématiquement conçus redondants en leur conférant des degrés de libertés supplémentaires dans l'espace articulaire, ainsi un manipulateur est

redondant quand le nombre n de ses articulation est supérieur à la dimension de l'espace de la tâche m .



Figure II.3 Exemple d'un bras manipulateur (KUKA KR6 ARC) à 7ddl.

Un manipulateur redondant est caractérisé par le fait qu'il peut avoir un nombre infini de configurations au niveau articulaire correspondant à plus de positions de l'organe effecteur dans l'espace du travail.

Alors que ce surplus de degrés de libertés complique la programmation et les stratégies de contrôle par contre il augmente considérablement l'utilité du robot.

Mathématiquement, un bras robotique est décrit par ses équations cinématiques et dynamiques, la cinématique d'un bras introduit l'étude des relations entre les positions, vitesses et les accélérations de ses différentes parties; l'analyse cinématique est nécessaire pour la planification et Les calculs dynamiques pour l'exécution des mouvements désirés du manipulateur.

Les équations dynamiques d'un bras décrivent son évolution dans le temps en réponse à des forces externes, et des couples agissant sur ses actionneurs. Cependant un système robotique n'est pas seulement un bras manipulateur, en plus du bras, le système renferme aussi une source d'énergie externe, un outillage de l'extrémité du bras, des capteurs externes et internes, des servomécanismes, un ordinateur interface et le contrôleur.

Le contrôleur d'un robot peut être pris comme le cerveau qui commande les mouvements mécaniques du bras : il est responsable, en se basant sur les modèles cinématiques et dynamiques du bras et les mesures captées, de la génération de directives contrôlant les actionneurs des articulations, nécessaire pour la génération du mouvement désiré [6].

II.2 Commande des Robots manipulateur

II.2.1 Introduction

La plupart des tâches demandées à un manipulateur impliquent un contact avec l'environnement. Ceci implique de réguler l'effort exercé par le robot. Les commandes en effort constituent donc l'essentiel des publications relatives à la commande des manipulateurs ; Ceci est à prendre spécifiquement en compte lors du choix de la structure de commande pour piloter le manipulateur.

Le problème de la commande d'un robot manipulateur peut être formulé comme la détermination de l'évolution des forces généralisées (forces ou couples) que les actionneurs doivent exercer pour garantir l'exécution de la tâche tout en satisfaisant certains critères de performance.

Différentes techniques sont utilisées pour la commande des bras manipulateurs. La conception mécanique du bras manipulateur a une influence sur le choix du schéma de commande. Un robot manipulateur est une structure mécanique complexe dont les inerties par rapport aux axes des articulations varient non seulement en fonction de la charge mais aussi en fonction de la configuration.

La plupart des robots utilisent des servomoteurs électriques comme actionneurs. Dans le cas de servomoteurs ayant de faibles rapports de réduction, ce sont les servomoteurs qui doivent compenser les effets des variations des forces d'inertie et de gravité. Dans le cas de servomoteurs avec de forts rapports de réduction, l'inertie vue par les moteurs varie beaucoup moins et il est alors possible de modéliser le robot par un système linéaire qui permet de découpler les articulations.

Dans le contexte de ce document nous considérons uniquement l'utilisation de servomoteurs avec de forts rapports de réduction comme actionneurs, ce qui produit des robots à articulations rigides. Le problème de la rigidité des articulations est évident lorsqu'on parle d'interaction avec l'environnement ou des collisions. Des imprécisions dans la

modélisation de l'environnement peuvent se traduire par des efforts de contact importants qui peuvent endommager les mécanismes internes du robot ou son environnement.

En robotique lorsque l'articulation est rigide, les blessures qu'un robot peut produire sont fonction de l'inertie du mouvement. Pour réduire les dommages produits par une collision, il est recommandé d'utiliser des robots ayant « une élasticité », on parle alors de robots avec des articulations flexibles. On peut considérer aussi des actionneurs élastiques, ou une action parallèle distribué.

Un chapitre du livre W. Khalil [7] est dédié à la commande de robots à articulations flexibles que nous recommandons au lecteur intéressé. Les travaux d'Albu-Schäffer et al utilisent la commande en couple pour chacune des articulations pendant que les travaux de Luca et al, considèrent la détection des collisions et la façon de réagir après la collision.

Deux types de mouvements apparaissent quand on parle de commande du bras manipulateur. Un premier type considère que les mouvements nécessaires pour la réalisation de la tâche sont exécutés dans l'espace libre. Le deuxième type considère des mouvements spécifiques avec des forces de contact pour l'organe terminal qui se déplace dans un espace contraint. Toute tâche de robotique est réalisée par une combinaison de ces deux types de mouvement. Pour simplifier la commande, les deux types de mouvement sont abordés séparément.

II.2.2 Commande dans l'espace libre

II.2.2.1 Commande par articulation

Cette technique est utilisée par des robots manipulateurs qui utilisent des servomoteurs avec de forts rapports de réduction. Lorsque le système présente un comportement linéaire, l'asservissement du mouvement peut être réalisé par des techniques classiques de commande. Nous parlons alors d'une commande décentralisée de type PID. Le schéma classique est amélioré avec des signaux d'anticipation pour corriger les effets de la force de gravité et de couplage.[2]

II.2.2.2 Commande jacobéenne

Cette technique est utilisée depuis les travaux de Whitney et elle est appelée de cette façon lorsqu'elle utilise la matrice jacobéenne inverse du bras manipulateur pour calculer les vitesses de consigne aux articulations. Elle est aussi connue sous le nom de commande à mouvement résolu.

Les approches les plus courantes sont de type : mouvement à vitesse résolu, mouvement à accélération résolu et mouvement à force résolu. La technique de mouvement résolu commande la position de l'organe terminal du manipulateur dans l'espace cartésien, par combinaison des mouvements de plusieurs articulations.

Ce type de schémas peut se présenter sous la forme d'un schéma cinématique quand les vitesses calculées aux articulations sont utilisées directement comme des consignes pour les boucles de commande de chaque articulation, ou sous la forme d'un schéma dynamique quand le schéma utilise le modèle dynamique pour découpler les articulations.

Son principal inconvénient est l'utilisation de l'inverse de la matrice jacobéenne, qui peut devenir singulière dans certaines conditions.[2]

II.2.2.3 Commande par découplage non linéaire

Cette technique est aussi connue sous les noms de commande dynamique ou commande par couple calculé. Lorsque l'application exige des évolutions rapides avec des contraintes dynamiques, la commande doit prendre en compte les forces d'interaction. Ce type de technique considère l'ensemble des articulations et, pour les découpler, utilise la théorie du découplage non linéaire. Cette théorie utilise le modèle dynamique du robot pour le calcul de la loi de commande, ce qui conduit à des lois de commande centralisées non linéaires. Des signaux d'anticipation peuvent être utilisés pour minimiser des effets non linéaires.

Ce type de technique permet la commande dans l'espace des articulations ou dans l'espace cartésien, avec l'avantage que les articulations sont découplées et peuvent évoluer à grandes vitesses avec de fortes inerties. Cette méthode dépend fortement du modèle du système, elle est très sensible aux imprécisions du modèle qui entraînent un découplage imparfait. Ceci constitue son principal inconvénient.[2]

II.2.2.4 Commande passive

Cette technique considère le robot comme un système passif, c'est à dire un système qui dissipe de l'énergie. De telles lois de commande permettent de modifier l'énergie naturelle du robot pour qu'il réalise la tâche. En utilisant le formalisme de Hamilton, la commande cherche à minimiser l'énergie du système en utilisant un bloc non linéaire passif dans la boucle de retour. La commande passive tend à être plus robuste que le découplage non linéaire, lorsque la technique ne recherche pas à annuler les non linéarités.[2]

II.2.2.5 Commande adaptative

Ce type de techniques vise à corriger les déficiences de la commande par découplage non linéaire, comme la connaissance approximative des paramètres du modèle du robot ou pour s'adapter aux différentes conditions opératoires. Ce type de schémas cherche à estimer ou ajuster en-ligne les valeurs des paramètres utilisées dans le calcul de la loi de commande. Un des travaux les plus intéressants sur ce sujet est la commande proposé par Slotine et al appelée commande de Slotine-Li ou commande adaptative passive. Plusieurs travaux sur la commande adaptative sont présentés dans [8].

Les avantages de ce type de techniques sont évidents, malheureusement la puissance de calcul demandée au système constitue un inconvénient important.[2]

II.2.2.6 Commande prédictive

Ce type de schémas, en utilisant le modèle du système et les consignes, est capable de prédire son évolution, de telle manière qu'il est capable d'agir en fonction de l'erreur de prédiction. Trois schémas différents sont proposés : point final fixe, horizon fini et une combinaison des deux précédents. Un grand avantage de ce type d'approche est lié au fait que l'erreur de prédiction n'est pas contaminé par les bruits de mesure mais la dépendance au modèle reste forte.[2]

II.2.2.7 Commande robuste

Dans le cas de paramètres fixes, il est connu que la technique de découplage non linéaire peut devenir instable en présence d'incertitudes. Si les paramètres du modèle ne sont pas connus de façon précise et si l'incertitude sur les paramètres admet des bornes connues, alors les techniques de commande robuste peuvent être utilisées. Par exemple, les travaux de Slotine [1], considèrent la technique de modes glissants appelés aussi commande de structure variable. Cette technique utilise une surface de glissement où la stabilité du système est garantie.[2]

II.2.2.8 Commande optimale

Pour réaliser une tâche, il peut exister un grand nombre de solutions. Dans ce cas, il peut être souhaitable de choisir une solution qui minimise un certain critère. La littérature présente différents types de critères pour la commande optimale : la commande en temps minimal, du domaine des neurosciences la minimisation du jerk pour maximiser la souplesse du mouvement, entre autres.[2]

II.2.3 Commande dans l'espace contraint

Lorsque l'organe terminal entre en contact avec une surface, la chaîne cinématique du robot est fermée sur l'environnement, et du fait de la raideur de l'ensemble, de faibles variations de position du point de contact peuvent induire des efforts importants. Lorsque le robot est contraint par l'environnement suivant toutes les directions, aucun déplacement de l'organe terminal n'est possible et il ne peut qu'exercer des efforts sur l'environnement; en revanche, lorsque le robot est dans l'espace libre, l'absence de tout contact empêche la génération d'efforts. Ainsi, les notions de commande en position et de commande en effort s'excluent mutuellement : on ne peut contrôler simultanément une force et un déplacement suivant une même direction.

La littérature propose cinq schémas de commande, les trois premiers n'utilisent pas de consigne d'effort : compliance passive, commande par raideur active et commande en impédance. Les deux suivantes utilisent la consigne d'effort : la commande hybride et la commande hybride externe.

Plus généralement, la fonction d'un robot manipulateur est de situer dans l'espace son organe terminal OT. La Situation de l'organe terminal est définie par la position et l'orientation d'un repère lié à son but. Une tâche simple peut être définie comme le mouvement libre réalisé par le robot manipulateur pour aller d'une situation initiale à une situation finale. Une tâche complexe peut être définie de la même façon, en précisant que le robot doit suivre un chemin prédéfini tout en respectant des contraintes cinématiques liées au mouvement. Dans les deux cas, le mouvement que le robot réalise devra être défini par des consignes qui correspondent à des trajectoires temporelles.[2]

Chapitre III :
Identification des
systemes
non linéaires
par les réseaux
de neurones

Chapitre III :

Identification des systèmes non linéaires par les réseaux de neurones

III.1 Introduction

La modélisation des systèmes non linéaires par réseaux de neurones a fait l'objet de nombreux travaux de recherche depuis une dizaine d'années à cause de la capacité d'apprentissage, d'approximation et de généralisation que possèdent ces réseaux [11][12]. En effet, cette nouvelle approche fournit une solution efficace à travers laquelle de larges classes des systèmes non linéaires peuvent être modélisés sans une description mathématique précise. L'identification, c'est l'opération de détermination du modèle dynamique d'un système à partir des mesures entrées/sorties. Souvent la sortie mesurée des systèmes est entachée du bruit. Cela est dû soit à l'effet des perturbations agissant à différents endroits du procédé, soit à des bruits de mesure. Ces perturbations introduisent des erreurs dans l'identification des paramètres du modèle [13][14][15]. Dans ce chapitre, nous allons résumer brièvement les différentes approches de la conception d'un modèle en utilisant les réseaux de neurones.

III.2 Identification des systèmes

L'identification d'un système consiste principalement à sélectionner la structure du modèle. Et l'estimation des paramètres de celui-ci. En d'autres termes construire un modèle acceptable montré en figure (III.1), qui produirait une sortie $\hat{y}(k)$ approximant $y(k)$ lorsqu'il est soumis à la même entrée $u(k)$ que celle du système à identifier et ceci dans le but d'optimiser un certain critère de performance basé sur l'erreur entre la sortie du système à identifier et la sortie du modèle d'identification [16][17][18][20]. Il est à rappeler qu'il existe plusieurs techniques d'identification des systèmes non linéaires, disponibles dans la littérature [21][22][20].

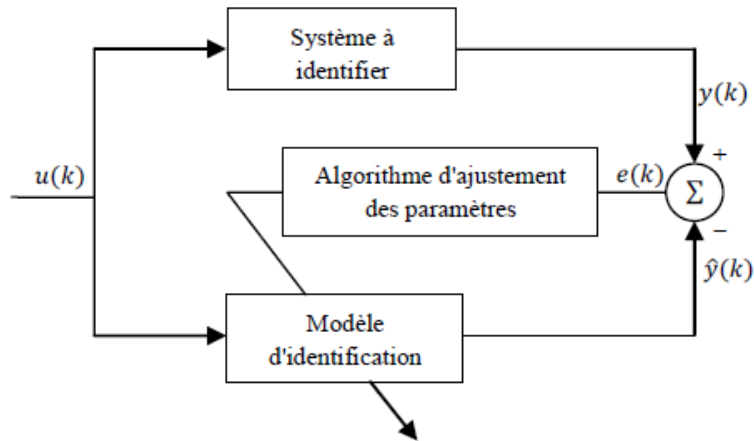


Figure III.1 schéma général d'Identification.

L'identification se fait en général en deux étapes: l'étape qualitative (caractérisation) et l'étape quantitative (estimation des paramètres).[23][24][20]

III.3 Les structures d'identification

Le principe général de l'identification est simple et consiste à placer en parallèle le modèle d'identification et le processus à identifier, comme le montre la Figure (III.2). Le modèle d'identification reçoit en entrée la commande $u(k)$ appliquée et éventuellement la sortie $y(k-1)$ précédente du processus. Il est entraîné à produire la nouvelle sortie (ou le nouvel état) $y(k)$ du processus, cette méthode d'identification est souvent appelée **méthode série-parallèle**. Elle est aussi souvent considérée comme plus stable car le modèle d'identification est régulièrement (recalculé) en utilisant l'état réel du processus, donné par :

$$\hat{y}(k+1) = \hat{f}[y(k), y(k-1), \dots, y(k-n+1); u(k), u(k-1), \dots, u(k-m+1)] \dots (III.1)$$

Par opposition à la méthode **parallèle** présentée au Figure (III.3). Dans cette dernière méthode, le modèle d'identification ne reçoit pas en entrée la sortie réelle $y(k-1)$ du processus mais la sortie $\hat{y}(k-1)$ qu'il a lui-même prédite au pas de temps précède.

Cette approche sera d'intérêt si la boucle donnant l'état du système est remplacée par une connexion récurrente du modèle d'identification. Ce dernier est alors libre de développer sa propre représentation de l'espace des états du processus. Elle est donnée par [25][26]:

$$\hat{y}(k+1) = \hat{f}[\hat{y}(k), \hat{y}(k-1), \dots, \hat{y}(k-n+1); u(k), u(k-1), \dots, u(k-m+1)] \dots (III.2)$$

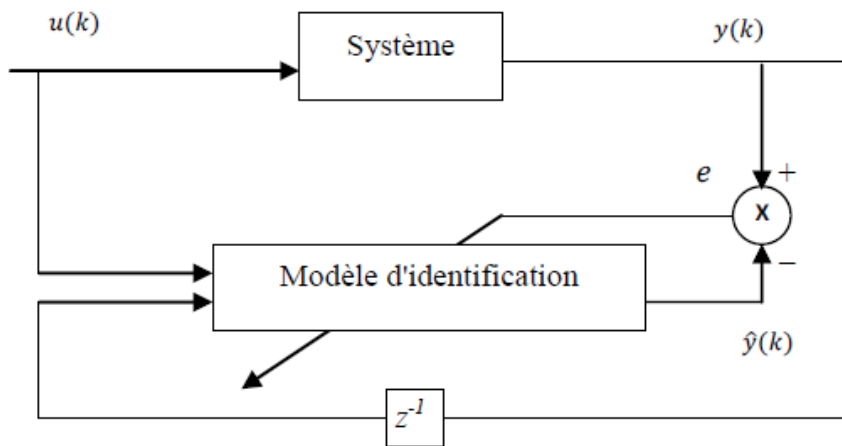


Figure III.2 Identification de processus par la méthode série-parallèle

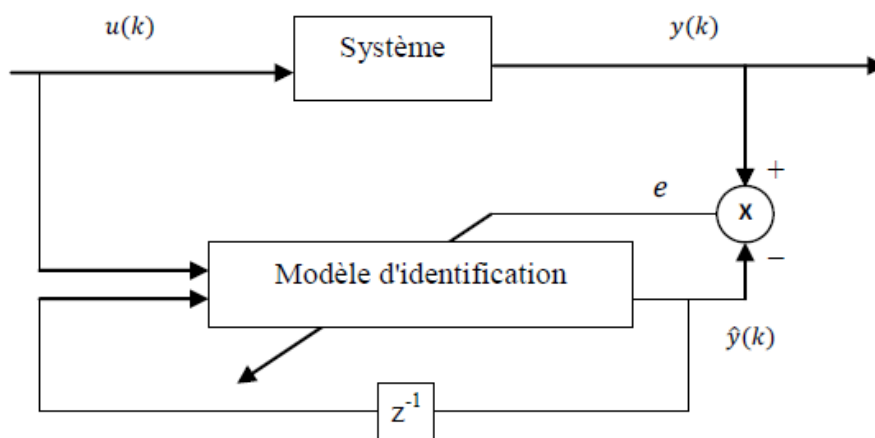


Figure III.3 Identification de processus par la méthode parallèle

III.4 Identification par réseaux de neurones

L'utilisation des réseaux de neurones pour la modélisation des systèmes non linéaires découle naturellement des aptitudes de ces derniers à l'approximation et la généralisation. La détermination du modèle d'identification neuronale d'un système comporte en général les étapes suivantes:

- acquisition des données d'apprentissage et de test,
- choix de la structure du modèle,
- estimation des paramètres du modèle,
- validation du modèle identifié.

La première étape fournit les données entrées/sorties susceptibles de permettre l'extraction d'un modèle de procédé significatif, la deuxième étape consiste à choisir la structure du modèle susceptible de représenter la dynamique du système, l'architecture du réseau de neurones et ses entrées. Les réseaux multicouches statiques sont les plus utilisés à

cause de la simplicité de leurs algorithmes d'apprentissage et leurs aptitudes à l'approximation et à la généralisation. Il n'existe pas de méthodes générales pour le choix du nombre de neurones sur chaque couche cachée ainsi que le nombre de ces dernières. Cependant, un réseau à une seule couche cachée est dans la majorité des cas suffisant [27][28]. En référence à la théorie des systèmes linéaires, plusieurs modèles non linéaires ont été proposés [27][29][30].

Si identifier un système permet à un modèle de reproduire son comportement, alors les modèles connexionnistes représentent une solution parmi d'autres. L'utilisation des réseaux de neurones ne permet cependant pas d'aboutir à une formulation analytique, ni même de connaître les valeurs des paramètres. L'utilisation d'un modèle connexionniste permet par contre d'obtenir un "mapping" entre l'espace des entrées et celui des sorties du système. Sa précision dépend du nombre de neurones impliqués et de la complexité inhérente de la structure du réseau. Les modèles connexionnistes ne sont pas des modèles paramétriques au sens conventionnel. Ils ont leur propre représentation et leur propres paramètres (les poids, les fonctions d'activation). A un stimulus d'entrée, ils "répondent" par la sortie qu'ils ont appris à y associer. Comme ils possèdent des capacités de généralisation, à une entrée qui n'a jamais été présentée, ils retournent une réponse qui est en quelque sorte une moyenne des sorties associées aux entrées les plus proches.[32]

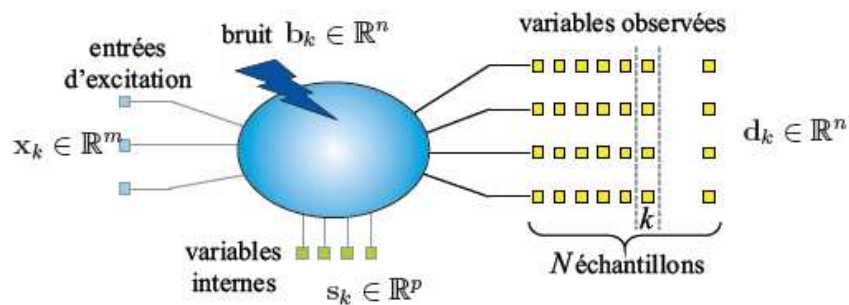


Figure III.4 Définition d'un processus.

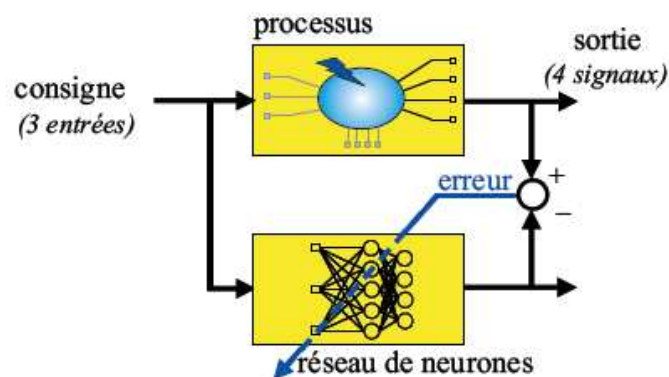


Figure III.5 Schéma d'apprentissage pour reproduire le comportement d'un processus.

L'apprentissage d'un processus passe généralement par un apprentissage supervisé, du moins pour un système défini par des entrées et des sorties. Ce principe est illustré par la Figure III.5. Le système impose alors ses dimensions au réseau de neurones qui est utilisé pour reproduire son comportement. Après apprentissage, pour un vecteur d'entrées à un instant donné, le réseau de neurones délivre un vecteur de sortie aussi proche que possible de celui du système. L'erreur entre sa propre sortie et celle du système sert à adapter ses poids de sorte que l'estimation soit encore meilleure à la prochaine itération.

Pour prendre en compte explicitement le caractère dynamique d'un système avec un modèle connexionniste, plusieurs techniques peuvent être utilisées. Une première méthode repose sur l'utilisation d'un réseau de neurones récurrent ou de tout autre réseau capable de capter, mimer et reproduire l'effet dynamique du système. Une autre solution consiste à fournir au réseau de neurones des signaux représentatifs du caractère dynamique du système. Ceci se fait généralement à l'aide de lignes à retard, leur nombre dépend de la dynamique

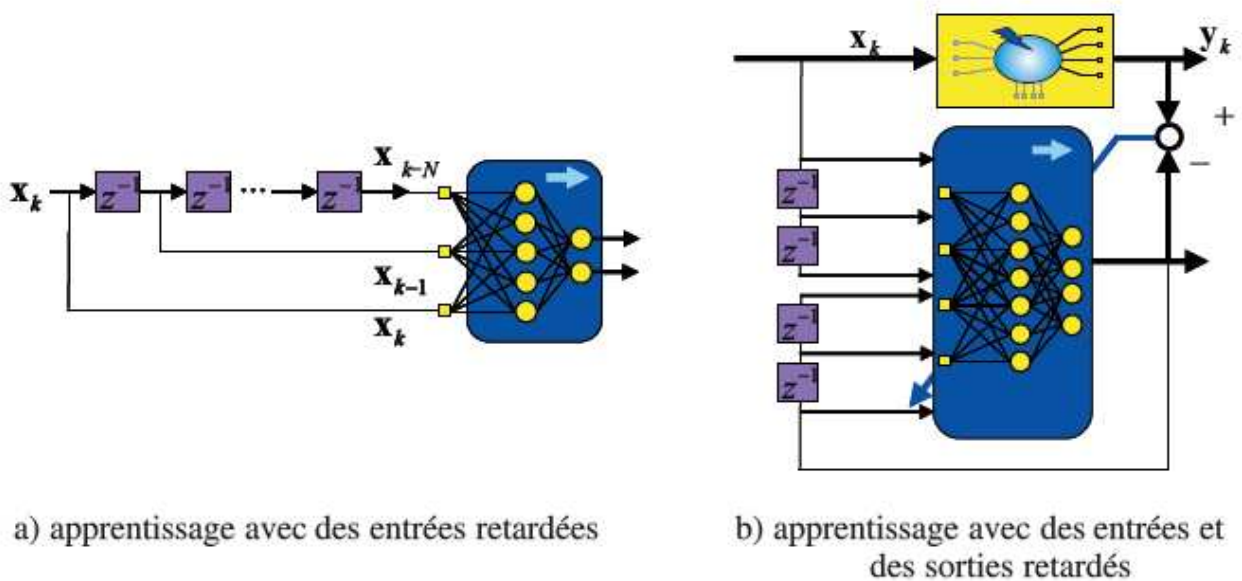


Figure III.6 Identification d'un système à l'aide de lignes à retard

du système. Pour être performant, le réseau de neurones doit disposer des informations pertinentes à travers les signaux des entrées ainsi que ceux des sorties. La Figure III.6 montre des exemples d'utilisation des lignes à retard.

Il est possible d'apprendre l'inverse du système. La Figure III.7 a) montre que pour estimer le système inverse, les sorties du système sont fournies en entrée du réseau de neurones, inversement, les entrées de commande du système deviennent les sorties de référence du réseau de neurones. Après apprentissage, le modèle connexionniste mime la

fonction inverse $f^{-1}(\cdot)$ d'un système ayant une fonction linéaire ou non linéaire $f(\cdot)$. Ainsi, en connectant le processus en série avec l'estimation de son inverse, la fonction de transfert de l'ensemble devient unitaire. Le réseau de neurone peut alors constituer un régulateur sans boucle de retour. L'intérêt de ce schéma réside essentiellement dans le fait qu'alors l'espace de commande est identique à l'espace de sortie à contrôler.

Il est parfois intéressant de pouvoir disposer du Jacobien du système qui est sa dérivée par rapport au temps. On conçoit aisément que l'apprentissage du Jacobien du système revient à estimer le lien entre les vitesses des entrées et des sorties. Des lignes à retard permettent d'obtenir les vitesses à partir des signaux et de leurs valeurs retardées. La Figure III.7 b) montre le lien effectué par le modèle connexionniste entre les déplacements ΔX_k et ΔY_k respectivement dans les espaces des entrées et des sorties (au lieu du lien entre des positions X_k et Y_k). Cette façon de procéder permet par exemple de mettre en œuvre une approche linéaire locale, en particulier en utilisant également le vecteur X_k en tant qu'entrée supplémentaire pour le réseau de neurones.

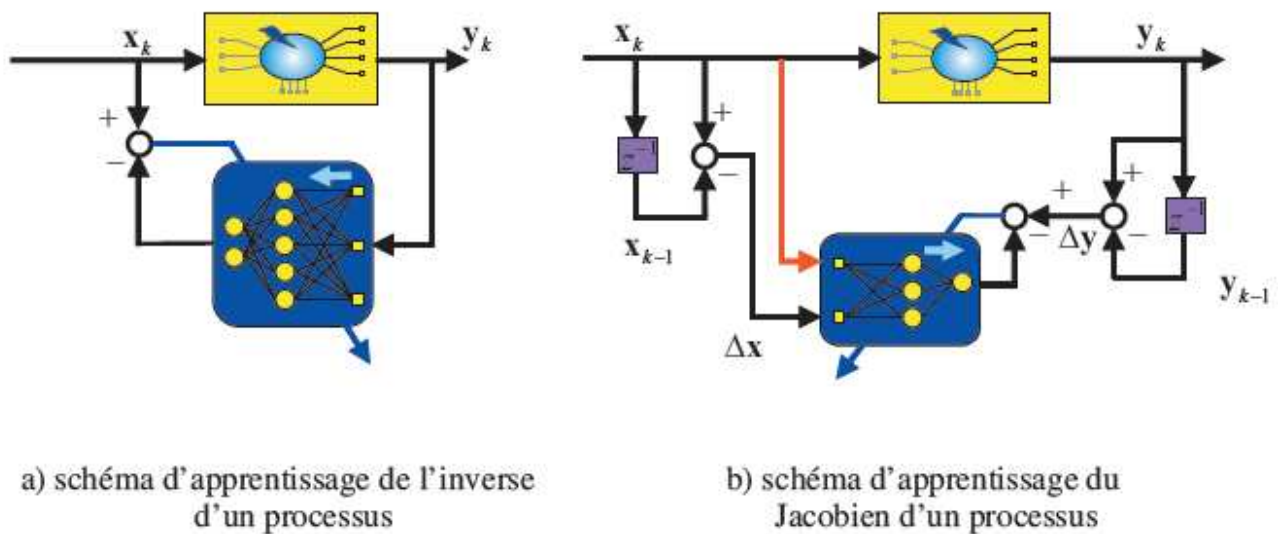


Figure III.7 Différents schémas neuronaux d'identification d'un système

Ces techniques d'apprentissage permettent de disposer des outils de base pour élaborer et mettre au point des lois de commande.

Chapitre IV :
Identification et
commande d'un
bras manipulateur à 2 ddl
avec les réseaux
de neurones artificiels

Chapitre IV :

Identification et commande d'un bras manipulateur à 2 ddl avec les réseaux de neurones artificiels

IV.1 Introduction

Malgré la constante augmentation de puissance des calculateurs, malgré les approches théoriques de plus en plus sophistiquées, un certain nombre de tâches résistent encore aux algorithmes et aux méthodes classiques de traitement des signaux et des données. Ces tâches relèvent typiquement du traitement, en temps réel, de très grands flots de données souvent multidimensionnelles et arrivant à des cadences élevées. Le grand nombre des données, leur variabilité, le fait qu'elles ne répondent pas à des modèles physiques connus nous laissent souvent démunis devant des tâches de caractérisation, de reconnaissance et de prise de décision.

Il y a des centaines d'exemples nous montrons à la fois combien on peut espérer de la modélisation du système nerveux mais aussi combien il sera difficile d'imaginer et de comprendre les divers aspects des problèmes de perception. Il paraît donc naturel d'essayer de comprendre comment les systèmes biologiques sont capables de telles performances, et si possible, de s'inspirer de leurs principes pour imaginer de nouveaux algorithmes ou de nouvelles machines plus efficaces que ceux dont nous disposons actuellement. Les techniques de réseaux de neurones relèvent d'une telle approche : comprendre les principes selon lesquels les systèmes biologiques traitent l'information et s'en inspirer pour élaborer de nouvelles techniques en sciences de l'ingénieur. C'est donc une double démarche, à la fois cognitive et synthétique où le monde biologique doit être considéré comme une source de référence et de connaissance.

D'un point de vue technique, il est clair que seuls les principes seront importants. Il ne sera généralement pas nécessaire, pour modéliser telle ou telle fonction, de simuler toutes les

molécules chimiques et les enzymes qu'elle implique, l'adéquation fine aux modèles biologiques ne sera retenue que dans la mesure où elle conduit à des réalisations techniquement économiques.

Les caractéristiques essentielles des réseaux de neurones réels que nous conserverons dans les modèles mathématiques étudiés, concernent le grand nombre de connexions, la non-linéarité des relations entrée-sortie et la faculté de "plasticité" ou d'adaptabilité. Ces caractéristiques, même simplifiées, leur confèrent déjà de multiples possibilités en traitement des signaux et des informations ainsi que la faculté d'apprendre à classer, à reconnaître des formes ou à réaliser des tâches complexes.

L'objectif de notre travail est l'identification et commande d'un bras manipulateur on utilisant les réseaux de neurones. Le robot avec son contrôleur et identificateur est simulé à l'aide du logiciel **MATLAB**.

IV.2 Le robot manipulateur

Soit l'exemple du robot manipulateur suivant :



Figure IV.1 Pelican : bras de robot expérimental de CICESE, laboratoire de robotique

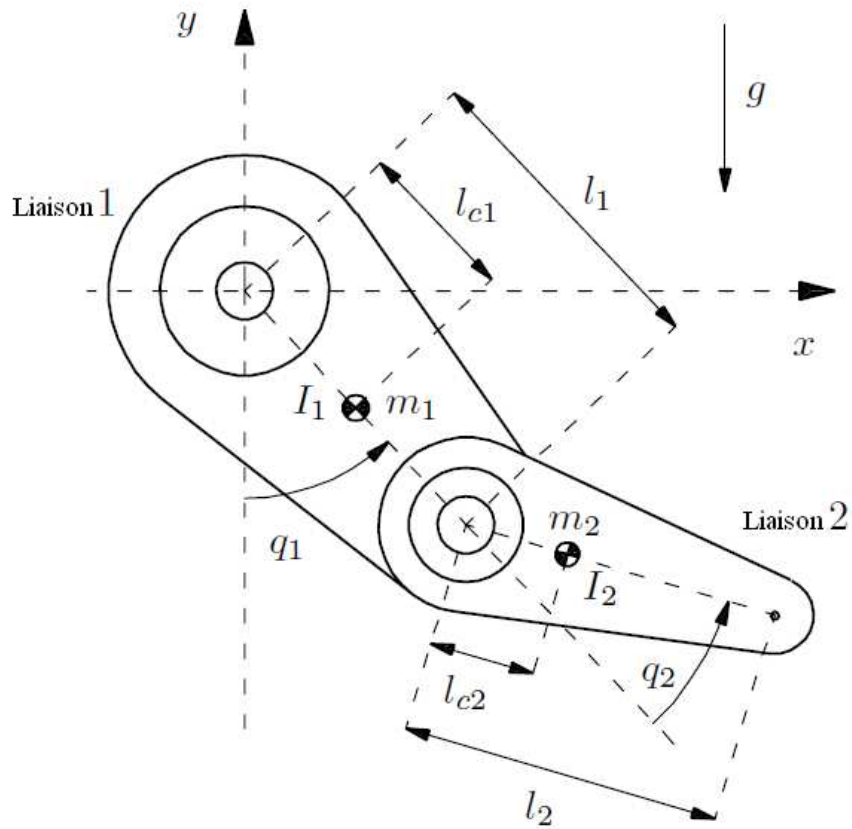


Figure IV.2 Diagramme du bras pelican (2 degrés de liberté)

Les paramètres du robot sont présentés dans le tableau suivant :

Tableau I: paramètres du robot (pelican)

Paramètre	Valeur
Masse de la liaison 1 (m_1)	6.5225 kg
Masse de la liaison 2 (m_2)	2.0458 kg
Longueur de la liaison 1 (l_1)	0.26 m
Longueur de la liaison 2 (l_2)	0.26 m
La gravité (g)	9.81 ms^{-2}
Distance vers le centre de masse de la liaison 1 (l_{c1})	0.0983 m
Distance vers le centre de masse de la liaison 2 (l_{c2})	0.0229 m
Moment d'inertie du centre de masse m_1 (I_1)	0.1213 kg m^2
Moment d'inertie du centre de masse m_2 (I_2)	0.0116 kg m^2

IV.3 Le modèle dynamique

IV.3.1 Introduction

Dans cette section nous allons écrire le modèle dynamique du bras en utilisant la formulation Lagrange-Euler (LE), et nous allons représenter la matrice d'inertie et le vecteur des forces centrifuges et Coriolis non linéaires et le vecteur des forces de charges gravitationnelles. Certainement, le modèle que nous dérivons ici s'en applique à tout les manipulateurs planaires qui ressemble au modèle pelican.[33]

IV.3.2 Formulation LE

Considérons le robot décrit dans la figure IV.2

L'énergie cinétique K du bras est décrit par la somme :

$$k(q, \dot{q}) = k_1(q, \dot{q}) + k_2(q, \dot{q}) \dots (IV.1)$$

Avec k_1 et k_2 sont les énergies cinétiques associées au centres de masses m_1 et m_2 respectivement .

Les coordonnées du centre de masse de la liaison 1 selon le plant (xy) :

$$x_1 = l_{c1} \sin(q_1) \dots (IV.2)$$

$$y_1 = -l_{c1} \cos(q_1) \dots (IV.3)$$

Le vecteur de vitesse v_1 de la même liaison est :

$$v_1 = \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} = \begin{bmatrix} l_{c1} \cos(q_1) \dot{q}_1 \\ l_{c1} \sin(q_1) \dot{q}_1 \end{bmatrix} \dots (IV.4)$$

Sachant que :

$$\|v_1\|^2 = v_1^T v_1 = l_{c1}^2 \dot{q}_1^2 \dots (IV.5)$$

Au final l'énergie cinétique correspondante à la 1^{er} liaison est obtenu par :

$$k_1(q, \dot{q}) = \frac{1}{2} m_1 v_1^T v_1 + \frac{1}{2} I_1 \dot{q}_1^2 = \frac{1}{2} m_1 l_{c1}^2 \dot{q}_1^2 + \frac{1}{2} I_1 \dot{q}_1^2 \dots (IV.6)$$

Les coordonnées du centre de masse de la liaison 2 suivant le plant (xy) :

$$x_2 = l_1 \sin(q_1) + l_{c2} \sin(q_1 + q_2) \dots (IV.7)$$

$$y_2 = -l_1 \cos(q_1) - l_{c2} \cos(q_1 + q_2) \dots (IV.8)$$

Le vecteur de vitesse v_2 de la même liaison (2) est :

$$v_2 = \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} l_1 \cos(q_1) \dot{q}_1 + l_{c2} \cos(q_1 + q_2) [\dot{q}_1 + \dot{q}_2] \\ l_1 \sin(q_1) \dot{q}_1 + l_{c2} \sin(q_1 + q_2) [\dot{q}_1 + \dot{q}_2] \end{bmatrix} \dots (IV.9)$$

En utilisant les deux formules trigonométriques $\cos(\theta)^2 + \sin(\theta)^2 = 1$ et $\sin(q_1)\sin(q_1 + q_2) + \cos(q_1)\cos(q_1 + q_2) = \cos(q_2)$ on conclut que :

$$\|v_2\|^2 = v_2^T v_2 = l_1^2 \dot{q}_1^2 + l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + 2l_1 l_{c2} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) \dots (IV.10)$$

Ce qui implique que :

$$\begin{aligned} k_2(q, \dot{q}) &= \frac{1}{2} m_2 v_2^T v_2 + \frac{1}{2} I_2 (\dot{q}_1 + \dot{q}_2)^2 \\ &= \frac{m_2}{2} l_1^2 \dot{q}_1^2 + \frac{m_2}{2} l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c2} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) \\ &\quad + \frac{1}{2} I_2 (\dot{q}_1 + \dot{q}_2)^2 \dots (IV.11) \end{aligned}$$

L'énergie potentiel u est décrit par :

$$u(q) = u_1(q) + u_2(q) \dots (IV.12)$$

Avec u_1 et u_2 sont les énergies potentiels associées aux masses m_1 et m_2 respectivement. Si l'énergie potentiel est égale à zéro lorsque $y=0$ alors :

$$u_1(q) = -m_1 l_{c1} g \cos(q_1) \dots (IV.13)$$

$$u_2(q) = -m_2 l_1 g \cos(q_1) - m_2 l_{c2} g \cos(q_1 + q_2) \dots (IV.14)$$

On obtient le Lagrangien par :

$$\begin{aligned} \mathcal{L}(q, \dot{q}) &= k(q, \dot{q}) - u(q) \\ &= \frac{1}{2} [m_1 l_{c1}^2 + m_2 l_1^2] \dot{q}_1^2 + \frac{1}{2} m_2 l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] \\ &\quad + m_2 l_1 l_{c2} \cos(q_2) [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \\ &\quad + [m_1 l_{c1} + m_2 l_1] g \cos(q_1) \\ &\quad + m_2 g l_{c2} \cos(q_1 + q_2) \\ &\quad + \frac{1}{2} I_1 \dot{q}_1^2 + \frac{1}{2} I_2 [\dot{q}_1 + \dot{q}_2]^2 \dots (IV.15) \end{aligned}$$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \dot{q}_1} &= [m_1 l_{c1}^2 + m_2 l_1^2] \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 \\ &\quad + 2m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_2 \\ &\quad + I_1 \dot{q}_1 + I_2 [\dot{q}_1 + \dot{q}_2] \dots (IV.16)\end{aligned}$$

$$\begin{aligned}\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{q}_1} \right] &= [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_1 \\ &\quad + [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_2 \\ &\quad - 2m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2^2 \\ &\quad + I_1 \ddot{q}_1 + I_2 [\ddot{q}_1 + \ddot{q}_2] \dots (IV.17)\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial q_1} = -[m_1 l_{c1} + m_2 l_1] g \sin(q_1) - m_2 g l_{c2} \sin(q_1 + q_2) \dots (IV.18)$$

$$\frac{\partial \mathcal{L}}{\partial \dot{q}_2} = m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + I_2 [\dot{q}_1 + \dot{q}_2] \dots (IV.19)$$

$$\begin{aligned}\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{q}_2} \right] &= m_2 l_{c2}^2 \ddot{q}_1 + m_2 l_{c2}^2 \ddot{q}_2 \dots (IV.20) \\ &\quad + m_2 l_1 l_{c2} \cos(q_2) \ddot{q}_1 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 \\ &\quad + I_2 [\ddot{q}_1 + \ddot{q}_2]\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial q_2} = -m_2 l_1 l_{c2} \sin(q_2) [\dot{q}_1 \dot{q}_2 + \dot{q}_1^2] - m_2 g l_{c2} \sin(q_1 + q_2) \dots (IV.21)$$

Le modèle dynamique du bras est obtenu on appliquant l'équation de Lagrange suivante :

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right] - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i \quad \text{avec } i = 1,2$$

On obtient le modèle dynamique au final :

$$\begin{aligned}\tau_1 &= [m_1 l_{c1}^2 + m_2 l_1^2 m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) I_1 + I_2] \ddot{q}_1 \\ &\quad + [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2] \ddot{q}_2 \\ &\quad - 2m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2^2 \\ &\quad + [m_1 l_{c1} + m_2 l_1] g \sin(q_1) \dots (IV.22) \\ &\quad + m_2 g l_{c2} \sin(q_1 + q_2)\end{aligned}$$

$$\begin{aligned}\tau_2 &= [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + I_2] \ddot{q}_2 \\ &\quad + m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1^2 + m_2 g l_{c2} \sin(q_1 + q_2) \dots (IV.23)\end{aligned}$$

Sachant que τ_1 et τ_2 sont des couples délivrés aux actionneurs des deux articulation 1 et 2.[33]

Le modèle dynamique est décrit sous l'équation matricielle suivante :

$$\underbrace{\begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix}}_{M(q)} \ddot{q} + \underbrace{\begin{bmatrix} C_{11}(q, \dot{q}) & C_{12}(q, \dot{q}) \\ C_{21}(q, \dot{q}) & C_{22}(q, \dot{q}) \end{bmatrix}}_{C(q, \dot{q})} \dot{q} + \underbrace{\begin{bmatrix} g_1(q) \\ g_2(q) \end{bmatrix}}_{g(q)} = \tau, \quad \dots (IV.24)$$

Avec :

$$M_{11}(q) = m_1 l_{c1}^2 + m_2 [l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos(q_2)] + I_1 + I_2$$

$$M_{12}(q) = m_2 [l_{c2}^2 + l_1 l_{c2} \cos(q_2)] + I_2$$

$$M_{21}(q) = m_2 [l_{c2}^2 + l_1 l_{c2} \cos(q_2)] + I_2$$

$$M_{22}(q) = m_2 l_{c2}^2 + I_2$$

$$C_{11}(q, \dot{q}) = -m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2$$

$$C_{12}(q, \dot{q}) = -m_2 l_1 l_{c2} \sin(q_2) [\dot{q}_1 + \dot{q}_2]$$

$$C_{21}(q, \dot{q}) = m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1$$

$$C_{22}(q, \dot{q}) = 0$$

$$g_1(q) = [m_1 l_{c1} + m_2 l_1] g \sin(q_1) + m_2 l_{c2} g \sin(q_1 + q_2)$$

$$g_2(q) = m_2 l_{c2} g \sin(q_1 + q_2)$$

Tel que :

q : vecteur des variables articulaires généralisées.

\dot{q} : vecteur des vitesses angulaires.

\ddot{q} : vecteur des accélérations angulaires.

M : matrice d'inertie.

C : matrice spécifiant l'effet de centrifuge et de Coriolis.

g : vecteur exprimant l'effet gravitationnel.

τ : vecteur des couples généralisés.

IV.4 La trajectoire désirée

On veut réaliser une commande pour que les deux liaisons du système bras robotique suis les trajectoires angulaires désirées données par les deux équations suivantes :

$$q_{1d}(t) = A \sin(2\pi ft) + q_0 \text{ rad} \dots (IV.25)$$

$$q_{2d}(t) = -A \sin(2\pi ft) + q_0 \text{ rad} \dots (IV.26)$$

Avec :

$$A = 0.45$$

$$f = 0.125 \text{ hz}$$

$$q_0 = 1$$

La trajectoire est illustrée sur les figures suivantes :

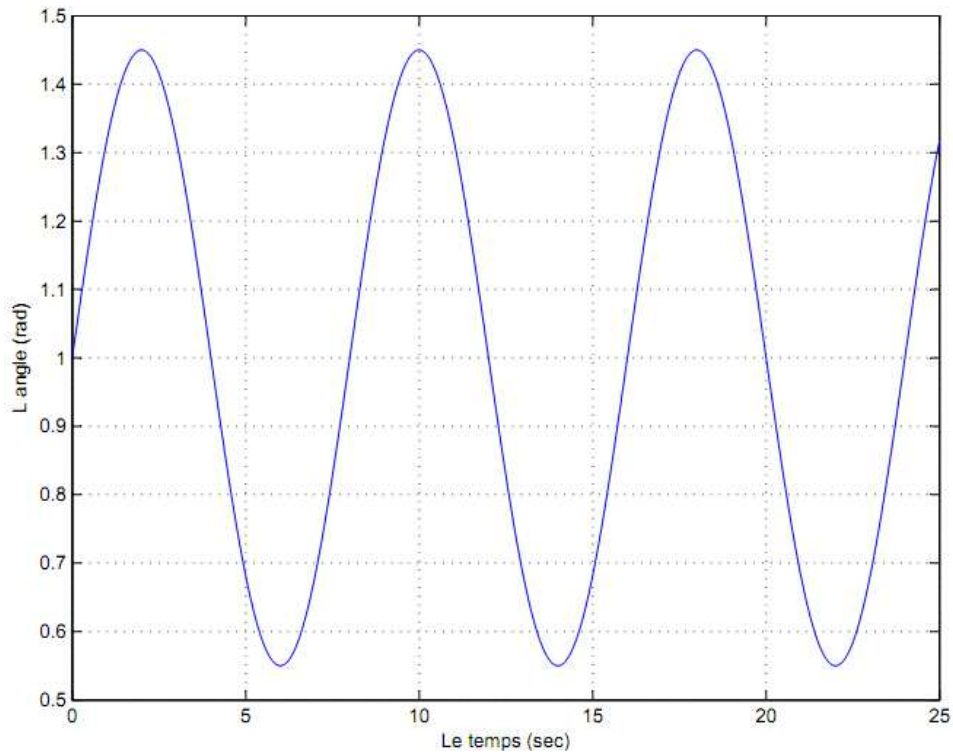


Figure IV.3 La trajectoire angulaire désiré (q_{1d}) pour la 1^{er} liaisons du bras.

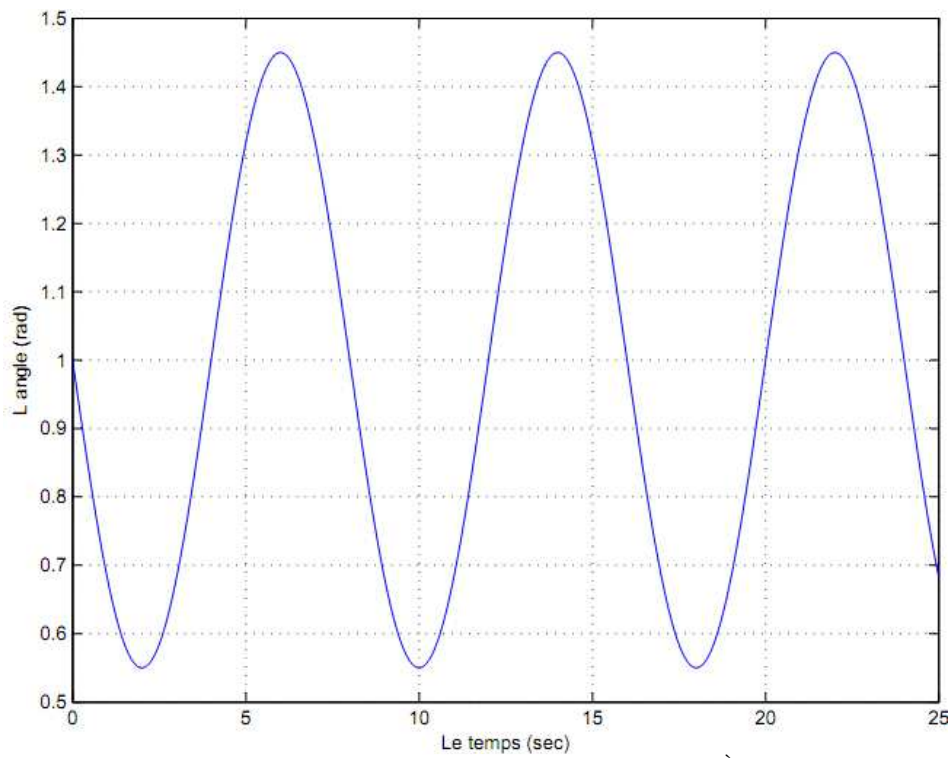


Figure VI.4 La trajectoire angulaire désiré (q_{2d}) pour la 2^{ème} liaisons du bras.

IV.5 Identification du système bras manipulateur

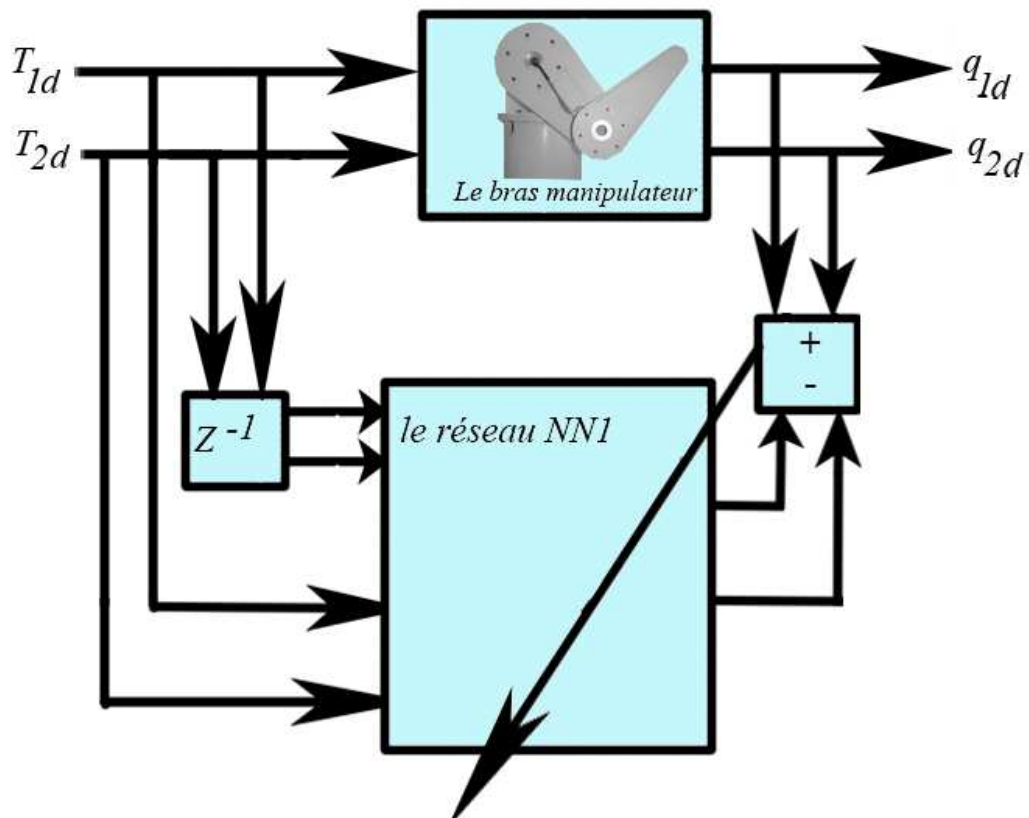


Figure IV.5 Schéma d'identification du modèle bras robotique.

Le but est d'Identifier le système dynamique (bras manipulateur) et cela on modélisant un modèle neuronal qui va se comporter comme le système après son apprentissage.

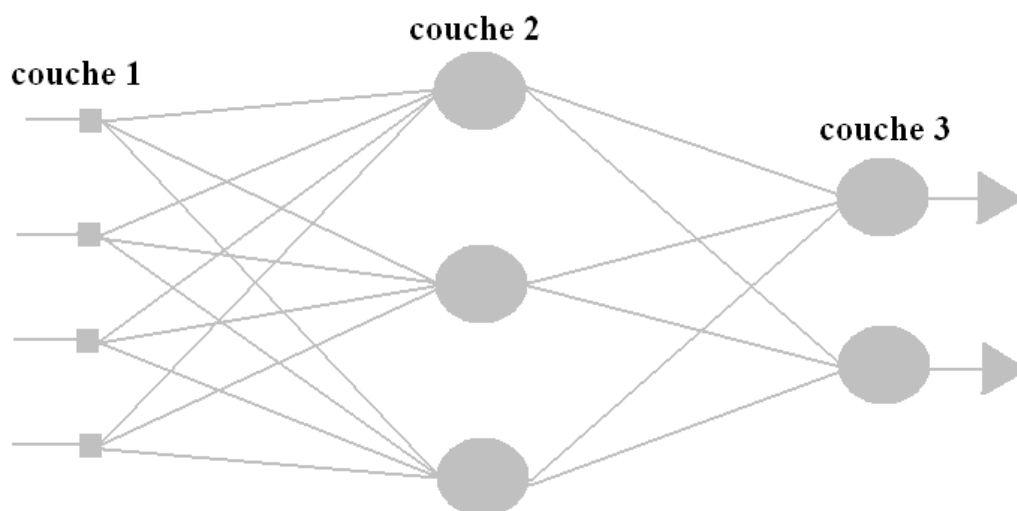


Figure IV.6 Schéma du réseau d'identification NN1.

Le réseau NN1 est constitué de trois couches (figure IV.6):

- La 1^{er} couche (celle des entrées) contient quatre entrées.
- La 2^{ème} couche est une couche cachée qui comporte trois nœuds.
- La 3^{ème} couche (celle des sorties) contient deux sorties.

Les entrées de ce réseaux se sont des couples retardés et non retardés, et ces sorties ce sont les angles approximant ceux qui sont formées par les deux liaisons du bras (Figure IV.5).

L'apprentissage du réseau NN1 se fait selon L'algorithme de rétro propagation du gradient de tel sorte à minimiser l'erreur d'identification « e_i » entre les sorties du modèle identifié (le modèle neuronal NN1) et les sorties du modèle réel du bras q_1 et q_2 .

IV.6 La commande adaptative neuronale

Il existe deux approches de contrôle adaptatif: l'approche directe et l'approche indirecte. Dans le contrôle adaptatif direct, montré en Figure IV.7, « e » est l'erreur de sortie (erreur de suivi) entre la sortie du système $y(k)$ et la sortie du modèle de référence $y_m(k)$: Les paramètres du contrôleur sont directement ajustés pour réduire une norme de l'erreur de sortie.

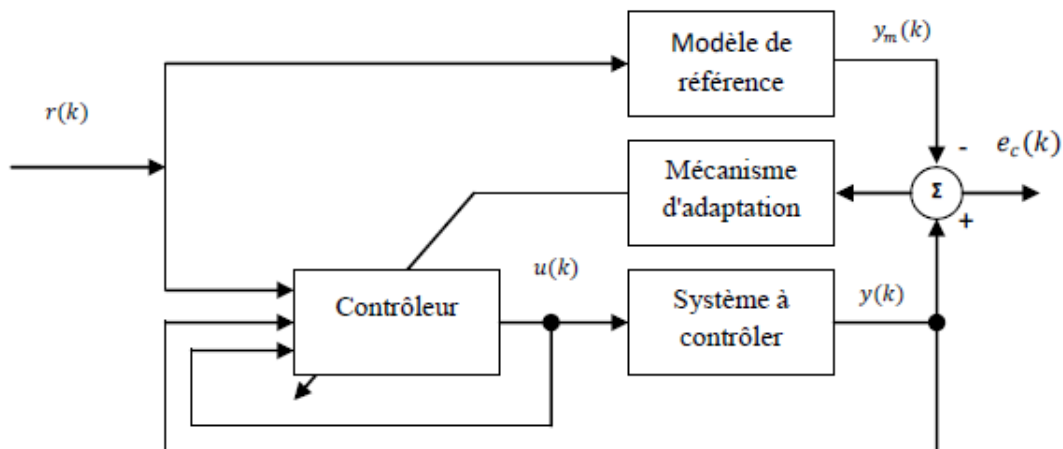


Figure IV.7 Contrôle adaptatif direct.

Dans l'approche indirecte, les paramètres du système à contrôler sont estimés et utilisés ensuite pour déterminer les paramètres du contrôleur ce qui est montré en Figure IV.8

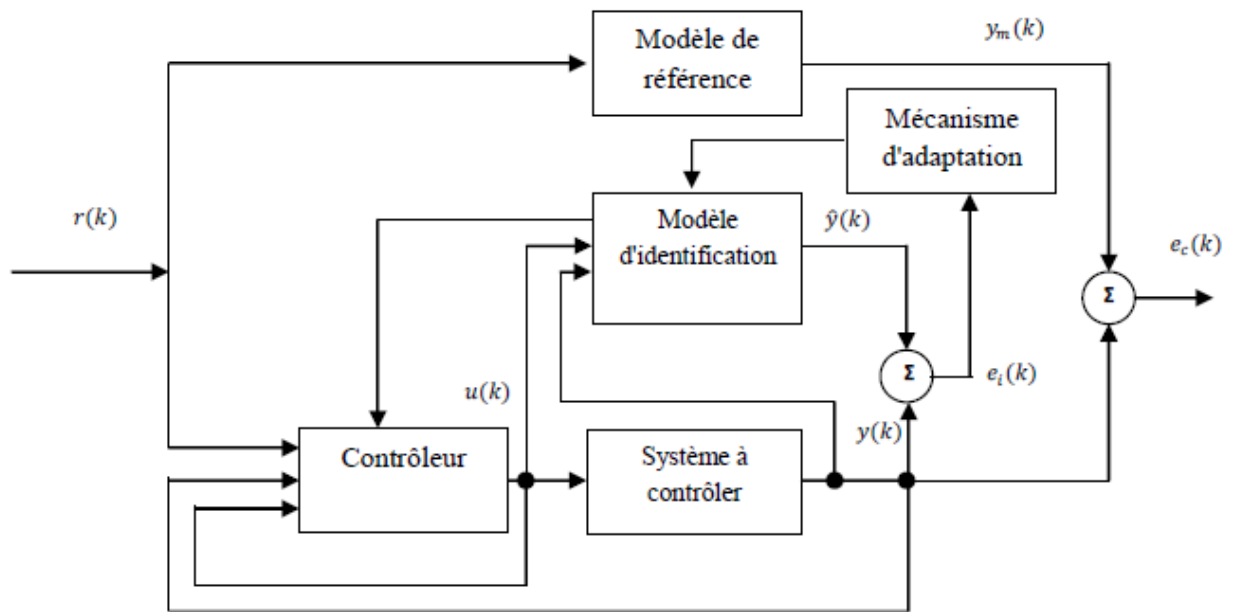


Figure IV.8 Contrôle adaptatif indirect.

IV.7 Application de la commande adaptative neuronale directe au modèle identifié du bras manipulateur

Le schéma de la commande est illustré sur la figure suivante :

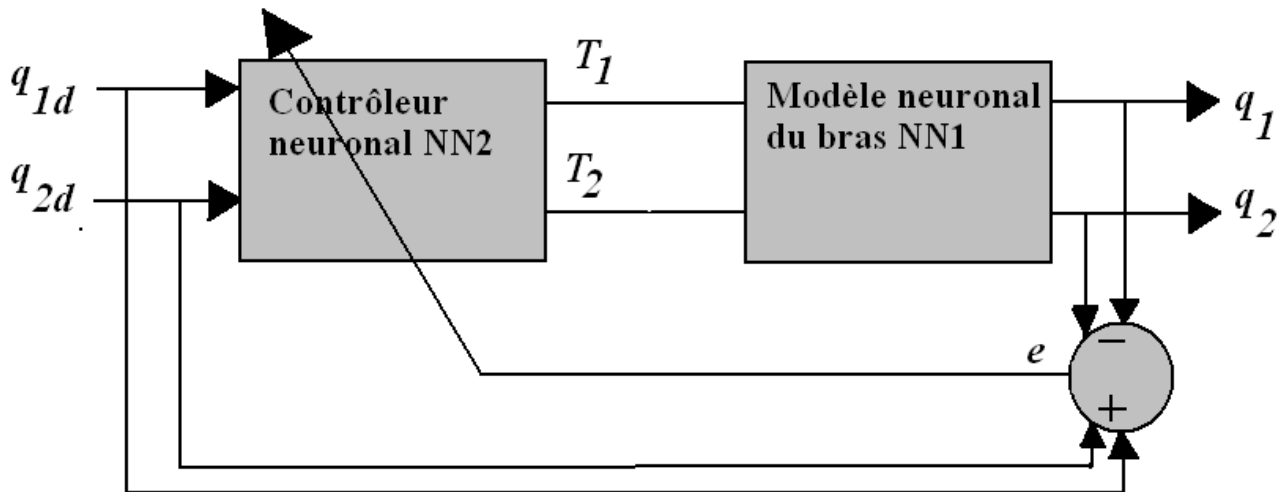


Figure IV.9 Schéma de la commande adaptative neuronale directe du modèle identifié du bras manipulateur.

L'objectif de la commande adaptative neuronale directe est la synthèse d'un contrôleur neuronal (NN2) capable de fournir la commande (les couples) nécessaire pour que les sorties du système NN1 (modèle neuronal du bras) suit les trajectoires de référence q_{1d} et q_{2d} (Figure IV.9).

Sachant que NN2 est un réseaux de neurone MLP (Figure IV.10) constitué de trois couches :

- La 1^{er} couche (celle des entrées) contient deux entrées.
- La 2^{ème} couche est une couche caché qui comporte trois nœuds.
- La 3^{ème} couche (celle des sorties) contient deux sorties.

Les entrées de ce réseaux sont les trajectoires désirées pour les deux liaisons du bras q_{1d} et q_{2d} , et les sorties se sont des couples T_1 et T_2 pour la commande du système.

L'apprentissage de ce réseau se fait selon L'algorithme de rétro propagation du gradient de tel sorte à minimiser l'erreur e entre la sortie du modèle identifié et la trajectoire de référence désiré q_d .

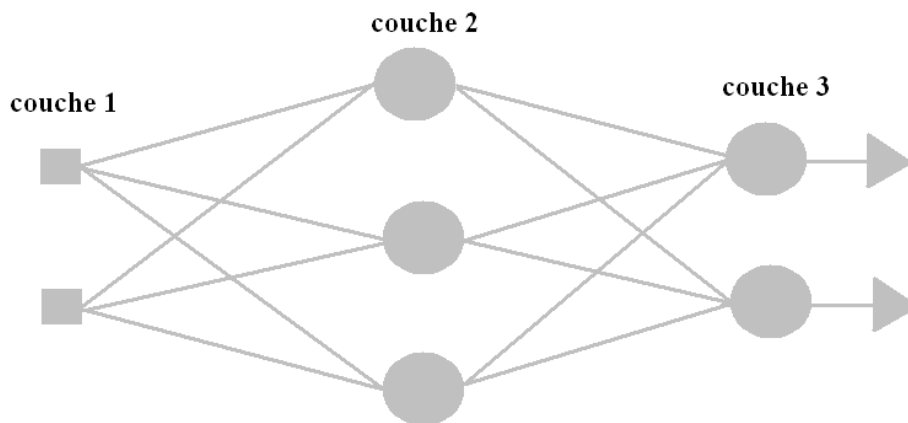


Figure IV.10 schéma du réseau contrôleur NN2.

IV.8 les résultats de la simulation

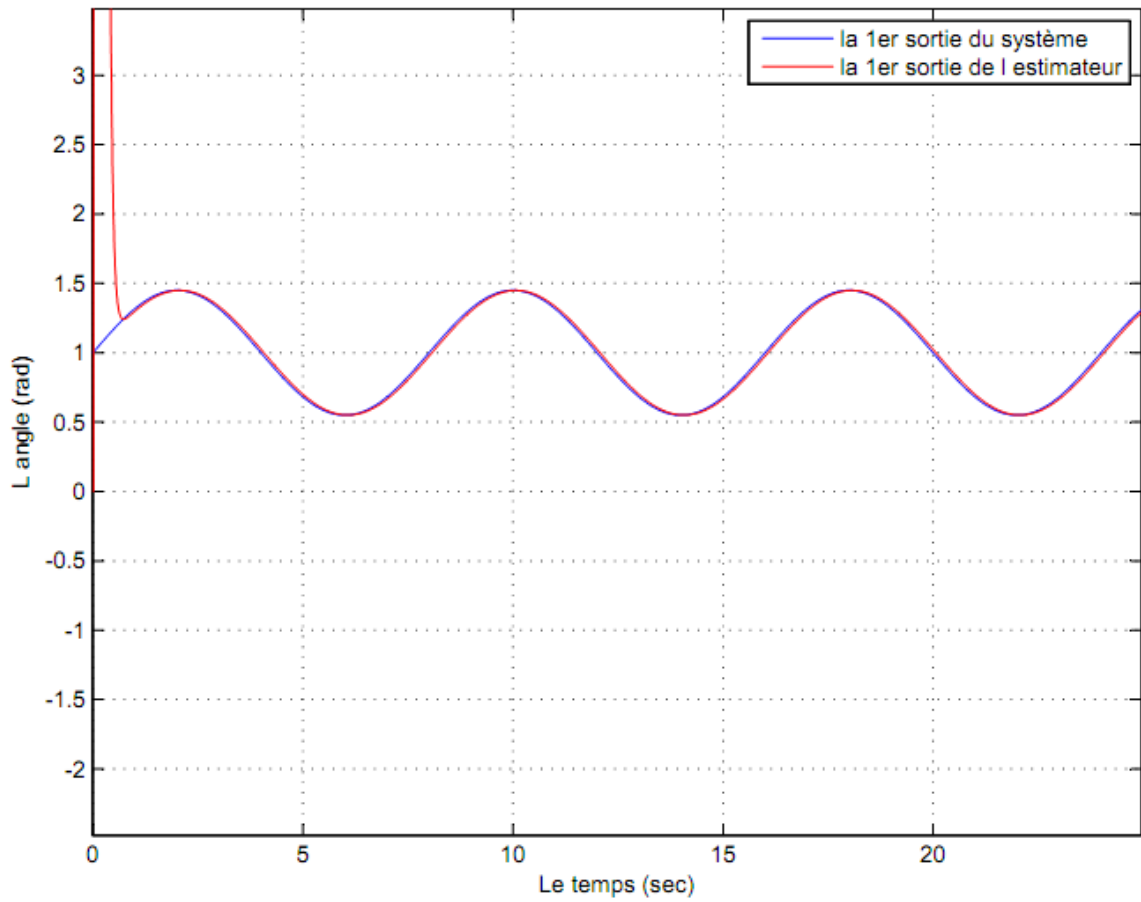


Figure IV.11 Résultats d'identification (1^{er} sortie).

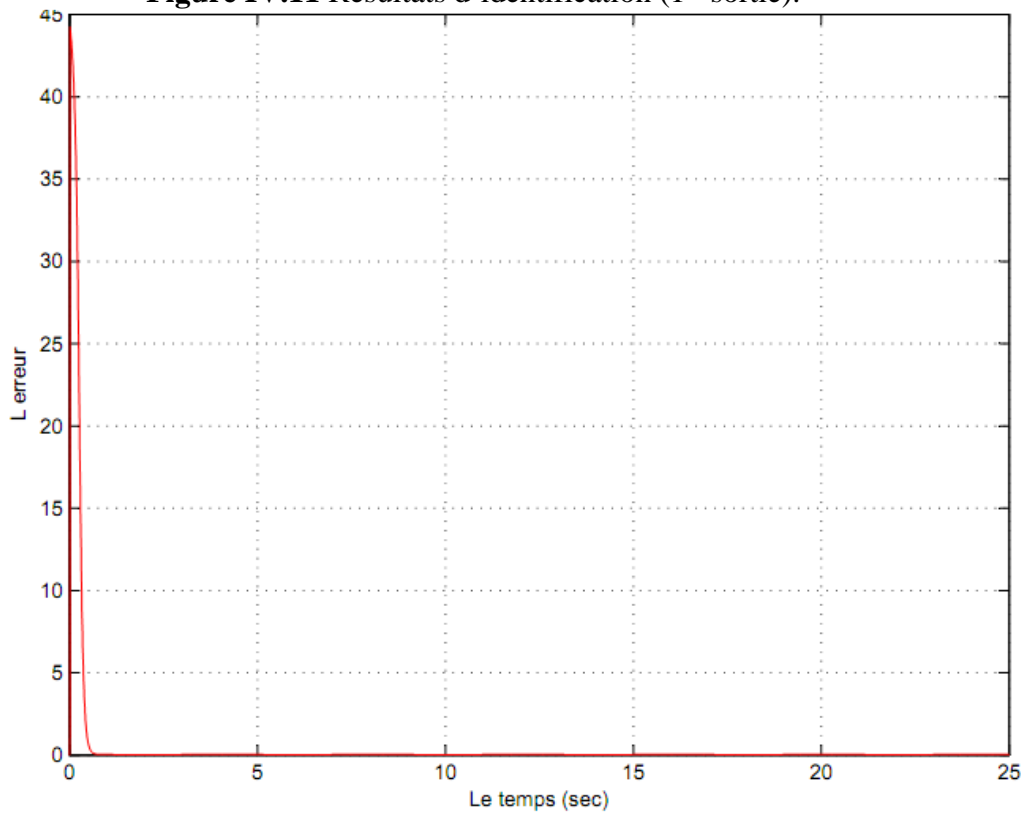


Figure IV.12 L'erreur d'identification pour la 1^{er} sortie.

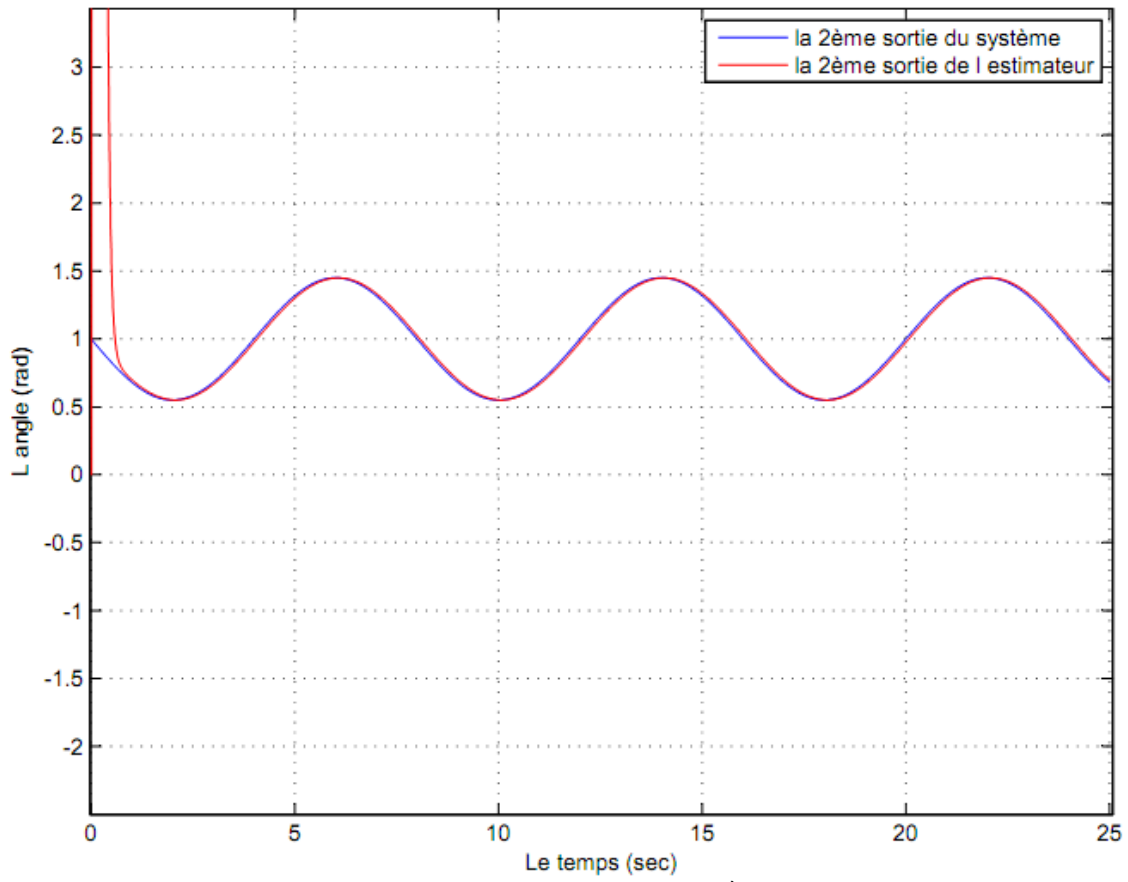


Figure IV.13 Résultats d'identification (2^{ème} sortie).

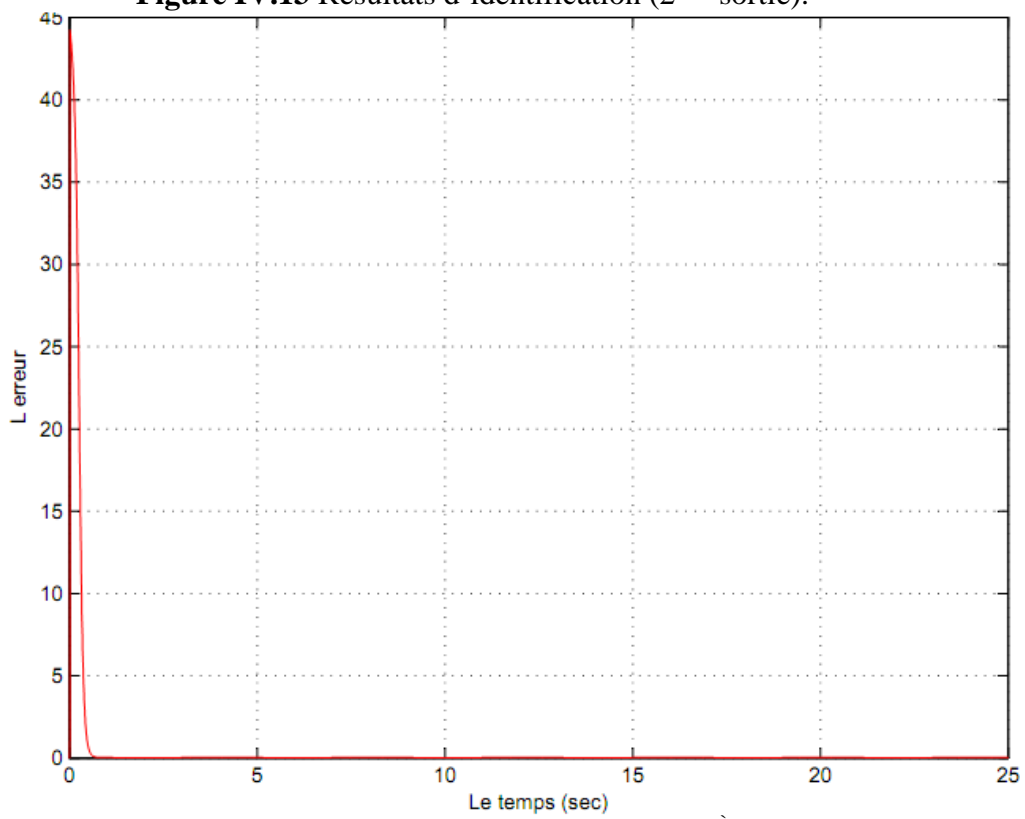


Figure IV.14 L'erreur d'identification pour la 2^{ème} sortie.

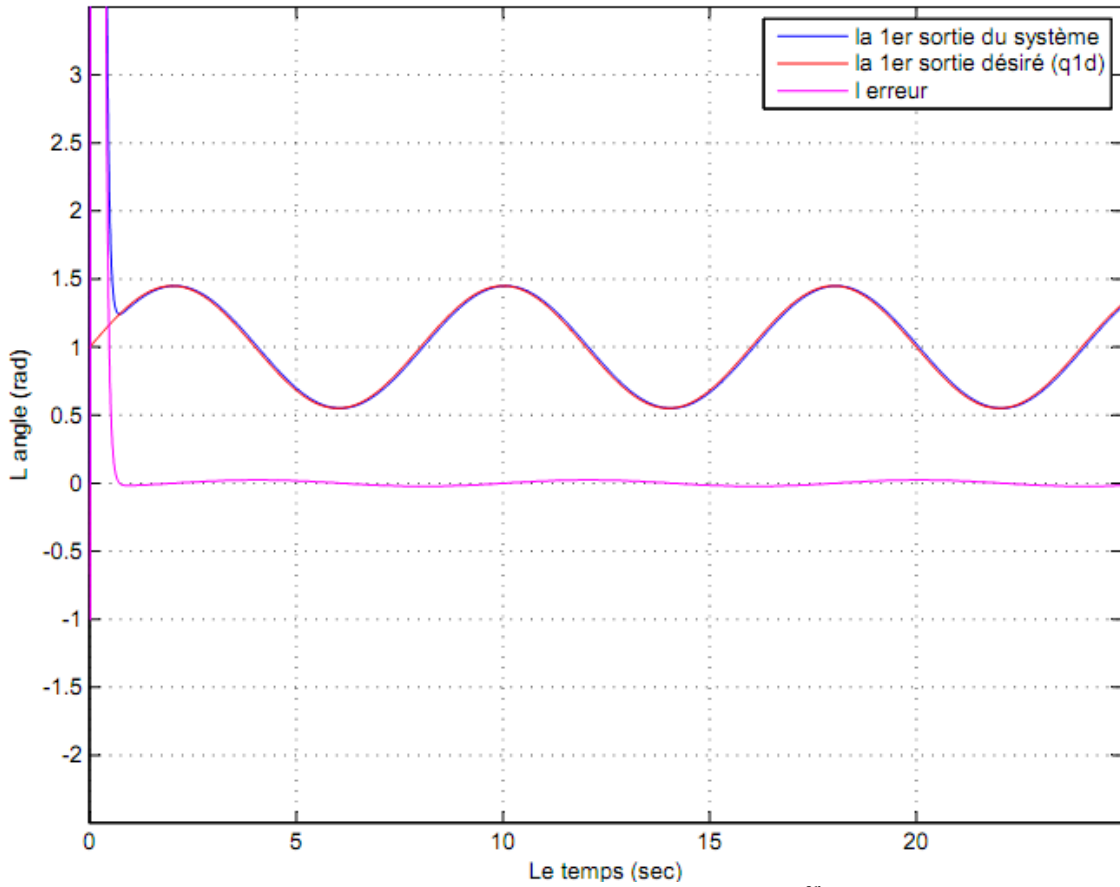


Figure IV.15 Résultat de Commande sur la 1^{er} sortie.

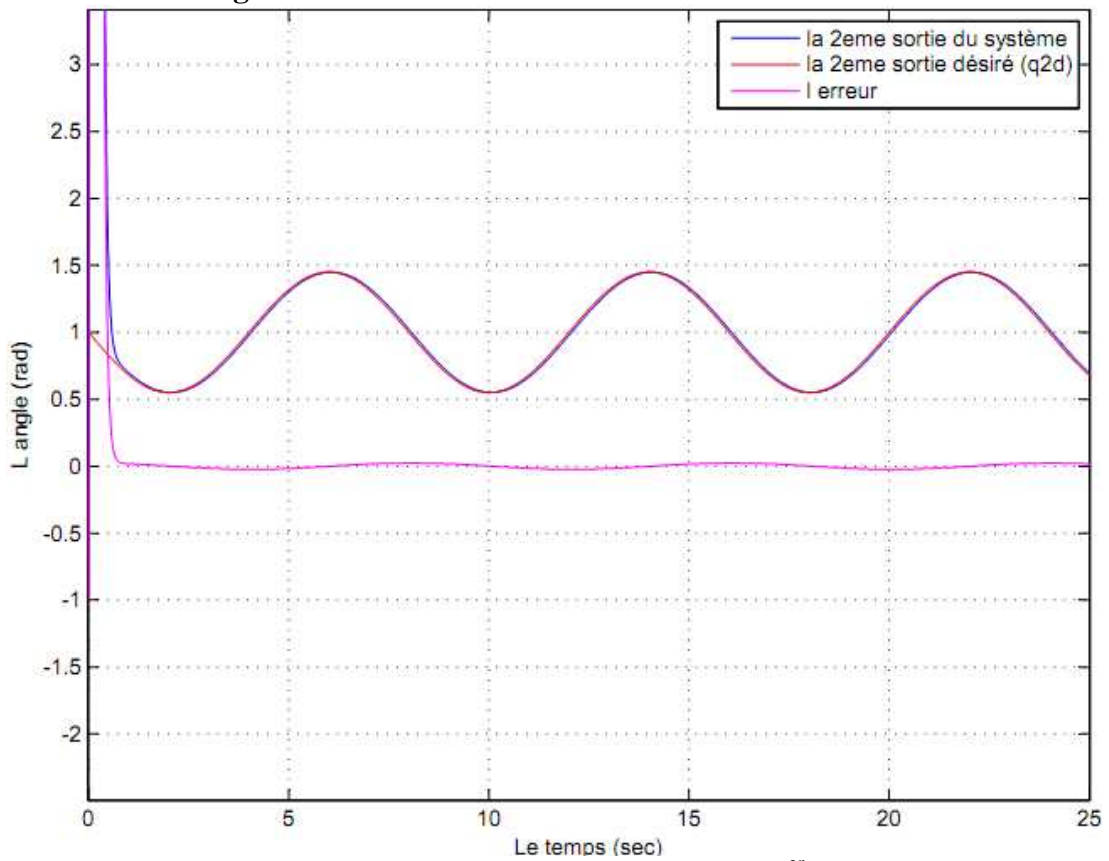


Figure IV.16 Résultat de Commande sur la 2^{er} sortie.

IV.9 Conclusion

La commande intelligente basée sur les réseaux de neurones pour ce système non linéaire nécessite un entraînement permettant une adaptation des poids des neurones des différentes couches du réseau d'identification afin de bien approximer le système (bras manipulateur à deux degrés de liberté), et une bonne adaptation des poids du réseau contrôleur pour une position donnée de l'organe effecteur du bras.

Conclusion générale

Le bras manipulateur flexible étudié dans le présent mémoire est un système non linéaire, du fait que sa modélisation dynamique abouti à des équations différentielles obtenues sous forme d'équations d'état en utilisant la théorie d'Euler-Lagrange, dont les coefficients sont variables suivant les paramètres mécaniques du bras et les configurations des mouvements désirés. Malgré ces complexités, on a pu implémenter un modèle pour une simulation de commande et identification intelligente sous l'environnement Matlab.

L'identification par réseaux de neurones du bras manipulateur nécessite une bonne adaptation des poids du réseau d'identification afin de bien approximer le système.

La commande intelligente basée sur les réseaux de neurones pour le modèle identifié du bras manipulateur nécessite un entraînement en ligne permettant une adaptation des poids des neurones des différentes couches du contrôleur neuronal pour une position donnée de l'organe effecteur.

Références bibliographiques

- [1] Jean Jacques E. Slotine, WEIPING AUTOR LI, «Applied Nonlinear Control», Prentice Hall, 1991.
- [2] GHENAM Abdelatif « Commande Compliante Intelligente d'un Bras Manipulateur Rigide pour des Applications de Chirurgie Médicale », Thèse de magister à l'université de Batna 2012.
- [3] Martin T. Hagan, Howard B. Demuth, Mark H. Beale, « Neural network design », Vikas Publishing House, 2003.
- [4] GUITANI Issam « Commande Adaptative Neuronale par Retour de Sortie des Systèmes Non Linéaires » thèse de magister à l'université de Constantine, 2007
- [5] Hicham CHAOUI, « Conception et comparaison de lois de commande adaptative à base de réseaux de neurones pour une articulation flexible avec non-linéarité dure », Thèse à l'université du QUÉBEC À TROIS-RIVIÈRES, 2002.
- [6] Arato.S.Deo, «Inverse Kinematics and Dynamic Control Methods for Robotic Systems », thèse, Houston .Texas,1995.
- [7] W. Khalil « Commande des robots manipulateurs», Hermes Science Publications, 2002.
- [8] H. Bruyninckx et J. Shutter. « Introduction to intelligent robotics.», Rapport technique, Katholieke Universiteit Leuven 2001.
- [9] H. A. Talebi, F. Abdollahi, R. V. Patel, K. Khorasani, « Neural Network-Based State Estimation of Nonlinear Systems », Springer, New York, 2010.
- [10] K.S. Narendra and K. Parthasarathy, « Identification and control of dynamical systems using neural networks », 1990.
- [11] KS. Narendra, « Neural networks for identification and control », Center for Systems Science, Yale University, Dec. 1998.
- [12] S. Chen and S. A. Billings, « Neural Networks for nonlinear dynamic system modeling and Identification », 1992.
- [13] Ioan D. Landau, « Identification des systèmes », Hermes, Paris, 1998.
- [14] R.Ben Abdenmour, P.Borne, M.ksouri, F.M'sahli,« Automatique: identification et commande numérique des procédés industriels »TECHNIP, Paris, 2001.
- [15] K. Kara, M. L. Hadjili, K. Benmahammed, « Modélisation neuronale des systèmes non linéaires en présence des perturbations » , Conférence Internationale sur les Systèmes de Télécommunications, d'Electronique Médicale et d'Automatique CISTEMA'2003, Tlemcen les 27, 28 et 29 Septembre 2003.
- [16] L. Ljung, « System identification: Theory for the user »,Pearson Education, 1998.
- [17] L. Ljung and T. Soderstrom, « Theory and practice of recursive identification » MIT Press, 1985.
- [18] P. Eykhoff, « System identification: Parameter and state estimation », John Wiley& Sons, USA, 1979.

- [19] MELAKHESSOU Lakhdar, « Contrôle et identification des systèmes non linéaires par les techniques neuronales », Thèse de Magister, Université de Batna.
- [20] I. Abdelmalek, « Identification et commande floues des systèmes non linéaires », Thèse de Magister, Université de Batna 1999.
- [21] R.Haber,L.Keviczky, « Nonlinear system identification. 2. Nonlinear system structure identification », kluwer academic publishers, 1999.
- [22] R.Isermann, M.Münchhof, « Identification of Dynamic Systems : An Introduction with Applications», Springer, 2011.
- [23]Jean-Pierre Richard,«Modélisation et identification des processus » TECHNIP , Paris,1992.
- [24] K. Kara, « Application des réseaux de neurones à l'identification des systèmes non linéaires » , Thèse de Magister, Université de Constantine 1995.
- [25] A. Aouich, « Rejection des perturbations dans les systèmes non linéaires : étude comparative » , Thèse de Magister, Université de M'Sila 2006.
- [26] A. Janczak « Identification of nonlinear systems using neural networks and Polynomial Models », Springer, 2005.
- [27] O.Nelles, «Nonlinear System Identification », Springer, 2001.
- [28] K. Hornik, M. Stinchcombe, H. White, « Multilayer feedforward networks are universal approximators » , Neural Networks vol. 2,1989.
- [29] G.P.Liu, « NONLINEAR IDENTIFICATION AND CONTROL: A Neural Network Approach », Springer, 2001.
- [30] M. Norgaard, O. Ravn, N. K. Poulsen and L. K. Hansen, « Neural networks for modeling and control of dynarnic systems », Springer, London 2000.
- [31] Eric GAUTHIER, « Utilisation des Réseaux de Neurones Artificiels pour laCommande d'un Véhicule Autonome », thèse de docteur à l'Institut national polytechnique de GRENOBLE 1999.
- [32]Patrice WIRA, « Approches neuromimétiques pour l'identification et la commande », thèse en vue d'obtenir L'Habilitation à Diriger Des Recherches à l'université de HAUTE ALSACE.
- [33] R.Kelly, V.Santibáñez and A.Loría, « Control of Robot Manipulators in Joint Space », Springer, 2005.