

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE

UNIVERSTE FERHAT ABBAS –SETIF 1  
UFAS (ALGERIE)

# THESE

Présentée à la Faculté de technologie  
Département d'Electronique

Pour l'obtention du diplôme de

## DOCTORAT EN SCIENCES

Spécialité : **Electronique**

Par

**Mr. Mohamed El Hossine DAACHI**

## THEME

### CONTRIBUTION A LA COMMANDE ET A L'IDENTIFICATION DE ROBOTS A ARCHITECTURES PARALLELES

Soutenue le 13/12/2012 devant la commission d'examen :

Mr. Rais El hadi BEKKA	Professeur à l'université de Sétif 1	Président
Mr. Djamel CHIKOUCHE	Professeur à l'université de M'Sila	Rapporteur
Mr. Hamimi CHEMALI	MCA à l'université de Sétif 1	Examineur
Mr. Kheireddine CHAFAA	MCA à l'université de Batna	Examineur
Mr. Amar MEZACHE	MCA à l'université de M'Sila	Examineur
Mr. Yacine AMIRAT	Professeur à l'université Paris Est	Invité

A la mémoire de mon frère Zahir

## Remerciements

---

J'exprime toute ma gratitude et reconnaissance à mon directeur de thèse Monsieur D. Chikouche, Professeur à l'université de M'sila, pour les précieux conseils qu'il m'a su donnés et l'attention et le sérieux avec lesquels il a suivi de près mes travaux de recherche.

Je remercie chaleureusement Monsieur R. E. Bekka, Professeur à l'université de Sétif1, pour m'avoir fait l'honneur de présider le jury.

Je remercie également Messieurs Hamimi Chemali, Maître de conférences A à l'université de Sétif1, Amar Mezache, Maître de conférences A à l'université de M'sila, Kheireddine Chafaâ, Maître de conférences A à l'université de Batna, pour l'honneur qu'ils me font d'avoir accepté d'examiner mon travail.

Mes vifs remerciements vont à Monsieur Yacine Amirat, Professeur à l'université Paris Est Créteil et directeur du laboratoire LISSI (**L**aboratoire **S**ignaux, **I**mages et **S**ystèmes **I**ntelligents), pour m'avoir accepté dans son laboratoire et m'avoir intégré dans l'équipe de recherche en robotique, ainsi pour ses précieux conseils, son suivi le long de mon séjour au laboratoire et pour l'honneur qu'il me fait en acceptant d'être invité.

Je tiens à remercier tout particulièrement Monsieur B. Daachi, maître de conférences à l'université Paris Est, pour m'avoir guidé tout au long de ma thèse. Je lui suis profondément reconnaissant pour son soutien permanent qu'il m'a manifesté et sans lequel je n'aurais pu mener à bien ce travail.

Je remercie également Monsieur B. Achili, Post-doctorant au laboratoire LISSI pour sa disponibilité et son soutien inestimable manifesté à mon égard.

Merci à S. Mefoued, doctorant à l'université Paris Est, pour sa collaboration et sa sympathie en partageant le même site expérimental.

J'exprime mes plus sincères remerciements à tous mes enseignants qui ont contribué à ma formation tout au long de mon parcours scientifique et à toute personne m'ayant aidé de près ou de loin dans la réalisation de ce travail.

Je ne saurais remercier suffisamment mes parents, mes frères et soeurs et tous mes amis pour leur soutien inestimable. Que tous soient assurés de ma profonde gratitude.

## *Résumé :*

Dans cette thèse, nous avons traité deux aspects relatifs aux systèmes mécatroniques : L'identification et la commande. En effet, les réseaux de neurones du type MLP ont été employés dans les différentes approches proposées dans cette thèse. Notons que le travail réalisé est purement expérimental. Les deux types de systèmes mécatroniques considérés sont le robot parallèle à liaisons C5 et le robot portable de type exosquelette du genou disponibles au laboratoire LISSI. En premier lieu, nous avons réalisé une identification neuronale de type boîte noire de la dynamique inverse du robot parallèle C5. Pour ce faire, trois schémas d'identification ont été testés et comparés. Concernant la commande, nous avons proposé une commande adaptative hybride moment/position du robot parallèle à liaison C5 directement dans l'espace de la tâche. Le modèle dynamique du robot en contact avec son environnement dans l'espace de la tâche considéré comme une boîte noire est estimé en ligne en utilisant un réseau de neurones de type MLP. Les paramètres du réseau de neurones sont adaptés en ligne via une étude de stabilité en boucle fermée au sens de Lyapunov. Une autre approche de commande neuronale est également proposée pour commander le robot portable. Dans la conception de cette approche de commande adaptative, le modèle dynamique est pris comme une boîte grise c.-à-d, seule la structure du système est connue. Les fonctions inconnues du modèle sont approximées en ligne via une analyse de stabilité en boucle fermée au sens de Lyapunov. L'approche proposée est testée sur une personne saine assise dans un mouvement de flexion/extension du genou.

***Mots clés :*** Identification, commande adaptative, réseaux de neurones, robot parallèle, exosquelette, stabilité.

---

# Table des matières

---

Liste des figures	vi
Introduction Générale	1
<b>1 Aperçu sur la commande de robots parallèles et de robots portables</b>	<b>5</b>
1.1 Introduction . . . . .	5
1.2 Robots à architecture parallèle . . . . .	6
1.2.1 Définition . . . . .	6
1.2.2 Quelques exemples de robots parallèles . . . . .	7
1.2.3 Le robot parallèle à liaisons C5 et son environnement . . . . .	9
1.2.4 Robots parallèles vs robots sériels . . . . .	11
1.2.5 Commande en position . . . . .	12
1.2.6 Commande hybride . . . . .	14
1.3 Robots portables : exosquelettes/orthèses . . . . .	16
1.3.1 Définition . . . . .	16
1.3.2 Exemples d'exosquelettes . . . . .	17
1.3.3 L'exosquelette développé au LISSI . . . . .	18
1.3.4 Commande des exosquelettes . . . . .	19
1.4 Systèmes de commande et dSPACE . . . . .	21
1.4.1 La carte dSPACE DS 1103 . . . . .	23
1.4.2 Caractéristiques techniques de la carte dSPACE DS 1103 . . . . .	23
1.5 Conclusion . . . . .	24

<b>2</b>	<b>Identification de la dynamique inverse du robot parallèle à liaisons</b>	
	<b>C5</b>	<b>25</b>
2.1	Introduction . . . . .	25
2.2	Aperçu sur les méthodes d'identification . . . . .	26
2.2.1	Méthodes ensemblistes pour l'identification . . . . .	28
2.2.2	Méthodes d'apprentissage pour l'identification . . . . .	29
2.2.3	Moindres carrées pour l'identification . . . . .	29
2.3	Problématique d'identification neuronale considérée . . . . .	33
2.4	Architectures neuronales et identification de la dynamique inverse . .	34
2.4.1	Génération de trajectoire pour l'identification . . . . .	34
2.4.2	Filtrage des positions mesurées et calcul des vitesses et des accélérations . . . . .	37
2.4.3	Présentation de la méthode d'identification . . . . .	37
2.5	Résultats expérimentaux . . . . .	40
2.6	Conclusion . . . . .	44
<b>3</b>	<b>Commande hybride moment/position du robot parallèle à liaisons</b>	
	<b>C5</b>	<b>46</b>
3.1	Introduction . . . . .	46
3.2	Formulation du problème . . . . .	47
3.2.1	Approximation neuronale et traitement des non linéarités . . .	50
3.3	Commande hybride moment/position pour le suivi de contour . . . .	53
3.3.1	Loi de commande et analyse de stabilité . . . . .	53
3.3.2	Analyse de stabilité . . . . .	55
3.4	Validation expérimentale . . . . .	56
3.5	Conclusion . . . . .	64
<b>4</b>	<b>Commande adaptative d'un robot portable de type exosquelette du genou</b>	<b>65</b>
4.1	Introduction . . . . .	65

---

4.2	Modèle dynamique du système . . . . .	66
4.3	Approximation neuronale . . . . .	67
4.4	Loi de commande et étude de stabilité . . . . .	68
4.4.1	Analyse de la stabilité lorsque le porteur de l'orthèse est passif	71
4.4.2	Analyse de stabilité lorsque le porteur de l'orthèse est actif . .	73
4.5	Résultats expérimentaux . . . . .	74
4.5.1	Le porteur de l'orthèse est passif ( $\tau_h = 0$ ) . . . . .	75
4.5.2	Le porteur de l'orthèse est actif ( $\tau_h \neq 0$ ) . . . . .	79
4.6	Conclusion . . . . .	82
<b>Conclusion et perspectives</b>		<b>83</b>
<b>A Analyse des systèmes par la méthode de Lyapunov</b>		<b>86</b>
A.1	Théorème de Lyapunov pour la stabilité des systèmes non-autonomes	86
A.2	Systèmes linéaires positifs . . . . .	87
<b>B Les réseaux de neurones à couches</b>		<b>89</b>
B.1	Les perceptrons multicouches (MLP) . . . . .	89
B.2	Algorithme d'optimisation . . . . .	90
<b>Références bibliographiques</b>		<b>92</b>

---

# Liste des figures

---

1.1	Hexapode ou plate-forme de Gough-Stewart. . . . .	7
1.2	Robot parallèle Hexa-M de Toyota. . . . .	7
1.3	Robot Delta FlexPicker . . . . .	8
1.4	Robot H4 de LIRMM . . . . .	9
1.5	Robot parallèle I4L . . . . .	9
1.6	Robot parallèle I4R . . . . .	10
1.7	Site expérimental du robot C5 . . . . .	10
1.8	(a) Représentation du robot parallèle à liaison C5. (b) Liaison C5. . .	11
1.9	Les tous premiers exosquelettes. . . . .	17
1.10	L'exosquelette Hercule . . . . .	18
1.11	Exosquelette de la société japonaise Kawasaki. . . . .	18
1.12	Exosquelette du LISSI . . . . .	19
1.13	Architecture matérielle de commande des deux systèmes robotisés considérés. . . . .	22
1.14	Catre dSPACE DS 1103 utilisée pour la commande de systèmes ro- botiques considérés. . . . .	23
2.1	Principe de la validation croisée. . . . .	32
2.2	Trajectoires excitantes pour les articulations 1 et 2. . . . .	35
2.3	Trajectoires excitantes pour les articulations 3 et 4. . . . .	36
2.4	Trajectoires excitantes pour les articulations 5 et 6. . . . .	36
2.5	Organigramme d'identification. . . . .	37
2.6	Premier schéma d'identification. . . . .	38
2.7	Deuxième schéma d'identification. . . . .	38

2.8	Troisième schéma d'identification. . . . .	39
2.9	Tableau comparatif . . . . .	40
2.10	Erreur de validation (généralisation) - axe 1 du robot. . . . .	41
2.11	Erreur de validation (généralisation) - axe 2 du robot. . . . .	41
2.12	Erreur de validation (généralisation) - axe 3 du robot. . . . .	42
2.13	Erreur de validation (généralisation) - axe 4 du robot. . . . .	42
2.14	Erreur de validation (généralisation) - axe 5 du robot. . . . .	43
2.15	Erreur de validation (généralisation) - axe 6 du robot. . . . .	43
3.1	Robot parallèle en contact avec l'environnement. . . . .	47
3.2	Démarche de commande expérimentale. . . . .	51
3.3	Schéma de commande hybride moment/position . . . . .	55
3.4	Résultat expérimental de la poursuite en moment . . . . .	58
3.5	Erreur de poursuite en moment . . . . .	59
3.6	Résultat expérimental de la poursuite en position dans le plan xy . . . . .	59
3.7	Erreur de poursuite en position dans le plan xy . . . . .	59
3.8	Sortie 1 du réseau de neurones . . . . .	60
3.9	Sortie 2 du réseau de neurones. . . . .	60
3.10	Sortie 3 du réseau de neurones . . . . .	60
3.11	Couple mesuré pour l'axe 1 du robot(Approche neuronale) . . . . .	61
3.12	Couple mesuré pour l'axe 2 du robot (Approche neuronale) . . . . .	61
3.13	Couple mesuré pour l'axe 3 du robot(Approche neuronale). . . . .	61
3.14	Couple mesuré pour l'axe 4 du robot ( Approche neuronale). . . . .	62
3.15	Couple mesuré pour l'axe 1 du robot (Contrôleur PID). . . . .	62
3.16	Couple mesuré pour l'axe 2 du robot (Contrôleur PID). . . . .	62
3.17	Couple mesuré pour l'axe 3 du robot (Contrôleur PID). . . . .	63
3.18	Couple mesuré pour l'axe 4 du robot (Contrôleur PID). . . . .	63
4.1	Commande neuronale adaptative de l'exosquelette. . . . .	69
4.2	Poursuite de trajectoire en position (Approche neuronale). . . . .	76
4.3	Poursuite de trajectoire en position (PID). . . . .	76

---

4.4	Erreur de poursuite de trajectoire en position. . . . .	77
4.5	Poursuite de trajectoire en vitesse (Approche neuronale). . . . .	77
4.6	Poursuite de trajectoire en vitesse (PID). . . . .	77
4.7	Couple de commande (Approche neuronale). . . . .	78
4.8	Couple de commande (PID). . . . .	78
4.9	Sortie du réseau de neurones. . . . .	78
4.10	Inertie (Approche neuronale). . . . .	79
4.11	Poursuite de trajectoire en position . . . . .	79
4.12	Erreur de poursuite de trajectoire en position . . . . .	80
4.13	Poursuite de trajectoire en vitesse . . . . .	80
4.14	Erreur de poursuite en vitesse . . . . .	81
4.15	Sortie du réseau de neurones . . . . .	81
4.16	Inertie estimée. . . . .	81
B.1	Structure générale d'un RNF non linéairement paramétré (MLP). . .	89

---

# Introduction Générale

---

De nos jours, les robots manipulateurs ont pleinement trouvé leur place dans la vie quotidienne de l'homme. Qu'ils soient à structure parallèle [Pacc 09][Achi 09a] ou sérielle [Lewi 04], les robots manipulateurs sont aujourd'hui largement déployés. Dans le domaine médical par exemple, ils sont utilisés pour l'assistance au geste chirurgical afin d'en améliorer la précision et également pour la rééducation des membres inférieurs et/ou supérieurs de l'homme (sous forme d'exosquelettes). Par ailleurs et pour ne citer que cela, ils ont aussi permis de soulager les opérateurs humains des tâches les plus pénibles, fréquentes et présentes dans les usines, les ports, etc. Selon la nature de la tâche à réaliser et les performances recherchées (précision, rapidité, dextérité, évitement d'obstacles, etc.), nous avons la possibilité de choisir entre un robot à architecture sérielle et un robot à architecture parallèle. Dans le cas de tâches nécessitant une grande dextérité et un espace de travail important, un robot à architecture sérielle est préféré à un robot à architecture parallèle. En revanche, lorsque la vitesse à laquelle une tâche doit être réalisée et sa répétabilité sont prédominantes, le choix est orienté vers une architecture parallèle plutôt que vers une architecture sérielle. Les exigences faites à ces robots sont diverses et variées et en conséquence leurs commandes sont appelées à évoluer en permanence. Les travaux de recherche se sont saisis de ce sujet et plusieurs études et réalisations ont été menées. Les commandes sont souvent jugées par leur capacité à s'adapter aux changements possibles de la dynamique du robot, de la tâche et de l'espace de travail. Des points comme les connaissances a priori et les hypothèses possibles sur le système ainsi que la présence de perturbations externes, sont en général déterminants dans le processus de conception de la commande.

La commande de robots à structure parallèle a fait l'objet de plusieurs travaux de recherche et de plusieurs validations expérimentales[Viva 04][Krut 03]. Des outils de l'intelligence artificielle ont été considérés pour l'élaboration des lois de commande, notamment les réseaux de neurones [Achi 12] et la logique floue [Khou 08]. On peut aussi rencontrer des commandes basées sur les modes de glissements etc. Chaque type de commande proposée possède ses caractéristiques en termes d'avantages (par exemple nombre réduit d'hypothèses sur le comportement dynamique du système) et d'inconvénients (par exemple stabilité du système en boucle fermée non assurée).

D'une façon générale, selon la nature de la tâche à réaliser, le robot peut être en mouvement libre comme il peut se trouver en contact avec son environnement, par exemple l'assemblage, l'usinage (ébavurage, ébarbage,...), le suivi de surfaces (polissage,...). Dans le premier cas, la commande en mouvement (position) toute seule est suffisante pour contrôler le robot, cependant, elle ne sera pas suffisante toute seule dans le cas où le robot entre en contact avec son environnement. Ceci est dû au fait qu'une commande d'un robot en mouvement libre, parfaitement stable, peut devenir instable en présence d'un contact (apparition de rebonds et détérioration de l'effecteur ou de l'environnement) d'où la nécessité de commander le robot également en effort. Dans ce contexte, deux types de commande ont été proposées dans la littérature: la commande en impédance et la commande hybride force/position. Dans le second type de commande, il est plus facile de réaliser l'étude de stabilité en boucle fermée inversement au premier type. Toujours concernant les robots manipulateurs, la plus part des commandes sont réalisées dans l'espace articulaire alors qu'il est plus pratique d'établir des lois de commande dans l'espace de la tâche afin de manipuler réellement des objets directement dans cet espace.

Par ailleurs, lorsque le robot se trouve en interaction directe avec un individu ou tout simplement porté par celui-ci, d'autres dynamiques doivent être considérées pour que la commande assure la réalisation d'une tâche bien précise. Ces dynamiques sont par exemples provoquées par les muscles du porteur. Concernant ce travail de thèse, nous avons pris le problème du contrôle/commande de ces robots portables

(exosquelettes ou orthèses) simplement comme un problème robotique. Tout l'aspect clinique n'a pas été traité.

L'identification et la commande sont les deux aspects que nous avons traités dans cette thèse. Le premier système mécatronique considéré est le robot parallèle à liaisons C5 développé au laboratoire LISSI<sup>1</sup>. Dans un second temps nous sommes intéressés à la commande en position et en vitesse de l'exosquelette du genou disponible également au laboratoire LISSI. Ainsi, nous pouvons résumer nos contributions dans les points suivants:

- Identification neuronale expérimentale du modèle dynamique inverse du robot parallèle à liaisons C5. L'identification réalisée est du type boîte noire (aucune connaissance n'est considérée sur le modèle dynamique du robot). Pour ce faire, trois schémas d'identification ont été testés [Daac 08].
- Commande hybride moment/position : la loi de commande proposée est élaborée directement dans l'espace de la tâche. Le modèle du robot est pris comme une boîte noire, estimé par un réseau de neurones du type MLP. L'approche de commande proposée a été testée sur le robot parallèle à liaisons C5 en contact avec son environnement et réalisant un suivi de contour. La stabilité en boucle fermée au sens de Lyapunov est vérifiée [Daac 12].
- Commande neuronale adaptative d'un robot portable de type exosquelette de genou. Il est à noter que le prototype d'exosquelette est destiné à la rééducation du membre inférieur et à l'assistance des personnes à mobilité réduite. Notre approche de commande est testée sur une personne saine. La trajectoire à suivre est proposée dans le sens de la rééducation du membre inférieur.

Nous avons choisi d'organiser ce travail de thèse en quatre chapitres suivis d'une conclusion et des perspectives.

Le chapitre 1 dresse un panorama sur les robots à structure parallèle et les robots portables de type exosquelette ainsi que les commandes ayant été appliquées dans

---

<sup>1</sup>Laboratoire **S**ignaux, **I**mages et **S**ystèmes **I**ntelligents. Université Paris-Est Créteil.

la littérature sur le même genre de systèmes. Les deux systèmes mécatroniques sur lesquels ont été menés nos manipulations, à savoir le robot parallèle à liaisons C5 et l'exosquelette destiné à la rééducation, sont également présentés dans ce chapitre. Le chapitre 2 est consacré à l'identification neuronale du modèle dynamique inverse du robot parallèle à liaisons C5. Les réseaux de neurones utilisés sont du type MLP. Il est à noter qu'aucune connaissance a priori n'est considérée sur le modèle dynamique du système.

Dans le chapitre 3, nous présentons une commande neuronale adaptative hybride moment/position du robot parallèle C5 dans l'espace de la tâche. Par son effecteur, le robot se trouve en contact avec son environnement lors de la réalisation de la tâche qui lui est assignée. Le modèle dynamique du robot est considéré comme une boîte noire. La stabilité en boucle fermée au sens de Lyapunov est garantie ainsi que la robustesse face aux perturbations externes et aux erreurs d'identification neuronale en ligne. L'approche proposée est traitée sur un environnement rigide.

Le quatrième chapitre quant à lui est réservé à l'implémentation d'une commande neuronale adaptative sur l'exosquelette. Pour ce système nous considérons la connaissance de la structure du modèle dynamique. Deux parties se distinguent dans le modèle. La première est liée à l'inertie et est identifiée de manière classique en utilisant la stabilité du système en boucle fermée. La deuxième est liée aux frottements secs et visqueux et à la force de gravité ainsi qu'aux autres dynamiques non modélisées. Nous terminons notre manuscrit par une conclusion et des perspectives.

---

# Chapitre 1

## Aperçu sur la commande de robots parallèles et de robots portables

---

### 1.1 Introduction

Dans ce chapitre, nous allons nous intéresser aux deux systèmes mécatroniques sur lesquels nous avons appliqué des schémas de commande. Le premier système est un robot à architecture parallèle et le deuxième est représenté par un robot destiné à être porté par un individu. Le rôle du premier système est de faire des tâches de manipulation d'objets en général et des tâches d'assemblage en particulier. Quant au deuxième système, il peut être à la fois utilisé pour l'assistance à des personnes dépendantes et pour la rééducation des membres inférieurs. Ce robot portable a une structure basique mais il est appelé à évoluer pour couvrir plus de degrés de liberté et offrir davantage de services. Par ailleurs, ce chapitre a également pour objectif de faire un panorama sur les commandes existantes dans la littérature et qui ont été appliquées au même type de systèmes mécatroniques. Les inconvénients et les avantages des schémas de commande retenus, sont également évoqués. Comme nous pensons qu'il est intéressant, pour le lecteur, d'avoir quelques informations sur d'autres robots (portables et manipulateurs parallèles), nous en avons sélectionné quelques-uns. Les domaines d'applications de ces robots choisis de la littérature, sont aussi succinctement décrits. Notre intérêt consiste à connaître les différentes

architectures existantes ainsi que les applications auxquelles elles sont destinées, afin d'enrichir nos perspectives et de situer un peu plus clairement nos idées à court et à long termes. Trois aspects nous paraissent très importants pour toute application robotique réussie à savoir la conception mécanique, la commande et l'étude de l'environnement dans lequel est sensé évoluer le robot. Par ailleurs, nous avons jugé utile de décrire l'architecture matérielle, représentée essentiellement par la carte dSPACE, ayant servi pour nos expérimentations dans les chapitres suivants [Achi 09a].

## 1.2 Robots à architecture parallèle

### 1.2.1 Définition

Un robot à architecture parallèle est un mécanisme en chaîne cinématique fermée dont l'organe terminal est relié à la base par plusieurs chaînes cinématiques indépendantes (figure 1.1). Les principaux avantages des robots parallèles se résument dans les points suivants :

- Possibilité de transporter des charges lourdes
- Précision
- Rapidité et souplesse

Les principaux inconvénients des robots parallèles se résument dans les points suivants

- Volume de travail limité et complexe
- Complexité de la commande
- Modèle géométrique et cinématiques directs difficiles à obtenir



Figure 1.1: Hexapode ou plate-forme de Gough-Stewart.

### 1.2.2 Quelques exemples de robots parallèles

Nous avons choisi de présenter cinq types de robots parallèles qui sont largement cités dans la littérature<sup>1</sup>. Le premier robot est une sorte de machine-outil à structure parallèle connue sous le nom de "Hexa-M de Toyota" (fig. 1.2). Ce robot a six degrés de liberté avec des actionneurs linéaires. C'est un robot qui s'apprête parfaitement bien pour manipuler des charges lourdes. Le deuxième robot est relativement léger

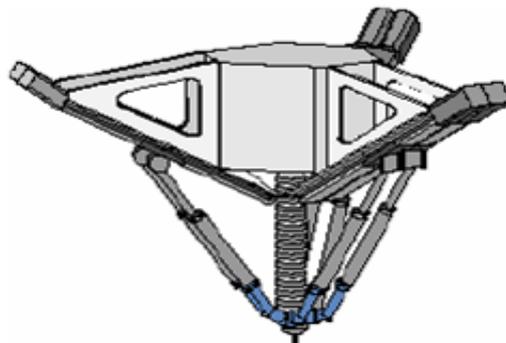


Figure 1.2: Robot parallèle Hexa-M de Toyota.

et très dynamique. Il peut atteindre des vitesses et des accélérations élevées. Sa

---

<sup>1</sup>Notre analyse est faite en collaboration avec B. Achili dont la thèse a été soutenue le 04 décembre 2009.

structure est de type Delta et ses actionneurs sont fixes à la base (fig. 1.3). Les domaines d'application de ce type de robot peuvent être en médecine, en agro-alimentaire ou encore dans l'usinage de haute précision, etc.



Figure 1.3: Robot Delta FlexPicker

Parmi les machines appartenant à la famille des robots légers, on peut citer le robot H4 (fig. 1.4). Il a été conçu au LIRMM<sup>2</sup> et il a quatre degrés de liberté. Ce robot est capable de réaliser des déplacements de la nacelle à une vitesse allant jusqu'à  $6\text{ m/s}$  avec une accélération maximale de  $130\text{ m/s}^2$ . Il peut être utilisé pour des opérations pick-and-place (prise et pose d'un objet). Son architecture mécanique est présentée dans [Viva 04]. Malgré les meilleures performances dynamiques que présente le robot H4, il reste sujet à une perte de rigidité à cause d'éventuelles collisions entre ses chaînes cinématiques. En outre, l'impossibilité de placer ces parallélogrammes à  $90$  degré les uns par rapport aux autres. Ce qui a conduit au développement d'une autre famille de robot appelée I4 à 4 degrés de liberté. On distingue deux types de robots dans cette famille, I4L et I4R (figures 1.5 et 1.6) à actionneurs linéaires et rotatifs respectivement. Le robot I4L est bien adapté à la manutention d'objets dans les chaînes d'assemblage à grande vitesse cependant, le robot I4R est destiné aux opérations pick and place. Cette nouvelle famille a

---

<sup>2</sup>Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier.

conduit à l'introduction d'une nouvelle nacelle équipée d'un système de transformation de mouvement (engrenages et pignons-crémaillères) adapté à réaliser une seule rotation. En revanche, dans le cas de deux rotations autour de deux axes or-



Figure 1.4: Robot H4 de LIRMM

thogonaux successifs, ce dispositif est jugé insuffisant. Une solution a été proposée dans [Krut 02] pour étendre le mouvement à deux rotations. Il s'agit d'un dispositif Twice inspiré du modèle de Zamanov. La particularité du mécanisme Twice est l'absence de singularité dans son espace de travail [Krut 03][Merl 01].



Figure 1.5: Robot parallèle I4L

### 1.2.3 Le robot parallèle à liaisons C5 et son environnement

Le site expérimental dans lequel se sont déroulées nos manipulations est constitué de deux éléments : le robot parallèle à liaisons C5 et le robot portique (fig. 1.7).



Figure 1.6: Robot parallèle I4R

Le robot parallèle (fig. 1.8-a), considéré comme un outil terminal, joue le rôle d'un poignet actif par rapport au portique. Le robot parallèle à liaisons C5 possède six

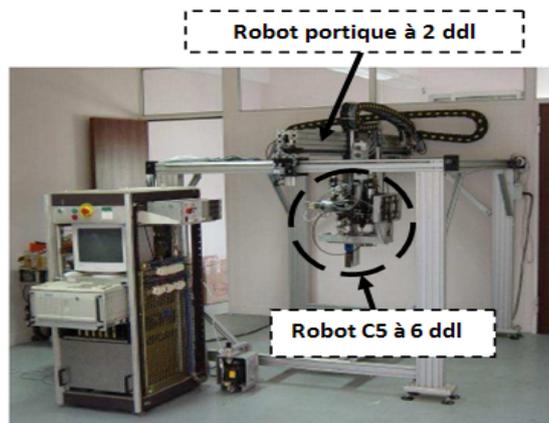


Figure 1.7: Site expérimental du robot C5

degrés de liberté. Il est constitué d'une partie fixe et d'une partie mobile. La base et l'organe terminal sont liés entre eux par un ensemble de six segments motorisés et disposés de façon cartésienne. Chaque segment est encastré dans la partie fixe au point  $A_i$  et relié à la partie mobile par une liaison C5 (trois rotations et deux translations). Une liaison C5 est composée d'une rotule couplée à deux tables croisées au point  $B_i$  qui réalisent deux translations orthogonales (fig. 1.8-b). L'étude théorique

de cette architecture a été proposée dans [Dafa 98]. Les rotules équipant les extrémités des vérins servent de lien entre les deux bâtis. Chaque vérin est équipé d'une vis à billes d'un pas de 5 mm. La liaison entre la sortie du réducteur et la vis à billes est assurée par une courroie crantée.

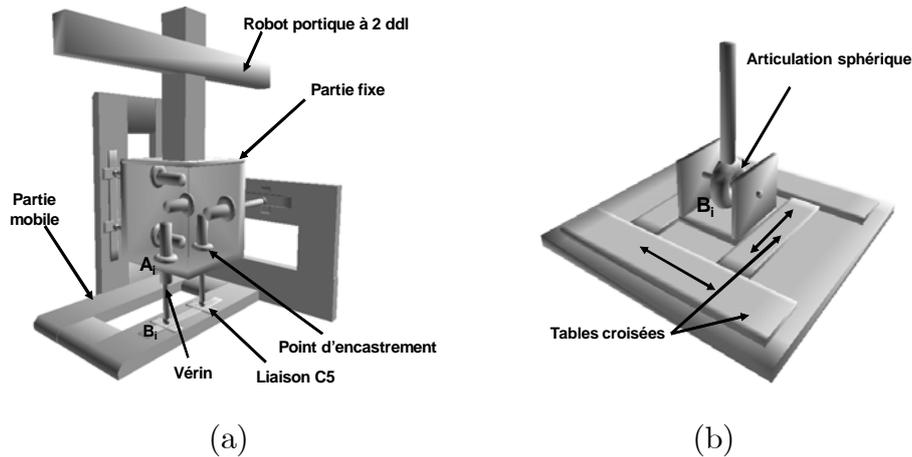


Figure 1.8: (a) Représentation du robot parallèle à liaison C5. (b) Liaison C5.

#### 1.2.4 Robots parallèles vs robots sériels

Les limitations liées à l'utilisation des robots sériels, ont conduit vers une nouvelle classe de mécanismes structurellement différente dans laquelle l'effecteur est relié à la base par au moins deux chaînes cinématiques indépendantes, ce qui constitue une chaîne cinématique fermée. Ce type de mécanisme est connu sous le nom de robot parallèle. Le fait que les erreurs de positionnement relatives aux actionneurs ne soient pas cumulées, permet plus de précision quant au positionnement de l'effecteur de ce type de robot. Par ailleurs, les effets liés à l'inertie se trouvent réduits et les vitesses avec lesquelles les tâches sont réalisées, augmentent sensiblement par rapport au cas des robots sériels. La commande en position des robots parallèles a suscité beaucoup d'intérêt dans la communauté roboticienne et plusieurs techniques de commande ont été étudiées (commande robuste, commande neuronale

à paramètres fixes, commande adaptative, etc.). La complexité d'un robot parallèle impose souvent de faire des hypothèses simplificatrices pour établir le modèle dynamique, ce qui rend les erreurs de modélisation inévitables. Il est alors judicieux d'y remédier à cela dans le processus de conception de la commande. De plus, si on considère les perturbations externes et la nature changeante de la tâche, la commande adaptative trouve *de facto* sa place avec bien entendu, le minimum de connaissances *a priori* et de mesures. Dans ce contexte, il a été proposé dans [Achi 09a] une méthodologie de commande associant la technique des modes glissants pour sa robustesse vis-à-vis des perturbations externes et les réseaux MLP pour leur caractéristique d'approximation parcimonieuse.

### 1.2.5 Commande en position

Plusieurs techniques de commande ont été proposées dans la littérature, la commande linéarisante [Midd 88], la commande adaptative [Slot 88], la commande à structure variable [Utki 78], la commande neuronale [Sun 01][Lewi 95] et la commande à inférence floue [Song 05][Gole 08]. La commande linéarisante nécessite la connaissance parfaite du modèle dynamique. Malgré sa sensibilité aux possibles variations des paramètres du système et aux perturbations externes, elle est considérée efficace pour les mouvements rapides.

La commande à structure variable par modes glissants est une technique inspirée des travaux des mathématiciens soviétiques *Fillipov* [Fill 60], *Emelyanov* [Emel 67] et *Utkin* [Utki 78]. Ces travaux ont été repris à partir des années 80 pour la commande des systèmes à structure variable [Slot 91][Mada 06] [Hong 08]. Cette approche de commande est non linéaire et présente l'avantage de robustesse par rapport aux perturbations externes et aux variations paramétriques. Elle constitue un moyen efficace pour la commande des systèmes non linéaires.

Dans la pratique, le modèle dynamique d'un système non linéaire (robot ou

autre), ne peut pas être précisément connu. Par conséquent, dans le but de contourner le problème des incertitudes dans le modèle dynamique, des techniques adaptatives et/ou de compensation sont des solutions qui ont fait leur preuve. Concernant la commande neuronale, deux approches ont été proposées, la première à paramètres fixes et la deuxième à paramètres adaptatifs. La première approche n'est efficace que s'agissant de systèmes ne changeant pas de dynamiques pendant l'exécution d'une tâche donnée. Quant à la deuxième approche, elle est appelée à maintenir ses performances même en cas de changement de dynamique. Ces méthodes neuronales peuvent être différenciées par la possibilité d'intégrer une ou plusieurs connaissances *a priori* dans le modèle, par le nombre de variables mesurées, par la garantie de la stabilité en boucle fermée et par le nombre de paramètres. Dans [Lew 95], un seul réseau de neurones est utilisé pour le suivi de trajectoire en position et en vitesse. Dans [Daac 06], la structure du modèle est considérée connue et toutes ses fonctions sont identifiées en temps réel en même temps que le calcul de la commande. Les paramètres neuronaux servant à identifier ces fonctions sont mis à jour de telle sorte à assurer la stabilité du système en boucle fermée au sens de Lyapunov. D'autres techniques utilisant le modèle identifié ou nominal ont été également proposées [Achi 08][Tso 00]. Dans [Mada 06][Hong 08], les auteurs présentent des approches basées sur la technique des modes glissants du premier ordre. La robustesse face aux perturbations externes et aux variations paramétriques sont des caractéristiques de ces approches. Leur principal inconvénient est la discontinuité du signal de commande. Afin de remédier à cette discontinuité (phénomène du chattering), plusieurs solutions ont été proposées dans la littérature. Dans [Slot 88], la fonction signe est remplacée par une simple saturation. Dans [Ertu 00][Topa 07], un réseau de neurones, couplé avec les modes glissants, est utilisé pour pallier à ce problème.

Dans [Ahme 99], une commande utilisant un observateur par modes glissants a été proposée pour réaliser la poursuite de trajectoire d'une classe de systèmes non linéaires triangulaires. C'est une commande basée sur une combinaison d'un

observateur robuste et de la technique du back stepping. Le travail présenté dans [Ahme 09] a été principalement axé sur l'identification de systèmes non linéaires à paramètres variables dans le temps. Les auteurs utilisent un prédicteur neuronal et un observateur par modes glissants [Ahme 07][Ahme 09].

### 1.2.6 Commande hybride

Lorsque l'effecteur du robot se trouve en contact avec l'environnement (meulage, assemblage, ébavurage, etc.), la commande en position seule n'est pas suffisante. Il est alors nécessaire de prendre en compte les forces qu'exerce le robot sur son environnement, conformément avec certaines exigences de la tâche à réaliser [Visi 10]. Par conséquent, la commande hybride force/position est nécessaire [Perd 91][Kara 07]. Ce type de commandes, a suscité l'attention des chercheurs dans le but de mieux contrôler le mouvement contraint du robot. Dans la littérature, plusieurs techniques de commande ont été développées et continuent à être explorées. Dans [Faro 09], les auteurs emploient la technique adaptative des modes glissants pour concevoir un contrôleur hybride force/position appliqué à une classe de robots flexibles. Pour l'étude de stabilité, quelques hypothèses ont été considérées dont la connaissance de la longueur de chaque segment, la plénitude du rang de la matrice jacobienne et la rigidité de l'environnement. Dans [Zili 06], il a été démontré que les performances d'une commande force/vitesse dans le suivi de contour de formes inconnues, dépendent de la configuration du robot et de la direction de la force de contact.

Les mêmes auteurs, proposent un algorithme de commande basé sur un apprentissage itératif afin de réaliser un suivi de contour de formes inconnues [Visi 10]. Les résultats expérimentaux ont été obtenus en employant un robot SCARA. Dans les deux cas, aucune preuve formelle de stabilité du système en boucle fermée n'est donnée.

En se basant sur le modèle dynamique, une stratégie de commande hybride force/position est proposée pour un manipulateur parallèle planaire à 3 DDL [Wu 09].

Il est à noter que dans leur proposition, les auteurs n'ont pas abordé la stabilité du système en boucle fermée. Dans [Toua 08], une approche neuronale pour la commande hybride force/position des robots parallèles a été proposée. Elle est réalisée en deux étapes. Dans un premier temps, un observateur neuronal hors ligne est développé pour estimer le comportement dynamique du système (robot + environnement). Dans un second temps, un contrôleur neural est mis en œuvre en ligne pour réaliser une tâche d'insertion. L'inconvénient de cette proposition est que la trajectoire de la force désirée est constante et proche de zéro. Aussi la stabilité du système en boucle fermée n'est pas abordée. Une autre approche de commande neuronale adaptative en force [Daac 01] a été proposée pour un vérin hydraulique (système de SISO<sup>3</sup>). Les entrées significatives dans le calcul de cette commande ont été fixées par le biais d'une identification a priori. Parmi les autres techniques de commande neuronale hybride force/position, on peut citer les travaux proposées dans [Kara 07]. L'inconvénient de cette approche est l'utilisation du réseau de neurones linéaire pour estimer la raideur du système. En outre, il existe d'autres méthodes dans la littérature où le modèle dynamique et cinématique inverse du robot sont nécessaires pour le calcul de la loi de commande. Dans ce cas, le signal de commande doit être calculé en temps réel à partir d'équations dynamiques complexes. Toutefois, cette inversion du modèle cinématique peut provoquer l'instabilité du système en boucle fermée. Afin d'éviter le calcul en ligne de la cinématique inverse du modèle, un nouvel intérêt pour la recherche dans les réseaux de neurones a été suscité.

Dans le chapitre trois, nous proposons un nouvel algorithme basé sur un réseau de neurones afin de commander, dans l'espace opérationnel, un robot parallèle en contact avec l'environnement. Les environnements dans lesquels évoluent les robots et les charges qu'ils ont à transporter ou à manipuler, sont différents. C'est d'ailleurs pour cette raison qu'un apprentissage a priori pour fixer les entrées pertinentes d'une fonction quelconque appartenant au modèle, trouve toute sa place. D'une part nous n'avons pas besoin d'utiliser une trajectoire riche en excitation et d'autre part nous

---

<sup>3</sup>SISO: Single Input Single Output = système à une seule entrée et une seule sortie.

avons la possibilité de réduire la complexité des commandes à construire. Avec l'association de deux outils, les réseaux de neurones et les modes glissants, nous réunissons deux caractéristiques importantes, l'approximation parcimonieuse et la robustesse face aux perturbations externes. La méthodologie globale laisse possible l'adaptabilité de la commande aux autres types de robots et plus largement, aux systèmes complexes.

## 1.3 Robots portables : exosquelettes/orthèses

### 1.3.1 Définition

Un exosquelette motorisé ou orthèse est un système mécatronique porté par un individu dans le même sens qu'une prothèse. Il se compose d'une partie mécanique, d'actionneurs et une interface de puissance pour la commande. Contrairement à une prothèse qui remplace une partie du corps humain, un exosquelette est utilisé pour aider, restaurer ou améliorer la motricité du sujet qui le porte. En outre, il peut être utilisé dans diverses applications telles que l'assistance aux personnes dépendantes dans leurs activités quotidiennes (se lever / s'asseoir, marcher, monter les escaliers, jardiner, etc.) ou pour la réhabilitation neuromusculaire des membres inférieurs/supérieurs. En médecine, et par rapport à un traitement manuel, l'exosquelette est supposé être capable de traiter le patient, même en l'absence du thérapeute. Cela conduit à augmenter la fréquence du traitement et à en réduire le coût. Par ailleurs, si l'exosquelette est équipé de capteurs appropriés, il est capable de réaliser des mesures pour évaluer l'état du patient. L'idée est de pratiquer une rééducation intensive ne pouvant pas être réalisée par le thérapeute soumis aux deux conditions : la fatigue et la disponibilité. Les exosquelettes ou les orthèses sont également d'un grand intérêt pour la simulation d'efforts dans un environnement virtuel ou pour soulever de lourdes charges. En général, le terme « exosquelette » est utilisé

pour décrire un dispositif qui augmente la performance d'un utilisateur valide, tandis que le terme « orthèse » est généralement utilisé pour décrire un dispositif utilisé pour aider une personne atteinte d'une certaine forme de pathologie [Herr 09].

### 1.3.2 Exemples d'exosquelettes

A la fin du dix-neuvième siècle, le chercheur N. Yagn dans son brevet [Yagn 90] s'est particulièrement intéressé au mécanisme permettant d'améliorer les performances liées à la locomotion du système humain. Les tous premières réflexions sur les exosquelettes sont illustrées par la figure (1.9).

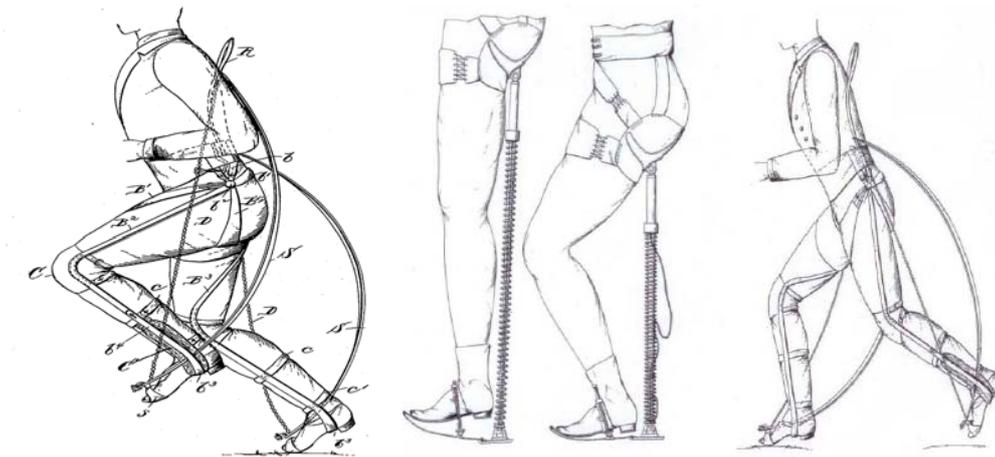


Figure 1.9: Les tous premiers exosquelettes.

L'exosquelette qui a attiré notre attention et connu sous le nom d'Hercule et il est destiné à porter de lourdes charges allant jusqu'à 100Kg. L'individu vêtu d'Hercule et portant des charges peut se déplacer librement et pratiquement sans effort. Ses applications immédiates sont militaires mais le projet se développe encore avec la prise en compte de membres supérieurs (bras) et vise des applications civiles (bâtiment et travaux publics, etc.). Hercule a été développé par des laboratoires Français (CEA<sup>4</sup>, ESME Sudria<sup>5</sup>). Toujours dans le domaine de l'assistance à l'effort,

<sup>4</sup>Commissariat à l'Énergie Atomique

<sup>5</sup>ex-Ecole Spéciale de Mécanique et d'Électricité



Figure 1.10: L'exosquelette Hercule

la figure 1.11 présente une combinaison robotique de membres inférieurs permettant à un porteur de soulever des charges allant de 30Kg à 40Kg avec un minimum d'efforts.



Figure 1.11: Exosquelette de la société japonaise Kawasaki.

### 1.3.3 L'exosquelette développé au LISSI

L'exosquelette/orthèse sur lequel ont porté nos manipulations au laboratoire LISSI, est une combinaison portée par un individu au niveau de son genou et s'étale sur sa cuisse et sa jambe. Avec sa configuration actuelle, cette orthèse est destinée pour la rééducation des membres inférieurs. Pour s'intéresser à d'autres problématiques, telle que la marche par exemple, l'orthèse en question est appelée à évoluer. La figure 1.12 représente un système composé d'une personne en position assise revêtue

d'une orthèse active du genou [Mefo 12a][Mefo 11a].



Figure 1.12: Exosquelette du LISSI

L'exosquelette est composé de deux segments reliés par un axe d'articulation aligné sur celui du genou. L'axe est actionné par un moteur du type Brushless permettant un mouvement synchronisé pour l'ensemble du système. L'exosquelette peut être soumis aussi à l'intention de son porteur dans un contexte d'assistance. Cette intention peut être estimée à travers la mesure des activités musculaires (électromyogrammes). Il peut également être soumis à des mouvements désirés imposés par le médecin rééducateur. Pour réaliser des mouvements de flexion/extension du genou, l'exosquelette est appelé à générer les couples nécessaires. L'orthèse est munie d'un certain nombre de capteurs embarqués tels que le potentiomètre et la jauge de contrainte. Pour estimer l'intention de la personne, d'autres capteurs embarqués comme le goniomètre et les électrodes EMG peuvent être utilisés. Le mouvement de l'exosquelette est contraint par celui du genou entre  $0^\circ$  et  $135^\circ$ , où :

- $0^\circ$  correspond à l'extension totale de la jambe,
- $90^\circ$  correspond à la position de repos,
- $135^\circ$  correspond à la flexion totale du genou.

### 1.3.4 Commande des exosquelettes

Afin que l'exosquelette ou l'orthèse puisse fonctionner correctement et ainsi atteindre les résultats escomptés en termes de réhabilitation [Zhou 08] ou d'assistance

[Bana 09], il doit être rigoureusement commandé. Dans la littérature, plusieurs études traitent le problème de la commande de l'exosquelette des membres inférieurs [Agui 12][Flei 08][Niki 07][Sait 05][Prat 04]. Dans [Jime 12], les auteurs ont mis l'accent sur les stratégies de commande des exosquelettes des membres inférieurs actifs et semi-actifs. Dans [Yeh 10], une orthèse du membre inférieur est conçue pour aider les personnes âgées ou les sportifs souffrant de blessures, à marcher et /ou à monter les escaliers. Dans [Llag 11], une approche d'optimisation physiologique statique pour résoudre le problème d'actionnement d'une orthèse musculaire est présentée. Egalement, une conception mécanique d'un nouveau principe de contrôle/commande d'une orthèse active du genou et de la cheville a été proposée. La conception et la commande d'un robot à trois degrés de liberté destiné aux exercices thérapeutiques pour la rééducation de membres inférieurs d'un patient, a été proposé dans [Akdo 11]. Une interface homme-machine avec une structure de contrôle basée sur un certain nombre de règles, a été développée pour commander ce robot (Physiotherabot). Par ailleurs des commandes de type PID qui sont sensibles à l'incertitude paramétrique et au changement de dynamiques sont également utilisés [Yu 10]. Les techniques de contrôle qui sont le plus souvent choisies ne sont pas adaptatives et sont parfois fondées sur une identification a priori.

Dans [Mefo 11b][Mefo 12b][Mefo 12c], une identification des paramètres dynamiques en utilisant la méthode des moindres carrés et une commande basée sur les modes glissants d'ordre supérieurs, ont été proposées pour une orthèse du genou. Le problème majeur de cette façon de faire est que l'étape d'identification dépend de l'individu et est à refaire pour tout autre individu. Dans [Rifa 12a], une commande adaptative d'un exosquelette active agissant au niveau du genou de son porteur, a été proposée. Son inconvénient c'est qu'elle est basée sur la connaissance exacte de la structure du modèle dynamique. Si on considère la nature de la personne vêtue de l'exosquelette ou de l'orthèse, son environnement, son intention et ses activités musculaires, les techniques issues de l'intelligence artificielle utilisées pour la conception de commandes, sont d'un intérêt certain.

Ainsi, les réseaux de neurones à couches considérés comme des approximateurs universels [Cybe 89][Horn 89], peuvent être utilisés pour approcher des fonctions et/ou paramètres inconnus du modèle. Comme une application immédiate, ces fonctions et ces paramètres sont alors utilisés en ligne dans le calcul de contrôleurs adaptatifs. Dans [Zhan 10], un réseau de neurones à bases radiales (RBF) est utilisé pour commander un exosquelette du membre inférieur destiné à la rééducation. Les paramètres du RBF sont adaptatifs et la stabilité du système est prouvée par l'approche de Lyapunov. Par ailleurs le fait de choisir les entrées de ce RBF comme les erreurs en position et en vitesse n'est pas justifié et l'hypothèse associée n'est pas vérifiée en pratique. Par ailleurs cette approche n'est validée que par des résultats de simulation. Dans [Pere 12], une solution basée sur les réseaux de neurones et la logique floue est proposée pour la résolution de la dynamique inverse d'un exosquelette à six degrés de liberté destiné pour les membres supérieurs. La méthodologie hybride est soumise à une procédure de validation croisée.

## 1.4 Systèmes de commande et dSPACE

La commande des deux systèmes étudiés est réalisée au moyen d'un micro-ordinateur qui intègre une carte dSPACE DS 1103, interface de puissance et un ensemble de capteurs. Cette carte a comme avantage principal la possibilité de tester et valider des lois de commande via un langage de haut niveau. Le logiciel connu sous le nom de « Control Desk », nous permet une facile exploitation de cette carte et nous offre la possibilité de réaliser un contrôle temps réel. En effet, la couche de commande qui gère l'asservissement des moteurs relatifs aux robots considérés (robot parallèle à liaisons C5 et exosquelette) est décrite d'abord par un modèle SIMULINK en utilisant des S-fonctions écrites en langage C compatible avec le module temps réel Real-Time Workshop de la carte dSPACE. Les informations délivrées par les capteurs dont sont équipés les deux systèmes robotisés, sont utilisées par la commande en temps réel. Ces informations sont les positions et les forces de contact pour le

robot parallèle à liaisons C5, et les positions de l'articulation du genou ainsi que les différents signaux liés à l'activité musculaire pour l'exosquelette (EMG<sup>6</sup> par exemple). Le résultat de la compilation du modèle SIMULINK génère tous les fichiers nécessaires à l'expérimentation en temps réel. Parmi ces fichiers, nous avons un fichier directement exploitable par le logiciel Control Desk destiné à être implanté dans le noyau temps réel de la carte dSPACE pour la gestion de l'application. Le noyau temps réel de la carte dSPACE supervise plusieurs tâches dont les tâches de contrôle pour la gestion des événements, les tâches de reprise et les tâches rapport.

Le schéma synoptique du système de commande valide, pour les deux systèmes considérés, est présenté par la figure 1.13. Il est à noter que la phase d'installation

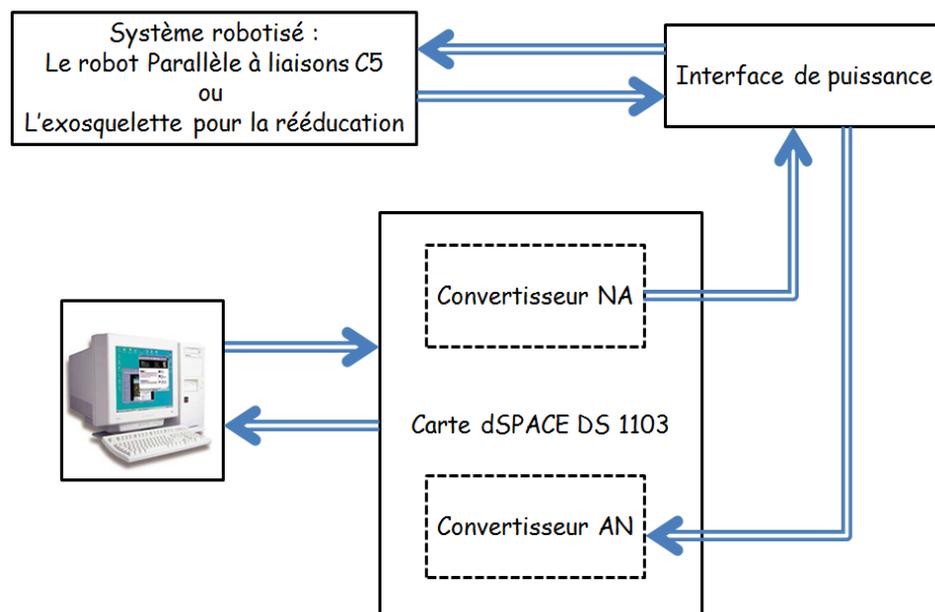


Figure 1.13: Architecture matérielle de commande des deux systèmes robotisés considérés.

des logiciels relatifs à la carte dSPACE, génère de manière automatique de nouvelles bibliothèques sous SIMULINK. Ce sont les bibliothèques dSPACE RTI CAN Blockset et dSPACE RTI1103 et contiennent les blocs d'entrées/sorties propres à la

<sup>6</sup>ElectroMyoGramme

carte.

### 1.4.1 La carte dSPACE DS 1103

La carte dSPACE DS 1103 est représentée dans la figure 1.14 . Elle est dotée d'un processeur rapide et permet une sélection complète d'interfaces d'E/S typiques. Elle est directement intégrée sur la carte mère d'un PC.



Figure 1.14: Carte dSPACE DS 1103 utilisée pour la commande de systèmes robotiques considérés.

### 1.4.2 Caractéristiques techniques de la carte dSPACE DS 1103

Nous résumons ici les principales caractéristiques de la carte dSPACE et qui montrent les différentes possibilités qu'elle offre :

- ▷ Carte contrôleur puissante pour prototypage rapide des lois de commande
- ▷ Choix d'E/S complet
- ▷ DSP esclave avec E/S supplémentaires
- ▷ Entièrement programmable à partir de l'environnement de schéma-bloc Simulink
- ▷ PowerPC 604e à 400 MHz

- ▷ 2Mo SRAM locale
- ▷ 32 Mo ou 128 Mo DRAM globale
- ▷ 16 canaux A/D, 16 bits
- ▷ 4 canaux A/D, 12 bits
- ▷ 8 canaux D/A, 14 bits
- ▷ Interface de capteurs inductifs de vitesse moteur (7 entrées)
- ▷ 32 canaux d'E/S numériques, programmables en groupes de 8 bits
- ▷ Interface série
- ▷ Interface CAN
- ▷ Génération PWM simples et triphasés
- ▷ 4 entrées de capture
- ▷ 2 unités ADC, chacune avec 8 entrées, 10 bits
- ▷ E/S numériques 18 bits

## 1.5 Conclusion

Nous avons fait le point sur les robots parallèles et les robots portables avec des exemples de systèmes réels issus de la littérature. Quelques applications de ses systèmes mécatroniques ont également été évoquées. Par ailleurs, nous avons exposé un certain nombre de méthodes de commande relatives à ces systèmes ayant fait l'objet de publications dans des revues et conférences internationales. Ceci nous permet de faire des propositions et de les appliquer aux deux systèmes mécatroniques considérés dans cette thèse. Par ailleurs, le contenu de ce chapitre permet d'éclaircir d'emblée les objectifs visés ainsi que la manière de les concrétiser. Dans le deuxième chapitre, nous nous intéressons à l'identification expérimentale de la dynamique inverse du robot parallèle à liaisons C5.

---

## Chapitre 2

# Identification de la dynamique inverse du robot parallèle à liaisons C5

---

### 2.1 Introduction

L'identification de systèmes non linéaires a fait l'objet de plusieurs travaux de recherche. Les techniques utilisées sont diverses et variées : la logique floue, les réseaux de neurones [Cybe 89][Nare 90], les moindres carrés [Gaut 01], etc. Chaque technique possède ses avantages et ses inconvénients. Des connaissances peuvent être intégrées par certaines techniques d'identification. Par exemple dans le cas de robots manipulateurs, l'utilisation des réseaux de neurones permet d'exploiter la structure du modèle dynamique qui est a priori connue. Une identification fiable permet de faciliter la conception de schémas de commande stables. L'identification est en général basée sur des données obtenues sur le système. En pratique et sur une manipulation réelle, ces données sont souvent affectées par un bruit de mesure. Deux étapes sont alors à accomplir pour obtenir une base de données exploitable et riche en excitation. La première étape consiste à choisir et à appliquer une commande nominale au système et recueillir les mesures qui en résultent. La deuxième étape consiste à

opérer un prétraitement sur ces données afin d'éliminer le bruit.

Dans un premier temps nous exposerons la méthode d'identification basée sur les moindres carrés. Cette méthode a fait l'objet d'une application au robot parallèle à liaisons C5 dans le cadre d'une thèse de doctorat soutenue en 2009 [Achi 09a]. Dans un second temps, nous détaillerons notre démarche d'identification basée sur les réseaux de neurones à couches. Les propriétés des réseaux de neurones à couches ont été largement étudiées dans la littérature. Hornik et Cybenko [Horn 89][Cybe 89] ont prouvé que si le réseau de neurones contient suffisamment de neurones cachés, il est capable d'approcher toute fonction continue sur un horizon fini. Souvent le nombre de neurones cachés est déterminé de manière heuristique et l'erreur d'approximation n'est jamais nulle. Dans la littérature, les méthodes d'identification neuronales ne considèrent pas la particularité de chaque système. En effet, la nature et le nombre d'entrées du réseau de neurones, peuvent être étudiés dans le but de réduire le nombre de paramètres d'une part et d'améliorer la qualité d'identification d'autre part. Pour le cas d'une identification boîte noire de la dynamique inverse d'un robot manipulateur, les entrées possibles sont les positions, les vitesses et les accélérations. Dans ce cas, il serait intéressant de comparer les résultats donnés par différentes architectures neuronales et c'est justement l'objet de ce chapitre.

## 2.2 Aperçu sur les méthodes d'identification

Pour réaliser l'identification d'un système dynamique, les mesures des entrées et des sorties correspondantes à celui-ci sont nécessaires. Dans la littérature plusieurs techniques d'identification ont été proposées et expérimentées sur des systèmes réels. Dans [Arms 86], les auteurs proposent une méthode d'identification de paramètres dynamique basée sur des essais expérimentaux sur chacun des corps d'une manière indépendante avant le montage du robot. On peut parler ici d'une technique d'identification décentralisée. Il est clair qu'il ne faut pas s'attendre à des miracles en appliquant cette technique d'identification car les corps d'un robot quel qu'il soit, ne sont jamais

dynamiquement indépendants. Par ailleurs, il est toujours possible de l'utiliser pour obtenir un résultat nominal d'identification à compléter par la suite.

Une autre technique d'identification utilise le système de Conception Assistée par Ordinateur (CAO) [An 85]. En effet, à l'aide d'un ordinateur et de la synthèse d'images, la CAO permet de concevoir des systèmes réels via des techniques de modélisation géométrique. En faisant l'hypothèse relative à la répartition homogène et uniforme des masses et selon certaines considérations géométriques sur les corps du robot, l'évaluation des paramètres de celui-ci devient possible. Comme dans la pratique ce genre de considérations et d'hypothèses n'est pas toujours vérifié, les méthodes concernées ne sont guère employées. D'autres approches plus réalistes ont été proposées [Wit 90][Gaut 90][Gaut 95][Kozl 98]. Les points suivants sont communs à toutes ces méthodes [Gaut 90]:

- Construction d'un système surdéterminé via l'échantillonnage du modèle dynamique le long d'un mouvement du robot.
- Utilisation de la forme factorisée QR en utilisant le procédé d'algèbre linéaire de décomposition en valeurs singulières.
- Trouver ou calculer un jeu de données riches en excitation de telle sorte à exciter le maximum de dynamiques du système à identifier.

Certaines de ces méthodes ont été appliquées pour l'identification de robots manipulateurs sériels mais peuvent également s'adapter et être utilisées dans le cas de robots à architectures parallèles. Lorsque nous pouvons connaître la structure du modèle dynamique d'un système mécatronique ainsi que les dynamiques qui y interviennent, il est possible de proposer une réécriture du modèle sous forme linéaire en les paramètres dynamiques à identifier.

La méthode du maximum de vraisemblance dans laquelle le modèle d'identification est parfait, a été employée dans [Swev 97] pour l'identification. Le bruit gaussien, de moyenne nulle, est supposé affecter les mesures mais les vitesses et les accélérations sont épargnées. Comme le modèle d'identification n'est jamais parfait dans la

pratique, une autre version robuste de maximum de vraisemblance est proposée dans [Cala 01] et tient compte des incertitudes relatives au modèle dynamique. Dans [Kubu 07][Kubu 08], les auteurs ont travaillé sur de méthodes d'identification en ligne afin de reconstituer les paramètres inertiels du robot. Le temps de calcul nécessité par ces méthodes lors de leur application en ligne, est de l'ordre de 1.5s [Kubu 08]. Toujours pour des applications en ligne, d'autres procédures d'identification sont proposées pour l'estimation d'autres dynamiques tels que la raideur, l'amortissement et l'inertie [Hadd 08a][Hadd 08b][Hadd 08c]. Le problème de l'incertitude ainsi que la variation paramétrique sont ainsi surmontés. La plupart de ces méthodes n'est pas orienté commande et permet simplement d'analyser le comportement du système considéré. Les modèles identifiés permettent par exemple de réaliser des schémas d'observation dans le but d'être appliqués dans le domaine du diagnostic.

Dans les paragraphes suivants, nous exposons plus en détail d'autres méthodes d'identification.

## **2.2.1 Méthodes ensemblistes pour l'identification**

Plusieurs méthodes ensemblistes appliquées à l'identification ont fait l'objet de propositions dans la littérature [Poig 03a] [Poig 03b]. Dans ces méthodes, les erreurs structurelles de modélisation sont prises en compte sans pour autant faire des hypothèses sur leur nature. Par exemple pour les systèmes poly-articulés, ces erreurs se manifestent par les jeux dans les articulations ainsi que par le bruit de mesure. L'algorithme ellipsoïdal et l'analyse par intervalle, que nous décrivons brièvement dans les sections suivantes, sont des exemples de ces méthodes ensemblistes.

### **2.2.1.1 Analyse par intervalle pour l'identification**

La méthode ensembliste basée sur l'analyse par intervalle [Poig 03b] consiste à manipuler des intervalles au lieu de manipuler des nombre ou chiffres réels. Dans la

thèse de V. Drevelle [Drev 11], l'analyse par intervalle sert principalement à manipuler les incertitudes. Elle peut être employée pour résoudre les équations linéaires et non linéaires. Elle peut être également utilisée dans les problèmes d'optimisation [L Ja 01] pour l'estimation des paramètres dynamiques [Poig 03b] où la fonction coût est non convexe. La solution est obtenue sous forme d'un vecteur d'intervalles et la largeur de chaque intervalle donne l'incertitude relative au paramètre identifié.

### 2.2.1.2 Algorithme ellipsoïdal pour l'identification

Le principe de l'algorithme ellipsoïdal [Poig 03a] consiste à englober le polyèdre des paramètres admissibles dans un ellipsoïde [Duri 00][MMil 96]. L'ellipsoïde est retenu pour son avantage d'être décrit simplement par trois paramètres : son centre, sa taille et son orientation. Les deux derniers paramètres sont représentés par une matrice définie positive.

### 2.2.2 Méthodes d'apprentissage pour l'identification

Le principe d'identification par les méthodes d'apprentissage consiste à réaliser une base de données d'apprentissage et une base de données de généralisation. Cette dernière doit balayer tout l'espace de travail du système pour garantir une bonne identification du système. Les paramètres de l'outil d'apprentissage choisi (système d'inférence floue, réseau de neurones, ou autre) doivent être correctement optimisés via un algorithme d'optimisation approprié (Marquardt-Levenberg pour les réseaux de neurones à couches par exemple). Pour plus de détail sur ces méthodes d'apprentissage, nous pouvons nous référer aux travaux [Nare 90][Zbik 94][Wang 11][Scha 98][Vija 02]

### 2.2.3 Moindres carrés pour l'identification

Cette méthode conventionnelle est réalisée en plusieurs étapes décrites dans les sections suivantes [Achi 09a]. Malgré que cette méthode n'est pas spécialement dédiée

aux systèmes fortement non linéaires, nous avons fait le choix de la décrire car elle a été appliquée au robot parallèle à liaisons C5 qui fait l'objet de cette thèse.

### 2.2.3.1 Choix du modèle dynamique pour l'identification

L'hypothèse admise pour le choix du modèle d'identification est qu'il soit linéaire en les paramètres dynamiques. Sa structure peut être celle donnée par l'équation suivante [Khal 99]:

$$\Gamma = W(q, \dot{q}, \ddot{q})\chi \quad (2.1)$$

Avec :

- $\Gamma$  représente le vecteur des couples appliqués au niveau des articulations
- $W$  est une matrice connue sous le nom de la matrice d'observation (ou parfois régresseur)
- $\chi$  vecteur contenant les paramètres dynamiques
- $q, \dot{q}, \ddot{q}$  représentent les positions, les vitesses et les accélérations articulaires

### 2.2.3.2 Calcul des paramètres de base du robot

Il s'agit de déterminer les paramètres de base du robot [Maye 90]. Ils peuvent être obtenus à partir des paramètres standards du robot en éliminant certains d'entre eux qui n'ont pas d'effet sur le modèle dynamique et en regroupant les autres par des combinaisons linéaires. Pour une analyse rigoureuse de la dynamique d'un système, faire le choix d'éliminer certains paramètres n'est sans doute pas judicieux.

### 2.2.3.3 Choix de la trajectoire excitante

Le choix de la trajectoire excitante est déterminant dans le processus d'identification. Avec une trajectoire représentative de la dynamique du robot (qui excite toutes les dynamiques) le résultat d'identification ne peut qu'être satisfaisant si la méthode

choisie a été correctement appliquée. De manière plus précise, une trajectoire excitante se traduit par un bon conditionnement de la matrice d'observation  $W$ . Le calcul de la trajectoire excitante optimale revient à résoudre un problème d'optimisation basé sur une fonction objective et les contraintes physiques du système. Dans la littérature, plusieurs types de trajectoire ont été proposés. Dans [Gaut 01], une trajectoire polynômiale de degré 5 est proposée. Le problème de calcul des coefficients du polynôme est posé comme un problème d'optimisation avec contraintes. Pour l'identification des paramètres dynamiques, il est nécessaire de filtrer la position. Ensuite les vitesses et accélérations sont calculées par l'algorithme de différence centrée pour éviter la distorsion en phase et en amplitude. Dans [Daem 96], l'auteur a proposé une autre méthode qui consiste à utiliser un polynôme de degré 7 tel que le point initial  $q_0$  et le point final  $q_f$  sont liés par un seul point intermédiaire  $q_1$ . Les vitesses et accélérations sont nulles aux points initiaux et finaux. Le même auteur a utilisé le profile de vitesse trapézoïdale pour générer une trajectoire excitante. Cette trajectoire a l'avantage d'être implémentée facilement dans le contrôle de robots industriels.

Le concept de trajectoire excitante périodique est introduit dans [Swev 96]. Cette trajectoire est basée sur une série de Fourier définie par la somme de deux fonctions *sinus* et *cosinus*. Bien que l'implémentation de cette trajectoire soit difficile, elle présente plusieurs avantages telle que la dérivée analytique de la série de Fourier nous permet d'obtenir précisément les vitesses et accélérations qui interviennent dans le calcul du régresseur. Les détails sur les autres avantages se trouvent dans [Swev 97].

Après l'identification des paramètres dynamiques, on doit procéder à une validation des valeurs obtenues. Dans la littérature plusieurs techniques peuvent être réalisées, celles les plus répandues sont décrites ci-dessous :

- Validation croisée : la base de données est divisée en plusieurs morceaux de données (sous-base ou plis). Par exemple les paramètres d'identification sont calculés en utilisant les données du premier morceau et la validation avec les

autres morceaux restants. Le processus est répété jusqu'à ce que l'erreur de validation soit la plus petite possible.

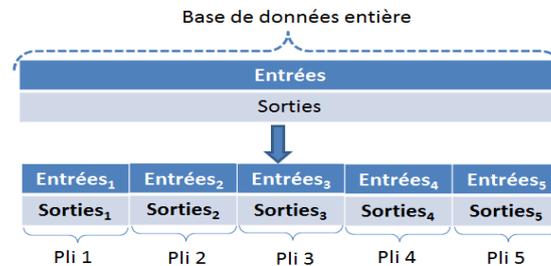


Figure 2.1: Principe de la validation croisée.

- Validation directe : elle se fait sur la base du calcul d'une erreur de prédiction relative au même mouvement employé lors de la phase d'identification.
- Validation par la commande: si la commande basée sur le modèle identifié donne des résultats satisfaisants, l'identification réalisée est définitivement adoptée. Cette façon de faire n'est pas forcément la meilleure surtout si la structure à commander n'est pas en mesure de supporter des essais répétitifs.

#### 2.2.3.4 Analyse et commentaires

Dans la thèse [Achi 09a], les résultats de cette méthode sont satisfaisants et ont même fait l'objet d'une commande en position du robot parallèle à liaisons C5. L'avantage de cette approche d'identification, de type boîte grise, est de faciliter l'analyse de la dynamique du système. En effet, on aura autant de bloc identifié que d'ensembles de paramètres représentant la même dynamique (paramètres liés aux frottements, paramètres liés à la gravité, paramètres liés à l'inertie, etc.). Aussi, la synthèse de lois de commande ne sera pas une tâche difficile avec bien sûr l'analyse de la stabilité du système en boucle fermée. L'inconvénient de la méthode réside dans le fait que la structure du modèle doit être connue ainsi que la caractéristique de la linéarité en les paramètres qui doit être admise également. Cependant ces deux

hypothèses ne sont pas toujours vérifiées ce qui oblige parfois à une linéarisation du modèle. Ceci conduit naturellement à faire des simplifications et dans certains cas à négliger des dynamiques qui peuvent s'avérer importantes. C'est d'ailleurs la raison pour laquelle la méthode en question n'est pas fortement choisie s'agissant de systèmes fortement non linéaires. Dans ce qui suit, nous allons nous pencher sur une technique neuronale pour identifier la dynamique de ce même robot à liaisons C5.

## 2.3 Problématique d'identification neuronale considérée

La problématique traitée dans cette partie du chapitre est l'identification de la dynamique inverse du robot parallèle à liaisons C5. Nous utilisons dans cette identification, les réseaux de neurones à couches du type perceptron multicouches (MLP). Nous nous sommes limités à des MLP à une couche cachée. Notre choix s'explique par le fait qu'il a été démontré, dans la littérature que n'importe quelle fonction continue pouvait être approchée avec une précision arbitraire par un réseau de neurones comportant une seule couche cachée [Horn 89][Cybe 89]. Les fonctions d'activation au niveau de la couche cachée sont de type sigmoïdal. Trois architectures neuronales seront étudiées et comparées. La première est une identification neuronale type boîte noire classique (un seul MLP). Dans la deuxième architecture, on essaiera de prouver par des données réelles, qu'il n'est pas possible de dissocier les dynamiques relatives aux axes du robot. On utilisera dans ce cas un réseau par axe avec, comme erreur permettant l'adaptation des paramètres neuronaux, la moyenne des erreurs d'identification concernant tous les axes. Les entrées de chaque réseau sont les variables généralisées de l'axe concerné. La troisième architecture est une version modifiée de la deuxième architecture avec les mêmes entrées pour tous les réseaux. L'algorithme qu'on a retenu pour l'optimisation des paramètres neuronaux

est celui de Marquard-Levenberg.

## 2.4 Architectures neuronales et identification de la dynamique inverse

En identification, les architectures neuronales pouvant être retenues sont variables. On peut parler par exemple des architectures boîte grise et boîte noire. Lorsqu'il est possible d'avoir des connaissances a priori sur le système (structure du modèle etc.) à identifier, il est intéressant de les associer aux réseaux de neurones identificateurs. On parle dans ce cas d'identification boîte grise. Les systèmes pour lesquels les seules informations disponibles sont les entrées et sorties, l'identification est dite boîte noire. Dans le présent travail, nous réalisons l'identification du type boîte noire et essayons de traiter tous les cas de figures possibles. L'identification consiste à obtenir un modèle pour représenter le comportement dynamique d'un système. Dans notre cas on utilise le réseau de neurones du type MLP pour identifier le comportement dynamique du robot parallèle à liaison C5. On essayera au travers de trois architectures neuronales différentes, de réaliser la procédure d'identification. Les résultats obtenus seront comparés. La procédure d'identification est composée d'une étape d'apprentissage et d'une autre étape de généralisation.

### 2.4.1 Génération de trajectoire pour l'identification

Pour toute identification, le choix ou le calcul d'une trajectoire riche en excitation, est déterminant pour tout le processus d'identification. Les mouvements articulaires avec une interpolation polynomiale entre les points de passage, définissent les mouvements du robot. Le principe consiste à calculer par optimisation non linéaire les coefficients des polynômes qui minimisent certains critères d'excitation, sous contraintes des positions, vitesses et accélérations admissibles.

Les mouvements excitants fournissent des données riches en information sur la

dynamique du système [Swev 97]. Dans notre cas le calcul de la trajectoire excitante, qui est en réalité une sorte de trajectoire modulée, est basé sur une méthode heuristique à base des séries de Fourier. La base de données représentant cette trajectoire contient les couples et les positions relatifs aux 6 axes du robot. Dans [Swev 97], la trajectoire excitante est basée sur la série de Fourier dont les paramètres sont issus d'une optimisation non linéaire. Dans notre cas, nous avons également utilisé la même série de Fourier pour calculer la trajectoire excitante du robot C5. En revanche, les paramètres de cette trajectoire sont calculés heuristiquement. Cette méthode est basée sur des contraintes du mouvement imposées par des limitations physiques du robot. Cette approche a permis d'aboutir aux courbes données par les figures 2.2, 2.3 et 2.4. Ces trajectoires ont été également utilisées dans [Achi 09a][Daac 08] et comme elles ont données de bons résultats d'identification par la méthode des moindres carrés, nous les avons reprises sans les modifier.

Pour la génération de données relative à la dynamique du robot parallèle à liaisons C5, les consignes sont remplacées par les trajectoires excitantes calculées. Pour éviter tout comportement non souhaité du robot lors de la génération de données pour l'identification, celui-ci est commandé en boucle fermée par un PID. Les couples appliqués et les positions mesurées sont ensuite enregistrés dans une base de données destinée à l'identification.

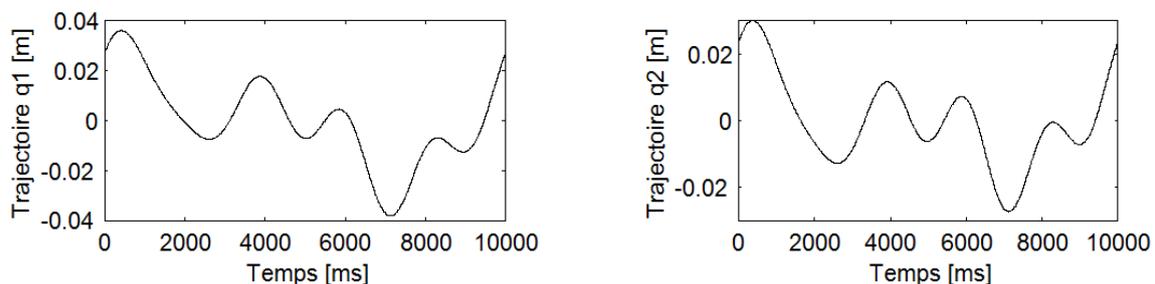


Figure 2.2: Trajectoires excitantes pour les articulations 1 et 2.

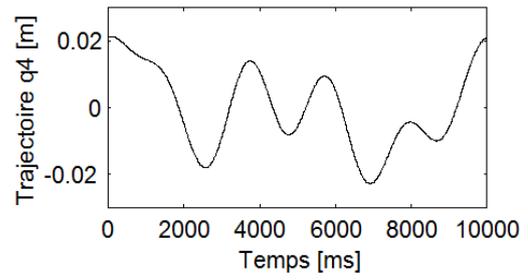
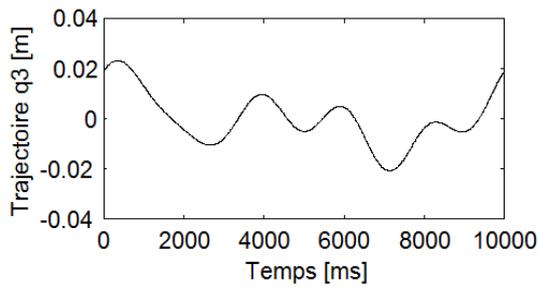


Figure 2.3: Trajectoires excitantes pour les articulations 3 et 4.

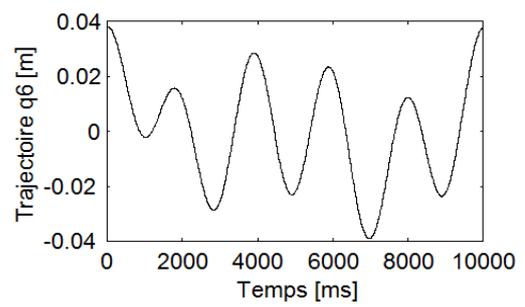
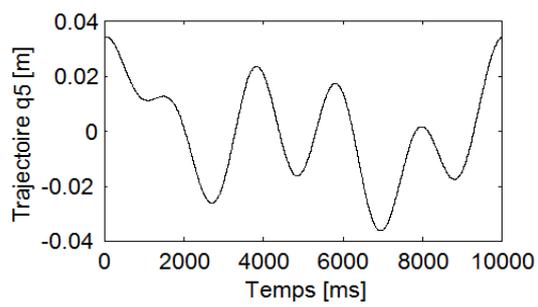


Figure 2.4: Trajectoires excitantes pour les articulations 5 et 6.

## 2.4.2 Filtrage des positions mesurées et calcul des vitesses et des accélérations

Afin d'éviter que du bruit soit interprété comme donnée réelle, filtrer les mesures obtenues constitue une étape importante pour garantir une bonne estimation des paramètres dynamiques. Le filtre utilisé est celui de Butterworth. Les vitesses et les accélérations sont calculées ensuite par le biais de l'algorithme dit de différence centrée permettant d'éviter la distorsion de phase et d'amplitude du signal.

## 2.4.3 Présentation de la méthode d'identification

L'organigramme de la figure 2.5 résume les étapes d'identification.

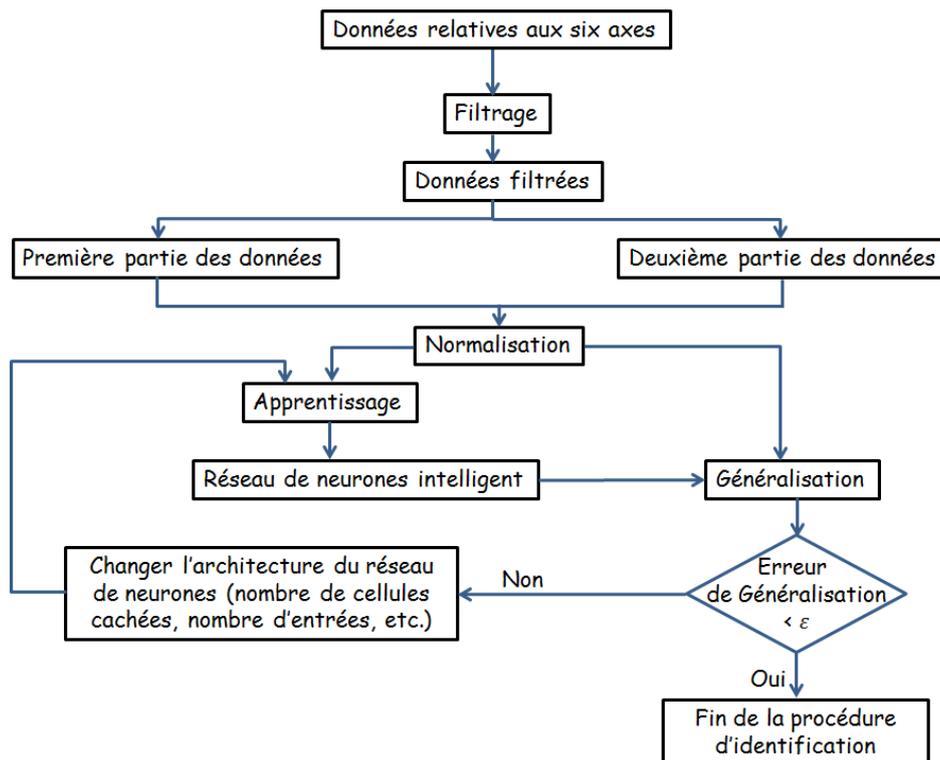


Figure 2.5: Organigramme d'identification.

Pour le calcul des paramètres neuronaux, on utilise l'algorithme de Marquard-Levenberg. Ainsi, nous avons expérimenté trois architectures neuronales. Le premier

schéma d'identification est composé d'un seul réseau de neurones. Les entrées du réseau sont les positions, les vitesses et les accélérations. Dans le deuxième schéma,

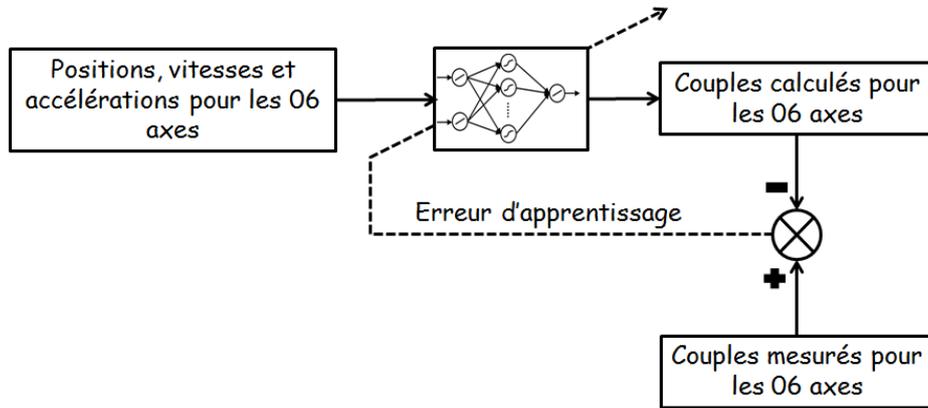


Figure 2.6: Premier schéma d'identification.

un réseau de neurones est réservé pour chaque axe avec comme entrées la position, la vitesse et l'accélération de l'axe concerné (figure 2.7). L'erreur qui permet de mettre à jour les paramètres neuronaux est la moyenne des erreurs partielles concernant les six axes. Dans le troisième schéma d'identification, nous utilisons également un

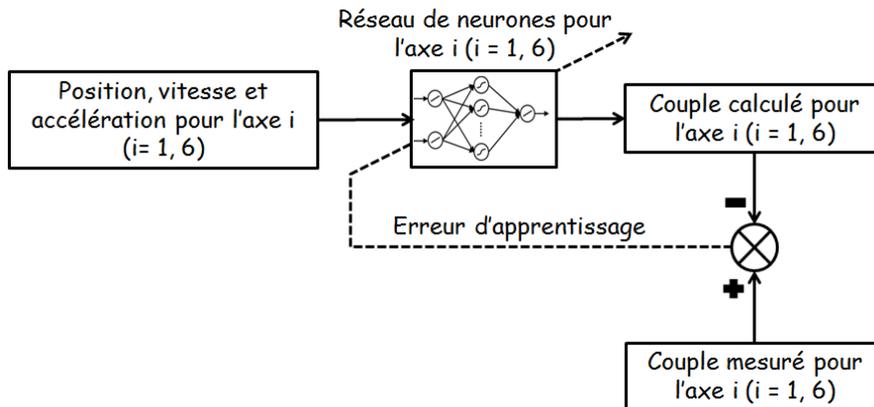


Figure 2.7: Deuxième schéma d'identification.

réseau de neurones par axe. Contrairement au deuxième schéma d'identification, les six réseaux possèdent les mêmes entrées (positions, vitesses et accélérations de

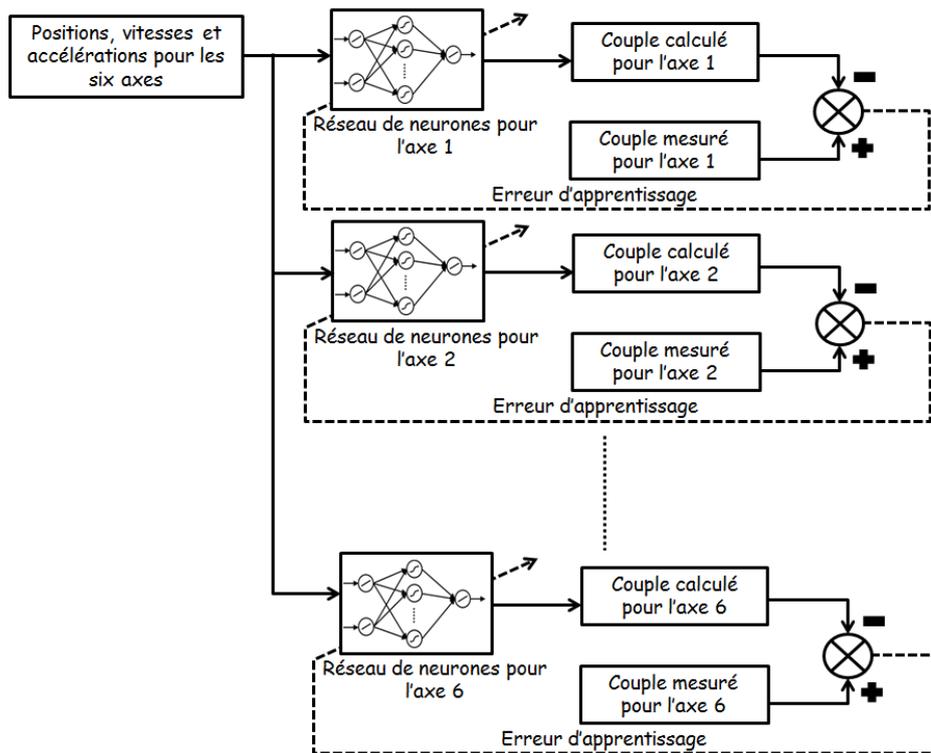


Figure 2.8: Troisième schéma d'identification.

tous les axes du robot). Aussi chaque réseau est indépendant de l'autre concernant la mise à jour des paramètres neuronaux (figure 2.8).

Pour mettre en oeuvre ces trois schémas d'identification de la dynamique du robot, et comme il s'agit d'un apprentissage neuronal supervisé, il est nécessaire :

- D'effectuer des mesures sur le robot réel. Ceci est réalisé via une commande nominale de type Proportionnelle Dérivée, ce qui permet d'éviter tout comportement anormal du robot et éviter ainsi tout dégât matériel pouvant être causé par une boucle ouverte.
- De faire un filtrage de ces données avec un filtre de Butterworth passe-bas, afin d'avoir les signaux des positions filtrées pour six axes du robot.
- Ensuite, à défaut de capteurs, on dérive la position pour avoir la vitesse, puis on dérive une deuxième fois pour avoir l'accélération.

## 2.5 Résultats expérimentaux

Dans cette phase d'identification, nous avons utilisé des données générées par la même manipulation. Avec comme données d'apprentissage la première moitié du signal et comme données de généralisation la deuxième moitié du signal. Pour une identification efficace, il est très important d'avoir les données d'apprentissage représentatives de la dynamique du système. La dynamique dont il s'agit d'identifier est la dynamique inverse. Le tableau de la figure 2.9 résume les principales caractéristiques de chaque méthode. Les paramètres retenus et présentés dans ce tableau donnent les meilleurs résultats d'identification concernant chaque méthode. C'est à dire que le nombre de cellules cachées est celui qui donne le meilleur résultat pour chacune des architectures retenues. Les illustrations données concernent uniquement

<i>Schéma d'identification</i> <i>Nombre de</i>	<b>1</b>	<b>2</b>	<b>3</b>
<b>Paramètres</b>	<b>144</b>	<b>144</b>	<b>114</b>
Réseaux de neurones	<b>1</b>	<b>6</b>	<b>6</b>
Neurones cachés dans chaque réseau	<b>6</b>	<b>6</b>	<b>1</b>
Entrées dans chaque réseau	<b>18</b>	<b>3</b>	<b>18</b>
Sorties de chaque réseau	<b>6</b>	<b>1</b>	<b>1</b>

Figure 2.9: Tableau comparatif

l'étape de généralisation. Pour faciliter l'étape de comparaison, les résultats obtenus des trois architectures sont présentés sur la même figure.

Avec un simple examen des figures (2.10, 2.11, 2.12, 2.13, 2.14, 2.15) on peut très bien exclure la deuxième méthode. Nous pouvons alors constater que chaque couple ne peut être correctement reconstitué qu'avec les variables articulaires de l'ensemble des articulations. C'est à dire qu'il n'est pas possible de découpler complètement la dynamique d'un axe par rapport à un autre. La première et la troisième architectures sont alors les seules qu'il nous est possible de retenir. En

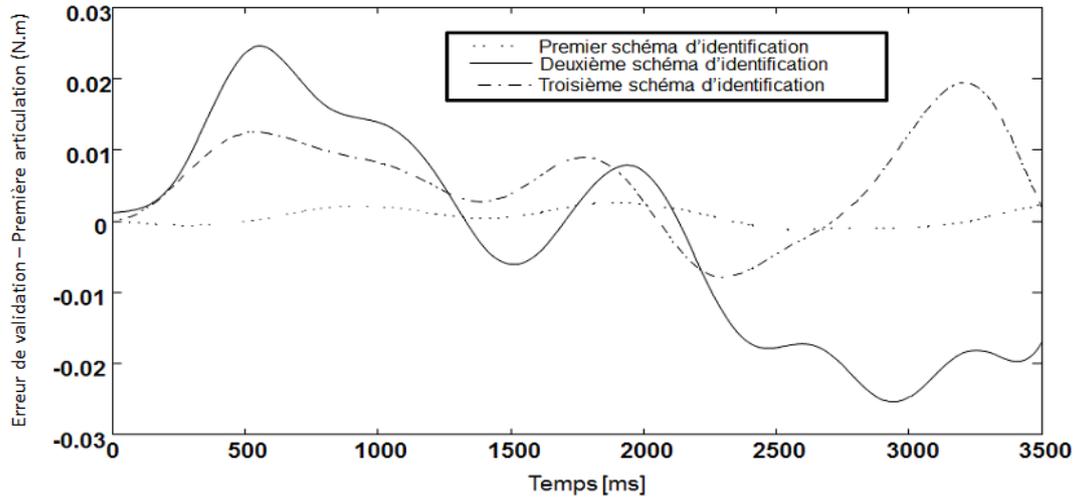


Figure 2.10: Erreur de validation (généralisation) - axe 1 du robot.

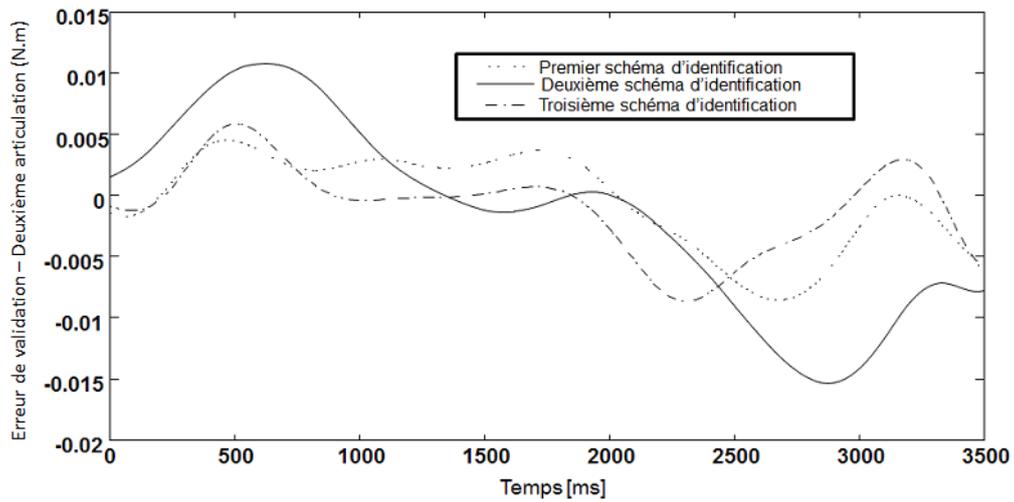


Figure 2.11: Erreur de validation (généralisation) - axe 2 du robot.

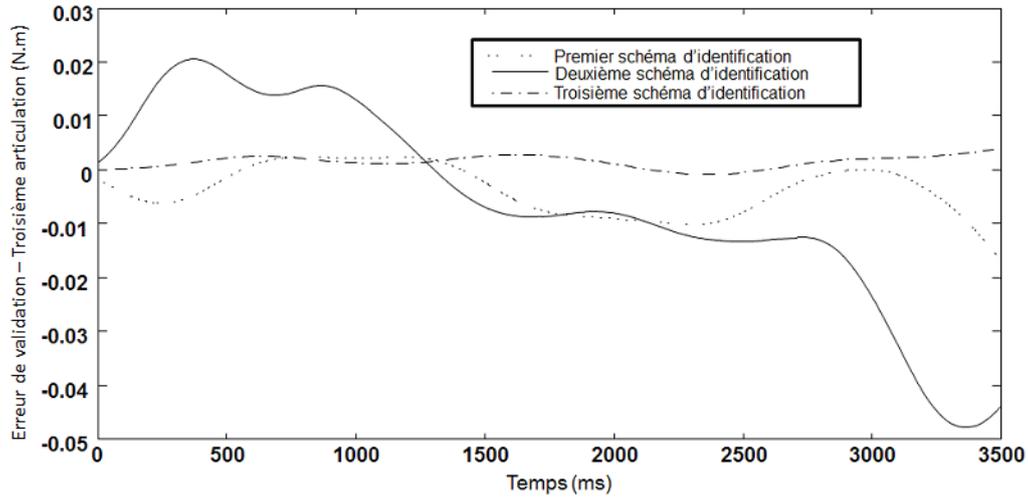


Figure 2.12: Erreur de validation (généralisation) - axe 3 du robot.

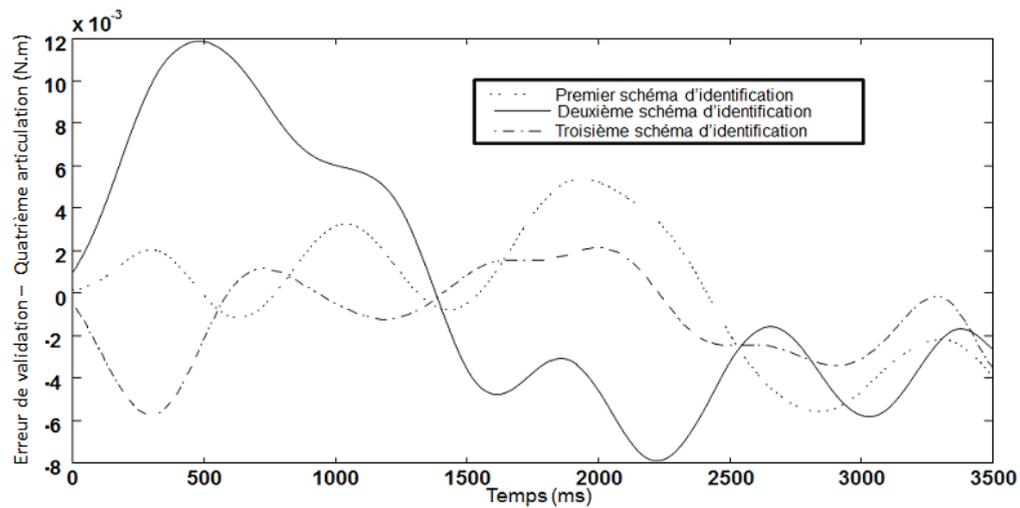


Figure 2.13: Erreur de validation (généralisation) - axe 4 du robot.

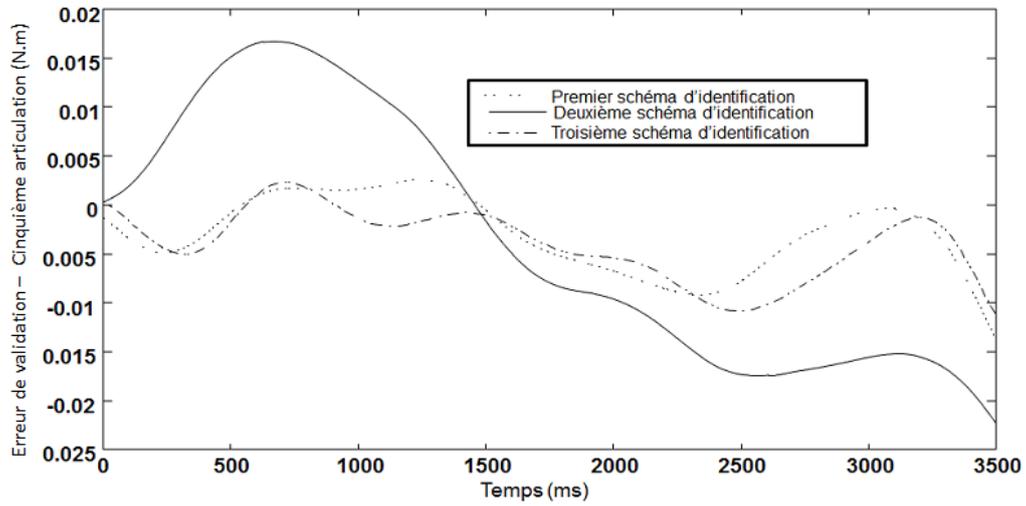


Figure 2.14: Erreur de validation (généralisation) - axe 5 du robot.

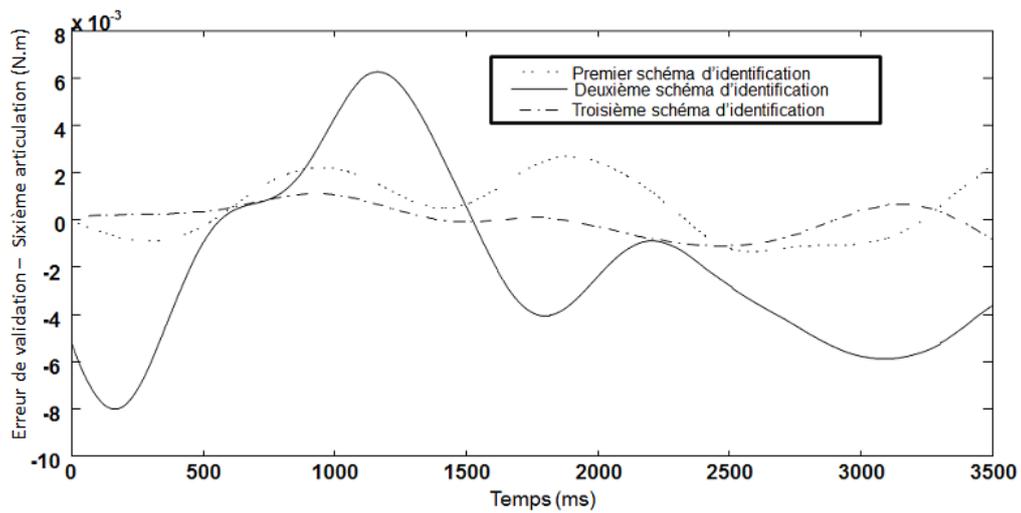


Figure 2.15: Erreur de validation (généralisation) - axe 6 du robot.

termes de nombre de paramètres, la troisième architecture est la meilleure. Lorsque les données ont été générées dans de bonnes conditions et ont été préparées correctement, les techniques d'apprentissage permettent souvent d'aboutir à de bons résultats d'identification.

Par ailleurs, il n'est pas admis qu'une identification puisse être valable définitivement et ce quel que soit sa qualité. En effet, tout système mécatronique est sensé réaliser des tâches différentes et évoluer dans des environnements variables. Ceci provoque des changements non négligeables dans sa dynamique et limite la longévité d'un résultat d'identification. Ce dernier peut être exploité efficacement au moment où il a été fait et perd de sa substance au fil du temps. Lorsqu'il s'agit d'une application en commande, le résultat d'une quelconque identification peut être très utile pour l'initialisation et doit être épaulé par des blocs de robustesse ou par des paramètres adaptatifs.

## 2.6 Conclusion

Dans ce chapitre, nous avons présenté l'identification expérimentale de la dynamique inverse d'un robot parallèle à liaison C5. Les réseaux de neurones à couches ont été utilisés pour cette identification. Les signaux utilisés pour l'identification sont des signaux modulés en fréquence et en amplitude. Ils sont formellement basés sur une série de Fourier. Cette caractéristique de modulation permet de couvrir une bonne partie de l'espace de travail et par conséquent de fournir des données riches en excitation.

Comme perspectives immédiates à ce travail, il serait intéressant d'appliquer le résultat de l'identification dans un schéma de commande. Dans ce cas, le schéma le plus utilisé est celui de la commande par anticipation. Là, il est important de faire une étude qui permet de rajouter un terme de robustesse qui fait face aux erreurs d'identification. Dans le chapitre qui suit, nous allons proposer une approche de commande hybride moment/position dans l'espace de la tâche et l'expérimenter sur

le robot parallèle à liaisons C5.

---

# Chapitre 3

## Commande hybride

### moment/position du robot

### parallèle à liaisons C5

---

#### 3.1 Introduction

Dans ce chapitre, nous proposons une commande adaptative hybride moment/position du robot parallèle à liaison C5 pour réaliser un suivi de contour. La commande est synthétisée dans l'espace de la tâche [Daac 12]. La pièce métallique sur laquelle le suivi de contour se réalise, a la forme d'une demi-sphère. Le réseau de neurones utilisé dans la commande est du type MLP et ses paramètres sont adaptatifs et leur mise à jour est basée sur des lois issues de l'étude de la stabilité du système en boucle fermée en utilisant l'approche de Lyapunov. Ce réseau a pour rôle d'approcher la dynamique du robot en contact avec l'environnement. Comme aucune connaissance a priori n'est considérée, l'approximation neuronale en question est de type boîte noire. Par ailleurs, aucune transformation de modèles n'est nécessaire et la commande est directement calculée dans l'espace opérationnel. L'effecteur du robot en contact avec la pièce métallique est représenté dans les figures 3.1. La commande par son caractère adaptatif est sensée fonctionner pour toute sorte d'environnement

et contrer les perturbations externes. Même si des changements de dynamiques interviennent au cours de l'exécution d'une tâche, théoriquement ils ne peuvent en aucun cas se manifester aux niveaux des performances en termes de suivi de contour.

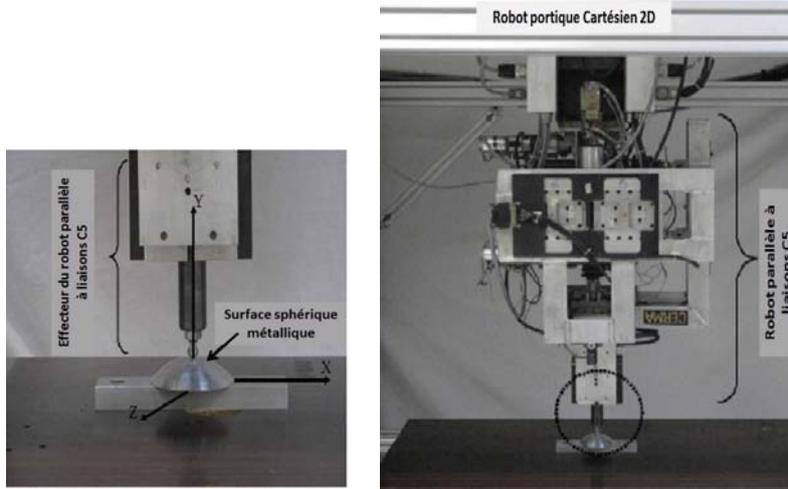


Figure 3.1: Robot parallèle en contact avec l'environnement.

## 3.2 Formulation du problème

Le modèle dynamique du robot dans l'espace articulaire en contact avec son environnement est donné par l'équation suivante [Khal 04][Achi 10]:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + H(q, \dot{q}) + J^T(q)F_e = \tau \quad (3.1)$$

Avec  $q \in R^n$  est le vecteur des positions articulaires,  $\dot{q} \in R^n$  est le vecteur des vitesses articulaires et  $\ddot{q} \in R^n$  est le vecteur des accélérations articulaires,  $M(q)$  est la matrice d'inertie,  $C(q, \dot{q})\dot{q}$  représente les forces de Coriolis et centrifuges,  $H(q, \dot{q})$  le vecteur des forces de gravitation et de frottements,  $\tau$  le vecteur des couples appliqués aux articulations.  $F_e$  représente le vecteur d'effort extérieur (forces et moments) exercé par le robot sur l'environnement.

La vitesse de l'effecteur du robot  $\dot{X} \in R^m$  et les vitesses articulaires  $\dot{q} \in R^n$  sont reliées par la matrice jacobienne  $J(q) \in R^{m \times n}$  tel que:

$$\dot{X} = J(q)\dot{q} \quad (3.2)$$

avec  $J(q) = \frac{\partial \phi(q)}{\partial q}$ .

Dans le cas du robot parallèle à liaisons C5, la dimension de l'espace opérationnel  $m$  est égale au nombre de degrés de liberté de l'espace articulaire  $n$ ;  $n = m = 6$ . Et par conséquent, la matrice jacobienne est une matrice carrée.

$X = \phi(q)$  est le modèle géométrique direct donnant la situation de l'organe terminal (position et orientation) dans l'espace opérationnel en fonction des variables articulaires. L'accélération de l'effecteur  $\ddot{X}$  est reliée à l'accélération articulaire  $\ddot{q}$  comme suit:

$$\ddot{X} = J\ddot{q} + \dot{J}\dot{q} \quad (3.3)$$

Si la matrice  $J$  est inversible, les équations (3.2) et (3.3) permettent le calcul des vitesses et des accélérations articulaires tel que:

$$\dot{q} = J^{-1}\dot{X} \quad \text{et} \quad \ddot{q} = J^{-1}\ddot{X} - J^{-1}\dot{J}J^{-1}\dot{X}$$

L'espace de la tâche du robot parallèle à liaison C5 est à l'abri des singularité [Dafa 98]. Et par conséquent la matrice jacobienne est inversible dans l'espace opérationnel du robot. Une étude détaillée sur les configurations singulières et l'espace de la tâche du robot C5 peut être trouvée dans [Dafa 98].

Le modèle dynamique dans l'espace de la tâche peut être exprimé comme suit:

$$J^{-T}(M(q)\ddot{q} + C(q, \dot{q})\dot{q} + H(q, \dot{q}) + J^T(q)F_e) = J^{-T}\tau \quad (3.4)$$

En utilisant les équations (3.2), (3.3) et (3.4), le modèle dynamique dans l'espace de la tâche s'exprime alors:

$$M^*(q)\ddot{X} + C^*(q, \dot{q})\dot{X} + H^*(q, \dot{q}) + F_e = \tau^* \quad (3.5)$$

avec:

$$\begin{cases} M^* = J^{-T} M J^{-1} \\ C^* = J^{-T} (C J^{-1} - M J^{-1} \dot{J} J^{-1}) \\ H^* = J^{-T} H \\ \tau^* = J^{-T} \tau \Rightarrow \tau = J^T \tau^* \\ F_e = J^{-T} f \end{cases}$$

Où  $f$  est le vecteur des forces articulaires. Le modèle dynamique dans l'espace opérationnel est fonction de  $q, \dot{q}, \ddot{q}, \dot{X}, \ddot{X}$  et  $F_e$ . Tant que  $\dot{X}, \ddot{X}$  et  $F_e$  sont fonctions de  $(q, \dot{q})$ ,  $(q, \dot{q}, \ddot{q})$  et  $(f, q)$  respectivement, le modèle donné dans (3.5) dépend de  $q, \dot{q}, \ddot{q}$  et  $f$ .

Pour la conception de la commande hybride force/position, le modèle dynamique du robot parallèle à liaisons C5 dans l'espace opérationnel vu comme une boîte noire est décrit par la fonction suivante:

$$\psi(r) = u \quad (3.6)$$

$\psi(r)$  est une fonction non linéaire inconnue,  $u$  est le couple de commande. Le vecteur  $r$  comprend les positions articulaires, vitesses, les accélérations et les forces  $(q, \dot{q}, \ddot{q}, f)$ .

La fonction  $\psi(r)$  est approximée par un réseau de neurones tel que:

$$\widehat{\psi}(r) = \hat{u} \quad (3.7)$$

Nous proposons un nouvel algorithme basé sur un réseau de neurones afin de commander, dans l'espace opérationnel, un robot parallèle en contact avec l'environnement. Les environnements dans lesquels peut évoluer le robot et les charges manipulées, ne sont pas toujours les mêmes. C'est la raison pour laquelle nous pensons qu'un apprentissage *a priori* pour fixer les entrées pertinentes d'une fonction quelconque appartenant au modèle, est justifié. D'une part, l'utilisation de trajectoire riche en excitation n'est plus une contrainte et d'autre part, cette approche permet de réduire la complexité des commandes à construire. La méthodologie globale laisse possible l'adaptabilité de la commande aux autres types de robots et plus largement, aux

systemes complexes. En supposant que le modèle en contact avec l'environnement n'est pas disponible, il est nécessaire de passer par une étape d'identification en utilisant un réseau de neurones MLP. Cette étape d'identification vise à déterminer le vecteur d'état du système. La recherche de ce vecteur d'état revient en quelques sortes à déterminer le degré relatif inconnu du système. Une fois ce degré relatif déterminé, l'application de la commande neuronale adaptative devient possible en exploitant le modèle identifié obtenu comme modèle initial. Les paramètres neuronaux sont ensuite mis à jour en ligne à partir d'un algorithme basé sur l'étude de stabilité du système en boucle fermée. Tel que prouvé dans [Achi 09b], un réseau de neurones avec comme vecteur d'entrée  $(q, \dot{q}, f)$  est suffisant pour estimer la fonction non linéaire  $\psi$ . Cette solution permet à la fois la réduction de la taille du réseau de neurones et évite l'utilisation des données d'accélération bruitées dans la loi de commande donnée ultérieurement. Dans ce qui suit, le vecteur  $r$  inclut uniquement les positions, les vitesses et les forces articulaires  $(q, \dot{q}, f)$ . L'algorithme de commande proposé est décrit dans l'organigramme 3.2.

### 3.2.1 Approximation neuronale et traitement des non linéarités

La commande hybride que l'on propose tient compte de l'hypothèse suivante :

Hypothèse: *La fonction  $\psi$  peut être approximée par un MLP de forme  $w_2^{*T} \varphi(w_1^{*T} r)$ , avec  $\varphi : R^p \rightarrow R^p$  est le vecteur de la fonction d'activation et  $w_1^{*T} \in R^{p, l}$ ,  $w_2^{*T} \in R^{h, p}$  sont les paramètres optimaux à approcher de manière adaptative ( $l$ , dimension du vecteur d'entrée du réseau de neurones  $r$ ;  $h$  est le nombre de sorties du réseau de neurones et  $p$  le nombre des neurones dans la couche cachée).*

Pour tout  $r$ , nous avons:

$$\begin{aligned} \left\| \psi(r) - \widehat{\psi}(w_1^*, w_2^*, r) \right\| &< \epsilon \\ \widehat{\psi}(w_1^*, w_2^*, r) &= w_2^{*T} \varphi(w_1^{*T} r) \end{aligned} \quad (3.8)$$

où

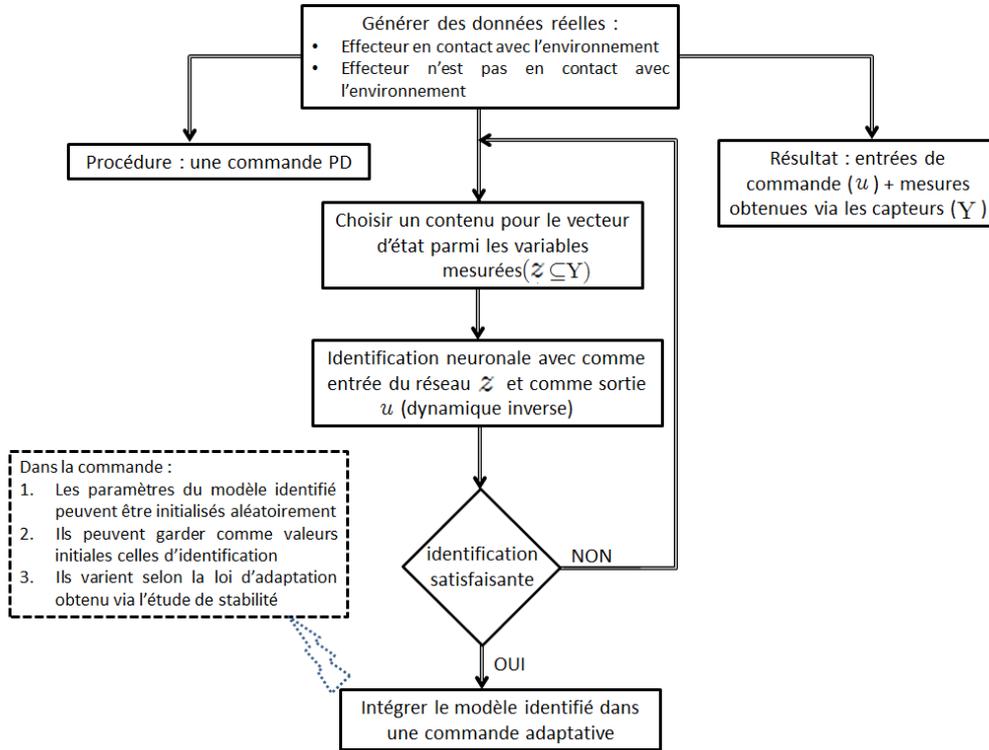


Figure 3.2: Démarche de commande expérimentale.

- $w_1^*$  : représente le vecteur des poids entre l'entrée du réseau de neurones et la couche cachée.
- $w_2^*$  : représente le vecteur des poids entre la couche cachée et la sortie du réseau de neurones.
- $\epsilon$  : erreur d'approximation neuronale, tel que ( $\|\epsilon\| \leq \bar{\epsilon}$ ), avec  $\bar{\epsilon}$  connu et suffisamment petit.

L'approximation neuronale de la fonction  $\psi$  peut être écrite sous la forme:

$$\hat{\psi} = w_2^T \varphi(w_1^T r) \quad (3.9)$$

- $w_1$  et  $w_2$  sont l'estimation de  $w_1^*$  et  $w_2^*$  respectivement.

Par rapport aux réseaux de neurones linéairement paramétrés, l'avantage des réseaux de neurones du type MLP est certainement le nombre relativement réduit de

paramètres. Il est clair que ce nombre dépend de la dimension de l'entrée, néanmoins cette dépendance n'est pas exponentielle. L'inconvénient de ce type de réseaux est leur paramétrisation non linéaire. Une alternative pour traiter ces non linéarités est d'utiliser le développement en série de Taylor-Young. La fonction  $\varphi$  peut être écrite sous la forme:

$$\varphi(w_1^{*T}r) = \varphi(w_1^T r) - \varphi'(w_1^T r)\tilde{w}_1^T r - O(w_1^T r) \quad (3.10)$$

avec  $\tilde{w} = w_1 - w_1^*$ .  $O(w_1^T z)$  converge vers zéro si  $\tilde{w}_1$  converge vers zéro;  $\varphi'$  représente la dérivée bornée de  $\varphi$ . En raison de simplification, les notations suivantes sont utilisées:

$$\begin{cases} \varphi = \varphi(w_1^T r) \\ \varphi^* = \varphi(w_1^{*T} r) \\ \varphi' = \varphi'(w_1^T r) \\ \tilde{\varphi} = \varphi(w_1^T r) - \varphi(w_1^{*T} r) = \varphi - \varphi^* \end{cases} \quad (3.11)$$

on obtient:

$$\psi - \hat{\psi} = w_2^{*T} \varphi^* - w_2^T \varphi + \epsilon \quad (3.12)$$

En ajoutant et en soustrayant  $w_2^{*T} \varphi$  dans l'équation (3.12), on aura:

$$\psi - \hat{\psi} = -w_2^{*T} \tilde{\varphi} - \tilde{w}_2^T \varphi + \epsilon \quad (3.13)$$

En ajoutant et en soustrayant  $w_2^T \tilde{\varphi}$  dans (3.13), on obtient:

$$\psi - \hat{\psi} = \tilde{w}_2^T \tilde{\varphi} - w_2^T \tilde{\varphi} - \tilde{w}_2^T \varphi + \epsilon \quad (3.14)$$

Utilisons le développement en série de Taylor-Young, nous obtenons:

$$\psi - \hat{\psi} = -w_2^T \varphi' \tilde{w}_1^T r - \tilde{w}_2^T \varphi + \epsilon + \epsilon_\varphi \quad (3.15)$$

avec  $\epsilon_\varphi$  représente les erreurs d'approximation dues au développement en série de Taylor-Young du premier ordre:

$$\epsilon_\varphi = \tilde{w}_2^T \varphi' \tilde{w}_1^T r - w_2^{*T} O(w_1^T r) \quad (3.16)$$

De (3.16), on peut noter que si:

$$\tilde{w}_1 \rightarrow 0 \text{ et } \tilde{w}_2 \text{ est borné, alors } \epsilon_\varphi \rightarrow 0$$

Sachant que:

$$\|\epsilon_\varphi\| \leq c_1 \|\tilde{w}_2\|_F \|\tilde{w}_1\|_F \|r\| + \|w_2^*\|_F \|O\| \quad (3.17)$$

$$\bar{\epsilon}_\varphi = c_1 \|\tilde{w}_2\|_F \|\tilde{w}_1\|_F \|r\| + \|w_2^*\|_F \|O\| \quad (3.18)$$

$c_1$  est une constante positive calculée de la dérivée de  $\varphi'$  et  $\|\cdot\|_F$  représente les normes de Frobenius.

avec:

$$\|w_1^*\|_F \leq \|w_1^*\|_{\max} \quad (3.19)$$

$$\|w_2^*\|_F \leq \|w_2^*\|_{\max}$$

L'équation (3.15) est utilisée pour calculer les lois d'adaptation des paramètres du réseau de neurones.

### 3.3 Commande hybride moment/position pour le suivi de contour

L'approche hybride proposée dans ce chapitre consiste en la commande simultanée du moment  $M_Z$  autour de l'axe  $z$ , et de la position dans le plan  $xy$ . La commande en position consiste en la poursuite de contour d'une partie d'une surface circulaire.

#### 3.3.1 Loi de commande et analyse de stabilité

La loi de commande proposée est donnée par l'équation suivante:

$$u = \hat{\psi} + \Lambda_1 E + \Lambda_2 \dot{E}. \quad (3.20)$$

avec:

$$E = \begin{bmatrix} s_1(x^d - x) + (1 - s_1)(F_x^d - F_x) \\ s_2(y^d - y) + (1 - s_2)(F_y^d - F_y) \\ 0 \\ 0 \\ 0 \\ s_6(\gamma^d - \gamma) + (1 - s_6)(M_Z^d - M_Z) \end{bmatrix}$$

- $\Lambda_1, \Lambda_2$  sont des matrices diagonales positives.
- $\hat{\psi}$  vecteur neuronal adaptatif.
- $\chi = (x, y, 0, 0, 0, \gamma)^T$  : vecteur des coordonnées opérationnelles mesurées;
- $\gamma$  est l'angle d'orientation autour de l'axe  $z$ .
- $\chi^d = (x^d, y^d, 0, 0, 0, \gamma^d)^T$  : vecteur des coordonnées opérationnelles désirées.
- $F = (F_x, F_y, 0, 0, 0, M_z)^T$  : torseur des forces mesurées.
- $F^d = (F_x^d, F_y^d, 0, 0, 0, M_z^d)^T$  : torseur des forces désirées.
- $s$  est la matrice de sélection définie par :

$$s = \begin{bmatrix} \bar{s}_1 & \bar{s}_2 & \bar{s}_2 \\ \bar{s}_2 & \bar{s}_2 & \bar{s}_2 \\ \bar{s}_2 & \bar{s}_2 & \bar{s}_3 \end{bmatrix} \quad \bar{s}_1 = \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix} \quad \bar{s}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \bar{s}_3 = \begin{bmatrix} 0 & 0 \\ 0 & s_6 \end{bmatrix}$$

- $s_1 = s_2 = 1$ , signifie que les directions  $x$  et  $y$  sont commandées en position.
- $s_6 = 0$ , signifie que la direction  $z$  est commandée en moment.

Le schéma de commande utilisé pour notre implémentation est représenté dans la figure 3.3.

Les signaux issus des capteurs de position et de force sont filtrés en ligne en utilisant un filtre passe bas montré dans la figure 3.3.

Les lois d'adaptation des paramètres neuronaux sont obtenues via une étude de stabilité du système en boucle fermée au sens de Lyapunov:

$$\begin{aligned} \dot{w}_2 &= \eta_2 \varphi E^t \\ \dot{w}_1 &= \eta_1 r E^t w_2^t \varphi' \end{aligned} \tag{3.21}$$

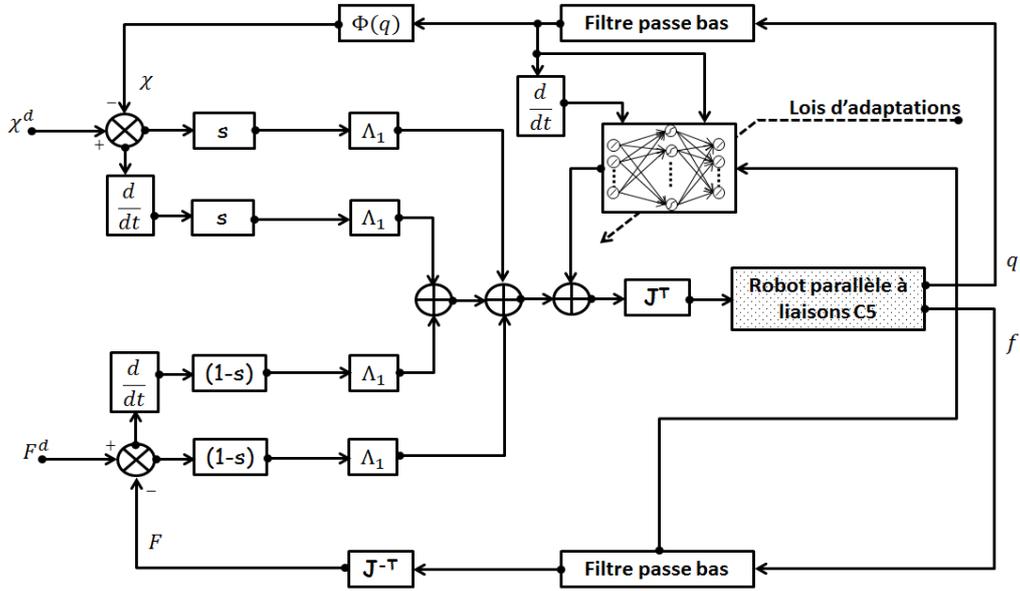


Figure 3.3: Schéma de commande hybride moment/position

Considérons le système en boucle fermée décrit par les équations suivantes :

$$\psi = \hat{\psi} + \Lambda_1 E + \Lambda_2 \dot{E} \quad (3.22)$$

$$\Lambda_2 \dot{E} = \psi - \hat{\psi} - \Lambda_1 E \quad (3.23)$$

$$\tilde{\psi} = \psi - \hat{\psi} \quad (3.24)$$

$$\Lambda_2 \dot{E} = \tilde{\psi} - \Lambda_1 E \quad (3.25)$$

### 3.3.2 Analyse de stabilité

Pour l'analyse de stabilité, considérons la fonction de Lyapunov positive suivante:

$$V = \frac{1}{2} E^t \Lambda_2 E + tr(\tilde{w}_1^t \eta_1^{-1} \tilde{w}_1) + tr(\tilde{w}_2^t \eta_2^{-1} \tilde{w}_2) \quad (3.26)$$

La dérivée peut être donnée comme suit:

$$\dot{V} = E^t \Lambda_2 \dot{E} + tr(\tilde{w}_1^t \eta_1^{-1} \dot{w}_1) + tr(\tilde{w}_2^t \eta_2^{-1} \dot{w}_2) \quad (3.27)$$

Utilisons les équations (3.15), (3.25) et (3.27), nous obtenons:

$$\begin{aligned}\dot{V} &= -E^t \Lambda_1 E - E^t \tilde{w}_2^t \varphi - E^t w_2^t \varphi \tilde{w}_1^t r \\ &\quad + tr(\tilde{w}_1^t \eta_1^{-1} \dot{w}_1) + tr(\tilde{w}_2^t \eta_2^{-1} \dot{w}_2) + E^t (\epsilon_\varphi + \epsilon) \\ &= \dot{V}_1 + \dot{V}_2 + \dot{V}_3\end{aligned}\tag{3.28}$$

avec:

$$\dot{V}_1 = -E^t \Lambda_1 E + E^t (\epsilon_\varphi + \epsilon)\tag{3.29}$$

$$\dot{V}_2 = -E^t w_2^t \varphi \tilde{w}_1^t r + tr(\tilde{w}_1^t \eta_1^{-1} \dot{w}_1)\tag{3.30}$$

$$\dot{V}_3 = -E^t \tilde{w}_2^t \varphi + tr(\tilde{w}_2^t \eta_2^{-1} \dot{w}_2)\tag{3.31}$$

De (3.29), nous obtenons:

$$\dot{V}_1 \leq -\lambda_{\min}(\Lambda_1) \|E\|^2 + \|E\|_1 \|(\bar{\epsilon} + \bar{\epsilon}_\varphi)\|_1\tag{3.32}$$

$\lambda_{\min}$  est la plus petite valeur propre de  $\Lambda_1$ .

$\dot{V}_1 \leq 0$  si:

$$\|E\|_1 \geq \frac{\|(\bar{\epsilon} + \bar{\epsilon}_\varphi)\|_1}{\lambda_{\min}(\Lambda_1)}\tag{3.33}$$

L'erreur  $E$  est bornée en boucle fermée. Via les lois d'adaptation données dans (3.21), nous obtenons:

$$\dot{V}_2 = -E^t w_2^t \varphi \tilde{w}_1^t r + tr(\tilde{w}_1^t \eta_1^{-1} \dot{w}_1) = 0$$

et

$$\dot{V}_3 = -E^t \tilde{w}_2^t \varphi + tr(\tilde{w}_2^t \eta_2^{-1} \dot{w}_2) = 0$$

Par conséquent  $\dot{V} = \dot{V}_1 \leq 0$  et le système est stable en boucle fermée.

### 3.4 Validation expérimentale

Afin de démontrer l'efficacité de l'approche de commande proposée, la commande hybride force/position a été implémentée sur le robot parallèle à liaisons C5 exécutant

la poursuite d'une partie d'une surface métallique sphérique. Une comparaison avec le contrôleur PID est également réalisée. Les paramètres du réseau de neurones sont initialisés aléatoirement entre 0 et 1. Les caractéristiques du réseau de neurones, les gains d'adaptation des paramètres neuronaux ( $\eta_1, \eta_2$ ) et les paramètres de la commande neuronale proposée ( $\Lambda_1, \Lambda_2$ ), qui sont aussi les gains proportionnel et dérivé du contrôleur PID respectivement, sont donnés dans le tableau suivant :

Nombre d'entrées du réseau de neurones	18
Nombre de sorties du réseau de neurones	3
Nombre de couches cachées du réseau de neurones	1
Nombre de cellules dans la couche cachée	3
$\text{diag}(\eta_1)$	0.5
$\text{diag}(\eta_2)$	0.5
$\text{diag}(\Lambda_1)$	20
$\text{diag}(\Lambda_2)$	0.01

Les trajectoires désirées ( $x_d, y_d$ ) et le moment désiré  $M_z^d$  sont définis par:

$$\begin{aligned}
 M_z^d &= 100 \text{ N.m} \\
 x^d(\text{cm}) &= 4 \cos\left(\theta + \frac{\Pi}{6}\right) \\
 y^d(\text{cm}) &= 4 \sin\left(\theta + \frac{\Pi}{6}\right) \\
 0 &\leq \theta \leq \frac{2\Pi}{3}
 \end{aligned}$$

Les résultats de poursuite en moment et l'erreur de poursuite en moment sont représentés dans les figures.3.4 et 3.5 respectivement. L'erreur moyenne et l'erreur relative moyenne sont  $0.826 \text{ N.m}$ ,  $0.8\%$  respectivement pour l'approche neuronale et  $15.21 \text{ N.m}$ ,  $13\%$  respectivement pour le contrôleur PID. Ces résultats montrent que la poursuite en moment en appliquant l'approche neuronale est satisfaisante inversement au cas du contrôleur PID. Idem pour la poursuite en position représentée dans la figure. 3.6. L'erreur en position dans le plan  $xy$  est représentée dans la

figure 3.7. Le maximum et le minimum d'erreur de poursuite en position sont de 1.1 mm et 0.54 mm respectivement pour l'approche neuronale et de 3.2 mm, 1.79 mm respectivement pour le contrôleur PID. Les sorties du réseau de neurones représentées dans les figures 3.8-3.10, contribuent au calcul des couples appliqués aux articulations. Les figures 3.11-3.14 représentent les couples issus de l'application de l'approche neuronale. Cependant, les figures 3.15-3.18 représentent les couples issus de l'application du contrôleur PID. D'après les figures représentant les couples, on constate bien qu'il n'y a pas de différence entre les couples appliqués au premier et au deuxième axe du robot pour les deux commandes (Approche neuronale et PID). Cela signifie qu'il n'y a pas de moment appliqué et par conséquent il n'y a pas de rotation autour de l'axe  $x$ . Cependant que la différence entre les couples appliqués aux troisième et au quatrième axes du robot est significative. Cela est dû à l'existence du moment caractérisé par une rotation autour de l'axe  $z$ . Pour conclure, les résultats obtenus par l'approche neuronale que ce soit en moment ou en position sont très satisfaisants par rapport à ceux obtenus par le PID. Les performances de l'approche proposée en font une approche intéressante du fait qu'il s'agit d'une commande hybride d'un système à architecture complexe dans l'espace de la tâche.

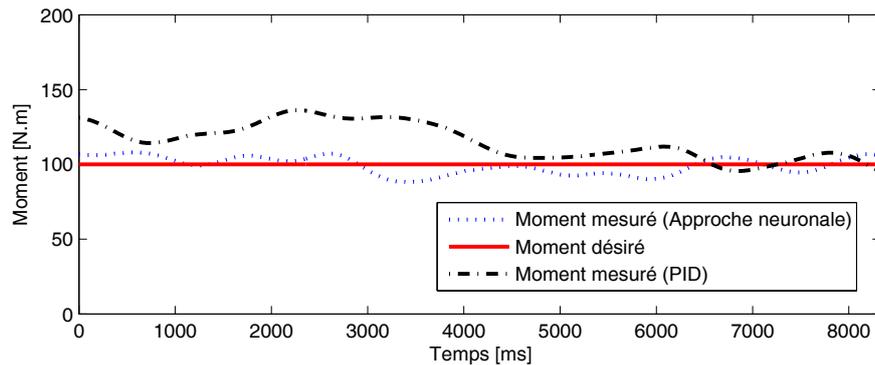


Figure 3.4: Résultat expérimental de la poursuite en moment

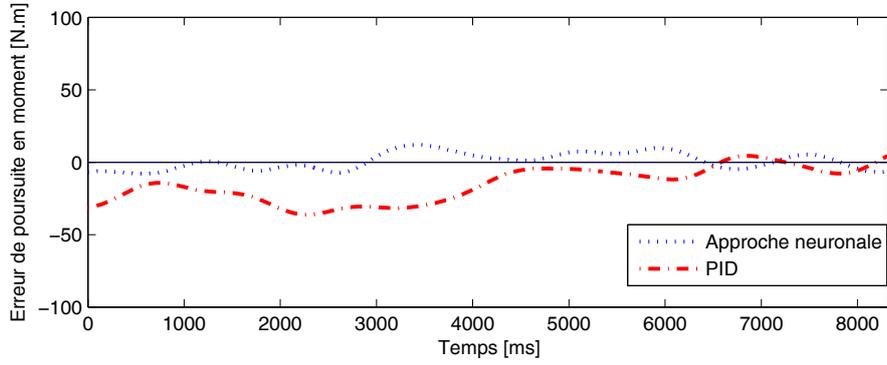


Figure 3.5: Erreur de poursuite en moment

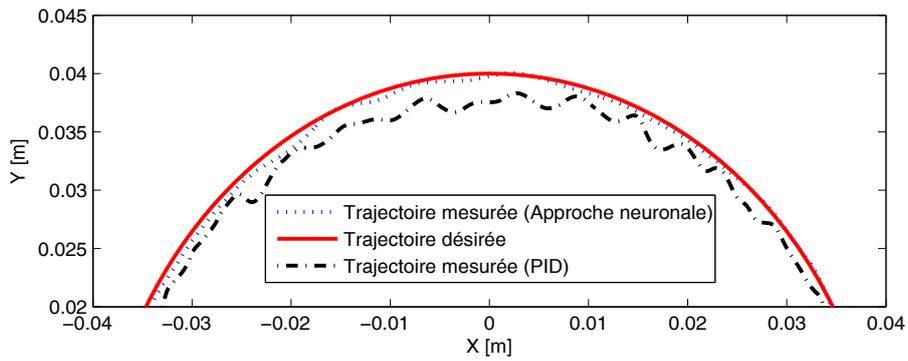


Figure 3.6: Résultat expérimental de la poursuite en position dans le plan xy

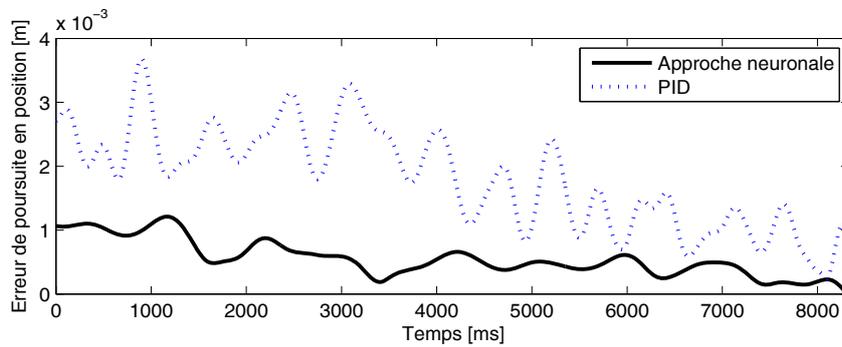


Figure 3.7: Erreur de poursuite en position dans le plan xy

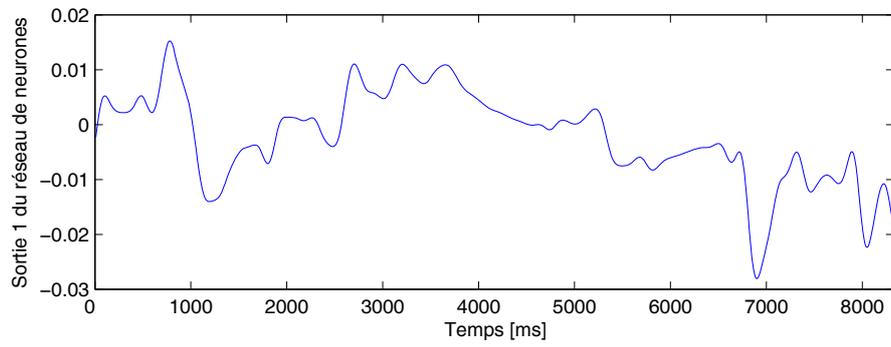


Figure 3.8: Sortie 1 du réseau de neurones

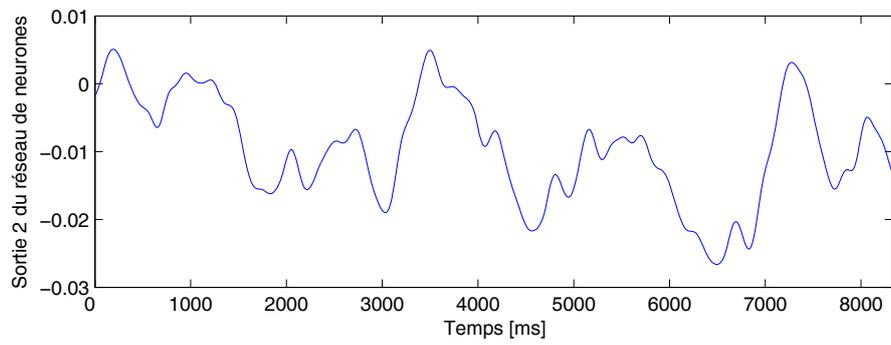


Figure 3.9: Sortie 2 du réseau de neurones.

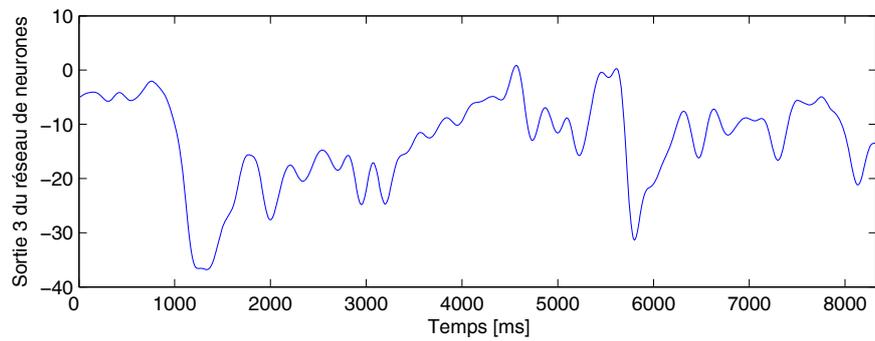


Figure 3.10: Sortie 3 du réseau de neurones

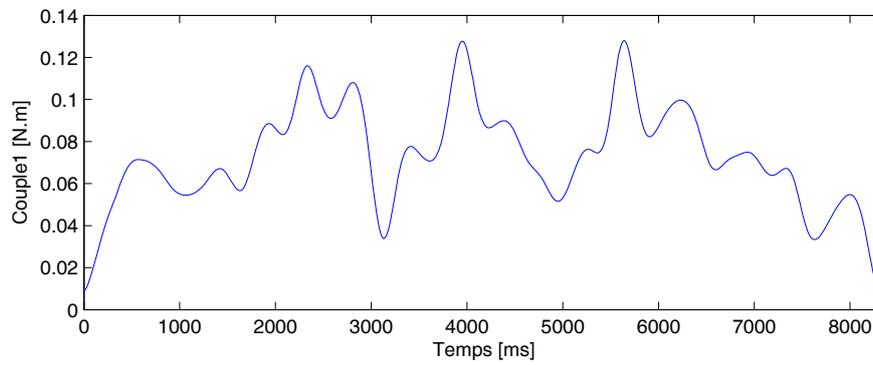


Figure 3.11: Couple mesuré pour l'axe 1 du robot(Approche neuronale)

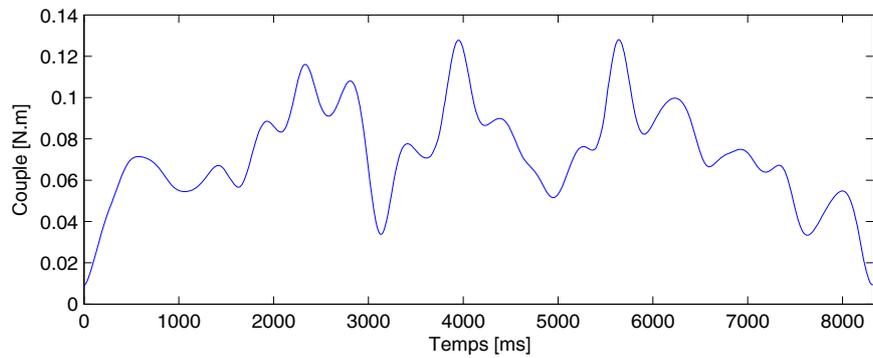


Figure 3.12: Couple mesuré pour l'axe 2 du robot (Approche neuronale)

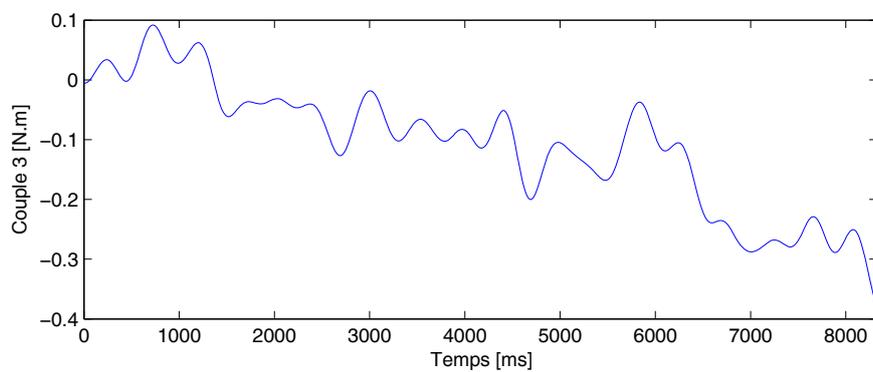


Figure 3.13: Couple mesuré pour l'axe 3 du robot(Approche neuronale).

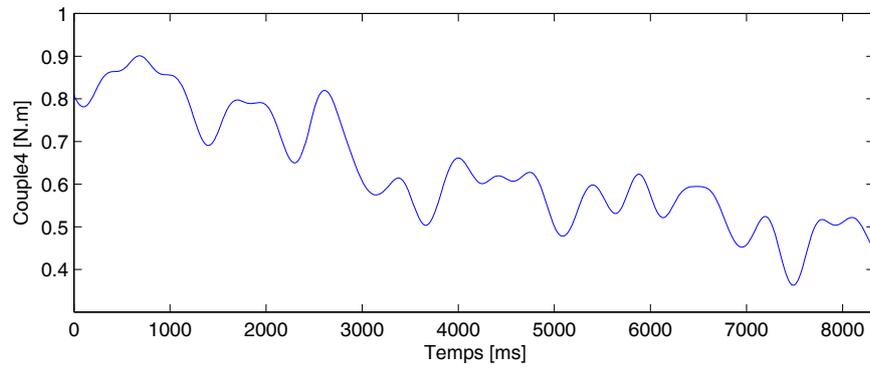


Figure 3.14: Couple mesuré pour l'axe 4 du robot ( Approche neuronale).

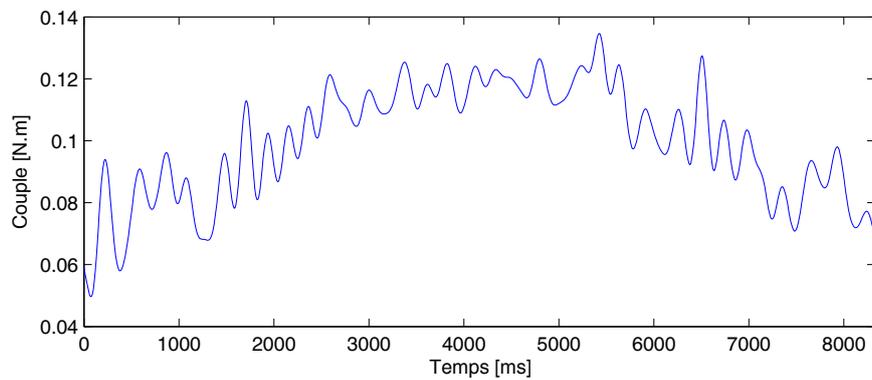


Figure 3.15: Couple mesuré pour l'axe 1 du robot (Contrôleur PID).

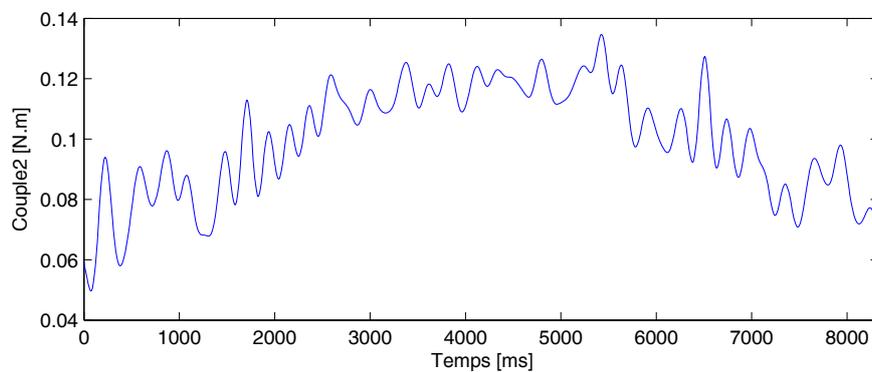


Figure 3.16: Couple mesuré pour l'axe 2 du robot (Contrôleur PID).

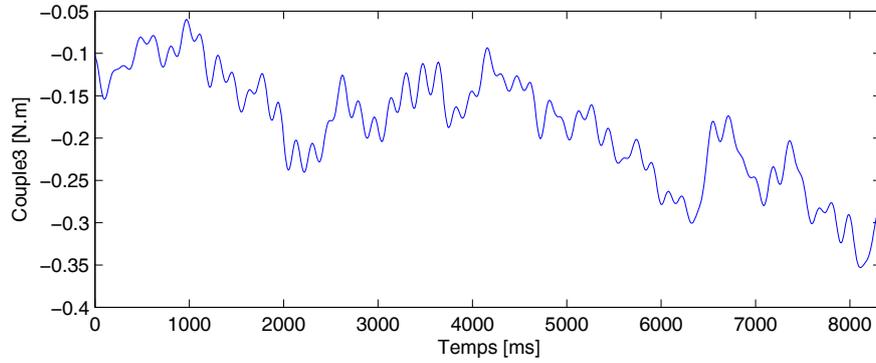


Figure 3.17: Couple mesuré pour l'axe 3 du robot (Contrôleur PID).

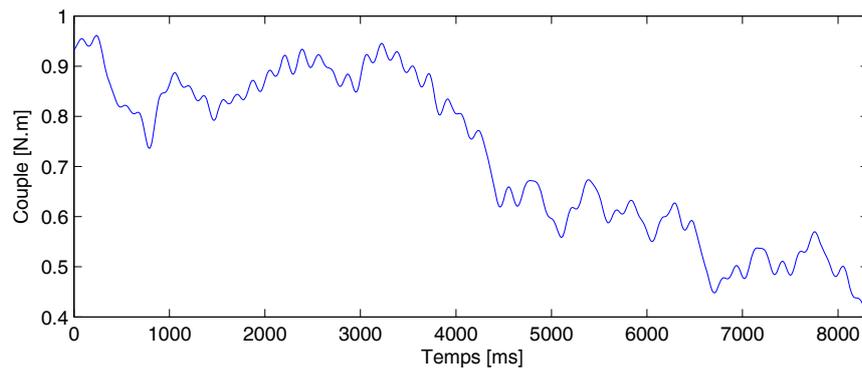


Figure 3.18: Couple mesuré pour l'axe 4 du robot (Contrôleur PID).

### 3.5 Conclusion

Dans ce chapitre, nous avons développé et validé expérimentalement une commande adaptative neuronale directement dans l'espace de la tâche. Cette commande est basée sur un réseau de neurones à paramétrisation non linéaire dont les paramètres sont adaptés en ligne selon une étude de stabilité en boucle fermée au sens de Lyapunov. Nous rappelons que, dans l'élaboration de notre approche de commande, l'accélération n'a pas été utilisée en vue d'éviter les mesures de l'accélération fortement bruitées et inexploitable. Les résultats de cette commande sont comparés à ceux du contrôleur PID. A travers les résultats expérimentaux obtenus, on peut facilement remarquer une nette amélioration de la qualité de poursuite apportée par la commande neuronale adaptative. L'avantage majeur de cette approche est qu'aucune connaissance à priori ni du modèle de robot ni de son environnement n'est considérée pour sa conception. L'approche proposée peut également pallier aux problèmes des paramètres incertains de l'environnement. Dans le chapitre suivant, nous allons adapter cette approche et l'implémenter sur un autre type de système robotisé qui est un robot portable de type exosquelette de genou.

---

# Chapitre 4

## Commande adaptative d'un robot portable de type exosquelette du genou

---

### 4.1 Introduction

Le robot parallèle à liaisons C5 que nous avons considéré jusqu'à présent est prévu pour des tâches de manipulation d'objets sans aucune interaction avec l'homme. Les schémas de commande et d'identification neuronale présentés dans ces chapitres, s'adaptent parfaitement à des problématiques de restauration de mouvement dans un contexte d'aide à la personne à travers des robots portables (wearable robots). Le rôle du réseau de neurones dans ce cas est d'approcher les fonctions inconnues du modèle dynamique résultant qui est représenté non seulement par les paramètres de l'exosquelette mais également par les paramètres anthropomorphiques ainsi que les paramètres musculo-squelettiques difficilement indentifiables. En effet, les méthodes conventionnelles de modélisation ou d'identification ne peuvent en aucun cas tenir compte de ces dynamiques variables d'un individu à un autre. Chaque personne a une morphologie unique et les mouvements à restaurer lui sont spécifiques et différents comparativement à d'autres personnes. Par ailleurs, le porteur de

l'exosquelette peut être actif en contribuant au mouvement comme il peut aussi être complètement passif en ne fournissant aucun effort musculaire. Les paramètres neuronaux peuvent être fixes ou adaptatifs mais nous nous sommes intéressés uniquement à l'aspect adaptatif.

## 4.2 Modèle dynamique du système

Le modèle dynamique du système composé par l'orthèse et le genou du porteur est donnée par [Mefo 11b, Rifa 12b]:

$$\tau + \tau_h = J\ddot{q} + \tau_g \cos(q) + A \text{sign}\dot{q} + B\dot{q} \quad (4.1)$$

où :

- $\tau$  représente le couple délivré par l'exosquelette,
- $\tau_h$  représente le couple délivré par le porteur de l'orthèse. Ce couple est borné :  $|\tau_h| \leq \delta \geq 0$ . Donc, la personne peut délivrer seulement un effort limité.
- $J = J_{or} + J_h$  est l'inertie du système (orthèse (*or*) + genou humain (*h*)),
- $\tau_g \cos(q) = (\tau_{g_{or}} + \tau_{g_h}) \cos(q)$  représente la force liée à la gravité,
- $A \text{sign}\dot{q} = (A_{or} + A_h) \text{sign}\dot{q}$  représente les frottements secs,
- $B\dot{q} = (B_{or} + B_h)\dot{q}$  représente les frottements visqueux,
- $q$  et  $q_d$  sont respectivement les positions mesurée et désirée,
- $\dot{q}$  et  $\dot{q}_d$  représentent respectivement les vitesses mesurée et désirée,
- $\ddot{q}$  et  $\ddot{q}_d$  représentent respectivement les accélérations mesurée et désirée

Le modèle dynamique du système (Orthèse + genou) décrit par l'équation (4.1) peut être écrit sous la forme:

$$\tau + \tau_h = J\ddot{q} + \beta(q, \dot{q}) \quad (4.2)$$

$$\beta(q, \dot{q}) = \tau_g \cos(q) + A \text{sign}\dot{q} + B\dot{q} \quad (4.3)$$

Le porteur de l'orthèse peut :

1. Exercer un effort musculaire résistif :  $\tau_h \neq 0$  et  $\tau_h \cdot \tau \leq 0$
2. Exercer un effort musculaire assistant :  $\tau_h \neq 0$  et  $\tau_h \cdot \tau \geq 0$
3. Etre complètement passif :

$$\tau_h = 0$$

Dans ce cas:

$$\tau = J\ddot{q} + \beta(q, \dot{q}) \quad (4.4)$$

- Dans la synthèse de la loi de commande proposée, on considère seulement connue la structure du modèle dynamique. C'est à dire  $J$  et  $\beta$  sont complètement inconnus. Ils seront estimés en ligne selon une étude de stabilité en boucle fermée. Pour l'estimation de  $\beta$ , un réseau de neurones de type MLP est utilisé. L'adaptation de ses paramètres se fait en ligne.

Concernant  $J$ , on utilise une simple loi d'adaptation que l'on détaillera plus loin.

### 4.3 Approximation neuronale

Le réseau de neurones considéré est le Perceptron Multicouches (MLP) avec une seule couche cachée et une sortie linéaire. La fonction à approcher est  $\beta(z)$ . Elle est approximée par un réseau de neurones de type MLP. Les composantes du vecteur  $z$  sont  $q$  et  $\dot{q}$ . Le problème de nonlinéarités présenté par le MLP sera traité de la même façon qu'en chapitre 3. La fonction à approximer  $\beta(z)$  par le MLP peut être écrite comme suit:

$$\beta(z) = w_2^{*T} \varphi(w_1^{*T} z) + \epsilon(z) \quad (4.5)$$

où  $\epsilon$  est l'erreur d'approximation neuronale, tel que ( $\|\epsilon\| \leq \bar{\epsilon}$ ), avec  $\bar{\epsilon}$  connu et suffisamment petit.  $w_1^{*T} \in \mathfrak{R}^{h_c \times 1}$  et  $w_2^{*T} \in \mathfrak{R}^{1 \times h_c}$  sont les poids des matrices entre l'entrée et la couche cachée et entre la couche cachée et la sortie du MLP respectivement. Avec :

- $h_c$  le nombre de cellules dans la couche cachée.

- La fonction d'activation  $\varphi$  a une forme sigmoïdale.

L'estimation neuronale de  $\beta(z)$  peut être exprimée telle que:

$$\hat{\beta}(z) = w_2^T \varphi(w_1^T z) \quad (4.6)$$

Les paramètres neuronaux  $w_1$  et  $w_2$  représentent l'estimation de  $w_1^*$  et  $w_2^*$  respectivement. Comme on peut le voir par la suite, ces paramètres seront ajustés en ligne selon une étude de stabilité en boucle fermée au sens de Lyapunov.

Le développement en série de Taylor de la fonction  $\varphi(w_1^{*T} z)$  autour des paramètres estimés ( $w_1^T z$ ) peut être écrit comme suit:

$$\varphi(w_1^{*T} z) = \varphi(w_1^T z) - \varphi'(w_1^T z) \tilde{w}_1^T z - O_s \quad (4.7)$$

$$\tilde{w}_1 = w_1 - w_1^* \quad (4.8)$$

où  $\varphi'(\vartheta) = \left. \frac{d\varphi(\varrho)}{d\varrho} \right|_{\varrho=\vartheta}$  et  $O_s$  sont des termes d'ordre supérieur avec:

$$O_s = (\varphi(w_1^T z) - \varphi(w_1^{*T} z)) - \varphi'(w_1^T z) \tilde{w}_1^T z \quad (4.9)$$

Après développement, on obtient:

$$\beta - \hat{\beta} = \tilde{\beta} = -w_2^T \varphi' \tilde{w}_1^T z - \tilde{w}_2^T \varphi + \epsilon - \epsilon_\varphi \quad (4.10)$$

$$\epsilon_\varphi = -\tilde{w}_2^T \varphi \tilde{w}_1^T z + w_2^{*T} O_s \quad (4.11)$$

$\epsilon_\varphi$  représente les erreurs d'approximation issues du développement en série de Taylor d'ordre 1.

## 4.4 Loi de commande et étude de stabilité

Notre objectif consiste à commander l'exosquelette avec le minimum de connaissances sur le modèle dynamique du genou/exosquelette dans un contexte de rééducation. Pour ce système, l'effort musculaire pouvant être délivré par le porteur de l'orthèse, est un élément nouveau par rapport aux robots manipulateurs. Cet effort

se traduisant par un couple, peut être traité comme une perturbation lorsqu'il est en opposition à ce qui est désiré ou comme un supplément de commande lorsqu'il est en harmonie avec le mouvement. Au travers des capteurs installés sur la personne (genou, cuisse, etc.), il est possible de récupérer l'intention du porteur pour la traduire en un mouvement désiré. Le modèle de l'exosquelette est un modèle dynamique pouvant être assimilé à un modèle d'un robot manipulateur. En conséquence, il est tout à fait possible d'adapter le schéma de commande proposé dans le chapitre précédent.

La Figure 4.1 illustre le schéma de commande proposé. La loi de commande

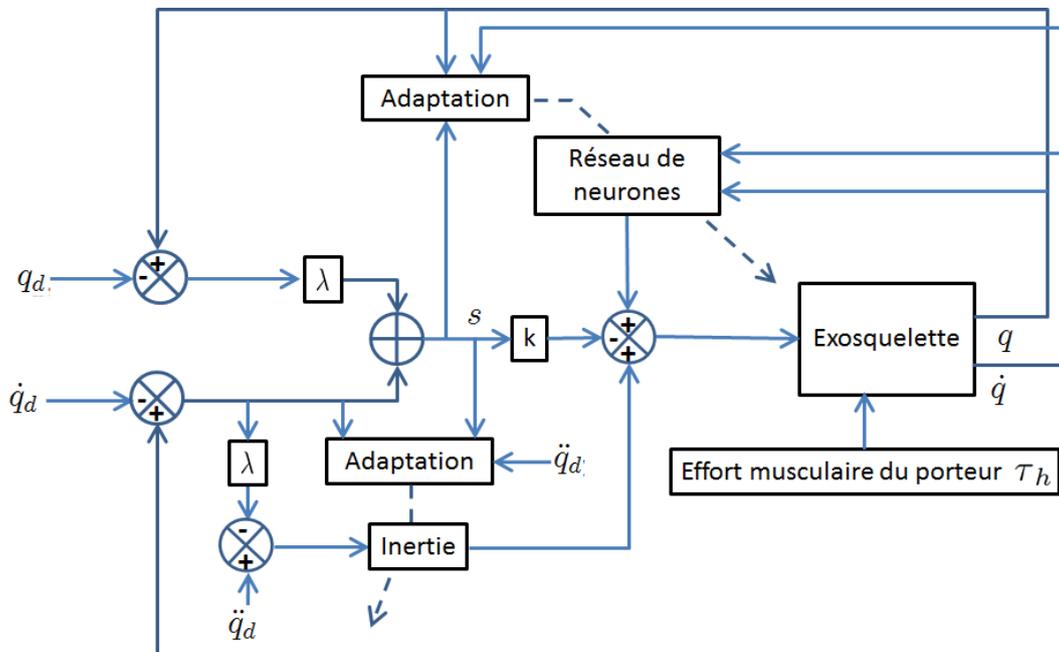


Figure 4.1: Commande neuronale adaptative de l'exosquelette.

proposée est donnée comme suit:

$$\tau = \hat{J}v + \hat{\beta}(q, \dot{q}) - ks \quad (4.12)$$

avec:

$$v = \ddot{q}_d - \lambda \dot{e}$$

$$s = \dot{e} + \lambda e$$

$$e = q - q_d$$

$$\dot{e} = \dot{q} - \dot{q}_d$$

$$\ddot{e} = \ddot{q} - \ddot{q}_d$$

où:

- $\lambda$  et  $k$  sont des gains (scalaires) positifs;
- $q$  est la position réelle du genou,  $q_d$  est la position désirée;
- $\dot{q}$  est la vitesse réelle du genou et  $\dot{q}_d$  est la vitesse désirée;
- $\ddot{q}$  est l'accélération réelle du genou,  $\ddot{q}_d$  est l'accélération désirée;

A partir de l'analyse de stabilité en boucle fermée, l'adaptation des paramètres neuronaux du contrôleur sont déterminés via une analyse de stabilité et sont donnés comme suit:

$$\begin{cases} \dot{\hat{J}} = -a_1 v s \\ \dot{w}_2 = -\mu_2 \varphi(w_1^T z) s \\ \dot{w}_1 = -\mu_1 z s w_2^T \varphi'(w_1^T z) \end{cases} \quad (4.13)$$

avec  $a_1$ ,  $\mu_1$  et  $\mu_2$  sont des gains positifs.

Pour l'analyse de stabilité, considérons le système en boucle fermée alors les équations suivantes peuvent être écrites:

$$J\ddot{q} + \beta(q, \dot{q}) = \hat{J}(\ddot{q}_d - \lambda \dot{e}) + \hat{\beta}(q, \dot{q}) - ks$$

$$\begin{aligned}
-ks &= J\ddot{q} - \hat{J}(\ddot{q}_d - \lambda\dot{e}) + \beta(q, \dot{q}) - \hat{\beta}(q, \dot{q}) - \tau_h \\
&= J\ddot{q} - (J - \tilde{J})(\ddot{q}_d - \lambda\dot{e}) + \tilde{\beta}(q, \dot{q}) - \tau_h \\
&= J\ddot{q} - J(\ddot{q}_d - \lambda\dot{e}) + \tilde{J}(\ddot{q}_d - \lambda\dot{e}) + \tilde{\beta}(q, \dot{q}) - \tau_h \\
&= J(\ddot{e} + \lambda\dot{e}) + \tilde{J}(\ddot{q}_d - \lambda\dot{e}) + \tilde{\beta}(q, \dot{q}) - \tau_h \\
&= J\dot{s} + \tilde{J}(\ddot{q}_d - \lambda\dot{e}) + \tilde{\beta}(q, \dot{q}) - \tau_h \\
J\dot{s} &= \tau_h - \tilde{J}(\ddot{q}_d - \lambda\dot{e}) - \tilde{\beta}(q, \dot{q}) - ks \tag{4.14}
\end{aligned}$$

$$J\dot{s} = \tau_h - ks - \tilde{\beta} - \tilde{J}v \tag{4.15}$$

avec  $\tilde{\beta} = \beta - \hat{\beta}$  et  $\tilde{J} = J - \hat{J}$ .

Dans ce qui suit, on montre que le système genou/exosquelette est stable dans deux cas. En premier temps, l'opérateur ne fournit aucun effort. Dans ce cas, le système est asymptotiquement stable. Dans le second cas, l'opérateur est considéré actif. le système est prouvé stable en tenant compte du couple bornée fournit de la part de l'opérateur.

#### 4.4.1 Analyse de la stabilité lorsque le porteur de l'orthèse est passif

Dans ce premier cas, le sujet porteur ne fournit aucun effort musculaire:

$$\tau_h = 0$$

Considérons la fonction de Lyapunov candidate positive et sa dérivée par rapport au temps:

$$V = \frac{1}{2}Js^2 + \frac{1}{2a_1}\tilde{J}^2 + \frac{1}{2}tr(\tilde{w}_1^T\eta_1^{-1}\tilde{w}_1) + \frac{1}{2}tr(\tilde{w}_2^T\eta_2^{-1}\tilde{w}_2) \tag{4.16}$$

$$\dot{V} = sJ\dot{s} + \frac{1}{a_1}\tilde{J}\dot{\tilde{J}} + tr(\tilde{w}_1^T\eta_1^{-1}\dot{\tilde{w}}_1) + tr(\tilde{w}_2^T\eta_2^{-1}\dot{\tilde{w}}_2) \tag{4.17}$$

En utilisant les équations (4.10), (4.14) et (4.17), on obtient:

$$\begin{aligned}
\dot{V} &= s(-\tilde{J}(\ddot{q}_d - \lambda\dot{e}) + w_2^T\phi'\tilde{w}_1^Tz + \tilde{w}_2^T\phi - \epsilon + \epsilon_\phi - ks) \\
&\quad + \frac{1}{a_1}\tilde{J}\dot{\tilde{J}} + tr(\tilde{w}_1^T\eta_1^{-1}\dot{\tilde{w}}_1) + tr(\tilde{w}_2^T\eta_2^{-1}\dot{\tilde{w}}_2)
\end{aligned}$$

$$\begin{aligned}
\dot{V} &= -s\tilde{J}(\ddot{q}_d - \lambda\dot{e}) + sw_2^T \varphi' \tilde{w}_1^T z + s \tilde{w}_2^T \varphi + s(\epsilon_\varphi - \epsilon) \\
&+ \frac{1}{a_1} \tilde{J} \dot{\tilde{J}} - sks + tr(\tilde{w}_1^T \eta_1^{-1} \dot{w}_1) + tr(\tilde{w}_2^T \eta_2^{-1} \dot{w}_2) \\
&= \dot{V}_1 + \dot{V}_2 + \dot{V}_3 + \dot{V}_4
\end{aligned}$$

avec

$$\begin{aligned}
\dot{V}_1 &= -sks + s(\epsilon_\varphi - \epsilon) & (4.18) \\
\dot{V}_2 &= sw_2^T \varphi' \tilde{w}_1^T z + tr(\tilde{w}_1^T \eta_1^{-1} \dot{w}_1) \\
\dot{V}_3 &= s\tilde{w}_2^T \varphi + tr(\tilde{w}_2^T \eta_2^{-1} \dot{w}_2) \\
\dot{V}_4 &= -s\tilde{J}(\ddot{q}_d - \lambda\dot{e}) + \frac{1}{a_1} \tilde{J} \dot{\tilde{J}}
\end{aligned}$$

En utilisant les lois d'adaptation donnée par 4.13, nous avons:

$$\begin{aligned}
\dot{V}_2 &= sw_2^T \varphi' \tilde{w}_1^T z + tr(\tilde{w}_1^T \eta_1^{-1} \dot{w}_1) = 0 \\
\dot{V}_3 &= s\tilde{w}_2^T \varphi + tr(\tilde{w}_2^T \eta_2^{-1} \dot{w}_2) = 0 \\
\dot{V}_4 &= -s\tilde{J}(\ddot{q}_d - \lambda\dot{e}) + \frac{1}{a_1} \tilde{J} \dot{\tilde{J}} = 0
\end{aligned}$$

En négligeant les erreurs d'approximation neuronale :

$$\dot{V}_1 = -ks^2 \leq 0 \quad (4.19)$$

Donc si  $s \neq 0$  alors  $\dot{V} < 0$  et si  $s = 0$  alors  $\dot{V} = 0$ . Par ailleurs, nous pouvons conclure que  $s$ ,  $\tilde{w}_1$ ,  $\tilde{w}_2$  et  $\tilde{J}$  sont bornés. Par le biais du lemme de Barbalat, nous pouvons dire que  $s$  tend vers 0 du fait que  $\ddot{V}$  est vérifiée bornée. Finalement, la loi de commande (4.12) assure que l'orthèse lorsqu'elle est en mouvement libre, est asymptotiquement stable ( $q \rightarrow q_d$  et  $\dot{q} \rightarrow \dot{q}_d$ ).

Considérons les erreurs d'approximation neuronale :

$$\dot{V}_1 = -ks^2 + s(\epsilon_\varphi - \epsilon) \quad (4.20)$$

$$\leq -ks^2 + |s||\epsilon_\varphi - \epsilon| \quad (4.21)$$

En utilisant l'inégalité de Young, nous pouvons écrire:

$$\begin{aligned} |s||\epsilon_\varphi - \epsilon| &\leq \frac{ks^2}{2} + \frac{(\epsilon_\varphi - \epsilon)^2}{2k} \\ \dot{V}_1 &\leq -ks^2 + \frac{ks^2}{2} + \frac{(\epsilon_\varphi - \epsilon)^2}{2k} \\ &\leq -\frac{k}{2}s^2 + \frac{(\epsilon_\varphi - \epsilon)^2}{2k} \end{aligned}$$

Après un temps fini d'adaptation,  $(\epsilon_\varphi - \epsilon)^2$  devient négligeable par rapport à  $2k$ . Alors, en utilisant la même analyse que celle dans le premier cas, il nous est possible de montrer la convergence des erreurs  $e$  et  $\dot{e}$  et prouver ainsi la stabilité du système en boucle fermée..

#### 4.4.2 Analyse de stabilité lorsque le porteur de l'orthèse est actif

Dans ce paragraphe, le sujet exerce un effort musculaire borné que l'on peut considérer comme un couple externe ( $|\tau_h| \leq \rho \geq 0$ ). La stabilité du système est étudiée en considérons la même fonction de Lyapunov donnée par (4.16). En utilisant les lois d'adaptation données par (4.13) et le couple externe  $\tau_h$ , la dérivée de  $V$  par rapport au temps est donnée par:

$$\begin{aligned} \dot{V} &= -ks^2 + \tau_h s + s(\epsilon_\varphi - \epsilon) \\ &= -(ks - \tau_h - (\epsilon_\varphi - \epsilon))s \\ &\leq -ks^2 + |\tau_h||s| + |s||\epsilon_\varphi - \epsilon| \end{aligned}$$

En utilisant l'inégalité de Young, nous pouvons facilement vérifier que:

$$\begin{aligned} |\tau_h||s| &\leq \frac{k}{4}s^2 + \frac{\tau_h^2}{k} \\ |s||\epsilon_\varphi - \epsilon| &\leq \frac{k}{4}s^2 + \frac{(\epsilon_\varphi - \epsilon)^2}{k} \end{aligned}$$

Donc:

$$\dot{V} \leq -\frac{k}{2}s^2 + \frac{\tau_h^2}{k} + \frac{(\epsilon_\varphi - \epsilon)^2}{k} \quad (4.22)$$

Comme dans le deuxième cas de la précédente section, après un temps fini d'adaptation,  $(\epsilon_\varphi - \epsilon)^2$  devient négligeable le comparant à  $k$ . Alors nous avons :

$$\begin{aligned}\dot{V} &\leq -\frac{k}{2}s^2 + \frac{\tau_h^2}{k} \\ |\tau_h| \leq \rho \geq 0 &\Rightarrow \frac{\tau_h^2}{k} \text{ borné} \\ \tau_h \rightarrow 0 &\Rightarrow \frac{\tau_h^2}{k} \rightarrow 0\end{aligned}$$

Il existe un scalaire  $\delta \rightarrow 0$  lorsque  $\tau_h \rightarrow 0$  tel que:

$$\dot{V} < 0 \quad \forall |s| \geq \delta. \quad (4.23)$$

Considérons la stabilité asymptotique relative au mouvement libre de l'orthèse ( $\tau_h = 0$ ) et les équations (4.16), (4.22) et (4.23), nous pouvons conclure que le système est stable en boucle fermée [Rifa 12b].

## 4.5 Résultats expérimentaux

La commande proposée a été implémentée sur le système représenté par l'exosquelette porté par un individu sain. Nous avons utilisé un réseau de neurones de type MLP ayant trois cellules cachées. Les fonctions d'activation utilisées sont des sigmoïdes. Les valeurs initiales des coefficients synaptiques sont données de façon aléatoire et comprises entre 1 et 0. Notre implémentation consiste en l'application de l'approche neuronale proposée sur une personne saine assise dans un mouvement flexion / extension du genou selon deux situations tel qu'il est précisé dans l'étude de stabilité: le porteur de l'orthèse reste inactif et ne fournit aucun effort dans le premier cas. Dans le second cas, la personne exerce un effort où une perturbation est également rajoutée pour s'assurer de la robustesse de l'approche proposée vis à vis des perturbations extérieures. Rappelons que si l'effort fourni est dans le sens de mouvement de l'orthèse, le couple total sera la somme des deux couples celui délivré par le moteur et le couple image de l'effort exercé de la part de la personne. Dans le cas où

l'effort fourni est dans le sens contraire du mouvement de l'orthèse, il sera considéré comme une perturbation extérieure ce qui nous permet de tester la robustesse de l'approche proposée.

La trajectoire à suivre dans les deux cas est définie par:

$$\theta(rd) = \frac{\Pi}{6} \sin(2\Pi ft) + \frac{\Pi}{4}$$

Les paramètres (poids) du réseau de neurones sont initialisés de façon aléatoire entre 0 et 1. Les caractéristiques du réseau de neurones, les gains d'adaptation ( $\eta_1, \eta_2, a_1$ ), les paramètres de la loi de commande neuronale proposée ( $k, \lambda$ ) et les gains proportionnel et dérivé ( $kp, kv$ ) du contrôleur PID sont donnés dans le tableau suivant:

Nombre d'entrées du réseau de neurones	2
Nombre de sorties du réseau de neurones	1
Nombre de couches cachées du réseau de neurones	1
Nombre de neurones dans la couche cachée	3
$\eta_1$	0.45
$\eta_2$	0.45
$a_1$	0.001
$k$	0.45
$\lambda$	65
$kp$	35
$kv$	5

#### 4.5.1 Le porteur de l'orthèse est passif ( $\tau_h = 0$ )

Dans ce cas, l'approche proposée est testée sur une personne saine et n'exerce aucun effort ( $\tau_h = 0$ ). La qualité de la poursuite de trajectoire en position, comme

on peut le voir sur les figure 4.2, 4.3 est très satisfaisante pour l'approche neuronale cependant qu'elle est moins bonne dans le cas du PID. Ce que l'on peut constater également sur la figure 4.4 représentant l'erreur de poursuite en position. La phase d'adaptation des paramètres neuronaux peut être constatée sur la figure 4.9 représentant la sortie du réseau de neurones. En visualisant les figures 4.5 et 4.6 représentant la poursuite en vitesse, on remarque facilement qu'en cas de l'approche neuronale, la poursuite en vitesse est satisfaisante inversement au cas du contrôleur PID qui est médiocre. Les couples relatifs aux deux contrôleurs sont donnés dans les figures 4.7 et 4.8 respectivement.

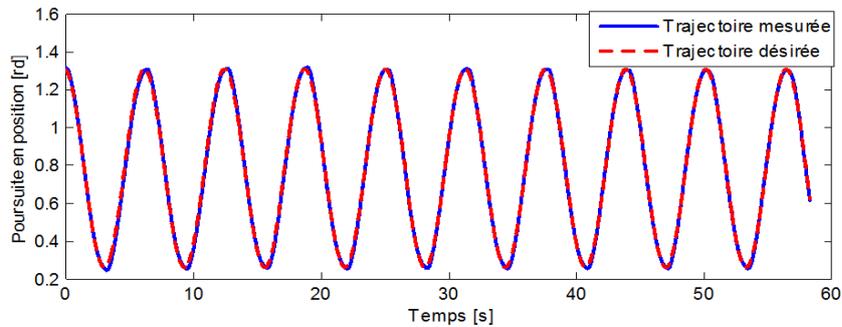


Figure 4.2: Poursuite de trajectoire en position (Approche neuronale).

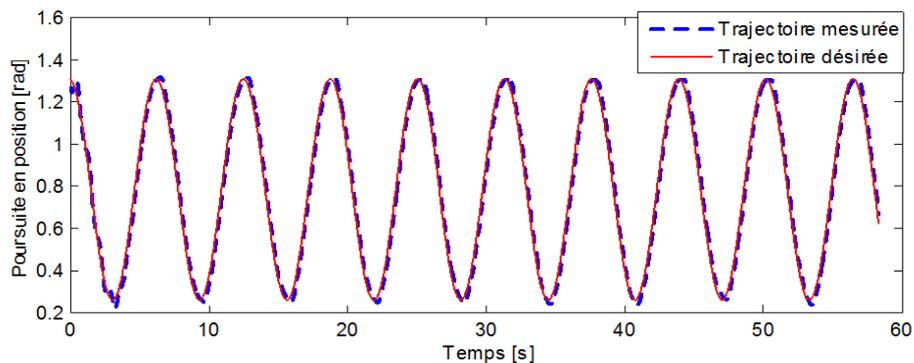


Figure 4.3: Poursuite de trajectoire en position (PID).

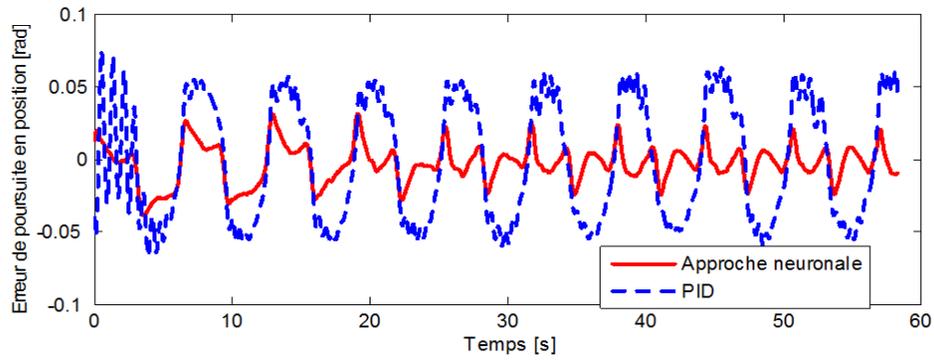


Figure 4.4: Erreur de poursuite de trajectoire en position.

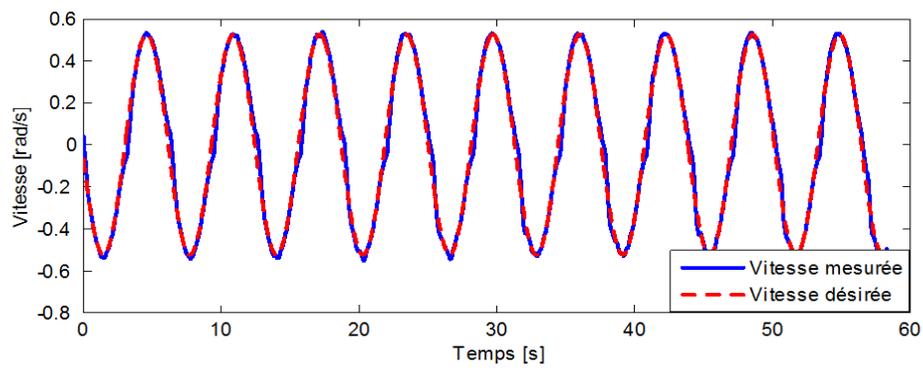


Figure 4.5: Poursuite de trajectoire en vitesse (Approche neuronale).

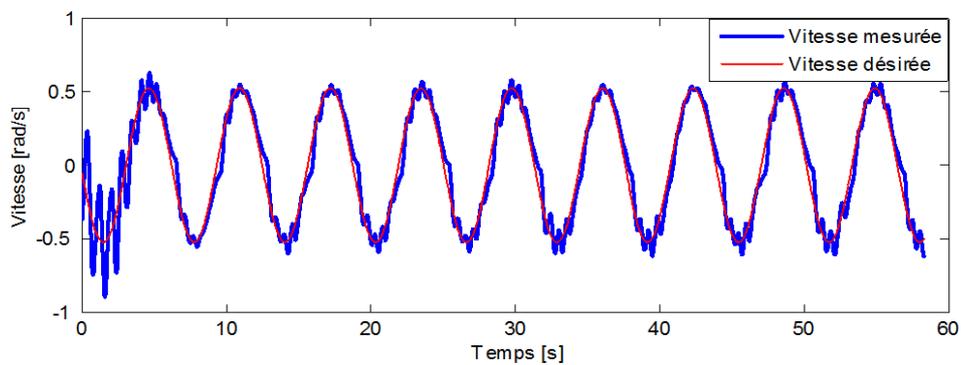


Figure 4.6: Poursuite de trajectoire en vitesse (PID).

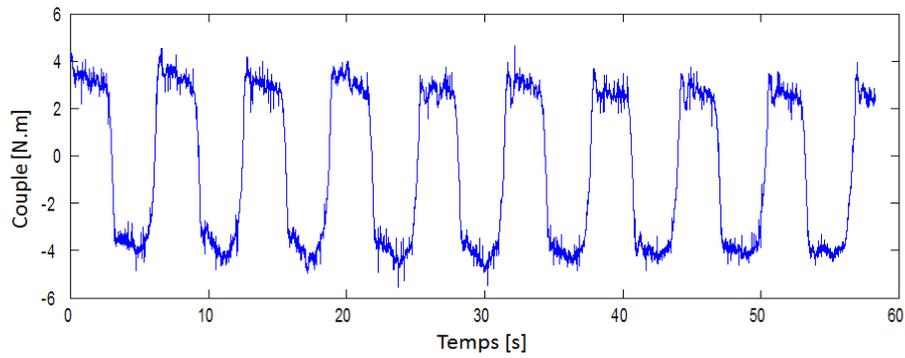


Figure 4.7: Couple de commande (Approche neuronale).

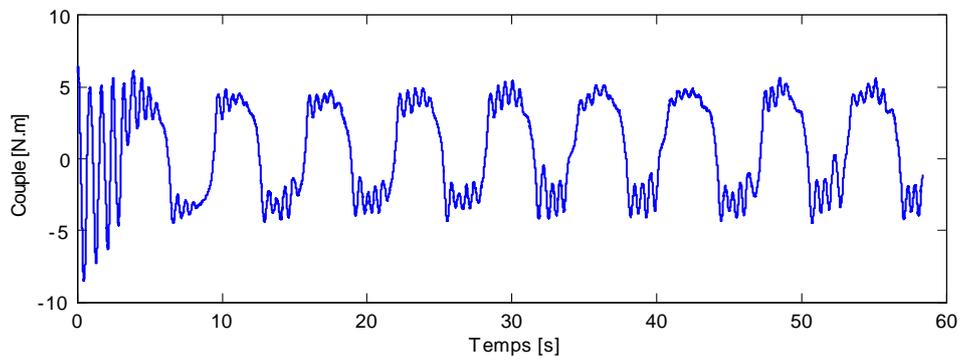


Figure 4.8: Couple de commande (PID).

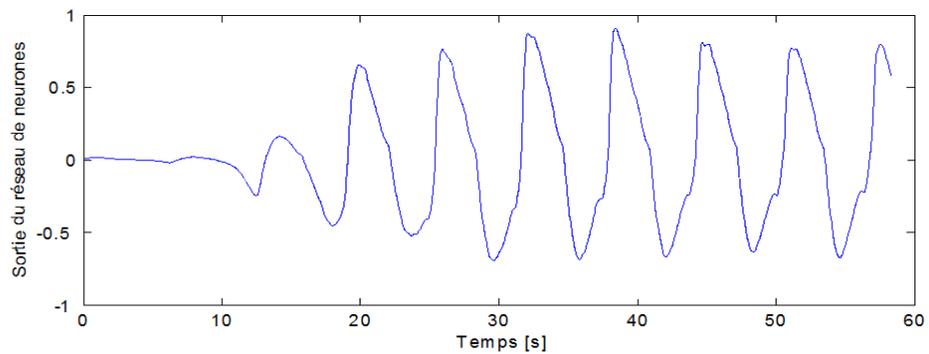


Figure 4.9: Sortie du réseau de neurones.

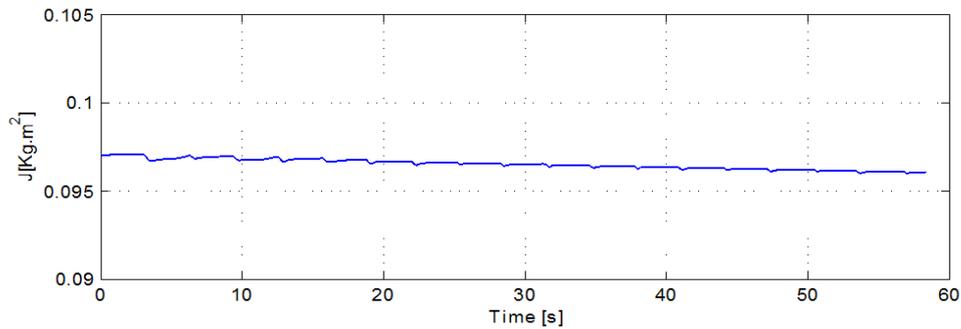


Figure 4.10: Inertie (Approche neuronale).

#### 4.5.2 Le porteur de l'orthèse est actif ( $\tau_h \neq 0$ )

Cette fois-ci, l'opérateur exerce un effort et par conséquent il contribue au mouvement de l'orthèse. Pour tester la robustesse de l'approche proposée vis à vis des perturbations extérieures, une perturbation est introduite entre 36 et 37 s. D'après les figures 4.11 et 4.12 représentant la poursuite en position et l'erreur de poursuite en position, on constate que la qualité de poursuite est toujours satisfaisante. Comme on peut voir sur la figure 4.11, la perturbation extérieure n'as aucun effet sur la poursuite en position.

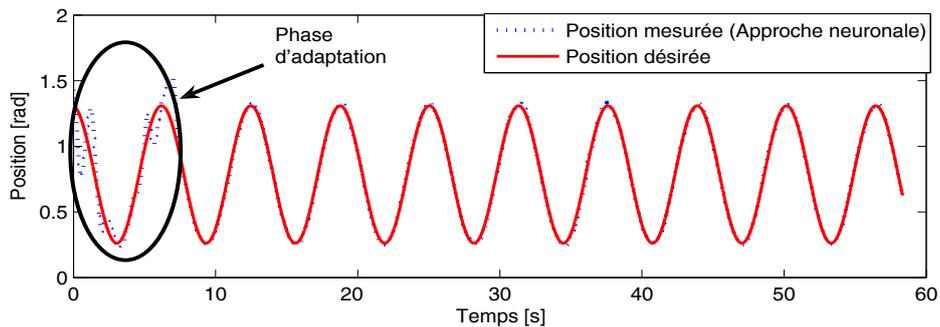


Figure 4.11: Poursuite de trajectoire en position

La poursuite en vitesse est également satisfaisante ce que l'on peut voir sur les

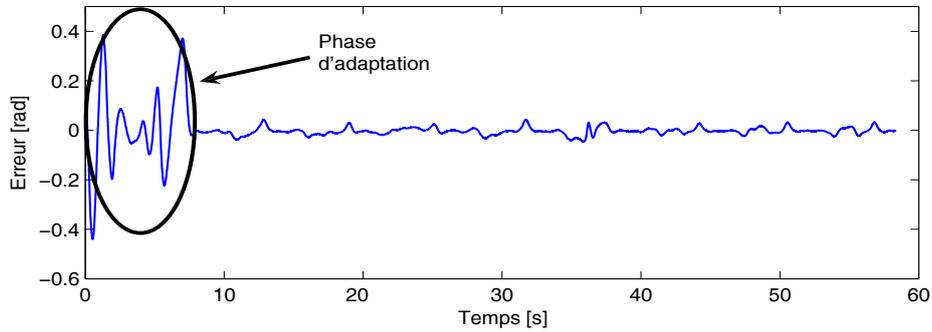


Figure 4.12: Erreur de poursuite de trajectoire en position

figures 4.13 et 4.14 représentant la poursuite en vitesse et l'erreur de poursuite en vitesse.

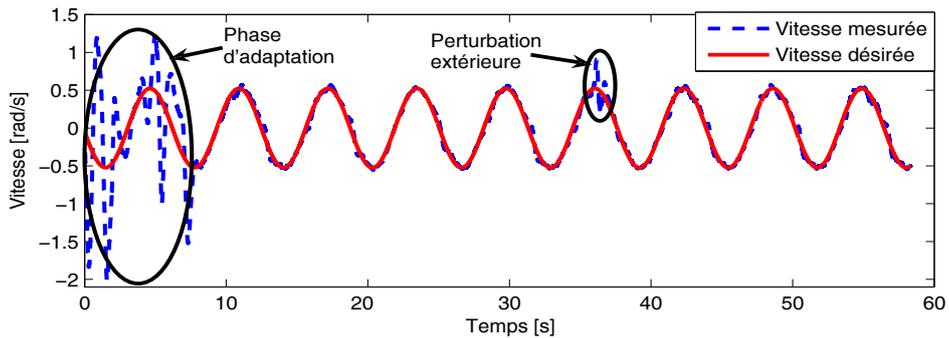


Figure 4.13: Poursuite de trajectoire en vitesse

L'effet de la perturbation extérieure, qui est rapidement rejetée, apparaît juste sur la vitesse entre 36 et 37 s. La sortie du réseau de neurones est donnée par la figure 4.15. L'inertie est représentée par la figure 4.16. Les résultats obtenus montrent la robustesse de l'approche proposée vis à vis des perturbations extérieures.

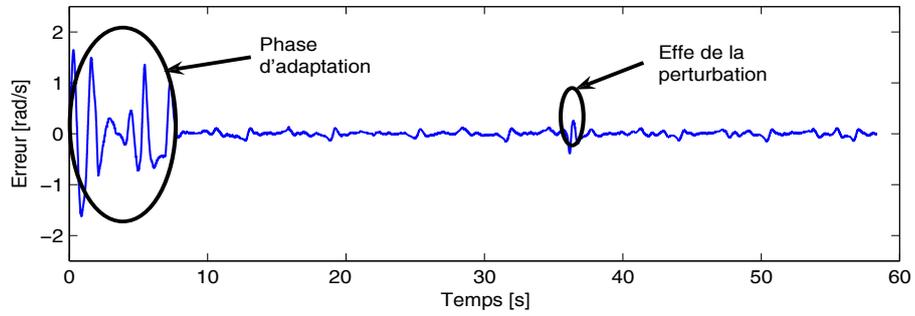


Figure 4.14: Erreur de poursuite en vitesse

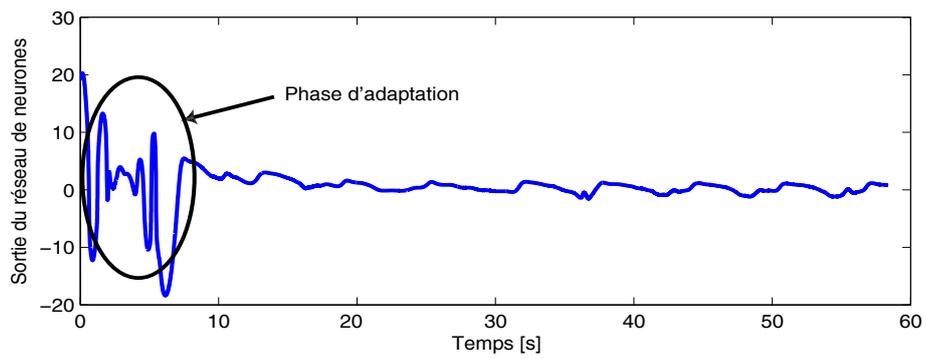


Figure 4.15: Sortie du réseau de neurones

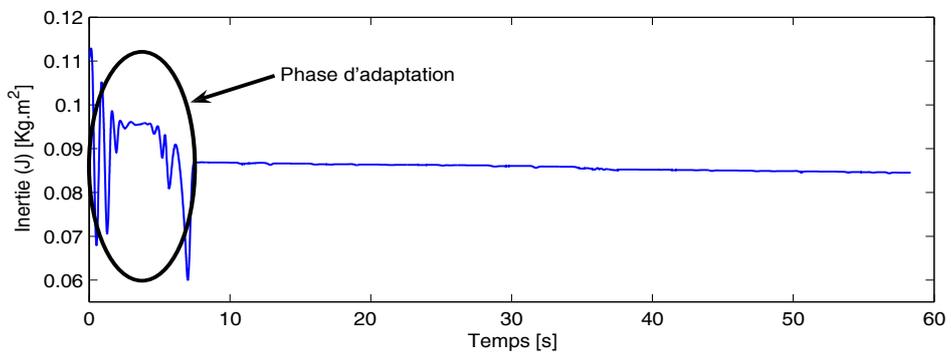


Figure 4.16: Inertie estimée.

## 4.6 Conclusion

Dans ce chapitre, nous avons adapté et expérimenté une loi de commande sur un autre système réel. Ce système a la particularité d'épouser directement une partie de l'homme représentée par son genou. Dans les lois de commande conventionnelles orientées modèle, il est prévu une identification du système complet et d'appliquer un schéma de commande utilisant les paramètres identifiés a priori. L'inconvénient de ces commandes est la dépendance entre les paramètres identifiés propres à chaque utilisateur et les performances de commande en termes de suivi de trajectoire. Ces commandes supposent en plus que l'identification soit parfaite et que les paramètres dynamiques ne sont pas variables. L'étape d'identification, réalisée hors ligne, peut être intéressante uniquement à l'initialisation dans un schéma de commande adaptative. L'aspect adaptatif, qui caractérise tous nos schémas de commande jusqu'à présent, a montré son efficacité sur ce nouveau système qui paraît simple en le comparant au robot parallèle à liaisons C5, mais en réalité la dynamique musculaire rend le modèle musculo-squelettique fortement non linéaire. En conséquence la complexité du système global augmente de façon considérable. Si on rajoute à cela le contact avec le sol pour des activités de locomotion (montée/descente des escaliers par exemple), d'autres problématiques liées à la modélisation de l'interaction avec le sol, la gestion de l'équilibre, il est indispensable de traiter d'autres aspects de la commande (commande force/position par exemple). Dans la dernière partie de ce manuscrit, représentée par la conclusion et perspectives, nous nous étalerons avec plus de détails sur les points que nous estimons possibles à réaliser dans l'immédiat.

---

# Conclusion et perspectives

---

Le travail réalisé dans cette thèse s'est basé sur l'aspect pratique où les approches proposées ont été implémentées sur deux prototypes réels disponibles au laboratoire LISSI : le robot parallèle à liaisons C5 et le robot portable de type exosquelette du genou destiné à la rééducation des membres inférieurs. En premier lieu, deux problèmes relatifs aux robots parallèles ont été traités à savoir l'identification et la commande. Ensuite, nous avons proposé une commande adaptative stable du type boîte grise pour réaliser des mouvements de flexion/extension de l'exosquelette. Ce type de mouvements est en général pratiqués par le thérapeute dans le cadre de la rééducation de membres inférieurs.

Les propositions réalisées dans cette thèse sont résumées dans les trois points suivants :

1. Identification expérimentale de la dynamique inverse du robot parallèle. Les schémas d'identification retenus sont de type boîte noire et sont basés sur les réseaux de neurones de type MLP. Trois schémas d'identification ont ainsi été proposés et les résultats obtenus sont considérés satisfaisants pour une application en commande ou pour faire de l'observation. Le réseau de neurones ayant été appliqué a bien appris la dynamique du robot.
2. Commande neuronale adaptative hybride moment/position. Elle a été développée et implémentée sur le robot parallèle C5. Le robot se trouve en contact avec une pièce métallique sphérique. La loi de commande est élaborée directement dans l'espace de la tâche. Le modèle du robot est pris comme une boîte noire. Dans sa conception, la loi de commande proposée ne considère aucune

connaissance ni sur le modèle du robot ni sur l'environnement. La nature de l'approche proposée est extensible et peut être appliquée sur d'autre type de robots.

3. Commande neuronale adaptative d'un exosquelette pour la rééducation. Le même principe adopté pour le robot parallèle est repris pour l'exosquelette du genou. La commande est du type boîte grise et elle considère l'effort pouvant être fourni par le porteur de l'orthèse dans le cas où celui-ci n'est pas complètement passif. Ceci veut dire que le porteur a la possibilité de contribuer au mouvement de flexion/extension. La robustesse face aux perturbations externes est également l'une des caractéristiques de la commande proposée. En utilisant l'approche de Lyapunov, la stabilité du système en boucle fermée a été formellement prouvée.

Ces travaux qui ont été principalement menés au laboratoire LISSI, conduisent à de nombreuses perspectives que nous pouvons résumer ainsi :

1. Pour les approches relatives au robot parallèle à liaisons C5, il serait intéressant de les tester sur d'autres types de robots à architectures parallèles. En effet, ceci pourra nous donner définitivement et de manière expérimentale une idée sur la pertinence de nos différentes propositions.
2. Concernant les travaux sur l'exosquelette que nous avons introduit dans le dernier chapitre, il serait intéressant de considérer d'autres types de mouvements autres que la flexion et l'extension comme la marche. Pour compléter nos schémas de commande, il est important d'examiner l'aspect contrôle en effort. Ce type de mouvement, qui nécessite de prendre en compte le contact avec le sol, met en jeu un modèle dynamique plus complexe avec beaucoup plus de paramètres qui ne sont pas systématiquement tous disponibles. De plus, l'exosquelette n'est considéré que pour un seul genou à la fois. L'introduction d'un deuxième exosquelette permettrait de traiter plus efficacement des problèmes de rééducation mais aussi compliquerait plus les lois de commande en

prenant compte l'aspect synchronisation des deux systèmes.

3. Contrôle/commande du type maître/esclave de l'exosquelette. Le médecin rééducateur est considéré comme maître et le porteur est considéré comme esclave. L'idée est d'essayer de reproduire côté esclave le même mouvement désiré et réalisé côté maître. Deux démarches peuvent être envisagées : la transmission de commande et/ou la transmission de consignes. Le maître doit disposer dans ce cas d'un système identique à celui de l'esclave. Par ailleurs, lorsque le médecin rééducateur a en charge plusieurs patients ayant des pathologies et des caractéristiques physiques différentes, qui en plus, peuvent être distants, les problèmes tels que la gestion du retard de communication et la robustesse vis-à-vis des changements de dynamiques sont des points importants à traiter.

Dans les annexes suivantes, nous essayons de reprendre un certain nombre de notions ayant servi dans la réalisation du présent travail. Il s'agit là d'un aperçu donnant la possibilité au lecteur d'avoir une idée précise sur les différents outils que nous avons employés dans cette thèse. Dans la plupart des cas, nous nous sommes appuyés sur deux références bibliographiques représentant des thèses déjà soutenues dans le domaine de la robotique [Achi 09a][Medd 98].

---

# Annexe A

## Analyse des systèmes par la méthode de Lyapunov

---

**Définition A.1** Une fonction scalaire variante dans le temps  $V(x, t)$  est localement positive définie si  $V(0, t) = 0$  et il existe une fonction positive définie invariante dans le temps  $V_0(x)$  telle que

$$\forall t \geq t_0, V(x, t) \geq V_0(x)$$

**Définition A.2** Une fonction scalaire  $V(x, t)$  est dite décroissante si  $V(0, t) = 0$ , et s'il existe une fonction  $V_1(x)$  définie positive invariante dans le temps telle que

$$\forall t \geq 0, V(x, t) \leq V_1(x)$$

### A.1 Théorème de Lyapunov pour la stabilité des systèmes non-autonomes

**Théorème A.3** Le point d'équilibre  $0$  est stable au sens de Lyapunov, s'il existe, dans une sphère  $B_{R_0}$  autour du point d'équilibre  $0$ , une fonction scalaire  $V(x, t)$ , dont les dérivées partielles sont continues, telle que  $V$  est positive définie et  $\dot{V}$  est négative semi-définie.

**Théorème A.4** *Le point d'équilibre 0 est uniformément stable si en plus de la stabilité,  $V$  est décroissante, le point d'équilibre 0 est uniformément asymptotiquement stable, si la seconde condition du théorème de stabilité est remplacée par la condition suivante :  $\dot{V}$  est négative définie.*

**Théorème A.5** *Si la région de l'espace  $B_{R_0}$  est remplacée par tout l'espace d'état et que toutes les conditions précédentes sont satisfaites et  $V(x, t)$  est radialement non bornée, alors le point d'équilibre 0 est globalement uniformément asymptotiquement stable.*

**Définition A.6** *La fonction  $V$  est dite fonction de Lyapunov du système si, dans un certain voisinage du point d'équilibre,  $V$  est définie positive et  $\dot{V}$ , sa dérivée au long des trajectoires du système, est négative semi-définie..*

## A.2 Systèmes linéaires positifs

**Fonction de transfert PR et SPR:** Soit la fonction de transfert suivante

$$h(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_0}$$

telle que les coefficients  $a_i$  et  $b_i$  sont réels et  $n \geq m$ .

**Définition A.7** *La fonction de transfert  $h(s)$  est dite positive réelle (PR) si*

- $h(s)$  réelle pour  $s$  réelle.
- $Re[h(s)] \geq 0$  quelque soit  $s$  telle que  $Re[s] \geq 0$ .

$h(s)$  est strictement positive réelle (SPR) si

- $\exists \varepsilon > 0$  tel que  $h(s - \varepsilon)$  est positive réelle.

**Théorème A.8** *Une fonction de transfert  $h(s)$  est strictement positive réelle (SPR) si et seulement si*

- $h(s)$  réelle pour  $s$  réelle.

- $h(s)$  est une fonction de transfert strictement stable.
- $\forall \omega, \operatorname{Re}[h(j\omega)] > 0$

---

# Annexe B

## Les réseaux de neurones à couches

---

### B.1 Les perceptrons multicouches (MLP)

Ils ont été introduits par Lewis [Lewi 04][Lewi 96][Cybe 89][Horn 89] pour des systèmes SISO et des systèmes discrets. Un réseau de neurones du type MLP à une couche cachée possède la structure générale suivante:

$$NN(x, w_1, w_2) = w_1^T \varphi(w_2^T x) \quad (\text{B.1})$$

où  $w_1$  et  $w_2$  sont deux vecteurs qui regroupent les paramètres du réseau. La figure (B.1) représente une structure générale du réseau utilisé.

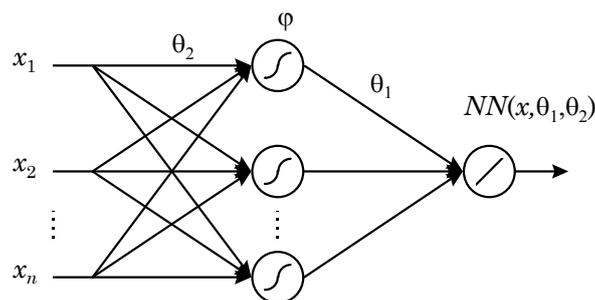


Figure B.1: Structure générale d'un RNF non linéairement paramétré (MLP).

**Remarque B.1** Pour l'approximation des fonctions non nulles à l'origine lorsque la fonction  $\varphi$  est nulle à l'origine il faut augmenter l'état  $x$  d'une valeur constante égale à 1.

Par rapport aux réseaux linéairement paramétrés, l'avantage des réseaux MLP est certainement le nombre relativement réduit de paramètres. Il est clair que ce nombre dépend de la dimension de l'entrée, néanmoins cette dépendance n'est pas exponentielle. D'un autre côté, ces réseaux présentent l'inconvénient d'être non linéairement paramétrés. En effet les techniques adaptatives avec des paramètres qui interviennent de façon non linéaire sont encore à leurs débuts et il n'y a pas suffisamment de méthodologie pour traiter complètement le sujet.

## B.2 Algorithme d'optimisation

Soit un ensemble de mesures d'un processus d'entrée  $\varphi(k)$  et de sortie  $y(k)$ , tel que

$$Z_N = \{(y(k), \varphi(k)) : k = 1, \dots, N\} \quad (\text{B.2})$$

Nous voulons modéliser ce processus par une fonction  $g(\varphi(k), \theta)$ . Le but est d'adapter le vecteur de paramètres  $\theta$  de sorte à minimiser l'écart entre la sortie du modèle et la sortie du processus au sens du critère quadratique.

$$\min_{\theta} J_N(\theta) = \frac{1}{2N} \sum_{k=1}^N \|y(k) - g(\varphi(k), \theta)\|^2 \quad (\text{B.3})$$

En général le minimum de  $J_N(\theta)$  ne peut pas être calculé analytiquement, la minimisation se fait par des procédures numériques.

**L'algorithme de base:** La plupart des algorithmes de recherche sont des algorithmes itératifs qui font une recherche du minimum local à partir du point initial. On a généralement la forme

$$\hat{\theta}_{i+1} = \hat{\theta}_i - \mu_i R_i^{-1} \nabla f_i \quad (\text{B.4})$$

où  $\hat{\theta}_i$  est une estimation des paramètres après la  $i^{\text{ème}}$  itération.

L'algorithme de recherche se base sur les trois valeurs:

- $\mu_i$  le pas.

- $\nabla \hat{f}_i$  une estimation du gradient  $\left. \frac{\partial J_N(\theta)}{\partial \theta} \right|_{\theta=\hat{\theta}_i}$ .
- $R_i$  une matrice qui modifie la direction de la recherche.

Il est important de noter qu'il y a deux types distincts de minimisation:

- *minimisation récursive*: La modification des paramètres ( $\mu_i R_i^{-1} \nabla \hat{f}_i$ ) est basée seulement sur la  $k^{\text{ième}}$  mesure ( $Z_N^k$ ) avec  $k = 1, \dots, N$ .
- *minimisation non récursive (batch)*: La modification des paramètres est basée sur l'ensemble des mesures  $Z_N$ .

**Les directions de recherche:** La base d'une optimisation locale est le gradient

$$\frac{\partial J_N(\theta)}{\partial \theta} = J'_N(\theta) = -\frac{1}{N} \sum_{k=1}^N (y(k) - g(\varphi(k), \theta)) h(\varphi(k), \theta) \quad (\text{B.5})$$

où

$$h(\varphi(k), \theta) = \frac{\partial}{\partial \theta} g(\varphi(k), \theta) \quad (\text{B.6})$$

Pour introduire une direction de recherche dans l'algorithme, on utilise la matrice  $R$ . Cette matrice  $R$  peut prendre plusieurs valeurs suivant la méthode choisie. En voici quelques unes

1. *Méthode de Newton*:  $R$  représente le hessien

$$\begin{aligned} R_i = \left. \frac{\partial^2 J_N(\theta)}{\partial \theta^2} \right|_{\theta=\hat{\theta}_i} &= J''_N(\hat{\theta}_i) = \frac{1}{N} \sum_{k=1}^N h(\varphi(k), \hat{\theta}_i) h^T(\varphi(k), \hat{\theta}_i) \\ &+ \frac{1}{N} \sum_{k=1}^N (y(k) - g(\varphi(k), \hat{\theta}_i)) \frac{\partial^2}{\partial \theta^2} g(\varphi(k), \hat{\theta}_i) \end{aligned} \quad (\text{B.7})$$

2. *Méthode du Gradient*:

$$R_i = I \quad (\text{B.8})$$

3. *Méthode de Gauss-Newton*:

$$R_i = H_i = \frac{1}{N} \sum_{k=1}^N h(\varphi(k), \theta_i) h^T(\varphi(k), \theta_i) \quad (\text{B.9})$$

4. *Méthode Levenberg-Marquard*:

$$R_i = H_i + \delta I \quad (\text{B.10})$$

où  $H_i$  est définie par (B.9) et  $\delta$  est un paramètre utilisé pour éviter la singularité des matrices mal conditionnées.

---

# Références bibliographiques

---

- [Achi 08] B. Achili, B. Daachi, A. Ali-Cherif, and Y. Amirat. “Compensateur neuronal adaptatif pour la commande d’un robot parallèle à liaison C5”. In: *Conférence Internationale Francophone d’Automatique (CIFA)*, Bucarest (Roumanie), Du 03 au 05 septembre 2008.
- [Achi 09a] B. Achili. “Contribution à l’identification et à la commande adaptative de robots à architecture parallèle”. Thèse de doctorat, Université Paris 08, 2009.
- [Achi 09b] B. Achili, B. Daachi, A. Ali-Cherif, and Y. Amirat. “A robust neural adaptive force controller for a C5 parallel robot”. In: *IEEE Int. Conference on Advanced Robotics (ICAR)*, Munich (Germany), June 22-26 2009.
- [Achi 10] B. Achili, B. Daachi, Y. Amirat, and A. Ali-Cherif. “Robust Adaptive Control For a Parallel Robot”. *International Journal of Control (Taylor Francis)*, Vol. 83, pp. 2107–2119, 2010.
- [Achi 12] B. Achili, B. Daachi, Y. Amirat, A. Ali-Cherif, and M. E. Daachi. “A Stable Adaptive Force/Position Controller for a C5 Parallel Robot: A Neural Network Approach”. *Robotica*, Vol. 30, No. 7, pp. 1177–1187, 2012.
- [Agui 12] G. Aguirre-Ollinger, J. Colgate, M. Peshkin, and A. Goswami. “Inertia compensation control of a one-degree-of-freedom exoskeleton for lower-limb assistance: initial experiments”. *IEEE Trans Neural Syst Rehabil Eng.*, Vol. 20, No. 1, pp. 68–77, 2012.
- [Ahme 07] T. Ahmed-Ali, G. Kennè, and F. Lamnabhi-Lagarrigue. “Nonlinear

- systems parameters estimation using neural network: Application to synchronous machine”. *Journal of Mathematical and Computer Modelling of Dynamical Systems*, Vol. 13, No. 6, pp. 365–382, 2007.
- [Ahme 09] T. Ahmed-Ali, G. Kenné, and F. Lammabhi-Lagarrigue. “Identification of nonlinear systems with time-varying parameters using a sliding-neural network observer”. *Neurocomputing*, Vol. 72, No. 7-9, pp. 1611–1620, 2009.
- [Ahme 99] T. Ahmed-Ali and F. Lammabhi-Lagarrigue. “Sliding observer-controller design for uncertain triangular nonlinear systems”. *IEEE Trans. on Automatic Control*, Vol. 44, No. 6, pp. 1244–1249., 1999.
- [Akdo 11] E. Akdogan and M. A. Adli. “The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherabot”. *Mechatronics*, Vol. 21, pp. 509–522, 2011.
- [An 85] C. H. An, C. G. Atkenson, and J. M. Hollerbach. “Estimation of inertial parameters of rigid body links of manipulators”. In: *Proceedings of the 24th Conference on Decision and control*, pp. 990–995, 1985.
- [Arms 86] B. Armstrong, O. Khatib, and J. Burdick. “The explicit dynamic model and inertial parameters of the Puma 560 arm”. In: *Proceedings of IEEE Conference on Robotics and Automation*, pp. 510–517, 1986.
- [Bana 09] S. Banala, S. Kim, S. Agrawal, and J. Scholz. “Robot assisted gait training with active leg exoskeleton (ALEX)”. *IEEE Trans Neural Syst Rehabil Eng.*, Vol. 17, No. 1, pp. 2–8, 2009.
- [Cala 01] G. Calafiore and L. E. Ghaoui. “Robust maximum likelihood estimation in the linear model”. In: *Automatica*, pp. 573–580, 2001.
- [Cybe 89] G. Cybenko. “Approximations by superpositions of sigmoidal functions”. *Mathematics of Control, Signals, and Systems*, Vol. 2, No. 4, pp. 303–314, 1989.
- [Daac 01] B. Daachi, A. Benallegue, and N. K. M’Sirdi. “A Stable Neural Adaptive

- Force Controller for a Hydraulic Actuator”. In: *Int. Conference on Robotics and Automation*, pp. 3465–3470, 2001.
- [Daac 06] B. Daachi and A. Benallegue. “A Neural Network Adaptive Controller for End-effector Tracking of Redundant Robot Manipulators”. *Journal of Intelligent and Robotic Systems*, Vol. 46, No. 3, pp. 245–262, 2006.
- [Daac 08] M. E. Daachi, B. Achili, B. Daachi, and D. Chikouche. “Neural Network Identification For a C5 Parallel Robot”. In: *Mediterranean Conference on Intelligent Systems and Automation. AIP Conference Proceedings*, Annaba (Algeria), 2008.
- [Daac 12] M. E. Daachi, B. Achili, B. Daachi, Y. Amirat, and D. Chikouche. “Hybrid Moment/Position Control of a Parallel Robot”. *International Journal of Control, Automation, and Systems*, Vol. 10, No. 3, pp. 536–546, 2012.
- [Daem 96] M. Daemi and B. Heimann. “Identification and compensation of gear friction for modeling of robots”. In: *In Proceeding of the 11th CISM-IFToMM Symposium on the Theory and Practice of Robots and Manipulators*, pp. 89–96, Udine, Italy, 1996.
- [Dafa 98] M. Dafaoui, Y. Amirat, C. Francois, and J. Pontnau. “Analysis and design of a six dof parallel robot. Modeling, singular configurations and workspace”. *IEEE Transactions on Robotics and Automation*, Vol. 14, pp. 78–92, 1998.
- [Drev 11] V. Drevelle. *Étude de méthodes ensemblistes robustes pour une localisation multisensorielle intégrée : Application à la navigation des véhicules en milieu urbain*. PhD thesis, 2011.
- [Duri 00] C. Durieu, E. Walter, and B. Polyak. “Set-membership estimation with the trace criterion made simpler than with the determinant criterion”. In: *12th IFAC Symposium on System Identification*, Santa Barbara, California, USA, 21-23 june 2000.

- [Emel 67] S. Emelianov. “Variable Structure Control Systems”. Nauka, Moscow, 1967.
- [Ertu 00] M. Ertugrul and O. Kaynak. “Neuro sliding mode control of robotic manipulators”. *Mechatronics*, Vol. 10, No. 1-2, pp. 239–263, 2000.
- [Faro 09] M. Farooq, D. Wang, and N. U. Dar. “Improved hybrid position/force controller design of a flexible robot manipulator using a sliding observer”. *Journal of Systems Engineering and Electronics*, Vol. 20, No. 1, pp. 146–158, 2009.
- [Fill 60] F. Fillipov. “Differential equations with discontinuous right-hand side”. *Matematicheskii sbornich*, pp. 99–128, 1960.
- [Flei 08] C. Fleischer and G. Hommel. “A human-exoskeleton interface utilizing electromyography”. *IEEE Transactions on Robotics*, Vol. 24, No. 2, pp. 872–882, 2008.
- [Gaut 01] M. Gautier and P. Poignet. “Extended kalman filtering and weighted least squares dynamic identification of robot”. *Control Engineering Practice*, Vol. 9, No. 1, pp. 1361–1372, 2001.
- [Gaut 90] M. Gautier. “Numerical calculation of the base inertial parameters”. In: *IEEE International Conference on Robotics and Automation*, pp. 1020–1025, Cincinnati, 1990.
- [Gaut 95] M. Gautier, W. Khalil, and P. Restrepo. “Identification of the dynamic parameters of a closed loop robot”. In: *IEEE International Conference on Robotics and Automation*, pp. 3045–3050, Nagoya, Japon, May 1995.
- [Gole 08] N. Goléa, A. Goléa, K. Barra, and T. Bouktir. “Observer-based adaptive control of robot manipulators: Fuzzy systems approach”. *Appl. Soft Comput.*, Vol. 8, No. 1, pp. 778–787, 2008.
- [Hadd 08a] A. Haddadi and K. Hashtrudi-Zaad. “A new online identification method for linear time-varying systems”. In: *in Proc. American Control Conference*, p. , Seattle, Washington, June 2008.

- [Hadd 08b] A. Haddadi and K. Hashtrudi-Zaad. “A new robust stability analysis and design tool for bilateral teleoperation control systems”. In: *in Proc. IEEE Int. Conf. Rob. and Auto*, pp. 663–670, Pasadena, CA, May 2008.
- [Hadd 08c] A. Haddadi and K. Hashtrudi-Zaad. “Online contact impedance identification for robotic systems”. In: *IROS*, pp. 974–980, 2008.
- [Herr 09] H. Herr. “Exoskeletons and orthoses: classification, design challenges and future directions”. *Journal of neuroengineering and rehabilitation*, Vol. 6, No. 1, pp. 21–30, 2009.
- [Hong 08] G. HongBo, L. YongGuang, L. GuiRong, and L. HongRen. “Cascade control of a hydraulically driven 6-DOF parallel robot manipulator based on a sliding mode”. *Control Engineering Practice*, Vol. 16, 2008.
- [Horn 89] K. Hornik, M. B. Stinchcombe, and H. White. “Multilayer feedforward networks are universal approximators”. *Neural Networks*, Vol. 2, No. 5, pp. 359–366, 1989.
- [Jime 12] R. Jiménez-Fabiàn and O. Verlinden. “Review of control algorithms for robotic ankle systems in lower-limb orthoses, prostheses, and exoskeletons”. *Medical Engineering Physics*, Vol. 34, pp. 397–408, 2012.
- [Kara 07] Y. Karayiannidis, G. A. Rovithakis, and Z. Doulgeri. “Force/position tracking for a robotic manipulator in compliant contact with a surface using neuro-adaptive control”. *Automatica*, Vol. 43, No. 7, pp. 1281–1288, 2007.
- [Khal 04] W. Khalil and O. Ibrahim. “General Solution for the Dynamic Modeling of Parallel Robots”. In: *Int. Conf. on Robotics and Automation*, April 26 - May 1, 2004.
- [Khal 99] W. Khalil and E. Dombre. *Modélisation, identification et commande des robots*. Hermès Science Publications, Paris, 1999.

- [Khou 08] A. Khoukhi, L. Baron, M. Balazinski, and K. Demirli. “A hierarchical neuro-fuzzy system to near optimal-time trajectory planning of redundant manipulators”. *Eng. Appl. of AI*, Vol. 21, No. 7, pp. 974–984, 2008.
- [Kozl 98] K. Kozlowski. *Modelling and Identification in Robotics*. Springer-Verlag, 1998.
- [Krut 02] S. Krut, O. Company, F. Marquet, and F. Pierrot. “Twice: A Tilting Angle Amplification System for Parallel Robots”. In: *in Proc. of IEEE ICRA: Int. Conf. on Robotics and Automation*, pp. 4108–4113, Washington, DC, USA, May 11-15 2002.
- [Krut 03] S. Krut. “Contribution à l’étude des robots parallèles légers, 3T-1R et 3T-2R, à forts débattements angulaires”. Thèse de doctorat - Université Montpellier II, 2003.
- [Kubu 07] D. Kubus, T. Kreger, and F. M. Wahl. “On-line rigid object recognition and pose estimation based on inertial parameters”. In: *In Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007.
- [Kubu 08] D. Kubus and F. M. Wahl. “Estimating Inertial Load Parameters Using Force/Torque and Acceleration Sensor Fusion”. In: *Robotik 2008, VDI-Berichte 2012*, pp. 29–32, Munchen, Germany, June 2008.
- [L Ja 01] O. D. L. Jaulin, M. Kieffer and E. Walter. *Applied interval analysis*. Springer-Verlag, London, 2001.
- [Lewi 04] F. Lewis, D. Dawson, and C. Abdallah. *Robot Manipulator Control Theory and Practice*. Marcel Dekker, INC, New York, 2004.
- [Lewi 95] F. L. Lewis, K. Liu, and A. Yesildirek. “Neural net robot controller with guaranteed tracking performance”. *IEEE Trans. on Neural Net*, Vol. 6, No. 3, 1995.
- [Lewi 96] F. Lewis, A. Yesildirek, and K. Liu. “Multilayer Neural-Net Robot Controller with Guaranteed Tracking Performance”. *IEEE Transactions*

*on Neural Networks*, Vol. 7, pp. 388–398, 1996.

- [Llag 11] J. F. Llagunes, R. Pàmies-Vilà, J., Alonso, and U. Lugrés. “Simulation and design of an active orthosis for an incomplete spinal cord injured subject”. *Procedia*, Vol. 2, pp. 68–81, 2011.
- [Mada 06] T. Madani and A. Benallegue. “Backstepping Sliding Mode Control Applied to a Miniature Quadrotor Flying Robot”. In: *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on*, pp. 700–705, 2006.
- [Maye 90] H. Mayeda, K. Yoshida, and K. Osuka. “Base parameters of manipulator dynamic models”. *IEEE Journal of Robotics and Automation*, Vol. 6, No. 3, pp. 312–321, 1990.
- [Medd 98] D. Y. Meddah. *Identification et commande neuronales de systèmes non linéaires : application aux systèmes robotisés*. PhD thesis, Université Paris 06, 1998.
- [Mefo 11a] S. Mefoued, S. Mohammed, and Y. Amirat. “Knee joint movement assistance through robust control of an actuated orthosis”. In: *IEEE International Conference on Intelligent and Robotic Systems*, San Francisco - USA, September 25-30 2011.
- [Mefo 11b] S. Mefoued, S. Mohammed, and Y. Amirat. “Knee joint movement assistance through robust control of an actuated orthosis”. In: *IROS*, pp. 1749–1754, 2011.
- [Mefo 12a] S. Mefoued, S. Mohammed, and Y. Amirat. “Commande Robuste d’un Robot Portable pour l’Assistance aux Mouvements des Membres Inférieurs d’un Sujet Humain”. In: *Conférence Internationale Francophone d’Automatique*, Grenoble, 4-6 juillet 2012.
- [Mefo 12b] S. Mefoued, S. Mohammed, and Y. Amirat. “Commande Robuste d’un Robot Portable pour l’Assistance aux Mouvements des Membres Inférieurs d’un Sujet Humain”. In: *CIFA*, pp. 671–676, 2012.
- [Mefo 12c] S. Mefoued, S. Mohammed, and Y. Amirat. “Sit-To-Stand Movement

- Assistance Using an Actuated Knee Joint Orthosis”. In: *IEEE International Conference on Biomedical Robotics and Biomechanics*, pp. 1753–1758, 2012.
- [Merl 01] J. P. Merlet and F. Pierrot. *Modélisation des robots parallèles*. Lavoisier, 2001.
- [Midd 88] R. Middleton and G. Goodwin. “Adaptive computed torque control for rigid link manipulators”. In: *System Control Lett.*, pp. 9–16, 1988.
- [MMil 96] M. Milanese, J. Norton, H. Piet-Lahanier, and E. Walter. *Bounding approaches to system identification*. Plenum, New York, 1996.
- [Nare 90] K. S. Narendra and K. Parthasarathy. “Identification and control of dynamical systems using neural networks”. *IEEE Trans. Neural Net.*, Vol. 1, No. 1, pp. 4–27, 1990.
- [Niki 07] J. Nikitczuk, B. Weinberg, and C. Mavroidis. “Control of electro-rheological fluid based resistive torque elements for use in active rehabilitation devices”. *Smart Materials and Structures*, Vol. 16, No. 2, pp. 418–428, 2007.
- [Pacc 09] F. Paccot. “Contributions à la commande dynamique référencée capteur de robots parallèles”. Thèse de doctorat, Université Blaise Pascal, Clermont II, 2009.
- [Perd 91] V. Perdereau. *Contribution à la commande hybride force-position : Application à la coopération de deux robots*. PhD thesis, Université Paris 06, 1991.
- [Pere 12] R. Pérez-Rodríguez, A. M.-C. no, U. Costa, J. Solana, C. Cáceres, E. Opisso, J. M. Tormos, J. Medina, and E. J. Gómez. “Inverse kinematics of a 6 DoF human upper limb using ANFIS and ANN for anticipatory actuation in ADL-based physical Neurorehabilitation”. *Expert Systems with Applications*, Vol. 39, pp. 9612–9622, 2012.

- [Poig 03a] P. Poignet, N. Ramdani, and O. Vivas. “Ellipsoidal estimation of parallel robot dynamic parameters”. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3300–3305, Las Vegas, Nevada, USA, 2003.
- [Poig 03b] P. Poignet, N. Ramdani, and O. Vivas. “Robust estimation of parallel robot dynamic parameters with interval analysis”. In: *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 6503–6508, Maui, Hawaii, USA, 2003.
- [Prat 04] J. E. Pratt, B. T. Krupp, and C. J. Morse. “The roboKnee : an exoskeleton for enhancing strength and endurance during walking”. *International Journal of Robotics and Automation*, Vol. 3, pp. 2430–2435, 2004.
- [Rifa 12a] H. Rifai, S. Mohammed, B. Daachi, and Y. Amirat. “Adaptive Control of a Human-Driven Knee Joint Orthosis”. In: *IEEE International Conference on Robotic and Automation (ICRA)*, St. Paul (USA), May 14-18 2012.
- [Rifa 12b] H. Rifai, S. Mohammed, B. Daachi, and Y. Amirat. “Adaptive control of a human-driven knee joint orthosis”. In: *ICRA*, pp. 2486–2491, 2012.
- [Sait 05] Y. Saito and K. Kikuchi. “Development of externally powered lower limb orthosis with bilateral servoactuator”. In: *International Conference on Rehabilitation Robotics*, pp. 394–399, 2005.
- [Scha 98] S. Schaal and C. G. Atkeson. “Constructive incremental learning from only local information”. *Neural Computation*, Vol. 10, No. 8, pp. 2047–2084, 1998.
- [Slot 88] J. Slotine and W. Li. “Adaptive manipulator control: a case study”. *IEEE Trans. Automat. Control.*, Vol. 33, pp. 995–1003, 1988.
- [Slot 91] J. J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, NJ, 1991.

- [Song 05] Z. Song, D. Z. J. Yi, and X. Li. “A computed torque controller for uncertain robotic manipulator system: Fuzzy approach”. *Fuzzy Sets and Systems*, Vol. 154, pp. 208–226, 2005.
- [Sun 01] F. Sun, Z. Sun, and P. Woo. “Neural network-based adaptive controller design of robotic manipulators with an observer”. *IEEE Trans. Neural Networks*, Vol. 12, pp. 54–67, 01.
- [Swev 96] J. Swevers, C. Ganseman, J. D. Schutter, and H. V. Brussel. “Experimental robot identification using optimized periodic trajectories”. *Mechanical Systems and Signal Processing*, Vol. 10, No. 5, pp. 561–577, 1996.
- [Swev 97] J. Swevers, C. Ganseman, D. Bilgin, J. D. Schutter, and H. V. Brussel. “Optimal robot excitation and identification”. *IEEE Transactions on Robotics and Automation*, Vol. 13, No. 5, pp. 730–740, 1997.
- [Topa 07] A. Topalov and O. Kaynak. “Neuro-adaptive SM tracking control of robot manipulators”. *Int. J. of adaptive control and signal processing*, Vol. 21, pp. 674–691, 2007.
- [Toua 08] Y. Touati, Y. Amirat, N. Saadia, and A. Ali-Chérif. “A neural network-based approach for an assembly cell control”. *Appl. Soft Comput.*, Vol. 8, No. 4, pp. 1335–1343, 2008.
- [Tso 00] S. Tso, Y. Fung, and N. Lin. “Analysis and real-time implementation of a radial-basis-function neural-network compensator for high-performance robot manipulators”. *Mechatronics*, Vol. 10, pp. 265–287, 2000.
- [Utki 78] V. Utkin. “Sliding Modes and their Application in Variable Structure Systems”. In: , p. , MIR, Moscow, 1978.
- [Vija 02] S. Vijayakumar, A. D’Souza, T. Shibata, J. Conradt, and S. Schaal. “Statistical Learning for Humanoid Robots”. *Autonomous Robots*, Vol. 12, No. 1, pp. 55–69, 02.
- [Visi 10] A. Visioli, G. Ziliani, and G. Legnani. “Iterative-Learning Hybrid

- Force/Velocity Control for Contour Tracking”. *IEEE Transactions on Robotics*, Vol. 26, No. 2, pp. 388–393, 2010.
- [Viva 04] A. Vivas. “Contribution à l’identification et à la commande des robots Parallèles”. Thèse de doctorat - Université Montpellier II, 2004.
- [Wang 11] D. Wang, X. Zeng, and J. A. Keane. “An Output-Constrained Clustering Approach for the Identification of Fuzzy Systems and Fuzzy Granular Systems”. *IEEE T. Fuzzy Systems*, Vol. 19, No. 6, pp. 1127–1140, 2011.
- [Wit 90] A. A. C. Canudas de Wit. “Parameters identification of robots manipulators via sequential hybrid estimation algorithms”. In: *Proceedings IFAC Congress*, pp. 178–183, Tallin, 1990.
- [Wu 09] J. Wu, J. Wang, L. Wang, and T. Li. “Dynamics and control of a planar 3-DOF parallel manipulator with actuation redundancy”. *Mechanism and machine theory*, Vol. 44, No. 4, pp. 835–849, 2009.
- [Yagn 90] N. Yagn. “Apparatus to facilitate walking, running, and jumping”. *U.S. Patent*, No. 420,179, 1890.
- [Yeh 10] T. Yeh, M. Wu, T. Lu, F. Wu, and C. Huang. “Control of McKibben pneumatic muscles for a power-assist, lower-limb orthosis”. *Mechatronics*, Vol. 20, No. 6, pp. 686–697, 2010.
- [Yu 10] W. Yu and J. Rosen. “A novel linear PID controller for an upper limb exoskeleton”. In: *IEEE International Conference on Decision and Control (CDC)*, 2010.
- [Zbik 94] R. W. Zbikowski. *Recurrent Neural Networks some Control Aspects*. PhD thesis, Faculty of Engineering of Glasgow University, 1994.
- [Zhan 10] F. Zhang, P. Li, Z. Hou, X. Xie, Y. Chen, Q. Li, and M. Tan. “An adaptive RBF neural network control strategy for lower limb rehabilitation robot”. In: *Proceedings of the Third international conference on Intelligent robotics and applications - Volume Part II*, pp. 417–427, Springer-Verlag, Berlin, Heidelberg, 2010.

- [Zhou 08] H. Zhou and H. Hu. “Human motion tracking for rehabilitation - A survey”. *Biomedical Signal Processing and Control*, Vol. 3, No. 1, pp. 1–18, 2008.
- [Zili 06] G. Ziliani, A. Visioli, and G. Legnani. “Gain scheduling for hybrid force/velocity control in contour tracking task”. *International Journal of Advanced Robotic Systems*, Vol. 3, No. 4, pp. 367–374, 2006.

## Abstract:

In this thesis, we have addressed two aspects in relation with mechatronic systems: Identification and control. Indeed, the MLP-NN (MultiLayer Perceptron Neural Network) is used in several approaches proposed in this thesis. Note that the realized work is purely experimental. The two mechatronic systems considered are the C5 links parallel robot and the wearable robot of exoskeleton type available in the LISSI laboratory. In the first time, we have achieved an identification neural black box of the inverse dynamics of C5 parallel robot. To do this, three identification schemes were tested and compared. On the control part, we have proposed an adaptive control hybrid moment / position of C5 parallel robot directly in the task space. The task space dynamic model of the robot in contact with its environment, seen as a black box, is estimated by a MLP-NN. An adaptation algorithm of the neural parameters resulting from a closed-loop stability analysis is proposed. Another approach of control is proposed to derive the exoskeleton. In this design approach, of the adaptive control, the dynamic model is taken as a gray box. Only its structure is known. The unknown functions of the dynamic model are approximated online. The neural parameters adaptation laws are obtained via stability study in Lyapunov sense of the system in closed loop. The proposed approach is tested on a healthy person in flexion / extension of the knee.

**Keywords:** Identification, adaptive control, neural networks, parallel robot, exoskeleton, stability.

## مُلخَص:

في هذه الأطروحة، قمنا بدراسة جانبيين متعلقين بالروبوت: تحديد النموذج الديناميكي والتحكم. لقد تمّ استعمال الشبكات العصبية ذات الطبقات في مختلف الحلول المقترحة في هذه الأطروحة. كما تجدر الإشارة إلى أنّ العمل المنجز كان تجريبيا بحتا.

النموذجان المدروسان في هذا العمل، يتمثلان في روبوت متوازي وروبوت محمول على شكل هيكل خارجي للأطراف السفلى للإنسان. النموذجان موجودان بالمخبر LISSI، جامعة باريس شرق.

بداية، قمنا بتحديد النموذج الديناميكي العكسي للروبوت المتوازي بشكل علبة سوداء باستعمال الشبكات العصبية. لتحقيق ذلك، تم دراسة ومقارنة ثلاث بُنى مختلفة.

فيما يخصّ التحكم، إقترحنا طريقة تحكم تكيفي هجين عزم \ حركة للروبوت المتوازي. الطريقة المقترحة جُربت على الروبوت مباشرة في الفضاء العملي. كما نشير هنا بأنّ النموذج الديناميكي للروبوت بالتماس مع محيطه في الفضاء العملي المعتبر كعلبة سوداء، تمّ تقديره خلال عملية التحكم باستعمال الشبكات العصبية ذات الطبقات. قوانين تكييف عناصر الشبكة العصبية حُصل عليها عن طريق دراسة الإستقرار في الحلقة المغلقة وفقا لنظرية ليابونوف.

طريقة أخرى للتحكم تعتمد هي الأخرى على الشبكات العصبية ذات الطبقات، تمّ تجربتها أيضا للتحكم في الروبوت المحمول. النموذج الديناميكي لهذا الأخير مُعتبر كعلبة رمادية، أي أنه معروف الصيغة فقط، و دواله المجهولة تمّ تقديرها خلال عملية التحكم عن طريق دراسة إستقرار النظام في الحلقة المغلقة وفقا لنظرية ليابونوف. طريقة التحكم هذه، تمّ تجربتها على إنسان سليم في وضعية جلوس أثناء قيامه بمدّ و ثني ساقه.

**الكلمات المفاتيح:** تحديد النموذج الديناميكي، التحكم التكيفي، الشبكات العصبية، الروبوت المتوازي، الروبوت المحمول، الإستقرار.