

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Thèse Intitulée :

**Contribution à la Commande Robuste des Systèmes. Théories et
Applications à la Commande de Robots manipulateurs à Muscles Artificiels
Pneumatiques.**

REZOUG Amar

Mémoire

Présenté pour l'obtention du diplôme de Doctorat ès Sciences en
Automatique.

Soutenu le 06-06-2013

Président de Jury :

BOUDJEMA	Farès	Professeur	ENP
-----------------	--------------	-------------------	------------

Examineurs :

SAARI	Hammoud	Professeur	ENSM
SAADIA	Nadia	Professeur	USTHB
BOUCHAFFRA	Djamel	Directeur de recherche	CDTA

Rapporteurs

TADJINE	Mohamed	Professeur	ENP
HAMERLAIN	Mustapha	Directeur de recherche	CDTA

Division Robotique & Productique, Centre de Développement des Technologies Avancées, Cité 20 août 1956, BP. N°
17, Baba Hassen 16303, Alger, Algérie.

Laboratoire de Commande des Processus, Département Automatique, Ecole Nationale Polytechnique d'Alger, 10 Av.
Hassen Badi, B182, Alger, Algérie.

Le 06 Juin 2013.

Remerciement

Les travaux de cette thèse ont été effectués conjointement au niveau de la division Robotique et Productique du CDTA, au niveau de Laboratoire Commande des Processus de l'Ecole Nationale Polytechnique d'Alger et au niveau du groupe GEPPETO de l'INSA-LAAS de Toulouse.

Je tiens à présenter mes sincères remerciements et ma profonde gratitude à mes Directeurs de Thèse les Professeurs Mohamed TADJINE et Mustapha HAMERLAIN pour leurs conseils et leurs soutiens durant ces années de formation doctorale.

Je remercie également le Professeur Bertrand TONDU de l'Institut National des Sciences Appliquées de Toulouse pour son chaleureux accueil ainsi que pour les précieuses directives tout au long de mon stage de formation au sien de son équipe de recherche.

Je remercie profondément le Professeur Farès BOUDJEMMA de m'avoir honoré d'accepter de présider le jury d'examen de ce travail de thèse.

Je remercie également les membres du jury, Monsieur Djamel BOUCHAFRA, Directeur de Recherche au CDTA, Nadia SAADIA, Professeur à USTHB, Hamoud SARRI, Professeur à l'ENSM, pour avoir accepté d'évaluer cette thèse.

Mes remerciements vont également au Ministère de l'Enseignement Supérieur et de la Recherche Scientifique Algérien pour m'avoir accordé une bourse de formation à l'étranger (PNE) pour l'année 2011-2012.

Je n'oublie pas de remercier Monsieur Brahim BOUZOUÏ le Directeur du CDTA pour m'avoir facilité les démarches administratives le long des années de ma formation doctorale.

Je remercie tous mes collègues du CDTA, notamment Monsieur, Hocine BOUTELLAA, Monsieur, Fayçal HADJELARBI et Madame, Hamida HAMOU pour leurs précieuses aides durant cette formation.

Un grand merci aussi à Mounir KAIS, Khelifa Baizid, Mohamed REZOUÏ et Abdallah REZOUÏ pour le soutien moral durant toutes ces dernières années.

Je dédie ce travail :

À mes parents,
À mes sœurs et frères,
À mon oncle Mohamed,
À toute la famille REZOUG.

ملخص العضلات الهوائية الاصطناعية هو نظام ديناميكي يتميز بسلوك غير خطي مما يجعل التحكم فيه بدقة و بسرعة عالية صعب التحقيق. هذا التحكم يصبح أكثر تعقيدا عندما يتعلق الأمر بالروبوتات التي تستعمل هذه العضلات من أجل الحركة. في هذه الأطروحة اهتمنا بتطوير قوانين تحكم شديدة من أجل التحكم في والوضعية الزاوية وتتبع مسار للروبوتات التي تستعمل هذا النوع من العضلات. نوعين من الروبوتات استعملنا في هذا البحث. الأول هو ذراع روبوت يتكون من ثلاث درجات حرية موجود على مستوي مركز البحث في التكنولوجيات المتقدمة (متتم) بالجزائر. الذراع الآلية الثانية هي روبوت يتكون من سبع درجات حرية موجود على مستوي المعهد الوطني للعلوم التطبيقية (موعت) بتولوز فرنسا. في البداية قمنا بقتراح نموذج رياضي خطي بالنسبة للروبوت "متتم", بالنسبة للروبوت "الموعت", لقد اقترحنا نموذجا رياضي خطي مكون من 2 درجات حرية الذين يمثلان القطع الأساسية للروبوت. بعد ذلك قمنا بتأليف اربع قوانين الأول يعتمد على المنطق الغامض من النوع الثاني المتكيف, ثلاث قوانين هجينة تم تأليفها بعد ذلك الأول هو قانون يستعمل المنطق الغامض و يعتمد على طريقة التحكم بالانزلاق الثاني يعتمد على الشبكة العصبية الاصطناعية. المنطق الغامض من النوع الثاني و طريقة التحكم بالانزلاق. الثالث هو قانون يستعمل طريقة التحكم بالانزلاق النهائية و المنطق الغامض من النوع الأول المتكيف. تم فحص اختبارات متانة كافة طرق تحكم.

كلمات مفتاحية : العضلات الهوائية الصناعية , اذرع الروبوتات , تحكم شديدة, التحكم بالانزلاق , المنطق الغامض من النوع الثاني.

Résumé : Le Muscle Artificiel Pneumatique (MAP) est un système dont le comportement dynamique est fortement non linéaire, ce qui rend sa commande avec précision et à vitesse élevée très difficile à réaliser. Cette nonlinéarité devient de plus en plus complexe lorsqu'il s'agit de robots actionnés par ce muscle. Dans ce travail de thèse, nous nous sommes intéressés au développement des nouvelles lois de commande robuste pour le contrôle en position et en poursuite de trajectoire des robots manipulateurs à MAPs. Deux robots manipulateurs ont été utilisés, le premier est celui à 3-dof du Centre de Développement des Technologies Avancées (CDAT) d'Alger, le deuxième robot utilisé est celui à 7-dof de l'Institut National des Sciences Appliquées (INSA) de Toulouse. Au début, nous avons proposé des modèles mathématiques pour les deux robots, dont le robot du CDAT a été approximé par un modèle linéaire. Les deux axes essentiels du robot de l'INSA ont été exploités, à cet effet, un modèle dynamique à deux axes est proposé. Quatre lois de commande ont été proposées. La première est une commande par logique floue de type-2 adaptative. Par la suite, trois commandes hybrides ont été proposées, la première est une commande par réseau de neurones à fonctions de bases radiales floue de type-2 basée sur le mode de glissement. La deuxième est une commande par logique floue de type-2 basée sur la condition d'attractivité de la commande par mode de glissement. La troisième commande est appelée commande par terminal sliding mode flou adaptatif. Des tests de robustesse ont été examinés pour toutes les commandes développées.

Mots clés : Muscle Artificiel Pneumatique, Robot Manipulateur, Commande Robuste, Mode de Glissement, Logique Floue de Type-2.

Summary: The Pneumatic Artificial Muscle (PAM) system has a higher nonlinear dynamic behavior which makes precision and high speed control very difficult to achieve. This nonlinearity is more complex in the case of robots actuated by PAM system. In this thesis, we are interested to develop new robust position control for robot manipulators driven by PAM. Two robot manipulators have been used, the first is the 3-DOF of the Center for Development of Advanced Technologies (CDAT), and the second is the 7-DOF of the National Institute of Applied Sciences (INSA) of Toulouse. At the beginning, we proposed mathematical models for both robots. The CDAT robot was approximated by a linear model. The two main axis of the INSA robot have been exploited, in this effect a two-axis dynamic model is proposed. Four controllers' have been proposed, the first is a fuzzy logic control of adaptive type-2. Subsequently, three hybrid controls have been proposed, the first is named radial basis function neural network type-2 fuzzy sliding mode controller. The second is a type-2 fuzzy logic control based on sliding mode control. The third command is called adaptive fuzzy terminal sliding mode controller. Tests of robustness were examined for all developed controllers.

Keywords: Pneumatic Artificial Muscle, Manipulator Robot, Robust Control, Sliding Mode, Type-2 Fuzzy Logic.

Références Personnelles

1- Articles Apparus dans des Journaux:

[1] **A. REZOUG, M. HAMERLAIN, M. TADJINE**, "Decentralized RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Manipulator Driven by Artificial Muscles". *International Journal of Advanced Robotic Systems*, vol 09, pp 1-12, November, 2012.

http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/decentralized-rbfnn-type-2-fuzzy-sliding-mode-controller-for-robot-manipulator-driven-by-artificial-

[2] **A. REZOUG, M. HAMERLAIN, M. TADJINE**, "Adaptive Interval Type-2 Fuzzy Controller For Robot Arm Actuated By Pneumatic Artificial Rubber Muscles". *The Mediterranean Journal of Measurement and Control*, Vol. 7, No. 3, pp. 260-272, August 2011.

<http://www.medjmc.com/papers%20archive/index.html>

2- Articles Présentés dans des Conférences Internationales :

[1] **A. REZOUG, M. HAMERLAIN, M. TADJINE**, "Adaptive RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Arm with Pneumatic Muscles", *IEEE International Conference on Robotics and Biomimetics (ROBIO 2012)*, China, 2012, pp. 1287-1292.

[2] **A. REZOUG, S. MAHJOUR, M. HAMERLAIN, M. TADJINE**, "Fuzzy Terminal Sliding Mode Controller for Robot Arm Actuated By Pneumatic Artificial Muscles", *8th Int. Multi-Conf. on Systems, Signals & Devices, Conference on Systems, Analysis & Automatic Control*, (SSD 2011), Tunisia, mars 2011, pp. 1-6.

[3] **A. REZOUG, M. HAMERLAIN, B. TONDU, M. TADJINE**, "Interval Type-2 Fuzzy Sliding Mode Controller for Robot Arm Driven By Artificial Muscles". *Special Session ANNIIP'11, 8th international conference on Informatics in control, Automation and Robotics (ICINCO 2011)*, The Netherlands, 2011, pp. 461-469.

[4] **A. REZOUG, A. MEDDAHI, K. BAIZID, M. HAMERLAIN, and M. TADJINE**, "Fuzzy Logic Controller for a Pneumatic Artificial Muscle Robot based on Sliding Mode Control", *14th IASTED International Conference on Robotics and Applications (IASTED 2009)*, USA, 2009, pp. 74-81

Table des Matières

INTRODUCTION GENERALE	1
CHAPITRE I : <i>Etat De L'art Sur Les Robots A Muscles Artificiels Pneumatiques</i>	7
I.1 INTRODUCTION.....	7
I.2. LES TYPES D' ACTIONNEURS A MUSCLES ARTIFICIELS	7
I.3 LE MUSCLE ARTIFICIEL PNEUMATIQUE.....	9
I.4 LES APPLICATIONS DES MUSCLES ARTIFICIELS PNEUMATIQUES.....	10
I.5 AVANTAGES ET INCONVENIENTS DES ROBOTS A MAPs.....	11
I.5.1 Les avantages du MAP.....	11
I.5.2 Les inconvénients du MAP	11
I.6 COMPOSITION D'UN MAP	12
I.7 PRINCIPE DE FONCTIONNEMENT D'UNE PAIRE DE MAPs PLACES EN ANTAGONISME.....	13
I.8 PLATES-FORMES EXPERIMENTALES UTILISEES DANS CETTE THESE.....	14
I.8.1 Description du Bras Manipulateur à MAP du CDTA.....	14
I.8.2 Caractéristiques du MAP FESTO Utilisé.....	17
I.8.3 Modélisation de robot à MAP du CDTA.....	18
I.8.4 Identification des axes du Robot du CDTA et Etude de Son Couplage.....	19
I.8.4.1 Procédure d'identification.....	19
I.8.4.2 Choix de Signal d'Excitation.....	19
I.8.4.3 Analyse de Couple par la Méthode RGA pour le Bras du Robot du CDTA.....	24
I.8.5 Problèmes Pratiques Liés au Robot du CDTA	26
I.9 DESCRIPTION DE ROBOT MANIPULATEUR A MAPS DE L'INSA.....	28
I.9.1 Problèmes Liés au Robot de L'INSA de Toulouse	30
I.9.2 Modélisation du Bras Anthropomorphe (Modèle de Simulation).....	30
I.10 CONCLUSION.....	32
CHAPITRE II <i>Etat De L'art De La Commande Par Mode De Glissement</i>	33
II.1. INTRODUCTION.....	33
II.2 LA COMMANDE A STRUCTURE VARIABLE.....	33
II.3 FORMULATION CLASSIQUE DU REGIME GLISSANT.....	34
II.3.1 Définition de Régime glissant idéal.....	34
II.3.2 Définition de Régime glissant réel.....	35
II.3.3 Explication du glissement.....	35
II.3.4 Condition d'existence du glissement.....	36
II.3.5 Choix des surfaces de glissement.....	38
II.4 COMMANDE A STRUCTURE VARIABLE EN REGIME GLISSANT.....	39
II.4.1 Equation d'état en mode de glissement exprimée à partir du gradient et du produit scalaire.....	39
II.4.2 Commande équivalente.....	41
II.5. EFFET DE BROUITEMENT.....	43
II.6. COMMANDE PAR TERMINAL SLIDING MODE.....	43
II.6.1. Synthèse de la Commande par Terminal Sliding Mode.....	44
II.6.2 Le Problème de Singularité.....	44
II.7 MODES GLISSANTS D'ORDRE SUPERIEUR (HOSM).....	45
II.7.2 Algorithmes glissants d'ordre 2.....	45
II.7.2.1 Commande Twisting.....	46
II.7.2.2 Commande Super twisting.....	48
II.8. LA COMMANDE PAR TERMINAL SLIDING MODE D'ORDRE SUPERIEUR.....	49
II.9 CONCLUSION.....	50

CHAPITRE III	<i>La Logique Floue de Type-2</i>	51
III.1 INTRODUCTION.....		51
III.2 NIVEAU D'INTERVENTION D'UN SYSTEME FLOU DANS UNE BOUCLE D'ASSERVISSEMENT		51
III.2.1 La commande.....		51
III.2.2 La supervision		52
III.2.3 L'observation		52
III.3 LES CARACTERISTIQUES DE LA COMMANDE PAR LOGIQUE FLOUE.....		53
III.4. LOGIQUE FLOUE TYPE-1.....		53
III.4.1 Les ensembles flous de type-1.....		53
III.4.2 Structure générale d'un régulateur flou de type-1.....		54
III.4.3 Types de contrôleur flou.....		55
III.5 EXTENSION DE CONTROLEUR FLOU DE TYPE TSUGENO (LE RESEAUX DE NEURONES A FONCTION DE BASE RADIALE (RBF)).....		55
III.6 SOURCES D'INCERTITUDES DANS UN SYSTEME A COMMANDER ET RAISONS DE L'USAGE DE LA LOGIQUE FLOUE DE TYPE-2.....		56
III.7 LA DEFERENCE ENTRE UN SYSTEME FLOU DE TYPE-1 ET DE TYPE-2.....		57
III.8 LOGIQUE FLOUE TYPE-2.....		58
III.8.1 Les Applications de la Logique Floue de Type-2.....		59
III.9 LES OPERATEURS DE LA LOGIQUE FLOUE TYPE-2.....		63
III.9.1 Union.....		63
III.9.2 Intersection.....		63
III.9.3 Complément d'un l'ensemble flou type-2.....		64
III.10 LE SYSTEME D'INFERENCE FLOUE DE TYPE-2 INTERVALLE.....		64
III.10.1 Fuzzification.....		65
III.10.2 Tables des règles.....		65
III.10.3 Inférence.....		65
III.10.4 Réduction de type.....		67
III.10.5. La défuzzification.....		69
III.11 CONCLUSION.....		69
CHAPITRE IV. <i>Limites Des Commandes Par Mode De Glissement, Et Par Mode De Glissement Flou</i>		71
IV.1 INTRODUCTION.....		71
IV.2. SYNTHESE D'UNE COMMANDE PAR MODE GLISSANT FLOU BASEE SUR UN MODELE IDENTIFIE.....		71
IV.2.1 La commande discontinue.....		73
IV.2.2. La condition d'existence d'une surface de glissement.....		73
IV.2.3 Résultats de simulation.....		73
IV.2.4 Interprétation des résultats.....		76
IV.3 COMMANDE FLOUE-Glissante.....		77
IV.3.1 Définition des fonctions d'appartenance des variable d'entrées/ sortie.....		77
IV.3.2 Inférence floue.....		78
IV.3.3 Résultats de simulation.....		79
IV.3.4 Interprétation.....		82
IV.4 CONCLUSION		82
CHAPITRE V...<i>commande Floue De Type-2 Intervalle D'un Robot Manipulateur à Muscles Artificiels Pneumatiques</i>		84
V.1 INTRODUCTION		84
V.2COMMANDE FLOUE DE TYPE-2 INTERVALLE D'UN ROBOT MANIPULATEUR A MAPs		84
V.3 COMMANDE ADAPTATIVE FLOUE DE TYPE-2 INTERVALLE POUR LE ROBOT MANIPULATEUR A MAPs		86
V.4 RESULTATS EXPERIMENTAUX		88
V.4.1 Résultats Expérimentaux Cas Découple.....		89
V.4.2 Commande Simultanée.....		92
V.4.3 Analyse des Performances.....		94
V.5 COMMANDE PAR LOGIQUE FLOUE BASEE SUR LES MODES DE GLISSEMENT.....		95

V.5.1 Conception De Controleur Flou De Type-2 Glissant.....	96
V.5.2 Commande Par Mode De Glissement Cas Multi Variables Lineaire.....	96
V.5.3 La surface de glissement dans le cas MIMO.....	97
V.5.4 conception de le loi de commande.....	98
V.6 COMMANDE PAR LOGIQUE FLOUE DE TYPE-2 INTERVALLE BASEE SUR LES MODE DE GLISSEMENT.....	98
V.7 RESULTATS EXPERIMENTAUX ET DISCUSSION.....	101
V.8 CONCLUSION.....	104

CHAPITRE VI... Commande Par Mode De Glissement Basée Sur Un Réseau De Neurones Et Un Contrôleur Flou De Type.....

VI.1 INTRODUCTION.....	105
VI.2 COMMANDE PAR MODE DE GLISSEMENT D'UN ROBOT MANIPULATEUR A N-DDL.....	105
VI.2.1. Analyse de stabilité.....	106
VI.3 COMMANDE PAR MODE DE GLISSEMENT D'UN SYSTEME MONO ENTREE MONO SORTIE.....	108
VI.4 ESTIMATION DE LA COMMANDE EQUIVALENTE PAR RBFNN.....	110
VI.4.1. L'architecture du RBFNN.....	110
VI.4.2 Estimation de la commande équivalente par le réseau de neurones à fonctions radiales.....	112
VI.4.3. Analyse de stabilité et adaptation des paramètres de la commande RBFNN.....	113
VI.5 LE CONTROLEUR FLOU DE TYPE-2 POUR L'AMELIORATION DES PERFORMANCES ET L'ATTENUATION DES BROUTEMENTS.....	115
VI.6 APPLICATION DE LA COMMANDE RBFT2FSMC SUR LE ROBOT A MAPs DU CDTA.....	116
VI.5.1 Simulation de la commande RBFT2FSMC.....	117
VI.5.2 Résultats expérimentaux.....	122
VI.6 CONCLUSION.....	124

CHAPITRE VII...Commande Par Terminal Sliding Mode Flou Adaptatif d'un Robot Manipulateur Anthropomorphe A Muscles Artificiels Pneumatiques

VII.1 INTROCUCTION.....	125
VII.2 LA SYNTHESE DE LA COMMANDE PAR TERMINAL SLIDING MODE NONSINGULIERE.....	125
VII.3 SYNTHESE DE CONTROLEUR PAR MODE DE GLISSEMENT TERMINAL FLOU.....	126
VII.4 RESULTATS DE SIMULATION ET DISCUSSION.....	128
VII.5 COMMANDE PAR TERMINAL SLIDING MODE FLOU ADAPTATIF DE ROBOT ANTROPOMORPHE DE L'INSA DE TOULOUSE	133
VII.6 COMMANDE PAR TERMINAL SLIDING MODE de robot de l'INSA.....	133
VII.7 PROBLEME DE BROUTEMENT.....	135
VII.8 CONTROLEUR FLOU POUR L'ATTENUATION DE BROUTEMENT	136
VII.9 COMMANDE ADAPTATIVE FLOUE DE ROBOT A MUSCLE ARTIFICIELS PNEUMATIQUES.....	136
VII.9.1 Mécanisme d'adaptation en ligne.....	138
VII.9.2 Résultats Expérimentaux.....	139
VII.10 TESTES DE ROBUSTESSE.....	142
VII.10.1 Mode régulation.....	142
VII.11. CONCLUSION.....	145

Conclusion Générale	146
---------------------------	-----

Liste des Figures

Figure I.1. Classification des muscles artificiels.....	8
Figure I.2. Les MAPs les plus répandus.....	9
Figure I.3. Exemples de robots à MAPs.....	11
Figure I.4. Quelque modèle du muscle artificiel toulousain.....	13
Figure I.5. Déférentes configurations de l'actionnement par MAPs.....	13
Figure I.6. Structure interne d'un axe du bras à MAP.....	14
Figure I.7. Le bras manipulateur à MAP du CDTA.....	15
Figure I.8. Vu des quatre blocs du système de commande du robot.....	15
Figure I.9. Schéma détaillé de composantes du robot du CDTA.....	17
Figure I.10. Muscles artificiels pneumatiques du FESTO.....	18
Figure I.11. Exemple d'un SBPA de longueur 7.....	20
Figure I.12. signal SBPA utilisé pour l'identification de modèle du robot.....	21
Figure I.13. Identification des deux axes avec prise en compte de couplage.....	23
Figure I.14. Schéma bloc de robot manipulateur avec interaction entre deux axes.....	23
Figure I.15. Réponses des axes du robot en boucle ouverte par rapport à des échelons de déférentes valeurs.....	27
Figure I.16. Bras manipulateur anthropomorphe de l'INSA.....	29
Figure I.17. Position initial du robot (a) épaule (b) coude.....	29
Figure I.18. Architecture du robot de l'INSA.....	29
Figure II.1. Schéma de principe d'un système à structure variable.....	34
Figure II.2. Glissement idéal.....	34
Figure II.3. Glissement réel.....	35
Figure II.4. Mode de glissement.....	35
Figure II.5. Espace d'état de dimension 3 et surface de glissement de dimension 2.....	36
Figure II.6. Schéma de principe montrant la région du mode de glissement et les champs de vecteurs associés.....	39
Figure II.7. Schéma de principe de la construction du champ de vecteur moyen f^0 dans une coupe de l'espace d'état.....	40
Figure II.8. Caractéristique dynamique de la grandeur de commande équivalente.....	42
Figure II.9. Plan de phase de la commande Twinting.....	48
Figure II.10. Plan de phase de la commande super Twisting.....	42
Figure III.1. Un SIF pour la commande.....	49
Figure III.2. Un SIF pour l'ajustement et/ou l'adaptation d'un autre contrôleur.....	52
Figure III.3. Un système d'inférence flou pour l'observation.....	52
Figure III.4. Schéma de principe d'un contrôleur flou type-1.....	53
Figure III.5. structure générale d'un réseau de neurones de fonctions à bases radiales.....	54
Figure III.6. Relation entre un système contrôleur flou de type-1 et celle de type-2.....	56
Figure III.7. Évolution de nombre de publications de la logique floue de type-2 en fonction des années depuis son apparition.....	58
Figure III.8. L'évolution de la logique floue de type-2 le long des années et leurs applications.....	59
Figure III.9. Fonction d'appartenance gaussienne type-2 (écart type variable).....	59
Figure III.10. Fonction d'appartenance gaussienne type-2 (moyenne variable).....	60
Figure III.11. Gaussienne Caractéristique de la fonction d'appartenance type-2.....	61
Figure III.12. Union de 2 gaussienne type-2 (a) fonction floue de type-2 (b) union de 2 fonctions.....	61
Figure III.13. Intersection de 2 gaussienne type-2, (a) fonction d'appartenance de type-2 (b) son complément.....	62
Figure III.14. Complément d'un ensemble flou type-2.....	63
Figure III.15. Schéma de principe d'un contrôleur flou type-2.....	64
Figure III.16. Méthode d'inférence type-2 Max-Min.....	65
Figure III.17. Exemple montrant la méthode de réduction de type Karnik-Mendel.....	67
Figure III.18. Organigramme de l'algorithme de réduction de type Karnik-Mendel.....	68
Figure IV.1. Commande par mode glissant classique de l'axe 1.....	68
Figure IV.2. Commande par mode glissant classique de l'axes 2 et 3.....	75
Figure IV.3. Les fonctions d'appartenances de l'erreur.....	76
Figure IV.4. Les fonctions d'appartenances de la variation de l'erreur.....	78
Figure IV.5. Les fonctions d'appartenances de la sortie k_{fuzzy}	78
Figure IV.6. Commande par mode glissant flou appliquée sur les deux axes 1 et 2 du robot.....	79
	82

Figure V.1. Les variables d'entrées (l'erreur et la variation de l'erreur sur les univers de discours).	85
Figure V.2. La distribution des fonctions d'appartenances de la sortie sur l'univers de discours.	85
Figure V.3. Commande floue de type-2 adaptative du robot manipulateur à MAPs.	88
Figure V.4. Résultats expérimentaux de la commande par logique floue statique et adaptative pour l'axe 1 (a) Position angulaire, (b) Signal de commande, (c) gain adaptatif.	90
Figure V.5. Résultats expérimentaux de la commande par logique floue statique et adaptative pour l'axe 2 (a) Position angulaire, (b) Signal de commande, (c) gain adaptatif.	91
Figure V.7. Réponses simultanées (a) axe 1 (b) axe 2.	93
Figure V.8. Signaux de commande cas simultanés (a) axe 1 (b) axe 2.	93
Figure V.9. Variations des gains d'ajustement de la commande AIT2FLC.	94
Figure V.10. les entrées de contrôleur IT2FSMC (a) surface de glissement (b) la dérivée de la surface de glissement.	99
Figure V.11. Les valeurs floues de la sortie pour la commande IT2FSMC.	99
Figure V.12. Schéma bloc de la commande IT2FSMC de robot à MAP.	101
Figure V.13. Résultats expérimentaux de la commande IT2FSMC sur l'axe 1 (a) repense en position angulaire IT2FSMC (bleu) and IT2FC (rouge) (b) signaux de commande IT2FSMC (bleu) and IT2FC (rouge) (c) surface de glissement.	102
Figure V.14. Résultats expérimentaux de la commande IT2FSMC sur l'axe 2 (a) repense en position angulaire IT2FSMC (bleu) and IT2FC (rouge), (b) signaux de commande IT2FSMC (bleu) and IT2FC (rouge) (c) surface de glissement.	103
Figure VI.1. Structure d'un réseau de neurones à fonction à bases radiales.	111
Figure VI.2. L'entrée sortie de contrôleur flou de type-2 pour le remplacement de la partie discontinue de la commande mode de glissement.	116
Figure VI.3. Schéma bloc de la commande RBFT2FSMC de robot manipulateur à muscles artificiels pneumatiques du CDTA.	117
Figure VI.4. Résultats de simulation de l'application des commandes RBFT2FSMC et RBFT1FSMC sur l'axe 1.	119
Figure VI.5. Résultats de Simulation de l'application de la commande RBFT1FSMC et RBFT1FSMC sur l'axe 2.	121
Figure VI.6. Résultats expérimentaux de l'application de la commande RBFT2FSMC sur l'axe 1 et 2 du robot.	123
Figure VII.1. (a) Les variables des entrées floues (a) l'erreur (e), (b) variation de l'erreur (\dot{e}).	127
Figure VII.2. variable floue de la sortie η_{fuzzy} .	127
Figure VII.3. Surface floue.	128
Figure VII.4. Commande par (a). Terminal Sliding Mode Flou, (b). Terminal sliding mode pour l'axe 1.	130
Figure VII.5. Commande par (a). Terminal Sliding Mode Flou, (b). Terminal sliding mode pour l'axe 2.	132
Figure VII.6. Les entrées/ sortie floues de type-1.	136
Figure VII.7. Schéma bloc de la commande par terminal sliding mode flou adaptatif.	137
Figure VII.8. Réponse du robot à une consigne égale à $(55^\circ, 100^\circ)$.	141
Figure VII.9. Poursuite de trajectoire de coude ($q_d = 100^\circ$).	142
Figure VII.10. Robustesse du robot vis-à-vis d'une charge de 200 g en mode régulation.	144
Figure VII.11. Robustesse du robot vis-à-vis une charge de 200 g en mode poursuite de trajectoire.	145

Liste des Tableaux

I.1 Caractéristiques du MAP FESTO utilisé	18
IV.1 Les labels de k_{fuzzy}	78
IV. Base de règles floues pour le détermination de gain k_{fuzzy}	79
V.1 Table de règles	85
V.2 Paramètres d'expérimentations	89
V. 3 Résultats des performances qualitatives	95
V.4 Table de règles	100
VI.1 Table de règles	116
VI.2 Paramètres des simulations	120
VI.3 Paramètres expérimentaux	122
VII. Table de règles floues	128

Liste des Abréviations

CDTA : Centre de Développement des Technologies Avancées ;

INSA : Institut National des Sciences Appliquées ;

MAP : Muscle Artificiel Pneumatique ;

SMC : Mode de Glissement ;

SIF : Système d'Inférence Floue ;

DLL : Degré de Liberté ;

FESTO : Société Allemande de Pneumatique Fabricant De Muscle FESTO ;

IT2FS: Système Flou de Type-2 Intervalle;

T1FS : Système Flou de Type-1 ;

FOU : Empreinte d'Incertitude ;

IT2FC : Contrôleur flou de Type-2 Intervalle ;

AIT2FC : Contrôleur Flou de Type-2 Intervalle Adaptatif ;

IT2FSMC : Commande par Mode Glissant Flou de Type-2 Intervalle ;

RBFT1FSMC : commande par mode de glissement basée sur un réseau de neurones à fonctions de bases radiales et un contrôleur flou de type-1

RBFT2FSMC : commande par mode de glissement basée sur un réseau de neurones à fonctions de bases radiales et un contrôleur flou de type-2

TSM : Terminal Sliding Mode ;

FTSMC: Terminal Sliding Mode Flou;

AFTSMC : Terminal Sliding Mode Flou Adaptatif.

Introduction Générale

Avec l'évolution technologique fulgurante et l'apparition de nouveaux systèmes de plus en plus complexes, la synthèse de leurs lois de contrôle-commande devient une tâche très lourde et difficile à réaliser. La modélisation des processus est une étape primordiale pour la mise en œuvre d'un contrôleur pour tout type de systèmes à commander. La nature non linéaire et la complexité des phénomènes physiques qui décrivent le comportement dynamique de ces nouveaux systèmes rendent leur description analytique très difficile à réaliser. De ce fait, les outils fondamentaux de synthèse des lois de commande utilisés dans le domaine linéaire deviennent caduques et d'autres méthodes plus sophistiquées sont devenues nécessaires. À cet effet, l'élaboration de nouvelles lois de commande a fait l'objet de nombreux travaux de recherche, et plusieurs solutions ont été proposées pour remédier aux problèmes de la commande des systèmes, non linéaires. Nous pouvons subdiviser les méthodes de commande des systèmes quelques soient leur nature et leurs descriptions mathématiques, en trois classes essentielles :

1. *Les techniques de commandes classiques*: On peut regrouper dans cette classe de nombreuses méthodes telles que la commande par PID, la commande par placement de pôles, la commande LQR et LQG, etc. Ces méthodes sont les plus anciennes. L'utilisation de ces dernières est généralement valable uniquement autour d'un point de fonctionnement. De plus, ces commandes présentent un autre inconvénient, celui de l'inexistence d'une possibilité de prouver analytiquement leur stabilité et/ou robustesse.
2. *Les commandes basées sur le modèle du système* : Cette classe regroupe l'ensemble des techniques dont la commande est synthétisée à partir du modèle du système à commander. Les performances réalisées par ces méthodes dépendent explicitement de l'exactitude du modèle utilisé pour l'élaboration de la commande. On trouve entre autres, la commande par linéarisation exacte d'entrée-sortie, la commande BackStepping, la commande par mode de glissement, etc.
3. *La commande par les techniques de l'Intelligence Artificielle (IA)*: Ce type de techniques est basé sur l'inspiration à partir du monde biologique. Les techniques de l'IA sont constituées de réseaux de neurones, de la logique floue et des méthodes d'optimisation méta-heuristiques. Du fait de son efficacité, cette classe de commande et ses hybridations augmente de jour en jour. L'inconvénient majeur de cette classe est la difficulté d'avoir analytiquement une garantie de la stabilité du système commandé. Elle demande généralement des moyens matériels et logiciels

sophistiqués pour leur mise en œuvre. Aussi, ces techniques se caractérisent par le nombre de paramètres important à ajuster

A cause des problèmes cités auparavant, nous avons focalisé notre effort dans ce travail sur la synthèse des commandes hybrides entre ces classes, ceci afin de compenser les inconvénients de l'une d'elles par les avantages des autres et vis versa. Dans ce travail, on s'intéresse à L'hybridation entre la commande par mode de glissement et les systèmes d'inférence floue et/ou les réseaux de neurones dont elles peuvent être regroupées en trois approches essentielles:

1. *Le remplacement de la commande discontinue* : Cette démarche est la plus connue, elle est souvent utilisée pour remédier au problème de broutement mais aussi pour augmenter la stabilité et la robustesse de la commande vis-à-vis des perturbations externes. Il y a un nombre important de travaux [17,23] qui utilisent cette approche au point où lorsqu'on parle de l'hybridation par mode glissant flou, cette approche vient à l'esprit systématiquement.
2. *Commande floue basée sur les modes de glissement* : Cette deuxième approche consiste à utiliser le principe de la commande par mode de glissement, notamment la condition d'attractivité, pour la synthèse du contrôleur flou. Elle permet de garantir la stabilité asymptotique du système à commander à l'aide d'une méthode heuristique [36]. Cette combinaison facilite l'élaboration de règles d'inférence floue. L'usage de la variable de glissement et sa dérivée permet d'impliquer la dynamique de la position, la vitesse et de l'accélération du système à commander dans la prise de décision du contrôleur, ce qui permet d'aboutir à une meilleure poursuite [35].
3. *Commande par mode de glissement basée sur la logique floue* : La dernière approche est moins utilisée. Elle consiste à estimer les différentes parties de la commande par mode de glissement par des systèmes d'inférence floue et/ou un réseau de neurones [42]. D'autres approches consistent à utiliser les systèmes d'inférence floue pour l'estimation des fonctions non linéaires du système à commander, et par la suite d'appliquer le mode de glissement sur le modèle estimé.

MOTIVATION

En (1950), [1-2] le Docteur. Joseph L. McKibben a remarqué la capacité de l'être humain à produire

des forces adéquates pour la manipulation des objets, ceci par l'usage de ces mains et bras actionnés par des muscles naturels. Cette observation l'a amené à penser à la réaliser un Muscle Artificiel Pneumatique (MAP) pour la motorisation d'orthèse pour les personnes handicapées [3]. Les MAPs ont les mêmes caractéristiques que les muscles naturels. En général, le muscle artificiel est un système contractile et compliant, stimulé par un signal externe qui est généralement l'air comprimé. Par définition, le MAP est un tube en caoutchouc recouvert d'une enveloppe tressée et attachée aux extrémités par des embouts métalliques. Le principe de motricité est le suivant: quand la pression de l'air contenu dans le tube augmente, le muscle se contracte longitudinalement et inversement, il s'allonge lorsque la pression d'air dans le muscle diminue. Depuis l'apparition du premier MAP, plusieurs muscles ont été développés tels que le muscle plissé [1], l'Air muscle [4], le muscle FESTO [5], le muscle INSA [6], etc. D'autre part, en se basant sur ces muscles, plusieurs robots ont été conçus, on peut citer les robots humanoïdes Lucy et ISAC dans [7] et [8] respectivement, le robot manipulateur à 7-ddl de l'INSA de Toulouse [6], les robots de la société Shadow Robot Company [1], le robot œil [9] et beaucoup d'autres.

Le muscle artificiel pneumatique et donc le robot à muscle a plusieurs caractéristiques attirantes : premièrement, les MAPs ont un poids léger et un rapport puissance/ poids très intéressant comparé aux moteurs électriques. Deuxièmement, l'énergie pneumatique utilisée par ces systèmes est gratuite et ne présente pas le risque d'incendie. Troisièmement, le coût de maintenance est faible. Ces systèmes permettent entre autre un contact sécurisé avec un opérateur humain. Cependant, il est très connu qu'à cause des variations inertielles, des frictions et des hystérésis, les MAPs et les robots à MAPs sont des systèmes fortement non linéaires, où la connaissance de leurs modèles avec des paramètres précis est quasiment inatteignable par l'usage des techniques de modélisation ordinaires. Malgré ces difficultés, la modélisation de MAPs a été l'objet de nombreux travaux de recherche depuis de nombreuses années. Au début, la majorité des chercheurs préféraient l'utilisation d'une approche basée sur la géométrie du muscle, de jour en jour, les phénomènes qui caractérisent ces actionneurs ont attiré l'attention, et récemment, certains chercheurs ajoutent même les propriétés des matériaux pour accentuer la validité de leur modélisation. On trouve entre autres les modèles de Chou et Hannaford [10], Tondu et Lopez [11], Tsagarakis, Caldwell *et al.* [12], Reynolds *et al.* [13], Kawashima *et al.* [14], Bertetto et Riggio [15] et celui de Lilly et Yang [28]. Par ailleurs, la prise en considération des phénomènes physiques liés au MAP augmente explicitement la complexité du modèle théorique, et engendre par conséquent des complexités pour la synthèse de la loi de commande. Ceci a poussé les chercheurs à se pencher vers les techniques de commande robuste, où l'objectif actuel dans ce domaine est de proposer des commandes robustes capables de fournir les

meilleures performances possibles à des vitesses élevées. Durant les dernières années, plusieurs lois de commande robuste ont été développées et appliquées sur des robots à MAP, on peut citer un PID dans [7] [8], une commande par mode de glissement classique et d'ordre supérieur [5], une commande neuronale [16], une commande par mode de glissement flou [17], et bien d'autres.

Bien que les différentes commandes citées sur le contrôle de la position du robot à MAP soient satisfaisantes et aient donné de bons résultats sur différentes formes de robots actionnés par des MAPs, néanmoins elles comportent certains points négatifs. Tout d'abord, les techniques de commande par PID ont prouvé leur limite face aux incertitudes paramétriques, aux perturbations externes, etc. La commande par réseau de neurones conventionnels souffre de la lourdeur de l'apprentissage en ligne et par conséquent, engendrent des difficultés de mise en œuvre en temps réel. La commande par mode de glissement classique présente aussi des inconvénients comme par exemple celui du broutement, la non convergence en temps fini, la nécessité d'avoir des informations sur les dérivées de la surface de glissement ainsi que la sensibilité aux bruits dans le cas de la commande par mode de glissement d'ordre supérieur. Une autre difficulté est celle des commandes non-linéaires basées sur le modèle du robot à muscle. Ces modèles sont difficiles à obtenir. De plus, l'implémentation de ces techniques amène à des opérations complexes et lourdes pour les calculateurs.

Notre travail de thèse consiste à appliquer des lois de commande robustes basées sur les nouvelles approches à structure variable par mode de glissement telles que le '*terminal sliding mode control*' et les nouvelles approches de la commande par les techniques de l'intelligence artificielle telles que la logique floue de type-2 et le réseau de neurones de fonctions à bases radiales, ainsi que leurs hybridations, avec l'objectif d'éliminer les inconvénients de certaines commandes par les avantages des autres et vis versa, sur des robots manipulateurs actionnés entièrement par des muscles artificiels pneumatiques. Deux robots ont été utilisés : le robot à 3-*ddl* du CDTA et le robot à 7 axes du groupe GEPETO du LAAS-CNRS de Toulouse, France.

CHRONOLOGIE DE LA THESE

En première année, nous avons effectué une recherche bibliographique sur la commande par logique floue de type-1 et de type-2 ainsi que sur la commande par mode de glissement classique et par terminal sliding mode. En deuxième année, nous avons mis en œuvre la simulation et l'expérimentation de ces techniques sur le robot du CDTA. Au cours de la troisième année, une

commande par réseau de neurones combinée avec un système d'inférence floue de type-2 basée sur le mode de glissement a été investie. En quatrième année, une commande par terminal sliding mode flou adaptatif a été proposée et appliquée sur le robot de l'INSA de Toulouse en France. Durant ces années, des valorisations ont fait l'objet de participations dans des conférences internationales et des publications.

ORGANISATION DU MEMOIRE

La thèse s'articule sur sept chapitres :

Chapitre 1 : Le chapitre contient un état de l'art sur les muscles artificiels pneumatiques. Une vue générale sur leur historique, leur mode de fonctionnement, leur utilisation actuelle, leur avantages et inconvénients, etc. Par la suite, une description matérielle et logicielle de deux robots utilisés dans cette thèse est présentée (le robot à muscles du CDTA et son homologue du l'INSA-LAAS de Toulouse, France). La troisième partie du chapitre est destinée à la proposition de modèles de simulation. Pour le robot du CDTA, le modèle choisi est issu d'une identification, en revanche pour le deuxième robot, un modèle dynamique à deux axes est proposé.

Chapitre 2 : Ce chapitre est consacré à un état de l'art sur les modes de glissement. Des commandes par mode de glissement classique, d'ordre supérieur et par terminal sliding mode ont été présentées.

Chapitre 3 : La logique floue de type-2 est un domaine qui est dans ses premiers balbutiements, où peu des travaux sont disponibles notamment en français. Ce chapitre est suffisamment détaillé afin de répondre aux besoins et aux questions des chercheurs du domaine. Une comparaison entre le cas traditionnel dit de type-1 et celui de type-2 est donnée pour montrer l'intérêt de cette nouvelle logique. L'extension du système flou de type-1 en réseau de neurones est aussi présentée.

Chapitre 4 : Les limites de la commande par mode de glissement classique, par logique floue de type-1 ainsi que leur hybridation sont montrées dans ce chapitre.

Chapitre 5 : Une commande par logique floue adaptative de type 2 est proposée et appliquée sur le robot du CDTA, par la suite une autre commande floue de type-2 basée sur la condition d'attractivité des modes de glissement est aussi proposée, appliquée et comparée au cas classique.

Chapitre 6 : Dans ce chapitre, nous proposons une commande basée sur une combinaison entre un réseau de neurones de fonctions à bases radiales et un système d'inférence floue de type-2. Cette commande a été développée sur la base de la commande par mode de glissement, la commande est appliquée sur le robot du CDTA.

Chapitre 7: L'application d'une commande par terminal sliding mode flou adaptatif a fait l'objet de ce chapitre. Cette commande est proposée et appliquée sur le robot de l'INSA de Toulouse.

Nous clôturons cette thèse par une conclusion générale qui permet de fixer les idées sur certains points qui concernent les commandes développées dans ce travail.

CHAPITRE I. *Etat De L'art Sur Les Robots A Muscles Artificiels Pneumatiques.*

I.1 INTRODUCTION

Le MAP est initialement inventé par le Docteur Joseph L. McKibben dans les années cinquante [1-2]. Jusqu'à nos jours, ce type de systèmes a porté plusieurs nominations telles que le 'Air Muscle' [4], le Muscle Artificiel de McKibben [2], le Muscle Artificiel Pneumatique [1], etc. La conception et la configuration du MAP lui ont procuré des caractéristiques tels que, une raideur variable, appelée aussi *compliance*, une flexibilité structurelle, un rapport puissance / poids très important, etc., souhaitables pour l'exploitation dans les systèmes robotisés. Néanmoins, malgré tous ces atouts, cet actionneur est régi par une relation fortement non linéaire entre la force de contraction, la pression interne et la longueur du muscle [13]. De plus, la structure intrinsèque des matériaux qui le composent engendre une hystérésis au moment de la contraction/extraction [1-3]. Ces handicaps génèrent des problèmes illimités de modélisation et par conséquent, un problème délicat de contrôle en position et en poursuite de trajectoires à vitesse élevée. Dans ce chapitre, nous présenterons les MAPs sous différents aspects : Les différents types de muscles, leurs avantages et inconvénients, leurs domaines d'application, etc. Par la suite, une description détaillée des robots du CDTA et de l'INSA utilisés dans ce travail sera présentée. Des modèles pour les deux robots seront proposés.

I.2 LES TYPES D'ACTIONNEURS A MUSCLES ARTIFICIELS

Le mot « muscle artificiel » est générique, une telle nomination a été attribuée à plusieurs systèmes d'actionnement, dont le principe de fonctionnement de chacun par rapport aux autres est quasiment différent. Contrairement aux actionneurs classiques, les muscles artificiels sont liés explicitement au domaine de la physique des matériaux et de la chimie. Avoir des connaissances en la matière est très important pour la compréhension et la maîtrise de leur fonctionnement. Depuis quelques années, de nombreux laboratoires de recherche ont commencé à s'intéresser au développement et à exploitation de cette classe d'actionneurs. Il existe aujourd'hui un nombre assez important de systèmes qui fonctionnent grâce à ces actionneurs. D'après [61], les muscles artificiels en général

sont divisés en deux grandes classes selon leur mode de mise en marche (mode de stimulation) tels que les muscles à alimentation interne et les muscles à alimentation externe (figure I.1).

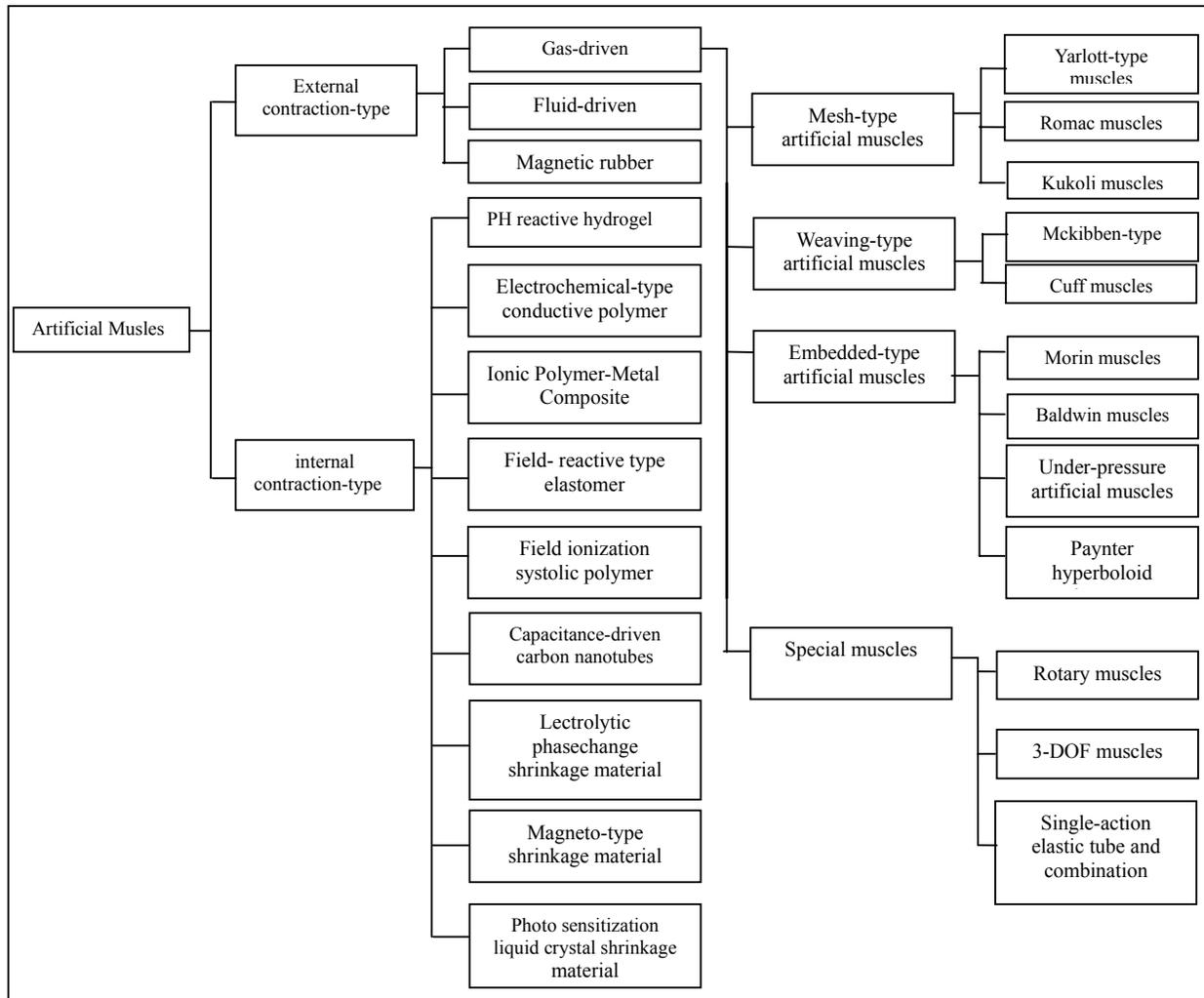


Figure I.1 Classification des muscles artificiels [61].

Les muscles artificiels les plus utilisés actuellement sont résumés par DIAZ GAZEL SERGIO dans [9].

I.3 LE MUSCLE ARTIFICIEL PNEUMATIQUE

La simplicité de fonctionnement et de mise en œuvre fait du MAP le plus classique parmi les actionneurs modernes dans ce domaine. Ce muscle est constitué d'une chambre à air qui engendre la force de traction quand il se gonfle ou se dégonfle par l'usage de l'air comprimé. En fonction de la technologie de fabrication, la figure I.2 montre les MAPs les plus répandus :



Figure I.2 Les MAPs les plus répandus

Les MAPs les plus répandus (figure I.2) sont essentiellement :

- Le muscle de la compagnie *FESTO* qui porte le nom de son fabricant [5]. Le MAP de FESTO est confectionné avec un tuyau en caoutchouc qui inclut un tressage de fibres aramides présentant une grande résistance à la rupture et une haute flexibilité. A notre connaissance, ce muscle est le plus demandé à l'échelle mondiale. Jusqu'à nos jours, le muscle FESTO a été utilisé par un nombre important de systèmes, comme c'est le cas pour le robot du CDTA. Il se caractérise essentiellement par sa force de traction très importante et sa durée de vie assez longue. Le seul problème avec ce muscle est l'inexistence d'une gamme de muscles miniaturisés.
- L'Air Muscle est commercialisé par la société anglaise *Shadow Robot Company* [15]. C'est le deuxième actionneur parmi les actionneurs les plus sollicités. Ce muscle est moins répandu par rapport au muscle FESTO, l'Air Muscle existe avec des tailles miniaturisées et il est utilisé précisément pour des robots miniatures, tel que le robot Hand [15].
- Le muscle plissé de l'université Vrije, (*Belgique*). Il est nouveau par rapport à ceux cités auparavant. Il est dédié actuellement à la recherche. Plusieurs robots ont été actionnés par ce muscle, parmi eux, le fameux robot Lucy [7].
- Remarque: Il existe d'autres MAPs qui sont moins répandus, on peut citer : *Rubbertuator* du fabricant de pneus japonais *Bridgestone* [1], le muscle canadien *Romac* [1], etc.

I.4 LES APPLICATIONS DES MUSCLES ARTIFICIELS PNEUMATIQUES

L'analogie entre le muscle biologique et le muscle de McKibben permet de préciser la notion de muscle artificiel. Dans la dernière décennie, plusieurs laboratoires de recherche se sont intéressés au développement et au contrôle des robots actionnés complètement par des MAPs, citons par exemple :

- Le robot grimpant Robine de l'université de Nashville, *Etats-Unis* [10];
- Le robot Exosquelette de réhabilitation de l'université de Salford, *Angleterre* [10];
- Le robot cafard de l'université de Case Western Reserve, *Etats-Unis* [10];
- Le robot humanoïde ISAC de l'université de Vanderbilt, *Etats-Unis* [8].
- Le robot bipède LUCY de l'université de Vrije, *Belgique* [7];

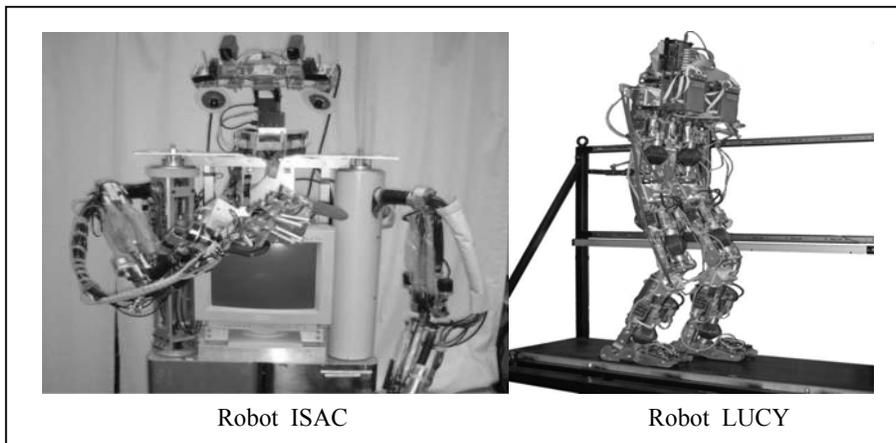
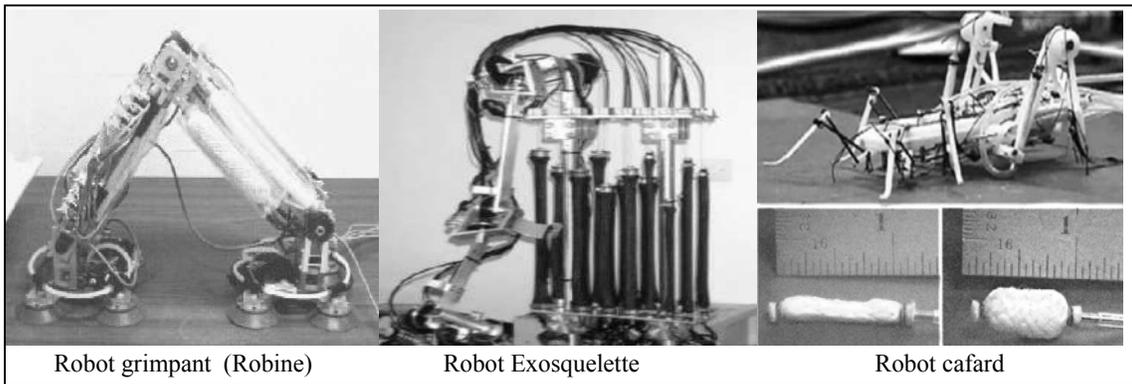


Figure I.3 Exemples de robots à MAPs

I.5 AVANTAGES ET INCONVENIENTS DES ROBOTS A MAPs

Tout type d'actionneurs, que ce soit pneumatique, hydraulique ou électrique présente des avantages et des inconvénients. L'énumération de ces derniers permet de choisir le système d'actionnement le plus adéquat, en fonction de l'application envisagée. Dans ce cadre, cette section vise à recenser des avantages et les inconvénients du MAP.

I.5.1 Les Avantages du MAP

L'utilisation de l'énergie pneumatique pour l'alimentation des MAPs offrent des propriétés pouvant s'avérer intéressantes pour les applications industrielles ou de service. Du point de vue sécurité, l'absence d'un fort courant électrique écarte les risques d'explosion et d'inflammation et réduit considérablement les interférences électromagnétiques. Les MAPs sont économiques, leur puissance massique est très élevée par rapport à un actionneur classique, le rapport puissance/poids est très important comparé au cas des actionneurs électriques. Ils engendrent des forces de contraction de même type que celles produites par les muscles humains. Ils présentent naturellement une faible rigidité qui peut être mise à profit pour limiter les conséquences des collisions et faciliter les tâches communes robot-robot ou robot-humain, etc.

I.5.2 Les Inconvénients du MAP

Le taux de contraction maximal du MAP est limité, ce qui réduit donc le volume de travail du robot. Le MAP n'offre pas une bonne autonomie énergétique, car sa dépendance d'une source pneumatique sous pression limite d'une part son application à la motorisation de bras à poste fixe et le rend, d'autre part, difficilement utilisable dans une application mobile [62]. La présence de frottement entre le tube en caoutchouc et la tresse synthétique entraîne une hystérésis, ceci engendre un effet néfaste sur le comportement de l'actionneur. La présence de non-linéarité dans ce type d'actionneurs ne permet pas d'obtenir un modèle de comportement précis, ce qui fait du MAP un système difficile à commander [34].

I.6 COMPOSITION D'UN MAP

Le MAP dépend de la technologie du fabricant. Afin de donner une idée sur le MAP. Dans ce qui suit, nous donnons une description du MAP Toulousain fabriqué au niveau de laboratoire de robotique de l'INSA de Toulouse. Ce muscle est composé des éléments suivants :

- **Le Tube en Caoutchouc :** C'est une chambre à air cylindrique en caoutchouc synthétique. Le principe de conversion entre les forces radiales et les forces de traction axiales du muscle impose que le tube interne en caoutchouc soit assez fin et qu'il reste en contact permanent avec la tresse, tout au long de la contraction. Cet aspect est très important pour que la transformation énergétique du muscle soit valide [59].
- **La Tresse Textile :** Cette tresse est constituée d'une gaine tubulaire en textile déformable, tressée symétriquement à double hélice par des fils inextensibles. Cela révèle un réseau de pantographes dont les sommets sont les nœuds d'entrecroisement des fils. Ce type de tressage permet à la gaine tubulaire de s'ouvrir au moment de la contraction du muscle, et de moduler le gonflement du tube interne en caoutchouc, en gardant sa forme cylindrique [59].
- **Les Embouts Métalliques :** Deux différents embouts métalliques sont fixés chacun à une extrémité du muscle, sur lesquels une jupe métallique est sertie, afin de garantir la cohésion entre la tresse et le tube en caoutchouc. L'un des embouts est relié à la chaîne de transmission des efforts de traction, pendant que l'autre est percé pour permettre l'alimentation en air comprimé du muscle ainsi que la fixation du muscle sur la structure du bras.



Figure I.4 Quelques variantes de MAPs Toulousains

I.7 PRINCIPE DE FONCTIONNEMENT D'UNE PAIRE DE MAPs PLACES EN ANTAGONISME

La mise sous forme antagoniste de deux MAPs permet de réaliser un modèle similaire au modèle humain biceps-triceps. Trois approches de mise en marche peuvent être envisagées :

1. Les deux muscles sont actionnés par une pression indépendante. Dans ce cas, les signaux de contrôle doivent être synchronisés.
2. Un seul muscle est commandé, le muscle antagoniste servant de ressort de rappel. Cette approche présente l'inconvénient de limiter les performances de l'actionneur.
3. La commande la plus utilisée et qui a fait l'objet de ce travail consiste à commander les deux muscles par une différence de pression. Ce type de configuration a été proposé par la société *Bridgestone* à la fin des années 80 [58].

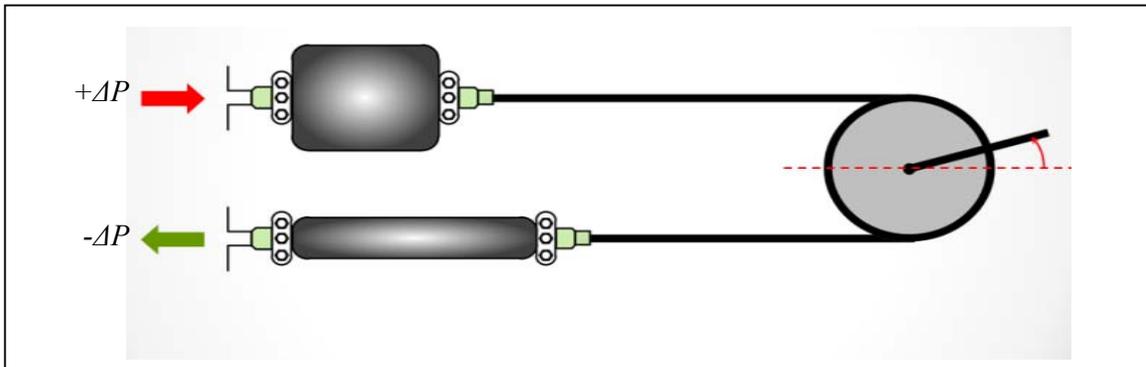


Figure I.5 Mise en antagonisme de deux MAPs

Le principe de la dernière méthode de mise en marche consiste à initialiser chaque articulation à une position initiale θ_0 . Pour ce faire, la pression dans les muscles antagonistes de chaque articulation est initialisée à une pression de départ P_0 . Le mouvement est produit par une variation opposée et simultanée de la pression ΔP dans les deux muscles, c'est-à-dire $P_1 = P_0 + \Delta P$ et $P_2 = P_0 - \Delta P$ (voir la figure I.5). La contraction d'un muscle se fait par l'augmentation de sa pression interne, l'allongement du muscle antagoniste se fait par une diminution de sa pression interne.

I.8 PLATES-FORMES EXPERIMENTALES UTILISEES DANS CETTE THESE

Deux plates-formes expérimentales ont été utilisées durant ce travail. La première est un bras manipulateur à 3-*dll* disponible au niveau de la DRP du CDTA, d'Alger. La deuxième plateforme est celle de l'INSA de Toulouse, il s'agit d'un robot manipulateur à 7-*dll*. Vue l'importance de ces systèmes pour la suite de ce manuscrit, nous avons consacré cette section à une description détaillée de leurs technologies.

I.8.1 Description du Bras Manipulateur à MAP du CDTA

Le banc d'essais du CDTA est un robot à trois (03) axes dont chaque articulation est de type rotoïde, chacune est actionnée par une paire de MAPs de type FESTO [5,49] montée en antagonisme. Le robot a été fabriqué par l'entreprise Robosoft [49,50]. Les anomalies physiques existantes dans ce prototype font de lui l'exemple parfait pour examiner les lois de commande développées au cours de ce travail. La figure I.6 montre la structure détaillée d'un axe du robot en configuration antagoniste :

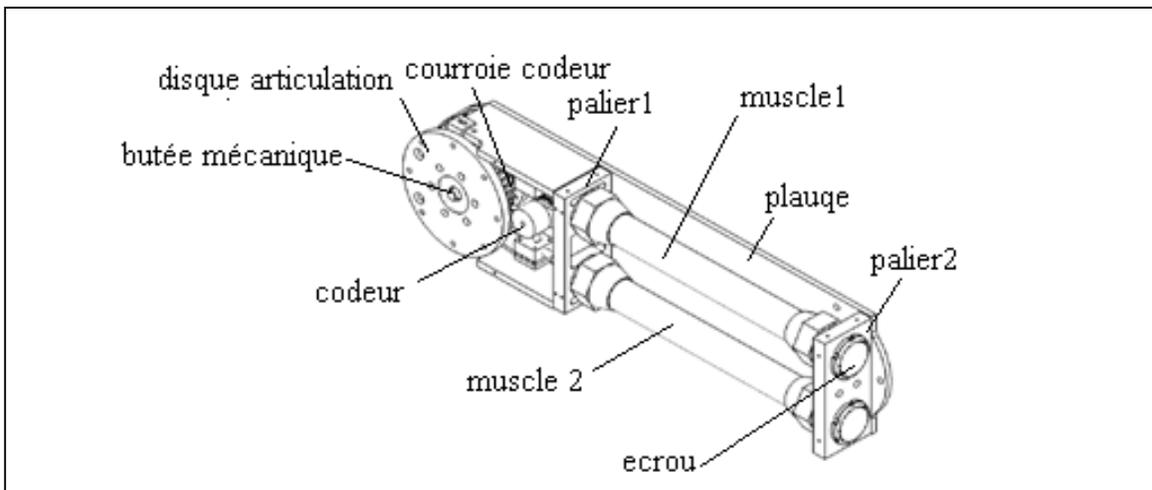


Figure I.6 Structure interne d'un axe du bras à MAP.

La figure I.7 représente une vue générale du bras manipulateur du CDTA. Afin d'avoir une similarité avec le bras ou la jambe humaine, sa base a été fixée sur une structure métallique en haut, ce qui positionne son organe terminal en bas.

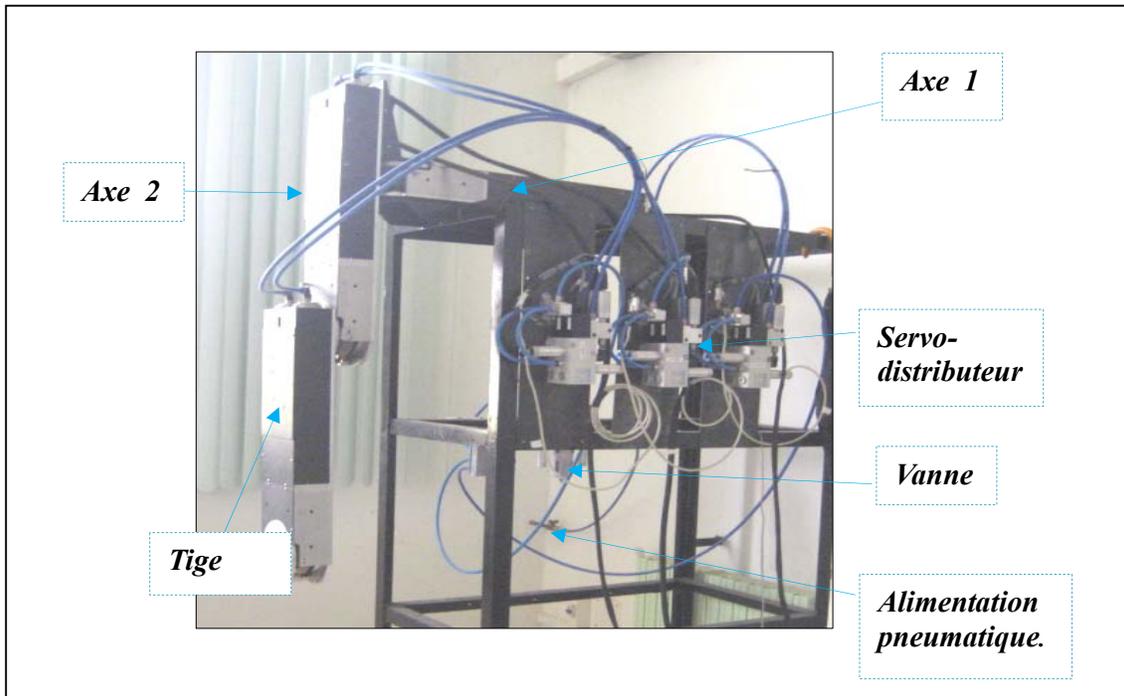


Figure I.7 Le bras manipulateur à MAP du CDTA.

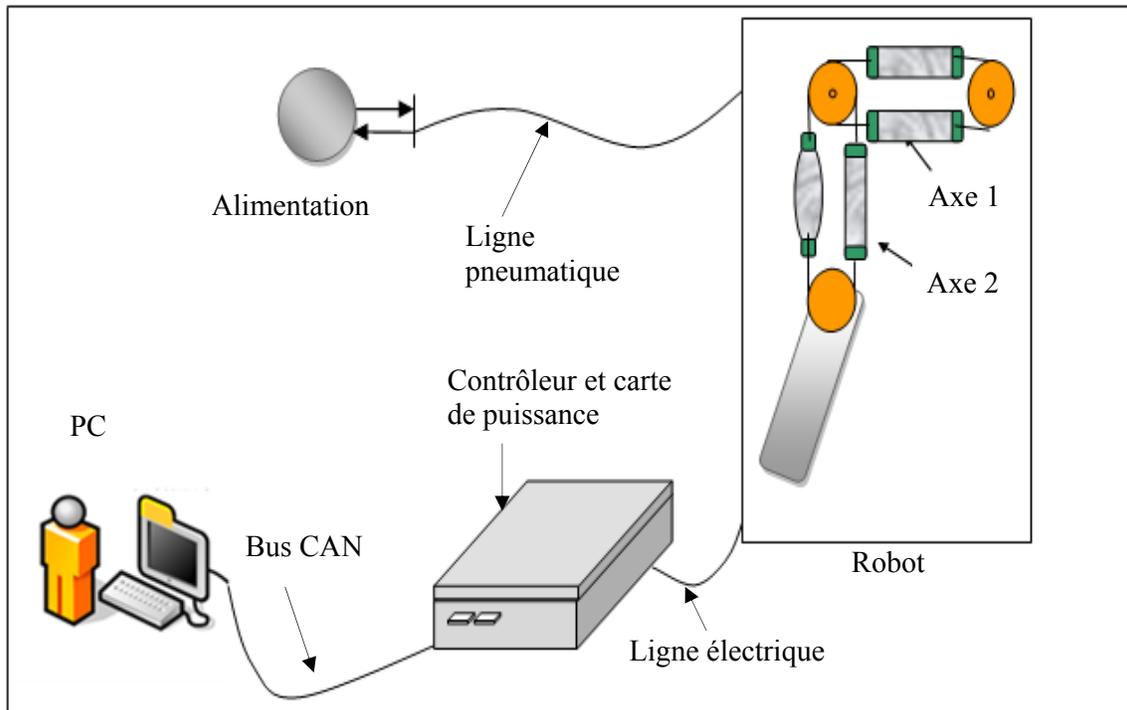


Figure I.8 Vue des quatre blocs du système de commande du robot [34].

Le banc est basé sur une architecture classique d'un système électropneumatique. A ce titre, il est composé de plusieurs éléments de technologies différentes, le schéma synoptique de la figure I.9

montre le circuit de l'énergie pneumatique, électrique et les différentes composantes du robot. Quatre grands blocs fonctionnels (figure I.8) peuvent être distingués dont les trois premiers ont été réutilisés (PC, carte de commande et robot) et le quatrième est créé, il consiste aux programmes de mise en marche. Les éléments physiques qui constituent le robot du CDTA sont :

1. Une source d'énergie pneumatique composée d'un compresseur délivrant de l'air filtré et réglé à quatre (04) Bar par une valve de type « MPYE-5-3/8-010-B ».
2. Des Servo-distributeurs de type « HEL-D-MAXI » de FESTO [49] permettant de convertir la commande calculée en Volt, en débit massique entrant aux chambres des muscles.
3. Des actionneurs (les MAPs) sont de type « MAS-40-300-N-AA-MOK » ont pour objectif de convertir l'énergie pneumatique issue des servo-distributeurs en des forces permettant le déplacement des axes du robot.
4. Une instrumentation composée de capteurs qui transforment l'information physique (pression, position) en information électrique. Ces informations sont conditionnées pour être compatibles avec le calculateur. Les capteurs de position pour chaque axe sont des codeurs incrémentaux de type « METRONIX, S48-8-3600ZO » fixés sur les articulations du robot.
5. Des calculateurs numériques qui délivrent la commande à appliquer sur les servo-distributeurs. Cette commande est calculée en fonction de la (ou les) consigne(s) et des informations provenant des capteurs. Ces cartes sont fabriquées par la société Robosoft [50], et construites autour du microcontrôleur MPC555 de Motorola. Toutes les cartes communiquent avec l'unité centrale du PC *via* des Bus CAN.
6. Des convertisseurs analogiques/ numériques de type « Advantech, PCI-1720 ». Son rôle est de convertir le signal issu du capteur en un paquet de données qui seront traitées ensuite par le calculateur.
7. Des convertisseurs numériques/analogiques de type « Analog Device DAC8420 » permettent de convertir les valeurs calculées par l'organe de commande en un signal analogique. Ce dernier va exciter les servo-distributeurs.
8. Le système à commander : c'est la structure mécanique du robot.

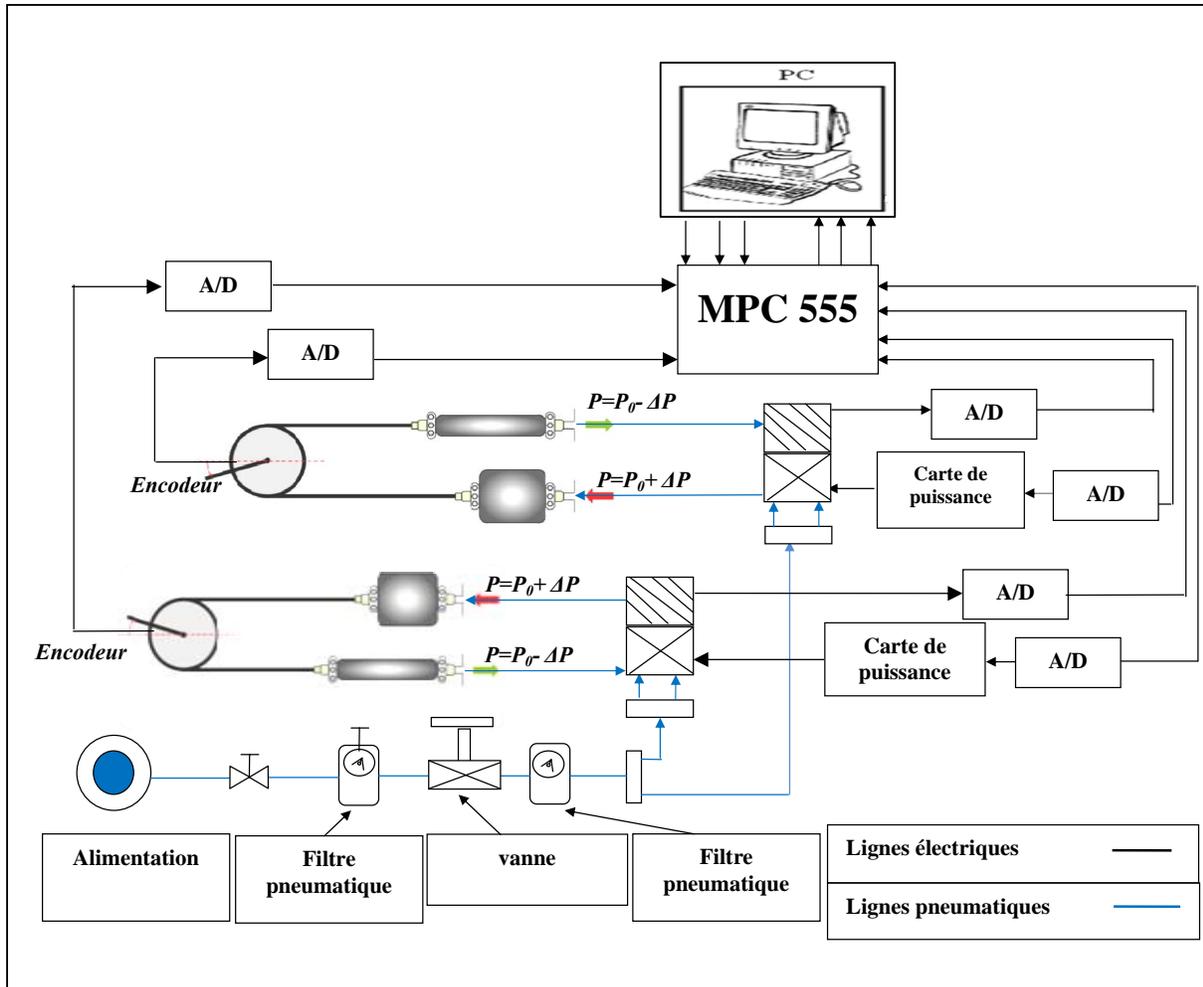


Figure I.9 Schéma détaillé du principe de fonctionnement du robot du CDTA.

I.8.2 Caractéristiques du MAP FESTO Utilisé

Comme nous l'avons mentionné auparavant, il existe un nombre important de types d'actionneurs à MAPs. Le bras manipulateur du CDTA est entièrement motorisé par des muscles fabriqués par l'entreprise Allemande FESTO [49]. En fonction de leurs caractéristiques physiques, ce groupe propose plusieurs modèles de muscles. Il confectionne des MAPs de 10mm, 20mm et 40mm de diamètre pour des longueurs au choix de 40mm jusqu'à 9000mm [62]. Ces muscles parviennent à produire une force maximale d'environ 630N, 1500N et 6000 N respectivement. Ils opèrent à une pression inférieure à 8.0 bars, avec un taux de contraction de 25% [62]. La figure I.10 représente quelques modèles de MAPs FESTO :



Figure I.10 Muscles artificiels pneumatiques du FESTO.

Les caractéristiques du muscle pneumatique FESTO nommé « MAS-40-300-N-AA-MOK » utilisé pour motoriser notre bras sont résumées dans le tableau I.2 ci-dessous :

Mode fonctionnement	A simple effet, en traction
Masse du muscle	1,185 Kg
Longueur	300 mm
Force sous 3 Bar	968,5 N
Degré de contraction	17 %
Pression max	6 Bar
T°c ambiante	+5°C / +60°C

Tableau I.1 : Caractéristiques du MAP FESTO utilisé

I.8.3 Modélisation du Robot à MAP du CDTA

Il est utile de mentionner ici que, malgré l'existence de plusieurs modèles dans la littérature pour les robots à MAPs [11-15], chacun de ces modèles reste spécifique au robot dans lequel la modélisation a été faite. Ce problème est posé à cause des différences technologiques entre les muscles utilisés pour l'actionnement d'un robot à l'autre. La configuration ainsi que l'équipement autour du robot pour sa mise en fonctionnement jouent un rôle primordial dans la détermination du modèle. Dans la littérature, afin d'avoir un modèle de simulation simplifié qui se rapproche du modèle réel, une deuxième alternative a été donnée. Elle consiste à utiliser un modèle de représentation (modèle issu de l'identification). Par exemple, dans [54] les auteurs utilisent un modèle de second ordre avec un retard pur pour deux axes d'un robot à 7-ddl. Dans [20] une approximation linéaire de la dynamique du robot SCARA actionné par des MAP par des équations différentielles du second ordre a été utilisée. Dans le travail [17], un modèle optimisé du second ordre en utilisant un algorithme génétique a été proposé pour une meilleure identification linéaire du

robot manipulateur à 2-*ddl* motorisé par des MAP. Les modèles obtenus dans ce cas donnent une description partielle de la dynamique du robot, les erreurs de modélisation seront compensées par une ou plusieurs commandes robustes.

I.8.4 Identification des Axes du Robot du CDTA et Etude de Son Couplage

Le modèle d'un robot manipulateur quelque soit sa technologie est un système non-linéaire multi-variable fortement couplé. En fonction de sa configuration et le point de fonctionnement souhaité, il peut être considéré comme un système totalement ou partiellement linéaire et/ou découplé [34]. Dans notre cas, la détermination du modèle est une tâche très lourde et extrêmement difficile à réaliser. Pour remédier à ce problème, nous avons fait appel aux techniques d'identification linéaire avec l'objectif d'avoir un modèle linéaire de simulation, ensuite les erreurs de modélisations seront compensées par l'usage d'une ou plusieurs lois de commande robuste. Le modèle linéaire présenté dans cette section est de forme multi-variable. Afin d'étudier le degré de couplage entre les axes du robot, une étude des interactions est faite à l'aide de la méthode appelée Matrice de Gains Statiques (Gain Array Methodology RGA en Anglais).

I.8.4.1 Procédure d'Identification

Un modèle de représentation linéaire est choisi afin de décrire partiellement la dynamique de robot du CDTA. En général, la procédure de modélisation/ identification se résume en cinq étapes telles que :

- Le choix du signal d'excitation ainsi que ses caractéristiques ;
- L'acquisition des données entrées /sorties du système ;
- Le choix de la structure du modèle ;
- L'estimation des paramètres ;
- La validation du modèle.

I.8.4.2 Choix du Signal d'Excitation

Afin d'obtenir un bon modèle linéaire du robot, le signal d'excitation doit couvrir une large plage des fréquences du système. Notre choix est porté sur la Séquence Binaire Pseudo Aléatoire (SBPA) [60]. À cause de sa ressemblance au bruit blanc, ce signal bien connu est souvent utilisé pour l'identification des systèmes [71]. La figure I.11 montre l'organigramme de génération d'un SBPA.

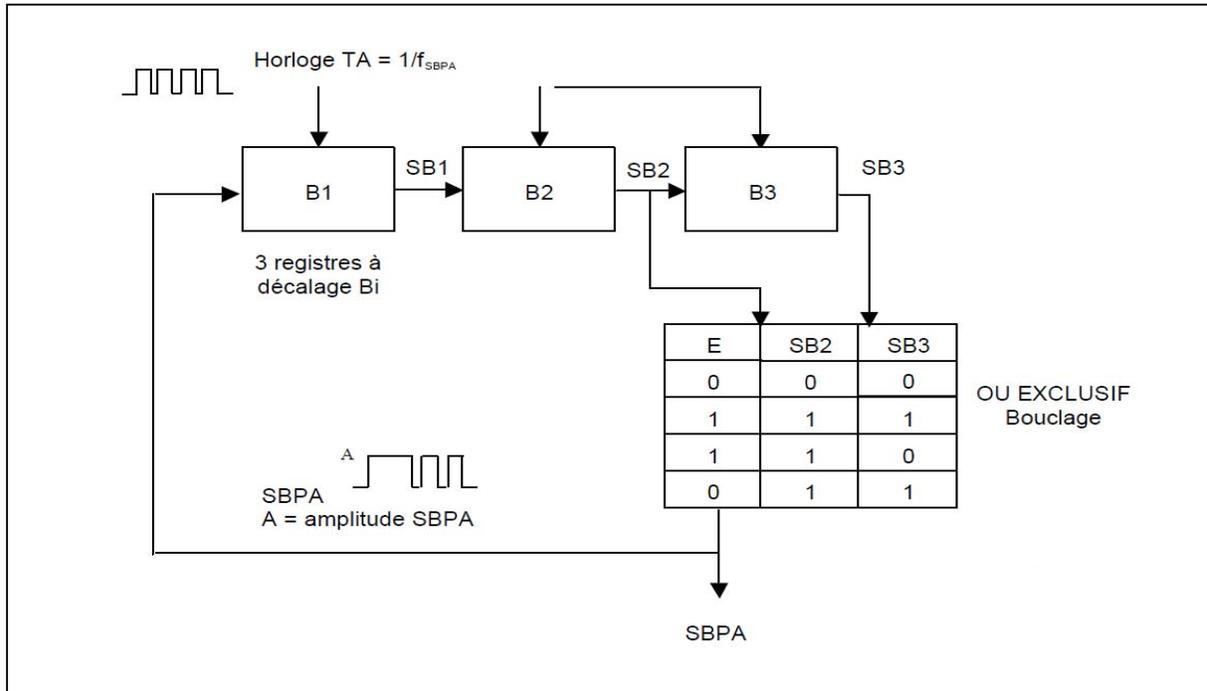


Figure I.11 Exemple d'un SBPA de longueur 7.

Le signal SBPA choisi doit vérifier l'égalité suivante [71]:

$$L = P \cdot 2^N - 1 \quad (I.1)$$

Où: L est la longueur de la séquence;

N est le nombre de cellules ;

P est une constante positive.

Après plusieurs essais infructueux, nous avons sélectionné les signaux d'entrée pour l'excitation des axes du robot (figure I.12.a et b), les paramètres de l'équation (I.1) sont $N = 9$, $P = 3$, alors $L = 1535$, ces paramètres sont valables pour les deux axes.

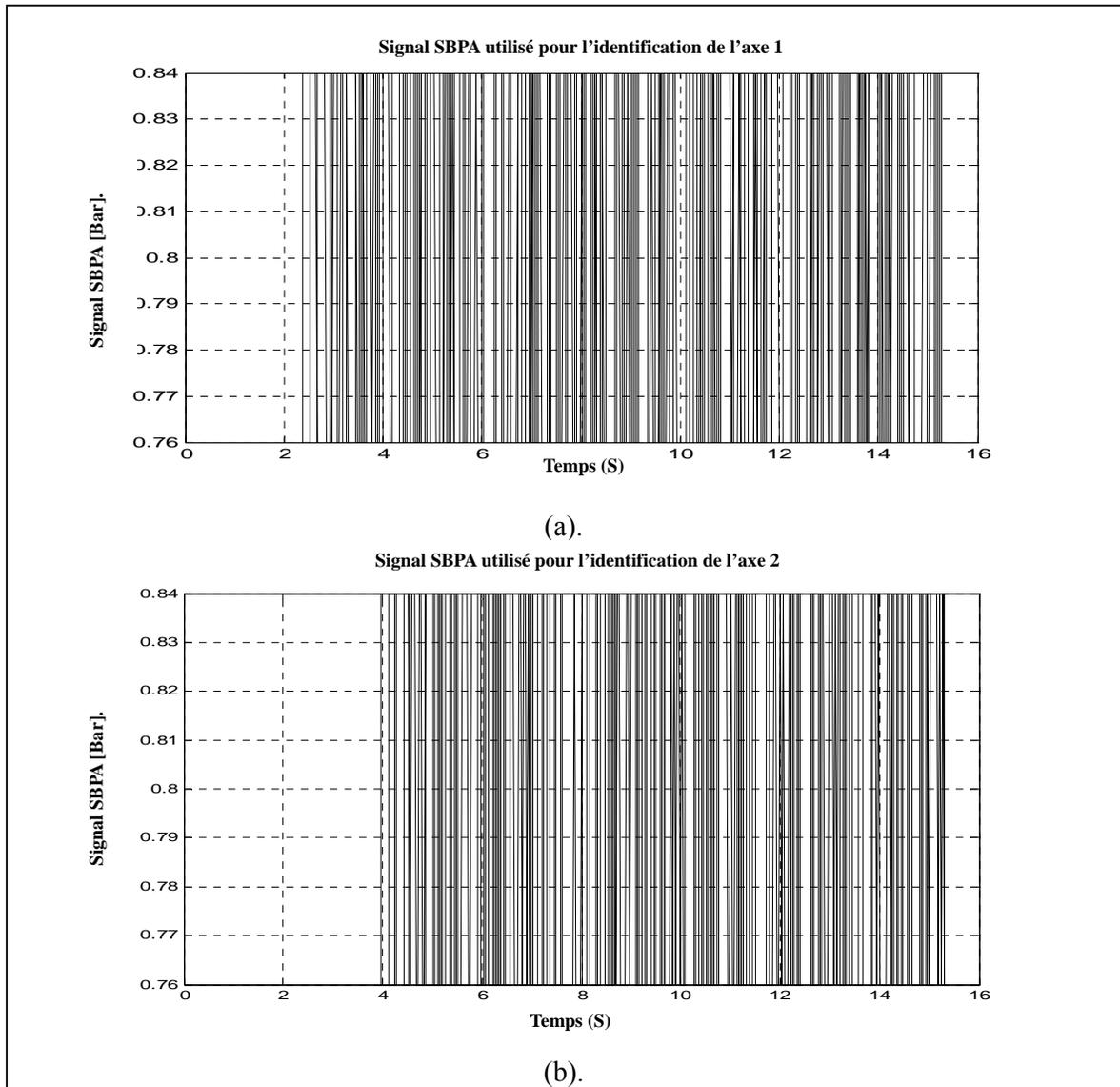


Figure I.12 Signal SBPA utilisé pour l'identification du modèle du robot.

Les signaux choisis ont été injectés aux axes du robot et les réponses en positions angulaires sont enregistrées. Ce signal correspondant à 14° pour l'axe 1 et 10° pour l'axe 2 est le point de fonctionnement dans lequel nous tenons à identifier les modèles des axes du robot. Afin d'isoler le couplage entre les axes, une excitation axe par axe a été effectuée, la dynamique du robot avec la prise en compte de l'interaction entre ces axes donc a été élaborée. Nous avons évalué différents degrés de modèles par l'utilisation de *Toolbox Ident* de *Matlab*. Le modèle multivariable du robot final obtenu autour du point de fonctionnement est décrit par l'équation (I.2). Nous représentons dans les figures I.13. a-d les résultats expérimentaux de l'identification des axes du robot obtenus, avec la prise en compte de l'interaction entre eux :

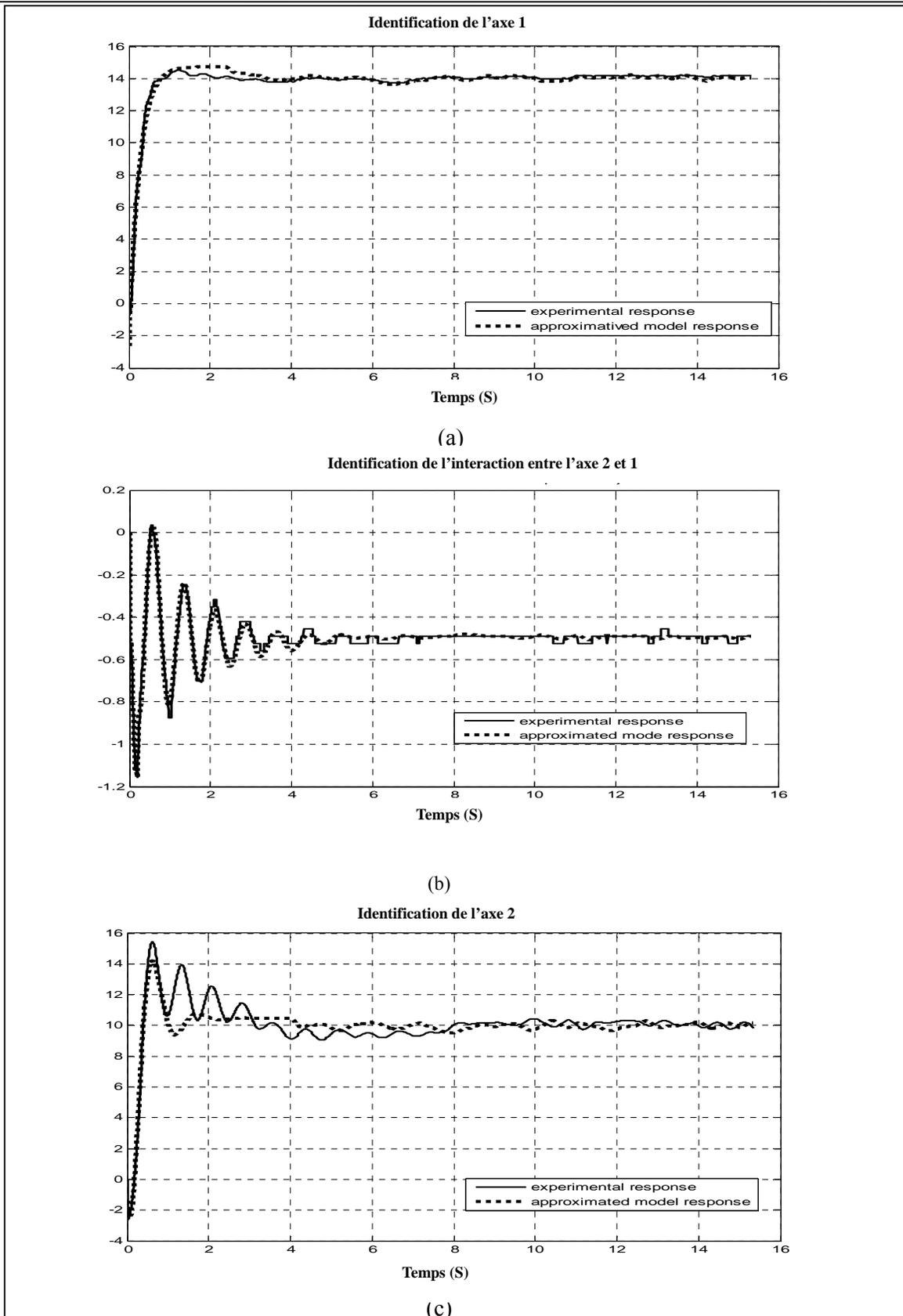


Figure I.13 Identification des deux axes avec prise en compte du couplage.

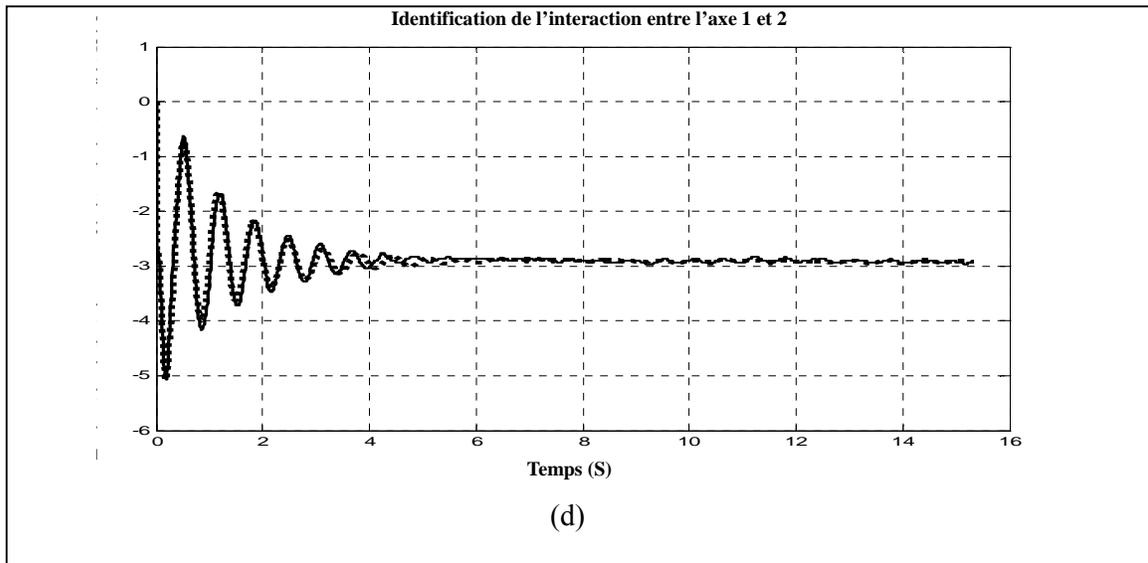


Figure I.13 Identification des deux axes avec prise en compte du couplage.

Les graphiques de validation du modèle sont illustrés dans la figure (I.13). Les lignes discontinues représentent les réponses des modèles estimés et les lignes continues sont les sorties du système réel issues des expérimentations. On peut constater à partir des résultats de l'identification que le modèle choisi pour modéliser l'axe 1 ainsi que les interactions entre les axes semblent parfaits, le modèle choisi pour l'axe 2 est moins bon, mais il représente davantage l'axe 2. Ce modèle de degré relatif 2 nous permettra de synthétiser des commandes par mode de glissement d'ordre 1. Notons ici que le modèle à choisir est généralement arbitraire; il est défini en fonction de la commande envisagée et les applications à réaliser. A cet effet, le modèle donné par l'équation (I.2) nous suffira pour la suite du travail. Le schéma multi-variable pour les deux axes du robot peut être représenté par le schéma bloc de la figure (I.14) suivante :

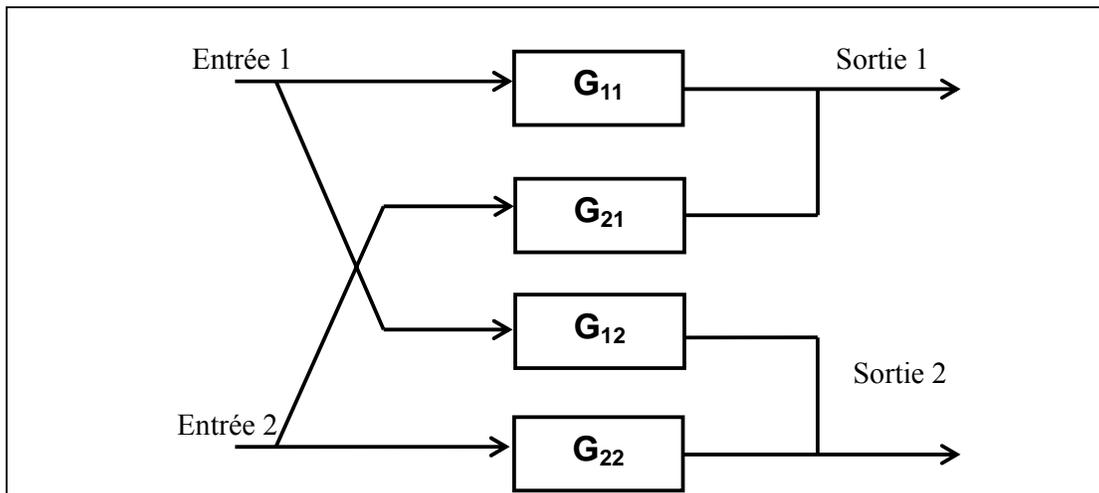


Figure I.14 Schéma bloc représentatif du robot manipulateur avec interaction entre deux axes.

$$G(s) = \begin{bmatrix} \frac{15.3217}{s^2 + 5.6654 \cdot s + 7.5873} & \frac{-1.5075}{(1 + 3.1663 \cdot s)(s^2 + 1.03 \cdot s + 41.4938)} e^{-0.085 \cdot s} \\ \frac{-0.1806}{s^2 + 1.1437 \cdot s + 29.5858} e^{-0.36 \cdot s} & \frac{2.0194}{s^2 + 2.9595 \cdot s + 16.2075} \end{bmatrix} \quad (I.2)$$

A partir de l'équation I.2, nous avons montré que la dynamique du robot peut être approchée par un modèle linéaire, mais avec des termes d'interaction relativement complexes. Afin d'examiner le degré de couplage entre les deux axes du robot, une étude par la méthode dite matrice des gains relatifs (Gain Array Methodology RGA en Anglais) est l'objet de la section suivante.

I.8.4.3 Analyse du Couple par la Méthode RGA pour le Bras du Robot du CDTA

La méthode d'étude d'interactions appelée RGA a été proposée par Bristol en 1966 [19][70]. Cette méthode est un outil d'aide à la prise de décision pour le problème de couplage d'entrées / sorties dans les systèmes multi-variables. L'examen de l'RGA permet d'isoler les sous systèmes et par conséquent d'avoir la possibilité de synthèse et d'implémentation des commandes de même nature ou de nature déférentes pour les sous systèmes du système global et d'une manière décentralisée. Malgré son ancienneté, la méthode RGA est restée très populaire est fréquemment utilisée. Ceci s'explique par les trois caractéristiques suivantes : facilité de mise en œuvre, richesse des informations produites et interprétation physique facile.

Le principe de la technique RGA repose sur le calcul de la matrice des gains statiques du système en boucle ouverte. Mathématiquement, l'équation générale de la méthode RGA est décrite de la façon suivante [19] :

$$\Lambda = G(0) \cdot (G(0)^T)^{-1} \quad (I.3)$$

Où:

$$\Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1n} \\ \lambda_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \lambda_{n1} & \dots & \dots & \lambda_{nn} \end{bmatrix} \quad (I.4)$$

Où G est une matrice de fonctions de transfert de dimension $n \times n$.

Après le calcul de la matrice donnée par l'équation I.3, et en fonction du résultat obtenu, quelques

règles de base d'arguments peuvent être énoncées [70]. Dans le cas des systèmes MIMO de dimension 2X2, nous pouvons figurer six cas :

1/ $\lambda_{ij} \approx 1$: Il n'y a aucune interaction entre la boucle de régulation de couple (y_i, u_j) et les autres boucles de régulation.

2/ $\lambda_{ij} = 0$: L'entrée j n'a aucun effet sur la sortie i .

3/ $0.5 < \lambda_{ij} < 1$: Il y a de l'interaction entre les boucles de régulation. Cependant, ce serait la sélection préférable car elle réduirait au minimum les interactions. Pour un système (2x2), si $0.5 < \lambda_{11} < 1$, la sortie y_1 doit être commandée par u_1 et si $0.5 < \lambda_{22} < 1$, la sortie y_2 doit être commandée par u_2 .

4/ $\lambda_{ij} = 0.5$: Il y a un niveau important d'interaction. Les autres boucles de régulation ont le même effet de l'entrée j sur la sortie i .

5/ $\lambda_{ij} > 1$: L'interaction est forte, donc on doit l'affaiblir. Cependant, ce serait la sélection préférable dans la configuration de commande.

6/ $\lambda_{ij} < 0$: Il y a de fortes interactions, la réponse de la boucle correspondante peut changer de sens de variation (système à réponse inverse) si les autres boucles sont fermées. De plus, la boucle elle-même peut être instable ou le système global devient instable si jamais la boucle considérée s'ouvre, d'où le couple correspondant ne doit pas être choisi dans la configuration de commande.

Le choix de la configuration de commande porte sur les couples ayant un gain relatif λ_{ij} proche de 1.

Après application de la RGA sur le modèle identifié du robot (équation (I.2)), nous avons abouti à la matrice de gains suivante :

$$\Lambda = \begin{bmatrix} 1.0145 & -0.0145 \\ -0.0145 & 1.0145 \end{bmatrix} \quad (I.5)$$

Le résultat de l'application de la méthode RGA sur le modèle du robot est donné par l'équation (I.5). D'après cette équation, nous pouvons observer que les gains de couplage sont proches de zéro, il est donc possible de considérer le robot comme un ensemble de sous-systèmes mono-entrée

mono-sortie (SISO). Ainsi, l'équation (I.2) devient :

$$G(s) = \begin{bmatrix} \frac{15.3217}{s^2 + 5.6654s + 7.5873} & 0 \\ 0 & \frac{2.0194}{s^2 + 2.9595s + 16.2075} \end{bmatrix} \quad (I.6)$$

Dans le domaine temporel, le système (I.6) peut être réécrit par le système d'équations différentielles (I.7) suivant :

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} a_{11} & 0 \\ 0 & a_{12} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} a_{01} & 0 \\ 0 & a_{02} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} b_1 & 0 \\ 0 & b_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (I.7)$$

Afin de simplifier sa forme, l'équation (I.7) peut être réécrite comme suit :

$$\ddot{Q} + A_1 \cdot \dot{Q} + A_0 \cdot Q = BU \quad (I.8)$$

Où : $Q = [q_1, q_2]^T$ est le vecteur de positions angulaires ;

$\dot{Q} = [\dot{q}_1, \dot{q}_2]^T$ est le vecteur de vitesses angulaires ;

$\ddot{Q} = [\ddot{q}_1, \ddot{q}_2]^T$ est le vecteur d'accélération angulaires ;

$U = [u_1, u_2]^T$ est le vecteur de commande.

Les paramètres identifiés des équations différentielles sont donnés comme suit :

$$\begin{aligned} A_0 &= \text{diag}(a_{01}, a_{02}) = \text{diag}(7.8573, 16.2075), \\ A_1 &= \text{diag}(a_{11}, a_{12}) = \text{diag}(5.6654, 2.9595) \\ B &= \text{diag}(b_1, b_2) = \text{diag}(15.3217, 2.0194) \end{aligned} \quad (I.9)$$

I.8.5 Problèmes Pratiques Liés au Robot du CDTA

Pour le robot à MAPs du CDTA, il existe plusieurs autres difficultés. A titre d'exemple, la contraction des muscles est limitée, ce qui réduit l'espace de travail du robot. Les muscles ne sont

pas parfaitement identiques. Dans ce système, il existe de légères fuites au niveau des conduites d'air comprimé, au niveau des distributeurs de pression et au niveau de la vanne. La pression injectée aux MAPs est supposée égale, chose qui n'est pas réalisable, car il existe toujours des légères différences de pression à l'intérieur de ces muscles. Les caractéristiques des MAPs changent avec l'augmentation du nombre d'expériences réalisées. A titre d'exemple, et afin d'analyser l'exactitude de ce robot, nous injectons quatre échelons de valeurs (20 mBar, 40 mBar, 60 mBar et 80 mBar) aux axes du robot. Chacun de ces échelons est appliqué plusieurs fois. Les réponses en position angulaire sont données dans la figure (I.15).

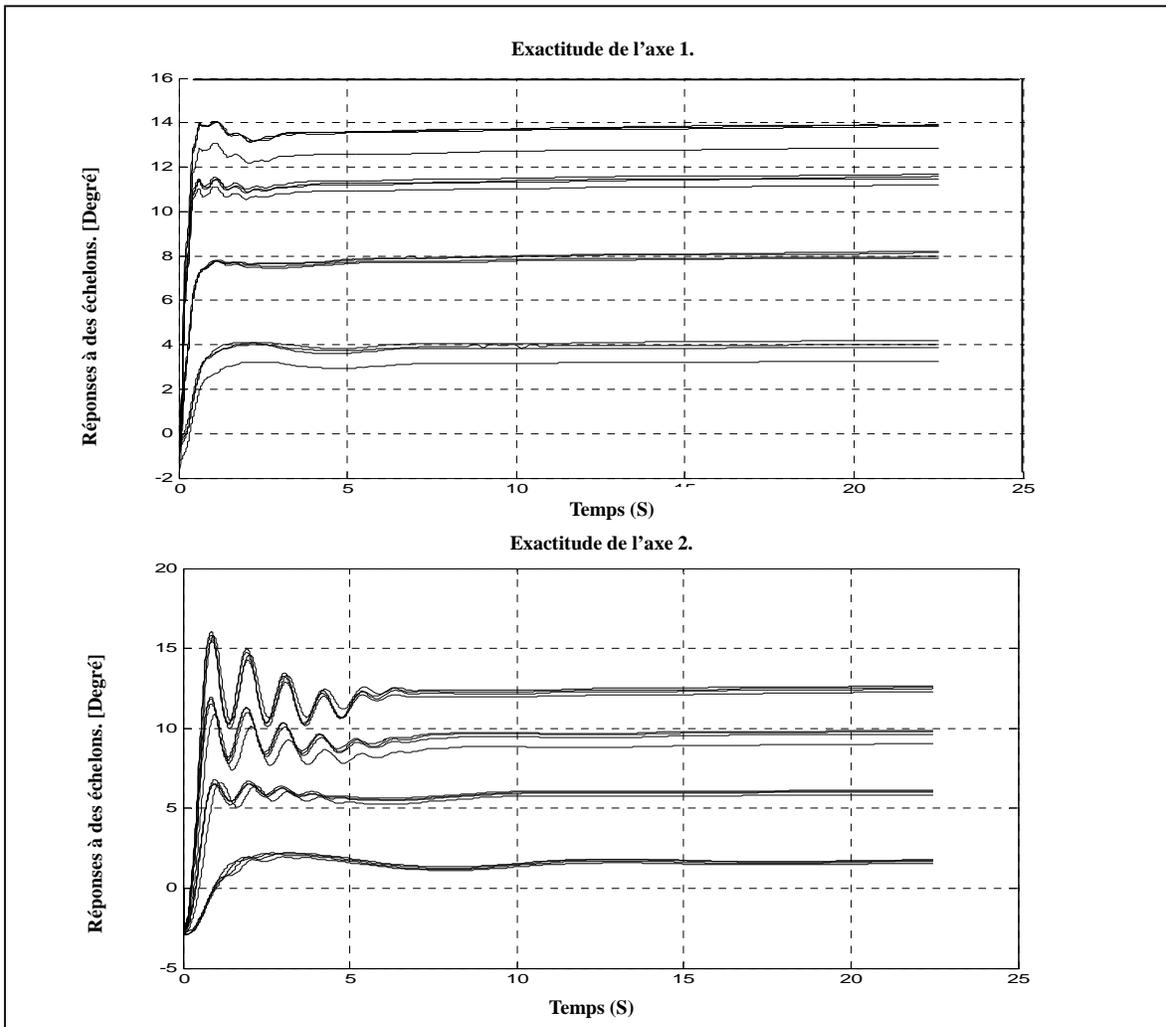


Figure I.15 Réponses des axes du robot en boucle ouverte par rapport à des échelons de différentes valeurs.

A partir de ces résultats, nous pouvons constater que le robot ne donne pas les mêmes réponses pour des entrées successives identiques pour chaque axe. Ce problème est dû aux variations des paramètres des actionneurs (MAP). Ces phénomènes modifient le comportement du système, c'est à

dire que les paramètres du système ne sont pas exactement connus et que des erreurs de modélisation et des incertitudes des paramètres du modèle peuvent apparaître. Tous les problèmes évoqués nous ont conduits à adapter la classe de commandes robustes.

I.9 DESCRIPTION DU ROBOT MANIPULATEUR A MAPS DE L'INSA

Présenté dans la figure I.16, le robot de l'INSA est un bras manipulateur à 7-*ddl*. Il est actionné entièrement par des MAPs fabriqués au niveau du laboratoire robotique de l'INSA en collaboration avec le laboratoire de mécanique du même établissement. Le principe de fonctionnement des deux robots (du CDTA et de L'INSA) est le même. Le robot pèse 6 kg, les axes sont en entraînement direct. Chaque articulation dispose d'un capteur de position angulaire analogique. Dans ce qui suit, nous allons présenter les éléments essentiels qui caractérisent la structure matérielle et logicielle de ce robot tels que la description de la structure mécanique du robot ainsi que les capteurs utilisés, l'architecture matérielle autour du robot permettant son fonctionnement et enfin l'architecture logicielle (système informatique).

I.9.1 Les Servovalves : Le modulateur de puissance est une servovalve permettant la conversion de l'intensité en une pression. Le robot de l'INSA se compose de 14 servovalves ; deux pour chaque axe. Ces servovalves sont commandées à l'aide de cartes de commande et de puissance intégrées. Elles délivrent un débit massique en fonction de la commande et des rapports de pression [10].

I.9.2 Cartes d'Entrée et de Sortie : il existe dans ce système deux types de cartes telles que :

Carte d'entrée : La carte CIODAS6402/12 est une carte analogique à 64 canaux unipolaires, elle est utilisée pour l'acquisition des données issues des capteurs de position situés au niveau des articulations du robot.

Carte de sortie : La carte CIODAC16/16 est une carte analogique avec 16 canaux. Cette carte est employée pour envoyer des signaux de commande aux actionneurs du robot.

I.9.3 Capteur de Position : Les capteurs de position que possède le robot sont des potentiomètres situés sur chacun des axes du robot. Ils sont fabriqués par la société BOURNS [10].

I.9.4 Le Système Informatique : L'architecture logicielle comprend deux ordinateurs dans une configuration maître/esclave. L'ordinateur esclave fonctionne sous un environnement temps réel V_XWorks. L'ordinateur maître fonctionne sous un environnement Windows NT. Un IDE Tornado est

utilisé pour le développement des lois de commande. La figure I.18 présente un schéma général de l'architecture matérielle.

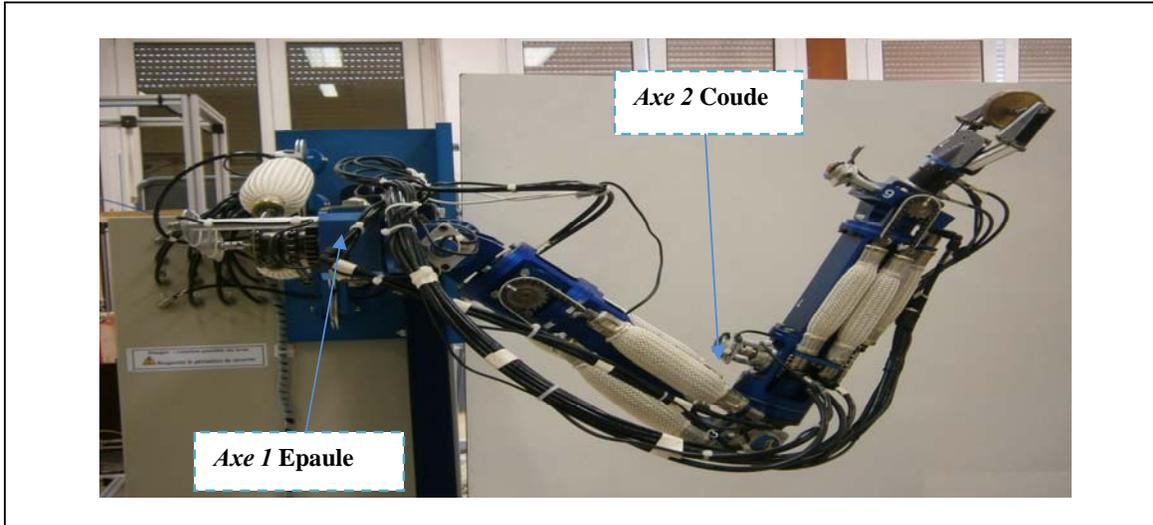


Figure I.16 Bras manipulateur anthropomorphe de l'INSA.

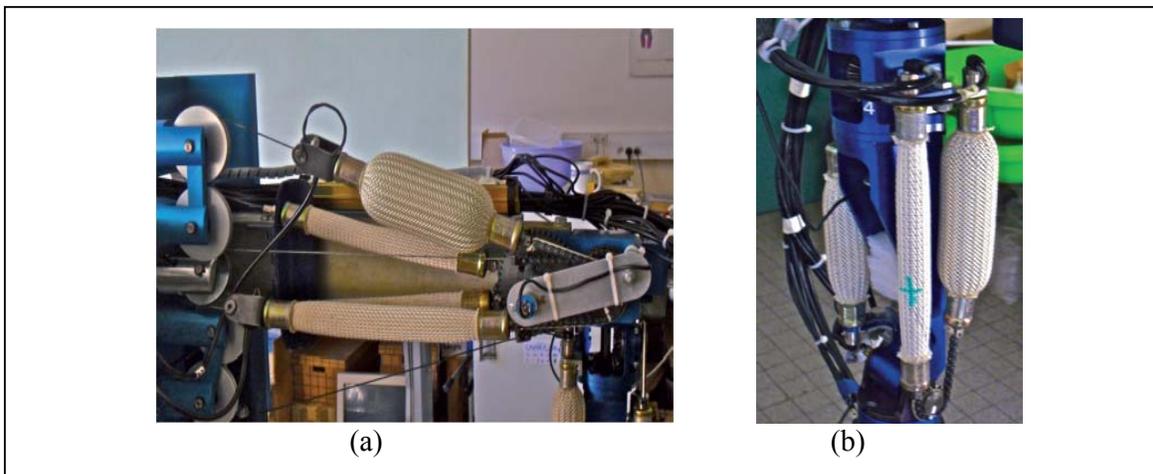


Figure I.17 Position initiale du robot (a) épaule (b) coude.

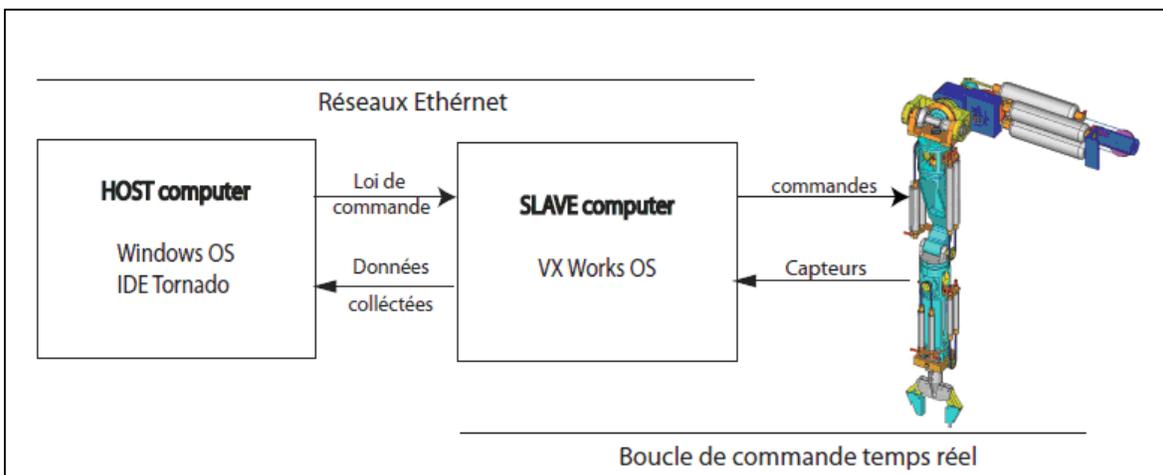


Figure I.18 Architecture du robot de l'INSA.

Par la suite, nous nous intéresserons uniquement aux articulations coude (axe 1) et épaule (axe 2). Il est utile de noter que les axes du robot sont stables en boucle ouverte. Le temps d'échantillonnage est fixé à 10 ms. Comme illustré dans la photo I.17, le volume maximal de la paire de muscles artificiels de l'axe 1 est plus important que celui de l'axe 2.

I.9.1 Problèmes Liés au Robot de L'INSA de Toulouse

Le système souffre de plusieurs sources de non-linéarités : premièrement, malgré son comportement mécanique proche de la biologie de l'être humain, le modèle de ce robot reste complexe et les travaux qui lui ont été consacrés n'ont abouti qu'à des modèles représentant partiellement les phénomènes physiques existants dans ce système. Deuxièmement, la durée de vie de ces muscles fabriqués au laboratoire est relativement courte, ce qui nécessite leur changement, ceci engendre naturellement un problème de détermination des nouveaux paramètres du modèle. Troisièmement, le retard dû aux temps de transmission des informations depuis et vers le robot. Quatrièmement, l'entraînement direct et l'action de la gravité rendent très difficile la commande de ce système. Enfin, la séparation physique entre les muscles de l'épaule n'est pas suffisante, ce qui provoque des frottements et par conséquent, des retards au niveau de la réponse de l'articulation.

I.9.2 Modélisation du Bras Anthropomorphe (Modèle de Simulation)

Dans cette section, nous étudions un cas simplifié du bras du robot anthropomorphe à 7-*ddl* de l'INSA. A cause des contraintes matérielles et logicielles ainsi que la complexité de la mise en œuvre d'une commande pour les sept axes simultanément, nous avons choisi de le considérer comme un robot 2-*ddl*. Malgré cette importante simplification, le système sous forme d'un robot à deux axes reste très complexe et représente un très bon exemple pour examiner les commandes robustes proposées dans ce travail de thèse, ceci notamment par des validations expérimentales. Un tel système dynamique schématisé par la Figure I.16 peut être régi par l'équation du modèle dynamique (I.10) qui est calculé par l'usage de l'équation de Lagrange (équation I.9) [63] :

$$\frac{d}{dt} \left(\frac{\partial L(q, \dot{q})}{\partial \dot{q}} \right) - \frac{\partial L(q, \dot{q})}{\partial q} = \Gamma(u) \quad (I.9)$$

Où q est le vecteur des coordonnées généralisées,

\dot{q} est le vecteur des vitesses généralisées,

$\Gamma(u)$ est le vecteur des forces généralisées.

$L(q, \dot{q})$ est la fonction de Lagrange qui présente la différence entre l'énergie cinématique et l'énergie potentielle du système.

L'application de l'équation (I.9) dans le cas d'un robot manipulateur à n -ddl en chaîne ouverte conduit à l'équation (I.10) [20] :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + T_d = \tau \quad (\text{I.10})$$

Avec :

$M(q) \in R^{n \times n}$ est la matrice d'inertie, qui est une matrice carrée de dimension $n \times n$ dépendant uniquement de la position du robot. Cette matrice est symétrique, définie positive, non singulière et bornée par: $m_{\min} \|x\|^2 \leq x^T Mx \leq m_{\max} \|x\|^2 \quad \forall x \in R^n$, avec m_{\max} et m_{\min} sont les valeurs propres maximales et minimales.

$C(q, \dot{q}) \in R^{n \times n}$ est le vecteur des forces centrifuges, de Coriolis et de friction.

$G(q) \in R^{n \times 1}$ est le vecteur de dimension $n \times 1$ qui présente les termes gravitationnels,

$q \in R^{nd}$, $\dot{q} \in R^{nd}$, $\ddot{q} \in R^{nd}$ sont les vecteurs de dimension $n \times 1$ représentant la position, la vitesse et l'accélération articulaire,

$\tau \in R^n$ est le vecteur de couple de force,

$T_d \in R^n$ présente les perturbations inconnues qui se caractérisent par leur borne supérieure telle que $\|T_d\| < \beta_d$, avec β_d est un vecteur positif.

Dans le cas du robot de l'INSA, à cause de la complexité de la dynamique de ce système, nous avons choisi de commander uniquement l'épaule et le coude. Dans ce cas, un modèle dynamique à 2-ddl est utilisé dont les matrices sont données comme suit :

$$\begin{aligned} M(q) &= \begin{bmatrix} (m_1 + m_2)l_1^2 + m_2l_2^2 + 2m_2l_1l_2 \cos q_2 & m_2l_2^2 + m_2l_1l_2 \cos q_2 \\ m_2l_2^2 + m_2l_1l_2 \cos q_2 & m_2l_2^2 \end{bmatrix} \\ C &= \begin{bmatrix} -m_2l_1l_2 \sin q_2 (2\dot{q}_1\dot{q}_2 + \dot{q}_2^2) \\ -m_2l_1l_2 \sin q_2 \dot{q}_1\dot{q}_2 \end{bmatrix} \\ G &= \begin{bmatrix} -(m_1 + m_2)gl_1 \sin q_1 - m_2gl_2 \sin(q_1 + q_2) \\ -m_2gl_2 \sin(q_1 + q_2) \end{bmatrix} \end{aligned} \quad (\text{I.11})$$

I.10 CONCLUSION

Bien que les MAPs présentent des qualités avérées et forment des moteurs potentiellement prometteurs, ils présentent néanmoins des non linéarités difficiles à surmonter; par conséquent, les commandes des robots dotés pour entraîner leurs articulations constituent une difficulté supplémentaire. Le MAP, par son comportement proche du muscle du squelette, possède un potentiel très important dans les robotiques de service et du médical. Dans ce chapitre, nous avons présenté en détail les deux robots utilisés dans le cadre de cette thèse ainsi que leurs modèles de simulation.

CHAPITRE II. *Etat De L'art De La Commande Par Mode De Glissement.*

II.1 INTRODUCTION

Le but de ce travail de thèse est de proposer une/ou plusieurs commandes qui permettent d'atteindre les meilleures performances possibles ainsi qu'une garantie absolue de la robustesse en boucle fermée de robots manipulateurs entièrement actionnés par des muscles artificiels pneumatiques. Ce chapitre est consacré à la présentation de la commande à structure variable par mode de glissement.

L'idée de base de la commande par modes glissants est de contraindre le système à se déplacer sur un ensemble prescrit appelé surface de glissement et ensuite d'y évoluer. Une telle approche présente deux principaux avantages. Tout d'abord, le régime glissant est insensible à la classe d'incertitudes pour laquelle la commande a été réglée. Ensuite, le comportement dynamique résultant, appelé régime glissant idéal, est complètement déterminé par le choix de cette surface [73]. La synthèse d'une loi de commande par modes glissants comprend deux phases :

- le choix de la surface de glissement : Il est défini en fonction des performances désirées en boucle fermée,
- La synthèse de la commande effectuée de façon à forcer les variables d'état du système à atteindre la surface de glissement, et ensuite, à y évoluer avec rejet des perturbations internes et/ou externe d'incertitudes.

II.2 LA COMMANDE A STRUCTURE VARIABLE

Par définition, la commande à structure variable est une commande dans laquelle les paramètres du régulateur peuvent prendre deux valeurs différentes. La commutation d'une valeur à l'autre se fait suivant le signe d'une fonction définie dans l'espace d'état du système correspondant à la surface de glissement, ceci dans le but d'obtenir de meilleures performances que celles obtenues par les régulateurs classiques. Ainsi, sous certaines conditions, la dynamique du système commandé peut être maintenue, sur la surface choisie dans l'espace des phases. D'autre part, quand le régime glissant sera atteint, la dynamique du système bouclé devient alors celle du système d'ordre réduit.

Le système bouclé est robuste vis à vis des perturbations extérieurs, des non linéarités et des variations des paramètres du système à commander. On peut dire que l'intérêt principal de la commande à structure variable se voit dans la simplicité de réaliser une commande de qualité pour des processus perturbés ou mal identifiés.

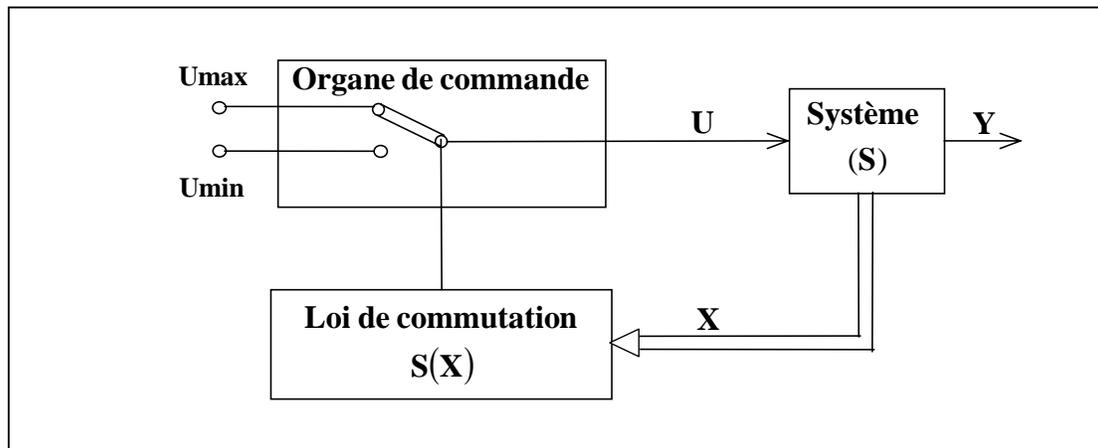


Figure II.1. Schéma du principe de la commande à structure variable [20].

II.3 FORMULATION CLASSIQUE DU REGIME GLISSANT

Dans la littérature, on distingue deux types de régime ou mode de glissement, idéal et réel.

II.3.1 Définition du régime glissant idéal

Le régime glissant idéal correspond à une oscillation de fréquence infinie et d'amplitude nulle, le point représentatif de l'évolution du système glisse parfaitement sur la surface de glissement S . Dans ce cas, la trajectoire en régime glissant est parfaitement décrite par l'équation $S(X) = 0$.

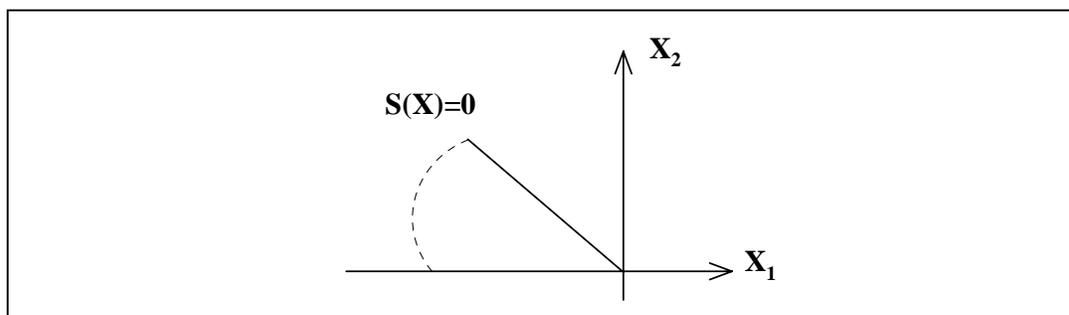


Figure II.2 Glissement idéal.

II.3.2 Définition du régime glissant réel

En pratique, l'organe de commutation est réalisé à partir de relais qui présentent des imperfections, comme les retards de commutation. La trajectoire de phase en régime glissant reste au voisinage de la surface de commutation, donnant naissance à des oscillations indésirables qui diminuent la précision du système, en gardant néanmoins la stabilité [75].

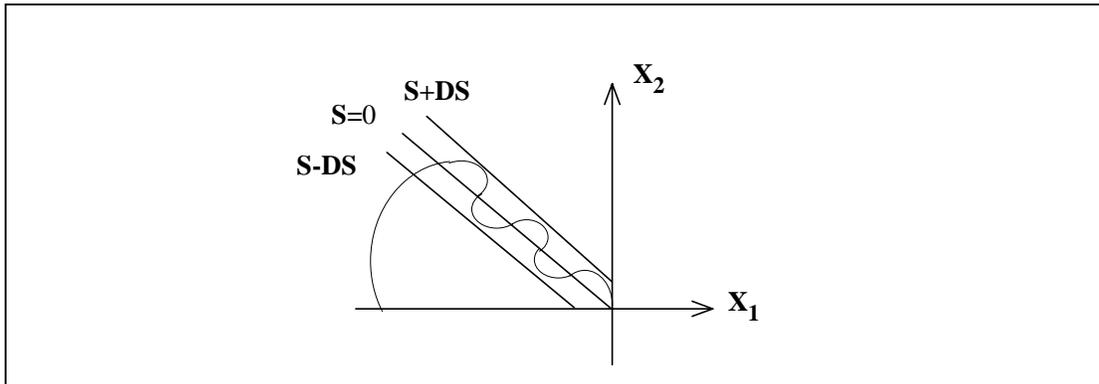


Figure II.3 Glissement réel.

II.3.3 Explication du glissement

Nous supposons qu'à un instant t_0 le point de fonctionnement du système coupe la surface de glissement avec la commande U_{\min} et dès qu'il atteint le point A , le seuil inférieur de basculement, la commande prend la valeur U_{\max} et de nouveau le point représentatif du système coupe la surface de glissement jusqu'à ce qu'il atteigne le point B ; le seuil supérieur de basculement. Une deuxième fois, la commande prend la valeur U_{\min} et le point de fonctionnement sera forcé à évoluer sur la surface de glissement et ainsi de suite. Donc, un mouvement continu à l'intérieur de la zone hystérésis du relais et lorsque le point représentatif atteint la zone de zéro, ce dernier continue à commuter sans s'éloigner du zéro (figure II.4) [75].

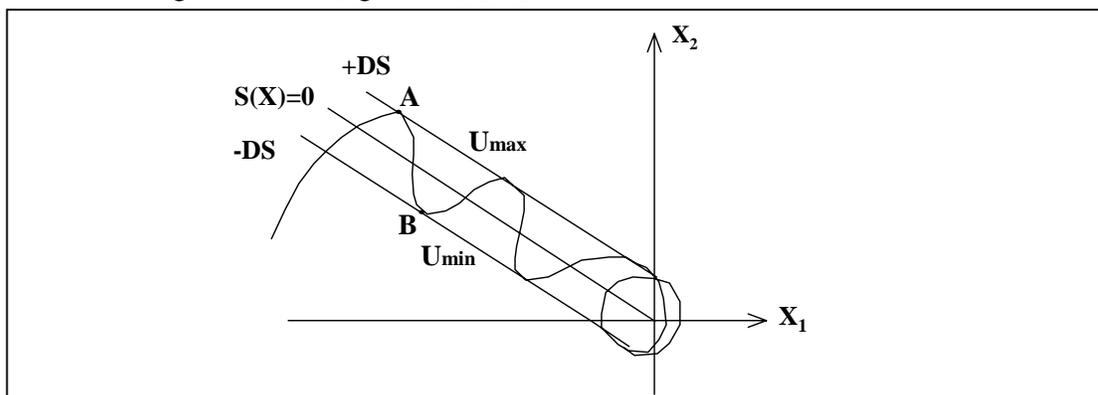


Figure II.4 Mode de glissement.

II.3.4 Condition d'existence du glissement [75]

Soit le système d'équations différentielles donnée par (II.1):

$$\frac{dX}{dt} = F(t, x_1, x_2, \dots, x_n) \quad (\text{II.1})$$

avec :

$$X = [x_1, x_2, \dots, x_n]^T, \quad F = [f_1, f_2, \dots, f_n]^T$$

Les fonctions $f_i(t, x_1, x_2, \dots, x_n)$ sont continues par morceaux et présentent des discontinuités sur l'hypersurface notée S d'équation : $S(X)=0$

Les fonctions f_i sont supposées définies dans un domaine G de l'espace d'état. La surface $S(X)=0$ sépare le domaine G en deux parties $G^+(S < 0)$ et $G^-(S > 0)$, ce qui nous donne au voisinage de la surface S deux valeurs de f soit f^+ et f^- . Nous appelons f_N^+ et f_N^- les projections respectives de f^+ et f^- sur la normale à la surface S , orienté de G^- vers G^+ .

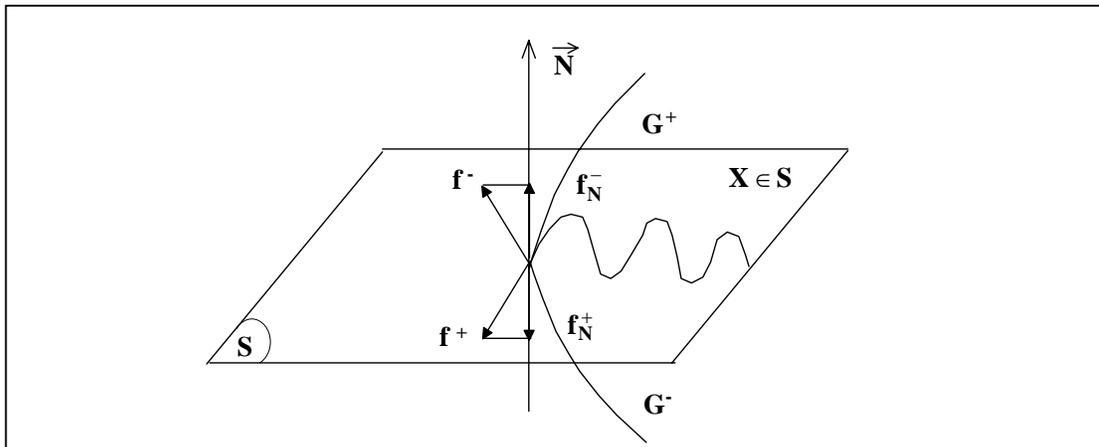


Figure II.5 Espace d'état de dimension 3 et de surface de glissement de dimension 2.

Théorème de Philipov

Le théorème suivant permet de préciser les conditions d'existence et d'unicité d'une solution $X(t)$ de l'expression (II.1) [74].

Considérons le système décrit par l'équation (II.1) satisfaisant la condition suivante:

$$\left| \frac{\partial f_i}{\partial x_j} \right| \leq K \quad \text{avec : } (i, j=1, 2, \dots, n)$$

K étant une constante arbitraire indépendante de X et de t , ces inégalités étant vérifiées pour tout X et t dans les domaines G^+ et G^- . Soit une surface S deux fois différentiable, chacune des composantes f_N^+ et f_N^- est continue sur $[t, x_1, x_2, \dots, x_n]$ pour $[x_1, x_2, \dots, x_n]$ appartenant à S et le vecteur $h = f^+ - f^-$ étant continûment différentiable. Si en chaque point de la surface S une des inégalités $f_N^+ < 0$ ou $f_N^- > 0$ est vérifiée (pas obligatoirement la même en tout point de S), il existe alors, dans le domaine G , une solution $X(t)$ pour le système (II.1), et les propriétés d'unicité et de dépendance de cette solution par rapport aux conditions initiales sont vérifiées de façon continue.

Ce théorème conduit à dire que si l'on vérifie à la fois $f_N^- > 0$ et $f_N^+ < 0$, la surface S sera attractive pour les trajectoires de phase $X(t)$ du système (II.1) au moins dans un petit voisinage de S dans l'espace d'état puisque de chaque côté de la surface, les vecteurs de vitesse sont dirigés vers la surface. Il y a donc un régime glissant limite sur la surface S .

La surface S est définie par :

$$S(x_1, x_2, \dots, x_n) = 0 \quad (\text{II.2})$$

En dérivant (II.2) par rapport à t , on aura :

$$\dot{S} = \sum_{i=1}^n \frac{\partial S}{\partial x_i} \cdot \frac{dx_i}{dt} = \sum_{i=1}^n \frac{\partial S}{\partial x_i} \cdot f_i(t, x_1, x_2, \dots, x_n) \quad (\text{II.3})$$

La dérivée \dot{S} apparaît comme le produit scalaire de la normale orientée à la surface S et du vecteur F . Or si les conditions du théorème précédent sont vérifiées, on déduit :

$$f_N^+ < 0 \text{ correspond à } S > 0 \text{ et } \dot{S} < 0$$

$$f_N^- > 0 \text{ correspond à } S < 0 \text{ et } \dot{S} > 0$$

$$\text{d'où : } f_N^+ < 0 \text{ et } f_N^- > 0 \Rightarrow S \cdot \dot{S} < 0 \quad (\text{II.4})$$

Donc la condition de glissement sous les deux formes classiques ($f_N^+ < 0$ et $f_N^- > 0$) et ($S \cdot \dot{S} < 0$) sont équivalentes. La condition (II.4) est l'inégalité fondamentale utilisée pour résoudre le problème de la synthèse d'un correcteur à structure variable. Alors un régime glissant existe sur une surface de glissement si, et seulement si, dans un voisinage de la surface de glissement, toutes les trajectoires du système sont dirigées vers elle. En d'autres termes :

$$\lim_{S(x) \rightarrow 0^-} \dot{S}(x) > 0 \quad \text{et} \quad \lim_{S(x) \rightarrow 0^+} \dot{S}(x) < 0 \quad (\text{II.5})$$

II.3.5 Choix des surfaces de glissement

Les surfaces de glissement sont choisies en général comme étant des hyperplans passant par l'origine de l'espace pour des raisons évidentes de stabilisation et de manière à définir une certaine évolution du système lors du régime glissant [20].

Les surfaces $S(X)$ sont données par :

$$S(X) = \sum_{i=1}^n c_i \cdot e_i = e_n + \sum_{i=1}^{n-1} c_i \cdot e_i \quad (\text{II.6})$$

avec : ($c_n = 1$)

on pose : $e_i = X_i - X_{id}$

avec :

X_d : état désiré.

c_i : coefficients de surface.

D'une façon générale, la stabilisation d'un système sur l'origine, par l'utilisation de régimes glissants, s'effectue en deux étapes distinctes :

- * Commander le système de façon à l'amener dans le voisinage de la surface de glissement à partir d'une position initiale arbitraire.

* Engendrer une commande permettant de faire évoluer le système en régime glissant sur cette surface. Le système sera alors plongé dans l'état d'un système réduit, déterminé par la surface de glissement (Etat oscillatoire).

II.4 COMMANDE A STRUCTURE VARIABLE EN REGIME GLISSANT [75]

Nous donnons ci-dessous deux expressions équivalentes pour la détermination de la trajectoire d'état $X(t)$, solution du système en mode de glissement.

II.4.1 Equation d'état en mode de glissement exprimée à partir du gradient et du produit scalaire :

Soit le système différentiel non linéaire dans lequel la commande U est indépendante :

$$\frac{dx}{dt} = f(U, x, t) \quad (\text{II.7})$$

Considérons une commande non linéaire discontinue :

$$U = \begin{cases} U^+(x, t) & \text{si } S(x, t) > 0 \\ U^-(x, t) & \text{si } S(x, t) < 0 \end{cases} \quad (\text{II.8})$$

Supposons que la partie droite de l'équation (II.7) présente une discontinuité à la traversée de la surface $S(x, t)$ (figure II.7).

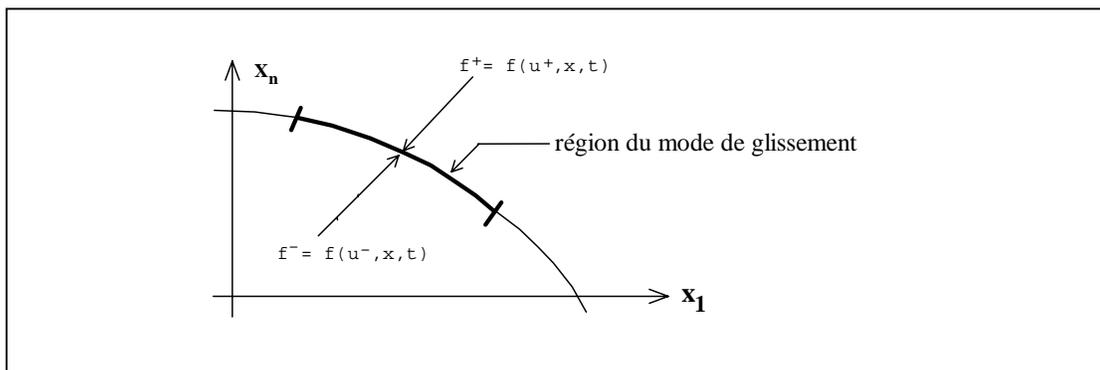


Figure II.6 Schéma de principe montrant la région du mode de glissement et les champs de vecteurs associés.

Soit une surface de glissement $S(x, t) = 0$ et une loi de commande donnée par (II.8), la trajectoire d'état $x(t)$ du système (II.7) est solution de l'équation suivante :

$$\frac{dX}{dt} = \mu \cdot f^+ + (1-\mu) \cdot f^- = f^0 \quad (\text{II.9})$$

avec : $0 \leq \mu \leq 1$ et $f^0 = [f_1^0, f_2^0, \dots, f_n^0]^T$

f^0 est appelée la vitesse d'état moyenne (c'est un champ de vecteur moyen) sur la trajectoire d'état en mode de glissement idéal. Ce champ de vecteur moyen est tangentiel à la surface de glissement, il est obtenu par la moyenne géométrique, ou par une combinaison connexe des champs de vecteurs de chaque côté de S .

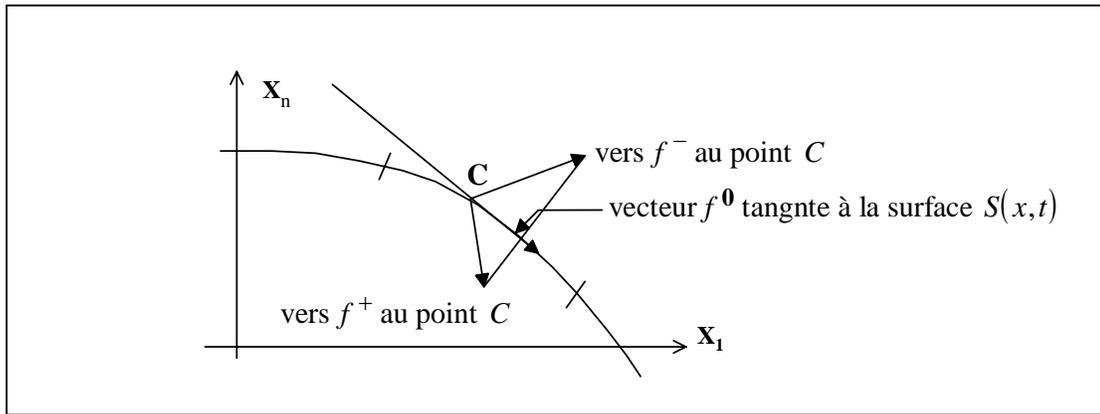


Figure II.7 Schéma du principe de la construction du champ de vecteur moyen f^0 dans une coupe de l'espace d'état.

Soit ∇S le gradient de $S(x,t)$ et $\langle \cdot, \cdot \rangle$ le produit scalaire de deux vecteurs. En dérivant $S(x,t) = 0$ par rapport au temps t , on aura :

$$\frac{dS(x,t)}{dt} = \left(\frac{\partial S}{\partial x} \right)^T \cdot f^0 + \frac{\partial S}{\partial t} = \langle \nabla S, f^0 \rangle + \frac{\partial S}{\partial t} = 0 \quad (\text{II.10})$$

De (II.9) et (II.10), on peut tirer :

$$\mu = \frac{\langle \nabla S, f^- \rangle + \frac{\partial S}{\partial t}}{\langle \nabla S, f^- - f^+ \rangle} \quad (\text{II.11})$$

En portant l'expression de μ dans (II.9), l'équation d'état qui décrit le système bouclé en régime

glissant est :

$$\frac{dx}{dt} = \left[\frac{\langle \nabla S, f^- \rangle + \frac{\partial S}{\partial t}}{\langle \nabla S, f^- - f^+ \rangle} \right] \cdot f^+ - \frac{\langle \nabla S, f^+ \rangle + \frac{\partial S}{\partial t}}{\langle \nabla S, f^- - f^+ \rangle} \cdot f^- = f^0 \quad (\text{II.12})$$

Cette équation détermine la trajectoire d'état, solution du système en mode glissement.

II.4.2 Commande équivalente

Soit le système mono-variable décrit par l'équation d'état (II.13) :

$$\frac{dx}{dt} = f(x, t) + g(x, t) \cdot U \quad (\text{II.13})$$

où x , f et g sont des vecteurs colonnes de dimension n .

La commande équivalente U_{eq} correspond au régime glissant idéal (ni seuil, ni retard, ni hystérésis). Ce glissement s'effectue sur la surface d'équation $S(x, t) = 0$ avec la condition $\dot{S}(x, t) = 0$.

Posons $f = [f_1, f_2, \dots, f_n]$ et $g = [g_1, g_2, \dots, g_n]$ et exprimons $\dot{S}(x, t)$, pour obtenir U_{eq} :

$$\frac{dS(x, t)}{dt} = \left(\frac{\partial S}{\partial x} \right)^T \cdot \frac{dx}{dt} + \frac{\partial S}{\partial t} = \left(\frac{\partial S}{\partial x} \right)^T \cdot [f(x, t) + g(x, t) \cdot U_{eq}] + \frac{\partial S}{\partial t} = 0$$

Si on suppose que la condition d'existence :

$$\left[\left(\frac{\partial S}{\partial x} \right)^T \cdot g(x, t) \right]^{-1} \neq 0 \quad (\text{II.14})$$

La commande équivalente peut être donnée par :

$$U_{eq}(x,t) = - \left[\left(\frac{\partial S}{\partial x} \right)^T \cdot g(x,t) \right]^{-1} \cdot \left\{ \left(\frac{\partial S}{\partial x} \right)^T \cdot f(x,t) + \frac{\partial S}{\partial t} \right\} \quad (\text{II.15})$$

En portant l'expression de $U_{eq}(x,t)$ dans l'équation (II.13), la trajectoire d'état $x(t)$ décrivant le système en régime glissant est solution de :

$$\frac{dx}{dt} = \left\{ I - g(x,t) \cdot \left[\left(\frac{\partial S}{\partial x} \right)^T \cdot g(x,t) \right]^{-1} \left(\frac{\partial S}{\partial x} \right)^T \right\} \cdot f(x,t) - g(x,t) \cdot \left[\left(\frac{\partial S}{\partial x} \right)^T \cdot g(x,t) \right]^{-1} \left(\frac{\partial S}{\partial t} \right)^T \quad (\text{II.16})$$

Les équations (II.12) et (II.16) sont équivalentes.

La commande équivalente U_{eq} a un sens physique, elle représente la valeur moyenne que prend la grandeur de commande lors de la commutation rapide entre U_{\max} et U_{\min} qui maintient l'état du système, sur la surface de glissement $S(x,t)=0$, défini comme suit :

$$U_{\min} = \text{Min} \{ U^+(x,t), U^-(x,t) \}$$

$$U_{\max} = \text{Max} \{ U^+(x,t), U^-(x,t) \}$$

$$U_{\min} < U_{eq}(x,t) < U_{\max}$$

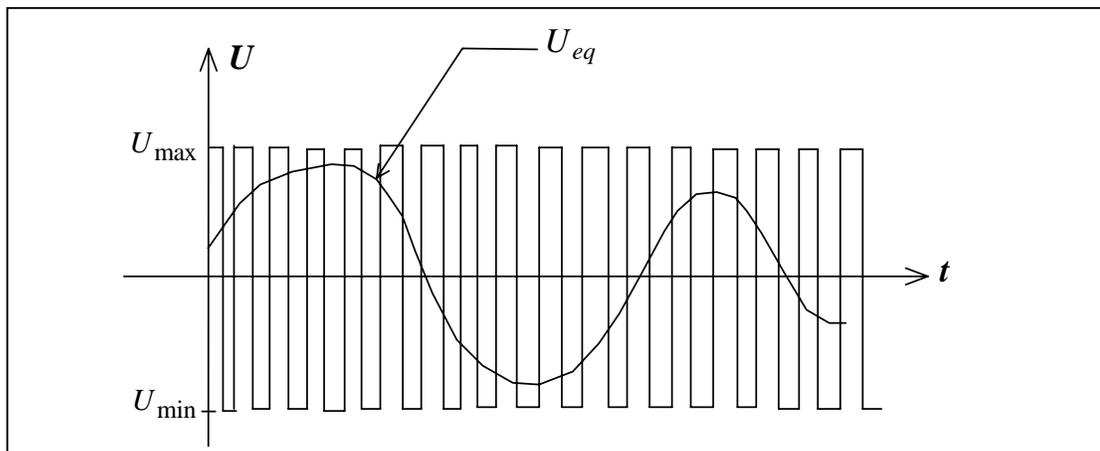


Figure II.8 Caractéristique dynamique de la grandeur de commande équivalente

II.5 EFFET DE BROUITEMENT

Le caractère discontinu de cette commande rend cette dernière robuste par rapport à certains types d'incertitude [69]. Néanmoins, cette même discontinuité est à l'origine du phénomène du *chattering*. Bien que l'utilisation, d'une part des fonctions dites sigmoïdes (saturation, arctangente, tangente hyperbolique, pseudo-signe) [3] et, d'autre part, de la commande à structure variable généralisée [52] permettent de réduire ce phénomène ; la précision de convergence et la robustesse s'en trouvent alors dépréciées.

II.6 COMMANDE PAR TERMINAL SLIDING MODE

Dans ces dernières années, une importante attention a été prêtée à la commande à structure variable par le Terminal Sliding Mode (TSM) et de nombreux travaux sur cette technique ont vu le jour [21-22]. Dans le cas du TSM, la surface de glissement choisie est de forme non linéaire et spécifique, de telle sorte que le système converge vers l'origine dans un temps fini. Autrement dit, la commande TSM force le système à évoluer de manière à assurer la convergence en temps fini. Comparé avec le cas classique qui utilise une surface linéaire, cette technique présente plusieurs avantages tels qu'une vitesse de convergence rapide et en temps fini, sa haute précision en raison de son accélération au moment du rapprochement au point d'équilibre, et enfin la possibilité de son usage pour les systèmes de degré relatif supérieur à 1 [21-22]. Le TSM est un concept qui a été introduit en 1994 par Man *et al.* [41] pour le contrôle d'un robot manipulateur. Le principe de base de la commande par TSM peut être brièvement défini comme suit:

Soit le système de second ordre non linéaire mono variable incertain suivant :

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= f(x) + g(x) + b(x) \cdot u,\end{aligned}\tag{II.17}$$

Où $x = [x_1, x_2]^T$ est le vecteur d'état du système, $f(x)$ et $b(x) \neq 0$ sont des fonctions non linéaires de x , $g(x)$ représente les incertitudes et les perturbations non mesurables, elle est bornée par $\|g(x)\| \leq I_g$ avec $I_g > 0$.

II.6.1. Synthèse de la Commande par Terminal Sliding Mode: A présent, la tâche principale est de synthétiser une commande u pour atteindre la consigne désirée dans un temps fini, afin de stabiliser x_1 et x_2 à zéro. Comme dans le cas de la commande par mode de glissement classique, le TSM nécessite le choix d'une surface de glissement. Dans ce cas, cette dernière est donnée par :

$$S = x_2 + k_e x_1^{q/p} \quad (\text{II.18})$$

Où $k_e > 0$. est une constante réelle positive, p et q sont des constantes entières définies positives avec $p > q$. Pour le système (II.1), la commande par TSM est synthétisée lorsque \dot{S} égale à zéro comme suit :

$$u = -b^{-1}(x) \cdot \left(f(x) + k_e \frac{q}{p} x_1^{((q/p)-1)} x_2 - \eta \operatorname{sgn}(S) \right) \quad (\text{II.19})$$

Où: $\eta > 0$.

Le temps de convergence du système au point d'équilibre est calculé en utilisant l'équation (II.20) :

$$t_s = -k_e \int_{x_1(t_r)}^0 \frac{dx_1}{x_1^{q/p}} = \frac{p}{k_e(p-q)} |x_1(t_r)|^{1-(q/p)} \quad (\text{II.20})$$

Par la prise en considération de la dérivée de s , la condition suffisante pour l'existence de TSM est :

$$S \cdot \dot{S} \leq -\eta \cdot |S| \quad (\text{II.21})$$

II.6.2 Le problème de singularité

La loi de commande (II.19) souffre d'un problème connu dans la littérature sous le nom de singularité. Afin de maîtriser ce problème qui consiste en l'existence d'une division par zéro quand $x_1 = 0$, la surface de glissement (II.19) a été modifiée pour être de la forme (II.22) et dénommée *surface de glissement terminal non singulière* [41]:

$$S = x_1 + \beta \cdot x_2^{p/q} \quad (\text{II.22})$$

Où: $\beta = 1/k_e$.

Pour le système (II.17), avec l'utilisation de la surface de glissement non singulière (NTSM) donnée par l'équation (II.22), la loi de commande (II.19) se transforme comme suit:

$$u = -b^{-1}(x) \cdot \left(f(x) + \beta \frac{q}{p} x_2^{2-p/q} - \eta \operatorname{sgn}(S) \right) \quad (\text{II.23})$$

Où $1 < p/q < 2, \eta > 0$. Dans ce cas, la surface de glissement (II.3) sera recherchée dans un temps fini et le problème de singularité ne se pose plus.

Où $x = [x_1, x_2]^T$ est le vecteur d'état du système, $f(x)$ et $b(x) \neq 0$ sont des fonctions non linéaires de x , $g(x)$ représente les incertitudes et les perturbations non mesurables d'une fonction bornée par $\|g(x)\| \leq I_g$ avec $I_g > 0$, u étant la commande.

II.7 MODES GLISSANTS D'ORDRE SUPERIEUR (HOSM)

Par la généralisation du régime glissant classique, en prenant en compte les dérivées successives de la fonction de glissement (la commande discontinue agit sur la plus haute dérivée de la fonction de commutation au lieu de sa première dérivée seulement comme dans le cas classique), les modes ou régimes glissants d'ordre supérieur (HOSM pour High Order Sliding Modes) permettent comme dans le cas classique de maintenir la propriété de robustesse, de réduire ou d'éliminer le phénomène de broutement indésirable et de lever la restriction sur le degré relatif (qui doit être unitaire dans le cas classique). Ces régimes glissants (HOSM) permettent aussi d'obtenir une meilleure précision de convergence en temps fini. Dans le cadre de ce travail, deux algorithmes de commande par régimes glissants d'ordre deux sont implémentés, le « Twisting » algorithme [78] et le « Super twisting » algorithme [76] dont l'avantage est qu'il ne requiert pas la connaissance de la dérivée de la fonction de glissement par rapport au temps $\dot{\sigma}$.

II.7.2 Algorithmes glissants d'ordre 2

L'objectif de ces algorithmes est de générer un régime glissant d'ordre 2 sur la fonction contrainte ou surface de glissement $\sigma = 0$ et d'aboutir en temps fini, à satisfaire $\sigma = \dot{\sigma} = 0$.

II.7.2.1 Commande Twisting

La convergence en temps fini vers l'origine du plan de phase $(\sigma, \dot{\sigma})$ est obtenue via la commutation de la commande entre deux valeurs qui assure qu'à chaque commutation, l'abscisse et l'ordonnée soient croisées de plus en plus près de l'origine.

- **Glissement idéal**

Selon le degré relatif par rapport à la fonction de glissement σ , la loi de commande est :

$$\begin{aligned} \text{Degré relatif}=1 \quad & \left(\frac{\partial \dot{\sigma}}{\partial u} \neq 0 \right) \\ v = \dot{u} = & \begin{cases} -u & \text{si } |u| > u_M \\ -a_m \operatorname{sgn}(y_1) & \text{si } y_1 y_2 \leq 0, |u| \leq u_M \\ -a_M \operatorname{sgn}(y_1) & \text{si } y_1 y_2 > 0, |u| \leq u_M \end{cases} \end{aligned} \quad (\text{II.24})$$

$$\begin{aligned} \text{Degré relatif}=2 \quad & \left(\frac{\partial \dot{\sigma}}{\partial u} = 0, \frac{\partial \ddot{\sigma}}{\partial u} \neq 0 \right) \\ v = u = & \begin{cases} -a_m \operatorname{sgn}(y_1) & \text{si } y_1 y_2 \leq 0 \\ -a_M \operatorname{sgn}(y_1) & \text{si } y_1 y_2 > 0 \end{cases} \end{aligned} \quad (\text{II.25})$$

u_M est une constante positive représentant la valeur maximale de la commande (imposée par des raisons pratiques).

Les conditions de suffisance sur les gains a_m , a_M pour assurer la convergence en temps fini vers l'origine du plan de phase $(\sigma, \dot{\sigma})$ sont :

$$\begin{cases} a_M > 4 \frac{K_m}{s_0} \\ a_m > \frac{\Psi}{K_m} \\ K_m a_M - \Psi > K_M a_m + \Psi \end{cases} \quad (\text{II.26})$$

Grâce à ces hypothèses, il peut être montré que les trajectoires du système (3.59), dans le plan de

phase $(\sigma, \dot{\sigma})$ restent inscrites à l'intérieur de deux trajectoires limites définies par les bornes des fonctions χ et ζ et convergent toutes deux vers l'origine en temps fini.

La borne supérieure du temps de convergence est établie, quel soit le quadrant du plan de phase (y_1, y_2) où se situent les conditions initiales, comme :

$$T_{\max tw} \leq t_{\text{int}} + \Theta_{tw} \frac{1}{1 - \theta_{tw}} \sqrt{|y_{\text{int}}|} \quad (\text{II.27})$$

y_{int} est la valeur de y_1 la première fois que l'axe des abscisses dans le plan (y_1, y_2) est traversé, t_{int} étant l'instant correspondant.

$$\Theta_{tw} = \sqrt{2} \frac{K_m a_M + K_M a_m}{(K_m a_M - \Psi) \sqrt{K_M a_m + \Psi}}$$

$$\text{et } \theta_{tw} = \sqrt{\frac{(a_m K_M + \Psi)}{(a_M K_m - \Psi)}} \quad (\text{II.28})$$

- **Glissement réel**

Cette forme ne nécessite pas la connaissance de $\dot{\sigma}$ et prend en compte la période d'échantillonnage τ supposée la même pour la mesure et la commande.

$\dot{\sigma}$ est en fait remplacée par $\Delta\sigma$.

$$\Delta\sigma = \begin{cases} 0 & k = 0 \\ (\sigma(k\tau) - \sigma((k-1)\tau)) & k \geq 1 \end{cases} \quad (\text{II.29})$$

Ce qui permet la réécriture de l'algorithme (3.61) sous la forme suivante,

$$v = u = \begin{cases} -a_m \operatorname{sgn}(\sigma) & \text{si } \sigma\Delta\sigma \leq 0 \\ -a_M \operatorname{sgn}(\sigma) & \text{si } \sigma\Delta\sigma > 0 \end{cases} \quad (\text{II.30})$$

Sous l'hypothèse (II.26) et la condition (II.27), la loi de commande (II.30) constitue un algorithme glissant réel d'ordre 2. Après établissement du régime glissant réel, les trajectoires d'état sont telles que :

$$|\sigma| = O(\tau^2) \text{ et } |\dot{\sigma}| = O(\tau) . \quad (\text{II.31})$$

Floquet *et al* [62] proposent une réduction du temps de convergence vers l'ensemble

($\sigma = \dot{\sigma} = 0$) pour le système (3.59), pour l'algorithme du Twisting comme :

$$u = -\alpha^2 y_1 - 2\alpha y_2 - \begin{cases} a_m \operatorname{sgn}(y_1) & \text{si } y_1 y_2 \leq 0 \\ a_M \operatorname{sgn}(y_1) & \text{si } y_1 y_2 > 0 \end{cases} \quad (\text{II.32})$$

$\alpha > 0$ permet de régler la vitesse de convergence.

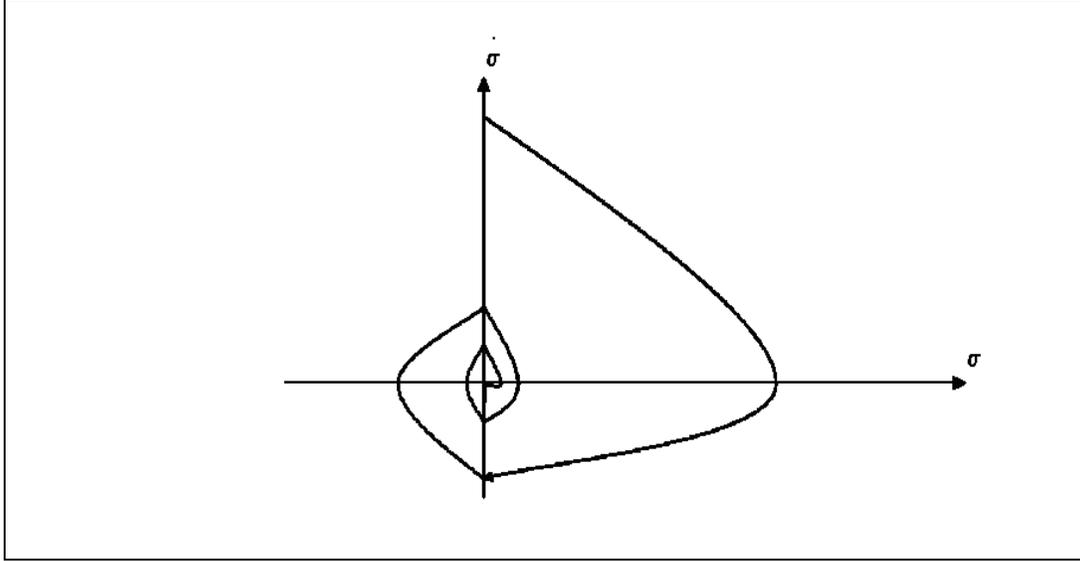


Figure II.9 Plan de phase de la commande Twisting

II.7.2.2 Commande Super Twisting

La commande Super Twisting a été proposée pour les systèmes de degré relatif 1 par rapport à la fonction contrainte σ [76]. Cette loi ne nécessite pas la connaissance de la dérivée de la variable de glissement $\dot{\sigma}$. Mathématiquement, cette commande se compose de deux parties :

$$u(t) = \int \dot{u}_1(t) dt + u_2(t) \quad (\text{II.33})$$

$$\dot{u}_1(t) = \begin{cases} -u & \text{si } |u| > u_M \\ -W \operatorname{sgn}(y_1) & \text{si } |u| \leq u_M \end{cases}$$

$$u_2(t) = \begin{cases} -\lambda s_0^\rho \operatorname{sgn}(y_1) & \text{si } |y_1| > s_0 \\ -\lambda |y_1|^\rho \operatorname{sgn}(y_1) & \text{si } |y_1| \leq s_0 \end{cases}$$

Pour la convergence les conditions sont données comme suit:

$$\begin{cases} W > \frac{\Psi}{K_m}, & \lambda^2 \geq 4\Psi \frac{K_M}{K_m^2} \frac{W + \Psi}{W - \Psi} \\ 0 < \rho \leq 0.5 \end{cases} \quad (\text{II.34})$$

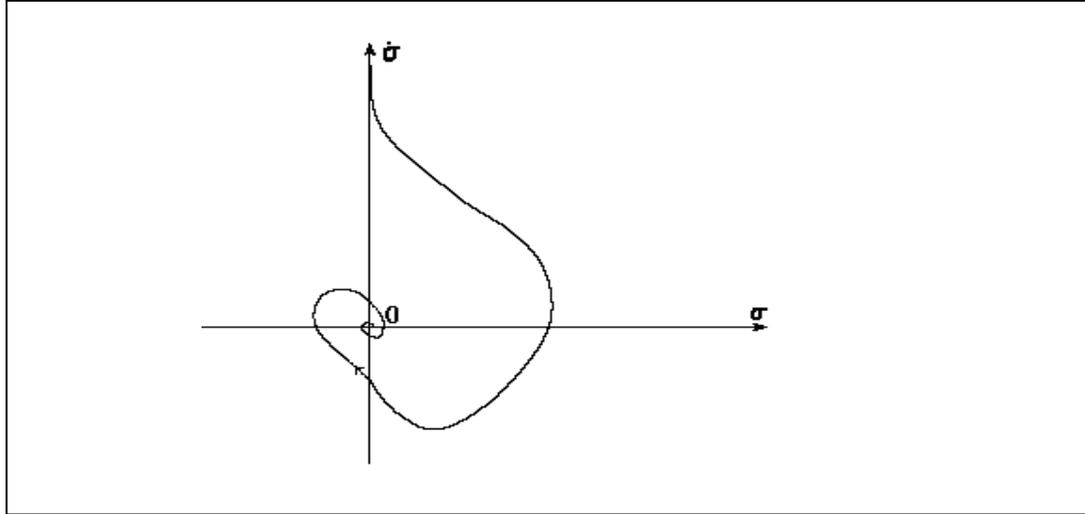


Figure II.10 Plan de phase de la commande Super Twisting.

II.8 LA COMMANDE PAR TERMINAL SLIDING MODE D'ORDRE SUPERIEUR

Le TSM d'ordre supérieur existe dans la littérature sur plusieurs algorithmes, le cas souvent utilisé est présenté comme suit [77]:

Soit le système non-linéaire multi-variable (II.35) :

$$y(x) = f(y^{(n-1)}, \dots, \dot{y}, y, t) + \Delta f(y^{(n-1)}, \dots, \dot{y}, y, t) + b(y^{(n-1)}, \dots, \dot{y}, y, t)u + d(t). \quad (\text{II.35})$$

La représentation d'état du système (II.35) s'écrit sous la forme suivante :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_n = f(x, t) + \Delta f(x, t) + b(x, t)u + d(t). \end{cases} \quad (\text{II.36})$$

Le vecteur des erreurs d'état est donné sous la forme suivante :

$$e = x - xd = [e^t, \dot{e}^t \dots e^{(n-1)}] \quad (\text{II.37})$$

Soit l'hyper-surface de glissement terminal suivante :

$$s(x, t) = A(e - v) \quad (\text{II.38})$$

Avec

$$v_i(t) = \begin{cases} \sum_{k=0}^n \frac{1}{k!} e_i(0)^{(k)} t^k + \sum_{k=0}^n \left[\sum_{l=0}^n \frac{a_{il}}{T^{j-l+n+1}} e_i(0)^{(l)} \right] t^{j+n+1} & \text{si } 0 \leq t \leq T \\ 0 & \text{si } t > T \end{cases} \quad (\text{II.39})$$

La loi de commande par globale terminal sliding mode est donnée par l'équation générale suivante :

$$\eta_{\Delta} = C(e - p) \quad (\text{II.40})$$

La variable de glissement dans le cas d'un système d'ordre supérieur est donnée par la forme suivante:

$$\begin{aligned} s_1 &= \dot{s}_0 + \alpha_0 s_0 + \beta_0 s_0^{q_0/p_0} \\ s_2 &= \dot{s}_1 + \alpha_1 s_1 + \beta_1 s_1^{q_1/p_1} \\ &\vdots \\ s_{n-1} &= \dot{s}_{n-2} + \alpha_{n-1} s_{n-2} + \beta_{n-2} s_{n-2}^{q_{n-2}/p_{n-2}} \end{aligned} \quad (\text{II.41})$$

La loi de commande dans ce cas est donnée par sa forme générale par l'équation suivante :

$$u(t) = \frac{1}{g(x)} \left[f(x) + \sum_{k=0}^{n-2} \alpha_k s_k^{n-k-1} + \sum_{k=0}^{n-2} \beta_k \frac{d^{n-k-1}}{dt^{k-n-1}} s_{(q_k/p_k)}^n + \phi s_{n-1} + s_{n-1}^{q/p} \right] \quad (\text{II.42})$$

La surface de glissement $s=0$, quand à elle, est atteinte en un temps $t_{s_{n-1}}$ tel que

$$t_{s_{n-1}} = \frac{p}{\varphi(p-q)} \ln\left(\frac{\varphi x_1(0)^{(p-q)/p} + \gamma}{\gamma}\right) \quad (\text{II.43})$$

II.9 CONCLUSION

Dans ce chapitre, nous avons essayé de faire un panorama des méthodes de commande robuste par mode de glissement. Au début, nous avons commencé par le cas classique, par la suite, nous avons abordé le cas de la commande par terminal sliding mode où la convergence en temps fini est l'objectif principal de la technique.

CHAPITRE III. *La Logique Floue de Type-2.*

III.1 INTRODUCTION

Le concept des sous ensembles flous "*Fuzzy Set Theory*" en Anglais fut introduit en 1965 par le Professeur Lotfi Zadeh de l'université de Berkeley, Etat Unis [46]. L'idée de la logique floue consiste à employer des grandeurs imprécises utilisées notamment dans le langage humain. Ces grandeurs permettent la représentation des classes vagues dont les frontières sont mal définies telles que l'âge d'une personne, la température d'une entité, l'éloignement d'un objet, etc. Du fait que la logique floue est exprimée simplement par des règles linguistiques de la forme Si-Alors, le nombre d'applications utilisant cette logique a augmenté considérablement ces dernières années. Cependant, avec la complexité de plus en plus élevée des systèmes à commander, cette logique a montré plusieurs limites, ce qu'a ramené à la proposition d'une nouvelle classe de la logique floue appelée « logique floue de type-2 ». Introduite en 1975 par l'inventeur de la logique floue (Lotfi Zadeh), cette nouvelle logique a été conçue avec l'objectif de repousser les limites de la logique floue classique nommée par la suite logique floue de type-1 [46]. Dans ce qui suit, nous allons introduire les notions de base de la logique floue de type-1 et de type-2 avec l'objectif de présenter tous les points clefs de cette nouvelle approche.

III.2 NIVEAU D'INTERVENTION D'UN SYSTEME FLOU DANS UNE BOUCLE D'ASSERVISSEMENT

Dans le domaine de l'automatique, la théorie mathématique stipule que les des systèmes d'inférence floue peuvent être regroupés grossièrement en trois grands modes :

III.2.1 La commande : Premièrement, la commande directe des processus (figure III.1). Dans ce cas, la base des connaissances floues agit directement sur les actionneurs du système à commander. Ceci afin d'assurer le suivi d'une consigne imposée. Le choix de la structure du contrôleur ainsi que sa base de règles de raisonnement n'est pas unique, il dépend de l'expertise de l'utilisateur et ses connaissances par rapport au système à régler. Dans ce cas, les informations sur les grandeurs à régler sont indispensables et sont généralement obtenues par des capteurs ou des observateurs.

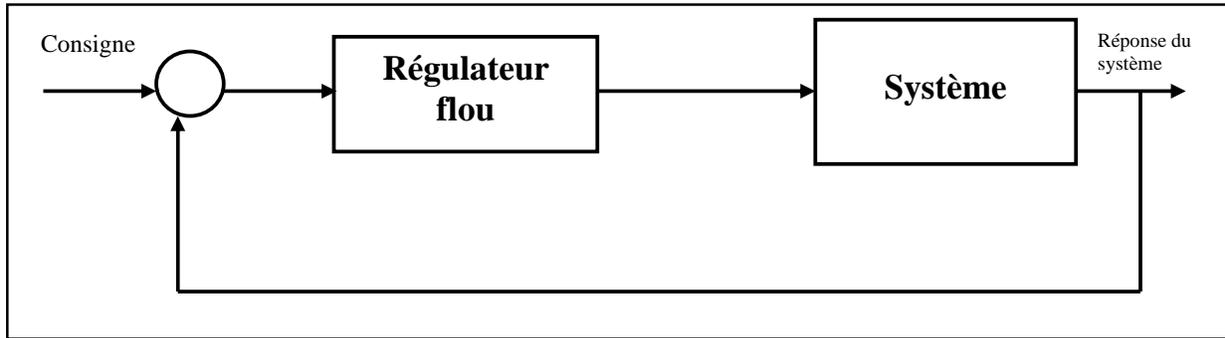


Figure III.1 Un SIF pour la commande

III.2.2 La supervision : Ce deuxième cas fait appel à la théorie de la logique floue pour la supervision des systèmes qui possèdent déjà des systèmes de commande. Le superviseur flou peut effectuer plusieurs tâches telles que l'ajustement des paramètres d'un régulateur utilisé (gain scheduling en Anglais), la surveillance par le choix du contrôleur dans une commande multi modèles, etc.

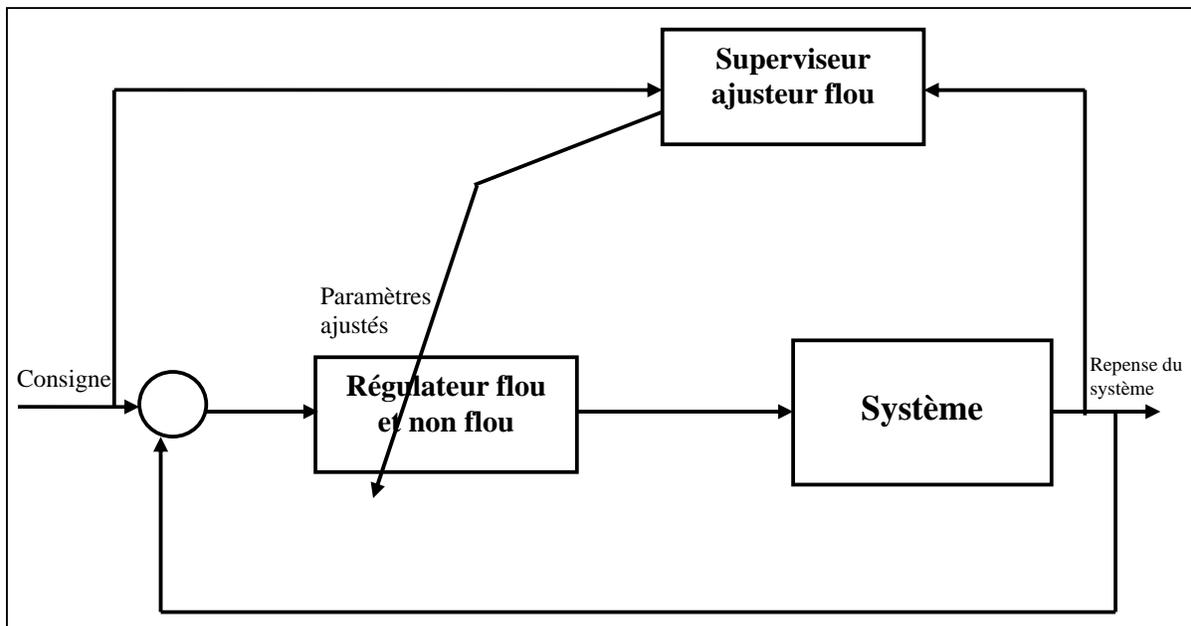


Figure III.2 Un SIF pour l'ajustement et/ou l'adaptation d'un autre contrôleur

III.2.3 L'observation : Le troisième volet de l'utilisation de la logique floue est l'estimation des variables d'états inaccessibles afin de mesurer les informations indispensables à la mise en œuvre de régulation. La simplicité de synthèse fait que l'algorithme le plus utilisé dans ce cas est celui de Takagi-Sugino d'ordre zero [72], le SIF (Système d'Inférence Floue) est utilisé aussi pour le diagnostic des défauts.

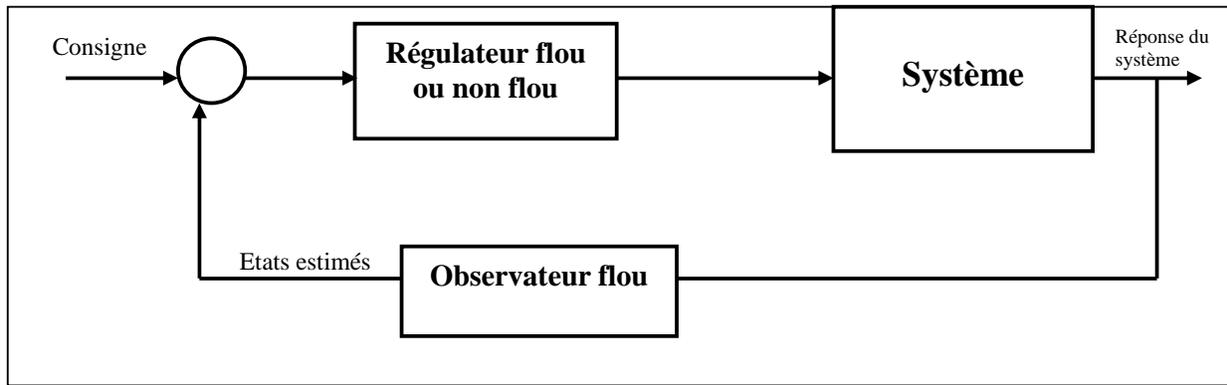


Figure III.3 Un SIF pour l'observation.

III.3 LES CARACTERISTIQUES DE LA COMMANDE PAR LOGIQUE FLOUE

L'utilisation de la logique floue pour la commande ne nécessite aucun modèle mathématique du système commandé. La connaissance issue de l'expertise humaine ou de l'expérience peut être prise en compte et la conception de la commande dans ce cas est relativement simple. Contrairement aux réseaux de neurones, un contrôleur flou est constitué d'une architecture dont les paramètres sont maîtrisables. Un contrôleur flou est utilisable aussi pour la modélisation des systèmes. Il est très simple de combiner un contrôle flou avec d'autres techniques de commande. Un SIF est utilisé pour l'adaptation des paramètres d'autres contrôleurs. Dans ces dernières années, les systèmes flous sont sollicités comme observateurs et/ou superviseurs. En revanche, il n'existe pas de directive précise pour concevoir un régulateur et un nombre important de paramètres doit être défini (le nombre de règles, fonctions d'appartenance ainsi que leurs paramètres, etc.). Dans un SIF, le nombre de règles augmente considérablement avec l'augmentation du nombre d'entrées. Il faut souligner aussi que son utilisation dans le cas multi-variable s'avère très complexe.

III.4 LOGIQUE FLOUE TYPE-1

III.4.1 Les ensembles flous de type-1

En logique binaire, la fonction d'appartenance caractérisant l'appartenance d'un élément x , de l'univers de discours X à un ensemble classique A , prend uniquement les valeurs 0 et 1 [79] :

$$A_{\mu}(x) \quad X \rightarrow \{0,1\}$$

$$\begin{aligned} x \rightarrow \mu_A(x) &= 1 && \text{si } x \in A \\ \mu_A(x) &= 0 && \text{si } x \notin A \end{aligned} \quad (\text{III.1})$$

Un sous ensemble flou est peut être perçu comme une généralisation du concept d'ensemble

classique. Dans la théorie des sous ensembles flous, les valeurs de la fonction d'appartenance $A_\mu(x)$ sont comprises dans l'intervalle $[0,1]$ [79] :

$$\begin{aligned} X &\rightarrow \{0,1\} \\ x &\rightarrow \mu_A(x) \end{aligned} \quad (\text{III.2})$$

Où un ensemble flou A est défini par l'ensemble de paires $(\mu_A(x), x)$, tel que:

$$A = \{ (\mu_A(x), x) / x \in X \} \quad (\text{III.3})$$

III.4.2 Structure générale d'un SIF de type-1

La détermination de la commande par un régulateur flou s'effectue en trois étapes : la fuzzification, l'inférence, et puis, la défuzzification (figure III.4)

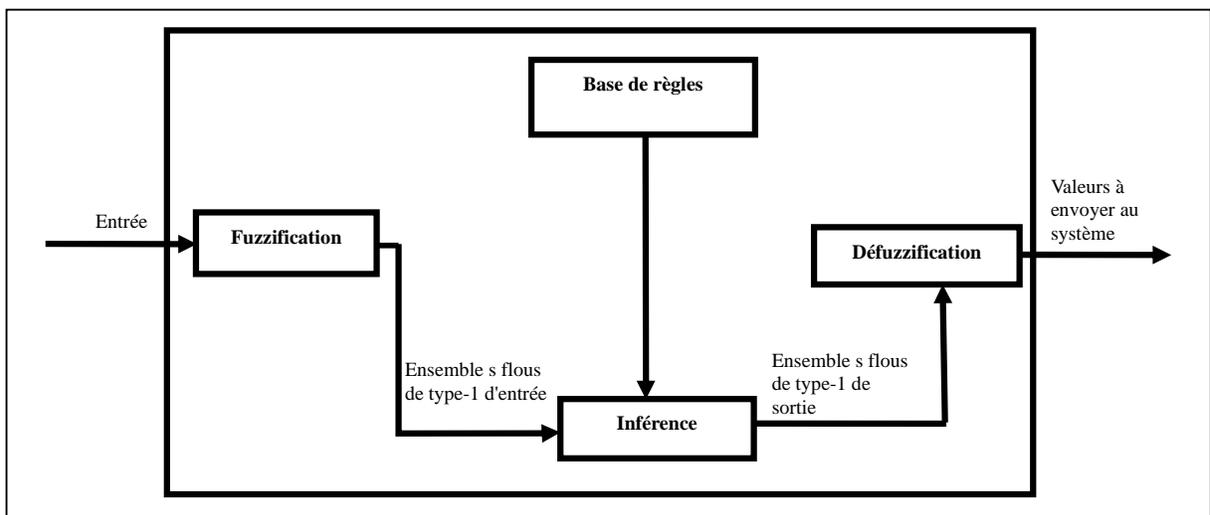


Figure III.4 Schéma du principe d'un contrôleur flou type-1 [72]

- 1) **La fuzzification** : Cette première étape consiste à fuzzifier les entrées, soit à transformer ces entrées en des variables linguistiques. Une variable linguistique est définie dans la base de connaissances par un ensemble de fonctions d'appartenance. La fuzzification permet d'établir le degré d'appartenance des entrées pour différentes variables linguistiques qui peuvent les caractériser. Postérieurement, ces variables linguistiques sont utilisées dans des règles de type "Si condition Alors décision" pour inférer des décisions.
- 2) **L'inférence** : Les règles linguistiques décrivent la politique de contrôle en faisant correspondre

une liste d'antécédents à une conséquence. Ces règles sont évaluées en parallèle et doivent toutes avoir été évaluées avant que le système ne puisse fournir une décision. Le mécanisme d'inférence est utilisé pour réaliser la conjonction des antécédents liés par des connexions "et" dans une règle, ainsi que pour unifier les résultats de règles associées à une même conséquence floue.

- 3) **la défuzzification** : cette étape consiste à faire correspondre les degrés flous associés aux conséquences en une valeur non floue qui présente la sortie du système. Les signaux d'entrées/sorties d'un contrôleur flou sont généralement modifiés par des gains appelés gains de normalisation pour les entrées et de dénormalisation pour les sorties, leur détermination sera souvent effectuée par tâtonnement, ou par une loi adaptative.

III.4.3 Types de contrôleur flou : deux principaux contrôleurs flous sont utilisés de nos jours. La différence entre les deux réside au niveau de la forme des conclusions des règles floues telles que:

1. **contrôleur de Mamdani** : Dans la méthode de Mamdani [72], les prémisses et les conclusions des règles sont floues. Cette méthode repose sur l'utilisation de l'opérateur min (minimum) pour l'inférence floue et de l'opérateur max (maximum) pour l'agrégation des règles. La défuzzification est généralement effectuée par la méthode du centre de gravité. Une variante de la méthode de Mamdani consiste à remplacer l'opérateur *min* de l'inférence floue par l'opérateur *prod* (produit).
2. **contrôleur de Takagi-Sugeno**: Dans la méthode de Takagi-Sugeno [72], les conclusions des règles sont des valeurs numériques calculées à partir des entrées. L'inférence floue est réalisée avec les opérateurs *min* et *prod*. La valeur finale est obtenue en effectuant une moyenne pondérée des conclusions.

III.5 EXTENSION DU CONTROLEUR FLOU DE TYPE TSUGENO (LE RESEAUX DE NEURONES A FONCTION DE BASE RADIALE (RBF)) :

Les réseaux de neurones à fonctions de bases radiales sont des réseaux de neurones caractérisés par leur structure très simplifiée et similaire à un système d'inférence floue de type Takagi-Sugeno [27]. Ce réseau est constitué d'une seule couche cachée dont les fonctions d'activation sont des fonctions à base radiale, généralement des Gaussiennes. Les entrées sont directement connectées aux neurones de la couche cachée. La fonction d'activation du neurone de la couche de sortie est l'identité. La fonction d'activation de la couche cachée est donnée par :

$$h_j(x) = e^{-\left(\frac{\|x-c_j\|}{2.\sigma_j^2}\right)^2} \quad (\text{III.4})$$

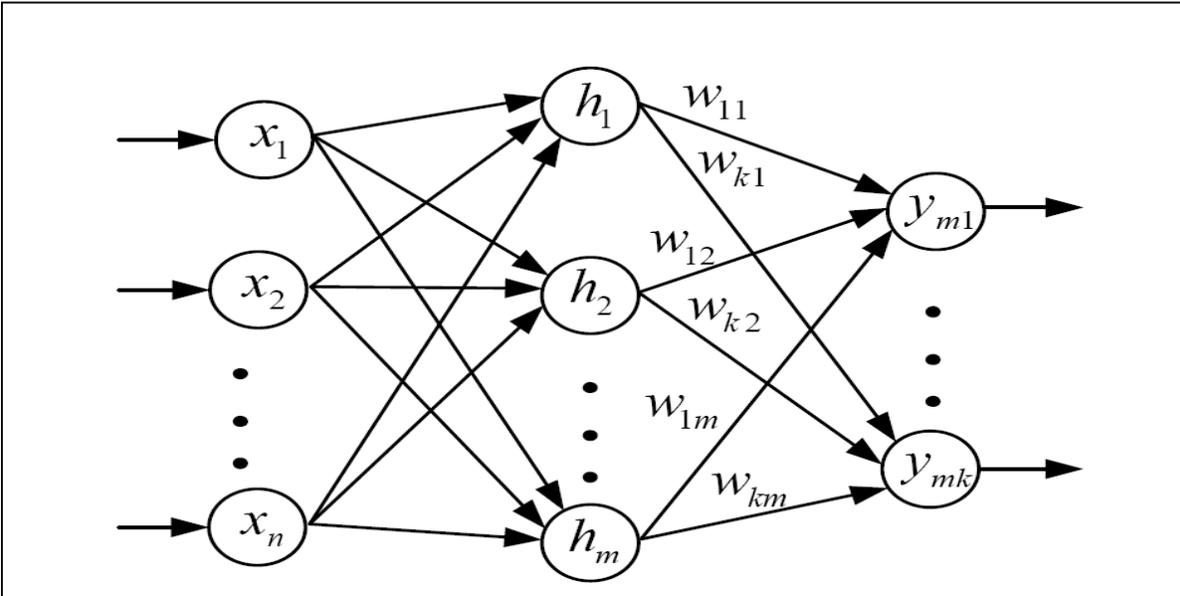


Figure. III.5 Structure générale d'un réseau de neurones de fonctions à bases radiales [80].

Les fonctions Gaussiennes sont définies par leur centre c_j et leur écart-type σ_j . Ces derniers avec les poids w_j sont les paramètres à optimiser en vue de l'apprentissage. La réponse de la fonction de base radiale ainsi positionnée diminue rapidement en fonction de l'éloignement du vecteur d'entrée par rapport au centre de la fonction à base radiale [57].

Nous avons choisi ce réseau afin d'éviter de passer par la lourde procédure d'apprentissage qui est nécessaire dans le cas des perceptrons multicouches.

III.6 SOURCES D'INCERTITUDE DANS UN SYSTEME A COMMANDER ET RAISONS DE L'USAGE DE LA LOGIQUE FLOUE DE TYPE-2

Il existe plusieurs sources d'incertitude dans les processus physiques. Les cinq principaux états qui engendrent des situations d'incertitude naturelles dans les systèmes à commander peuvent être résumés dans les cas suivants :

1. Les paramètres du système à commander ne sont pas connus avec certitude ;

2. L'imperfection des instruments de mesure (exemple : résolution dégradée des capteurs, résolution des cartes d'entrées/sorties, etc.) ;
3. A haute fréquence, l'ordre et la structure du système sont généralement inconnus ;
4. A cause de la présence des non linéarités, les paramètres du modèle peuvent varier en fonction du changement des conditions de travail (exemple : identification d'un système pour une réponse à des échelons de différentes amplitudes) [20].
5. L'erreur de raisonnement qui résulte de la fatigue de mise en œuvre et de l'incapacité de savoir atteindre l'objectif désigné par l'utilisateur [65].

À cet effet, les systèmes flous de type-1 dont les fonctions d'appartenance sont des sous ensembles flous de type-1 sont incapables de prendre en compte de telles incertitudes de règles. Nous introduisons dans ce qui suit une nouvelle classe de systèmes flous appelée système flou type-2, dans laquelle les valeurs d'appartenance des prémisses ou/et des conséquences sont elles-mêmes des ensembles flous type-1. Les ensembles flous type-2 sont très efficaces dans les circonstances où il est difficile de déterminer exactement les fonctions d'appartenance pour les ensembles flous [66].

III.7 LA DIFFERENCE ENTRE UN SYSTEME FLOU DE TYPE-1 ET DE TYPE-2:

L'arrivée graduelle de la logique floue de type-2 dans le domaine de l'engineering a permis de découvrir de jour au jour la capacité de cette logique à résoudre des problèmes liés aux limitations de cas classique. Dans le paragraphe suivant, nous allons nous focaliser sur les avantages essentiels apportés par la logique floue de type-2 comparée avec celle de type-1 :

1-Un "ensemble flou de type-2 intervalle" peut mieux modéliser les incertitudes intra-personnelles (selon Mendel "l'incertitude intra-personnelle est l'incertitude d'une personne sur un mot") et les incertitudes inter-personnelles (selon Mendel " l'incertitude inter-personnelle est l'incertitude d'un groupe de gens sur un mot"). Ces incertitudes sont liées au langage naturel. Parce que le degré d'appartenance d'un IT2FS est un intervalle au lieu d'un certain nombre fixe dans le cas d'un T1FS. Mendel [33] a également montré qu'IT2FS est un modèle scientifiquement correct pour modéliser l'incertitude linguistique, alors que T1FS n'est pas capable de le modéliser.

2- La capacité de l'empreinte d'incertitude (FOU) à couvrir les incertitudes permet la présentation des domaines de variables d'entrées / sorties avec moins de sous-ensembles flous, c'est-à-dire que l'utilisation des sous-ensembles flous de type-2 intervalle pour représenter les entrées et/ou sorties du contrôleur flou engendrera systématiquement la réduction de la base de règles comparativement au cas où les systèmes flous de type-1 sont utilisés, cela également

rend plus facile l'élaboration de la base de règles et augmente aussi la robustesse.

3- Les systèmes d'inférence floue de type-2 peuvent réaliser des relations d'entrée-sortie plus complexes qui ne peuvent pas être obtenus par le cas classique (type-1) [46].

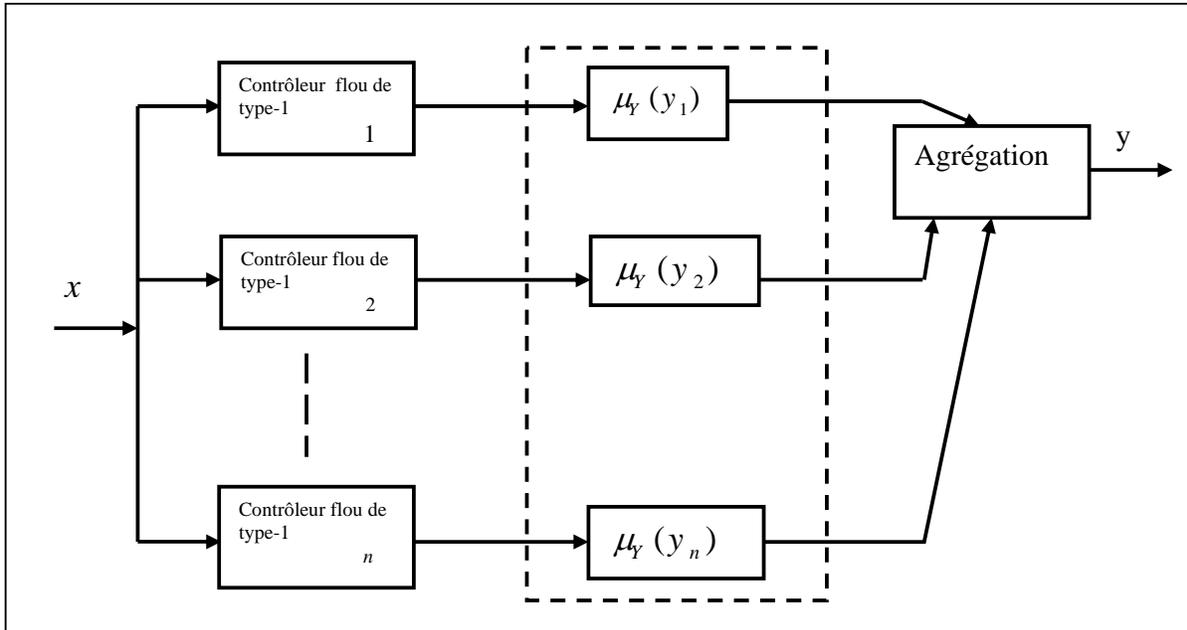


Figure III.6 Relation entre un SIF de type-1 et un SIF de type-2 [46]

III.8 LOGIQUE FLOUE TYPE-2

La notion de la logique floue type-2 a été introduite en 1975 par le professeur L. Zadeh [46] comme une généralisation de la logique floue classique (dite récemment de type-1). L'objectif principal de la logique floue de type-2 est d'incorporer des incertitudes sur les règles pour compenser les principales sources d'erreur des systèmes flous de type-1. Nous pouvons dire que le système flou de type-2 est une généralisation de type-1. Dans ce sens, l'incertitude réside non seulement dans les variables floues, mais aussi dans les fonctions d'appartenance [52]. La fonction d'appartenance de type 2 a trois dimensions par l'implication de l'empreinte d'incertitude (FOU). Cette dernière est une nouvelle dimension permettant de maîtriser le maximum d'incertitude du système commandé [46]. Actuellement, la logique floue de type-2 est un domaine de recherche très actif et plusieurs applications ont été réalisées en se basant sur cette approche, notamment la navigation des robots mobiles [26], le traitement d'images [24], la reconnaissance des formes et la commande des systèmes [47-48], etc. Dans ce chapitre, nous nous sommes intéressés à la logique floue type-2 pour la commande de robots manipulateurs actionnés par des muscles artificiels pneumatiques. Dans la plupart des applications réelles, nous ne pouvons pas déterminer les formes exactes des fonctions

d'appartenance à cause des incertitudes sur les données, c'est-à-dire que la dynamique du système est non maîtrisable. Nous pouvons dire aussi que les ensembles flous type-1 sont le premier ordre d'approximation des incertitudes et les ensembles flous type-2 constituent le deuxième ordre, ce qui rend possible la définition des ensembles flous type-n, mais la complexité du traitement croît énormément.

III.8.1 Les Applications de la Logique Floue de Type-2

Depuis son apparition en 1975, la logique floue de type-2 n'a pas cessé d'attirer l'attention des chercheurs. Au début, avec le manque d'une théorie complète, le souci était toujours de la compléter, de la simplifier et de la mettre en évidence pour l'exploitation dans les différents domaines de l'engineering. Plusieurs équipes de recherche avaient investi le domaine. Le premier système d'inférence floue type-2 compilé a été synthétisé en 1999 [53] par N. N. Karnik et J. M. Mendel de l'Université de Southern Californie. Après la publication de leur fameux article " *Type-2 Fuzzy Logic Systems*" sur les différentes méthodes de réduction de type qui pose un sérieux problème dans les systèmes d'inférence floue de type-2. Les premières applications réelles dans la commande sont apparues en 2004 [9]. Le diagramme donné dans la figure III.7 montre l'intérêt qui a été porté à la logique floue de type-2 ces dernières années (depuis son apparition jusqu'à 2006) :

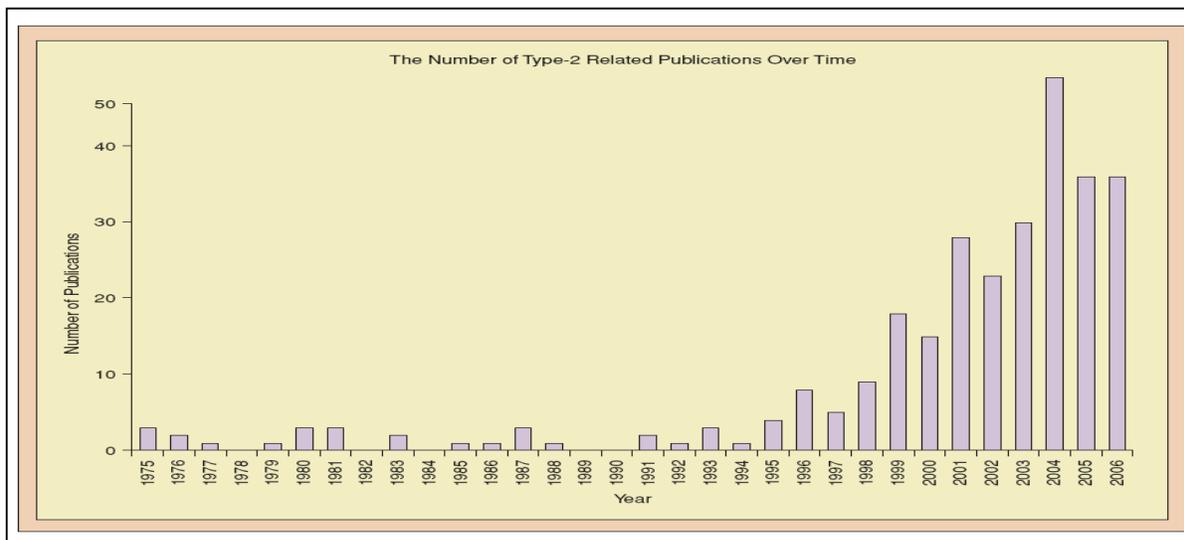


Figure III.7 Évolution de nombre de publications de la logique floue de type-2 en fonction des années depuis son apparition.

La recherche dans le domaine de la logique floue de type-2 s'active actuellement sur deux principaux champs, le premier concerne l'application de la commande sur les différents types de

systemes. On trouve entre autre l'application sur un bioréacteur [25], la commande de robot mobile [38], robots rigide et flexible [37][36], commande visuelle d'un robot humanoïde [34], DC-DC convertisseur [41], le traitement d'images [14], et d'autres. Le deuxième champ de recherche du domaine concerne l'amélioration et l'éclaircissement de la théorie de logique floue de type-2, où il y a beaucoup de travaux, notamment sur la réduction de type et les formes des fonctions d'appartenance [26][29][48][51], afin de minimiser le temps de traitement. La figure III.7 montre que les chercheurs dans la logique floue de type-2 ont commencé timidement à s'intéresser au domaine depuis son apparition en 1975 jusqu'à 1998. Par la suite, une forte attention a été portée au domaine, où une moyenne d'une trentaine de publications apparaît chaque année jusqu'à 2006. Malgré que ce domaine soit encore dans ses débuts, les travaux sont en constante hausses.

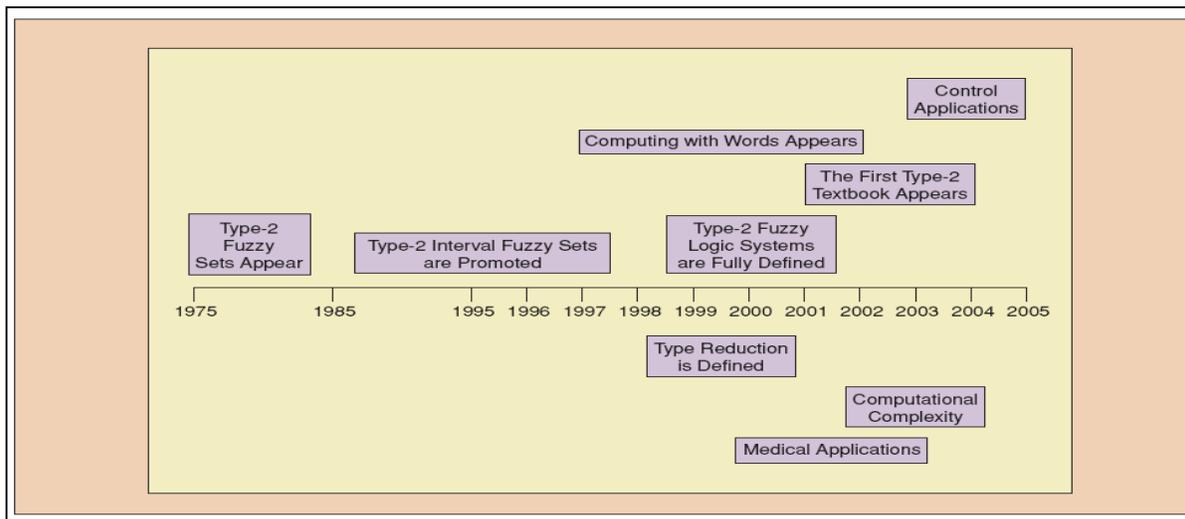


Figure III.8 L'évolution de la logique floue de type-2 le long des années et ses applications [63].

Définition 1: Fonctions d'appartenance type-2 : Comme dans la logique floue de type-1, le choix de la forme des fonctions dans le cas de la logique floue de type-2 est arbitraire. Habituellement, de nombreuses formes de fonctions d'appartenance de type-2 peuvent être utilisées telles que : Gaussienne, Triangulaire, Trapézoïdale, etc. Parmi les avantages de la logique floue de type-2, la possibilité de combiner différentes formes de fonctions d'appartenance, pour présenter une valeur floue, on trouve entre autre des combinaisons des triangles et des Gaussiennes [81], etc. Afin de présenter les propriétés des sous-ensembles flous de type-2 et de simplifier la compréhension de la théorie de la logique floue de type-2, nous avons choisi d'utiliser une fonction de type gaussienne. Ce choix est justifié par l'existence de deux paramètres ajustables (la moyenne et l'écart type), et aussi la simplicité de sa programmation, ce qui va simplifier la compréhension de cette théorie [46]. L'équation d'une forme Gaussienne est définie par :

$$\mu(x) = \exp\left(-\frac{1}{2} \frac{(x - m)^2}{\sigma^2}\right) \tag{III.5}$$

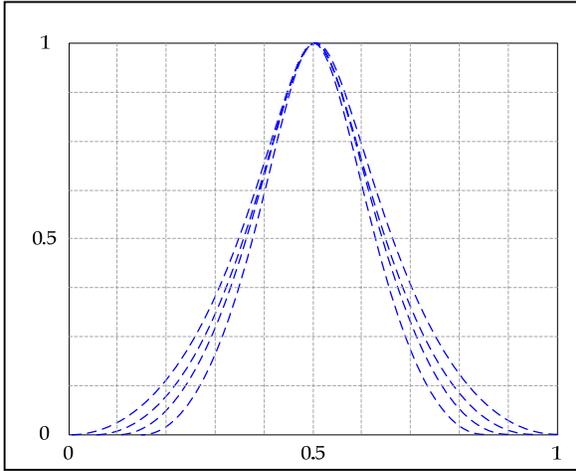


Figure III.9 Fonction d'appartenance Gaussienne type-2 (écart type variable)

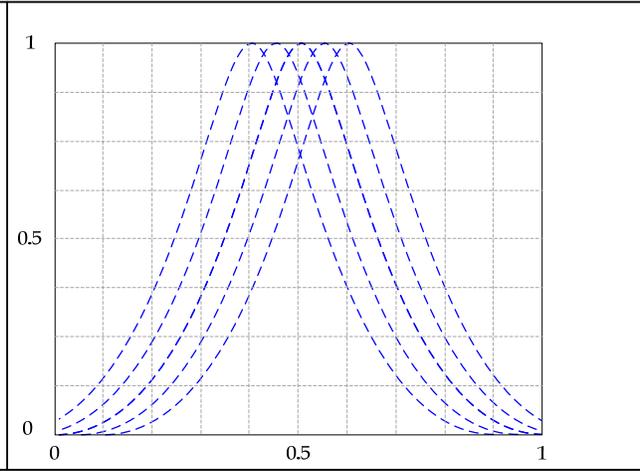


Figure III.10 Fonction d'appartenance Gaussienne type-2 (moyenne variable)

Définition 2 : Fonction d'appartenance primaire : Une fonction primaire (Primary Membership Function en Anglais) est une fonction d'appartenance de type-1 incluse dans le domaine de variation de la fonction d'appartenance type-2 [11]. Autrement dit, une fonction primaire est une fonction d'appartenance de type-1 appartenant au FOU.

Définition 3 : L'ensemble flou de type-2 : Théoriquement, un ensemble flou de type-2 \tilde{A} dans X est donné par [3] [5]:

$$\tilde{A} = \left\{ (x, u), \mu_{\tilde{A}}(x, u) / \forall x \in X, \forall u \in J_x \subseteq [0,1] \right\} \tag{III.6}$$

Une autre représentation mathématique de l'ensemble flou de type-2 peut être obtenue comme suit :

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x \subseteq [0,1]} 1/(x, u) = \int_{x \in X} \left[\int_{u \in J_x \subseteq [0,1]} 1/u \right] / x \tag{III.7}$$

Où $x \in X$ est appelé "variable floue primaire", $u \in J_x$ est la variable floue secondaire, \int représente l'union des variables x et u , $J_x \subseteq [0,1]$ est la fonction d'appartenance primaire de x , pour les ensembles flous de type-2 intervalle, le second degré d'appartenance de \tilde{A} est toujours égale à 1 [5][33].

Définition 4: l’empreinte d’incertitude (en Anglais Footprint of Uncertainty (FOU)) représente l’incertitude sur l’ensemble \tilde{A} . Mathématiquement, cette empreinte représente l’union de toutes les fonctions floues premières, telle que :

$$FOU(\tilde{A}) = \bigcup_{\forall x \in X} J_x = \left\{ (x, u) : u \in \left[\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x) \right] \right\} \tag{III.8}$$

Avec

$\underline{\mu}_{\tilde{A}}(x) = \underline{FOU}(\tilde{A})$ et $\overline{\mu}_{\tilde{A}}(x) = \overline{FOU}(\tilde{A})$, $\forall x \in X$ représentent respectivement les fonctions d’appartenance floues supérieures et inférieures.

Définition 5: Fonctions d'appartenance supérieure et inférieure : Un ensemble flou de type-2 est complètement caractérisé par des fonctions d'appartenance supérieure et inférieure (upper et lower membership functions, en Anglais), elles sont toutes les deux des fonctions d'appartenance de type-1. La fonction supérieure est définie comme la plus grande fonction d'appartenance primaire, alors que la fonction d'appartenance inférieure est définie comme étant la plus petite fonction d'appartenance primaire de type-1. Ces fonctions sont montrées dans l’exemple de la figure III.11.

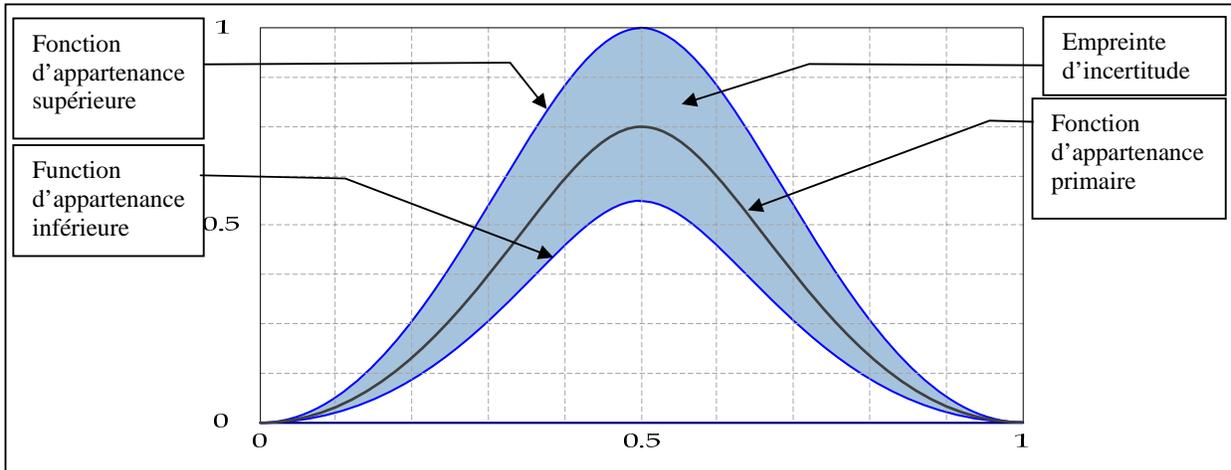


Figure III.11 Caractéristique de la fonction d’appartenance type-2 (exemple : fonction Gaussienne)

Les deux fonctions supérieure et inférieure présentent en quelque sorte les frontières de l’empreinte d’incertitude.

III.9 LES OPERATEURS DE LA LOGIQUE FLOUE TYPE-2

III.9.1 Union : l'union de deux ensembles flous type-2 \tilde{A}_1, \tilde{A}_2 est un ensemble flou type-2 défini par le maximum entre les fonctions d'appartenance primaire, l'écriture mathématique de cette définition est la suivante :

$$\tilde{A}_1 \cup \tilde{A}_2 = \int_{x \in X} \tilde{A}_1 \cup \tilde{A}_2 / X \tag{III.9}$$

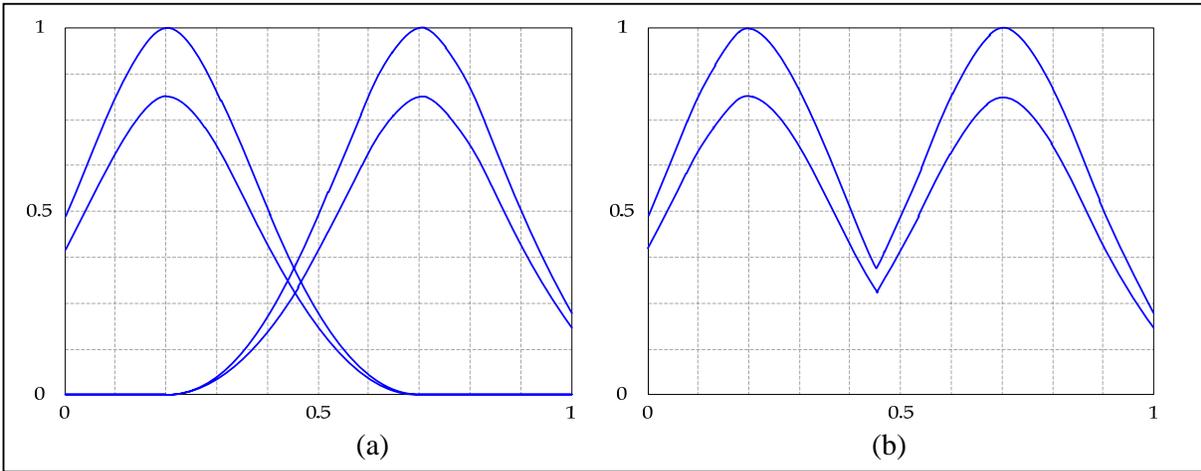


Figure III.12 Union de deux fonctions d'appartenance floue de type-2 (a) fonctions floues de type-2 (b) union de deux fonctions.

III.9.2 Intersection : L'intersection de deux ensembles flous type-2 \tilde{A}_1, \tilde{A}_2 est ensemble flou type-2 défini par le minimum entre les fonctions d'appartenance primaire. L'écriture mathématique de cette définition est la suivante :

$$\tilde{A}_1 \cap \tilde{A}_2 = \int_{x \in X} \tilde{A}_1 \cap \tilde{A}_2 / X \tag{III.10}$$

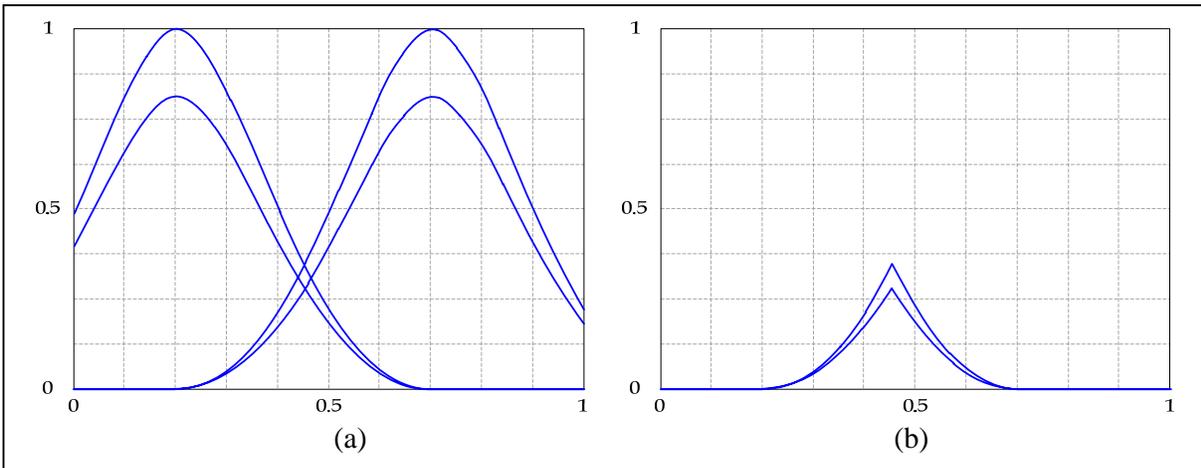


Figure III.13 Intersection de deux fonctions d'appartenance floue de type-2, (a) deux fonctions d'appartenance de type-2 (b) sont intersection.

III.9.3 Complément d'un l'ensemble flou type-2 : le complément d'un ensemble flou type-2 \tilde{A} est un ensemble flou type-2 défini comme le complément de toutes les fonctions d'appartenance primaires, exactement comme les ensembles flous type-1, l'écriture mathématique de cette définition est la suivante :

$$\tilde{A}' = \int_{x \in X} \mu_{\tilde{A}'}(x) / X \quad (\text{III.11})$$

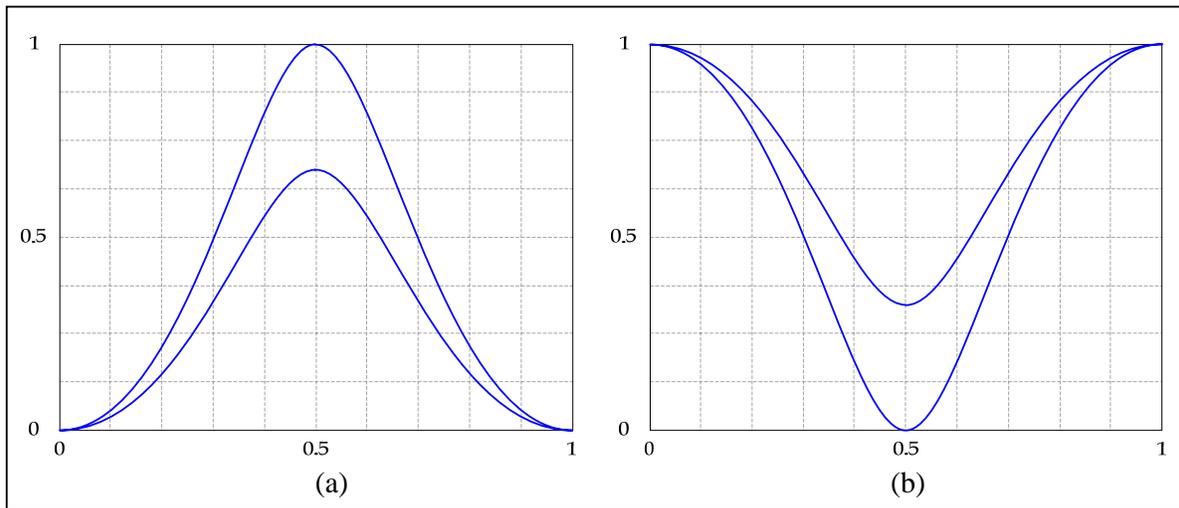


Figure III.14 Complément d'un ensemble flou type-2

III.10 LE SYSTEME D'INFERENCE FLOUE DE TYPE-2 INTERVALLE

Le principe de la logique floue ne change pas en passant des systèmes flous de type-1 aux systèmes flous de type-2. En général, ils ne changent pas pour n'importe quel type- n . Un type d'ordre supérieur indique un degré de fuzzification plus élevé « a higher degree of fuzziness ». C'est la nature des fonctions d'appartenance qui change, et bien évidemment seules les opérations qui dépendent de ces fonctions changent. La structure des règles dans le cas d'un système flou type-2 est la même que dans le cas d'un système flou type-1. Néanmoins, deux différences principales distinguent un contrôleur flou de type-2 d'un contrôleur classique. Ces différences consistent dans les formes des fonctions d'appartenance et l'incorporation d'une étape entre l'inférence et la défuzzification qui permet de transformer les données qui sont des ensembles flous de type-2 en des ensembles flous de type-1. La structure générale d'un système flou de type-2 est donnée par la figure III.15.

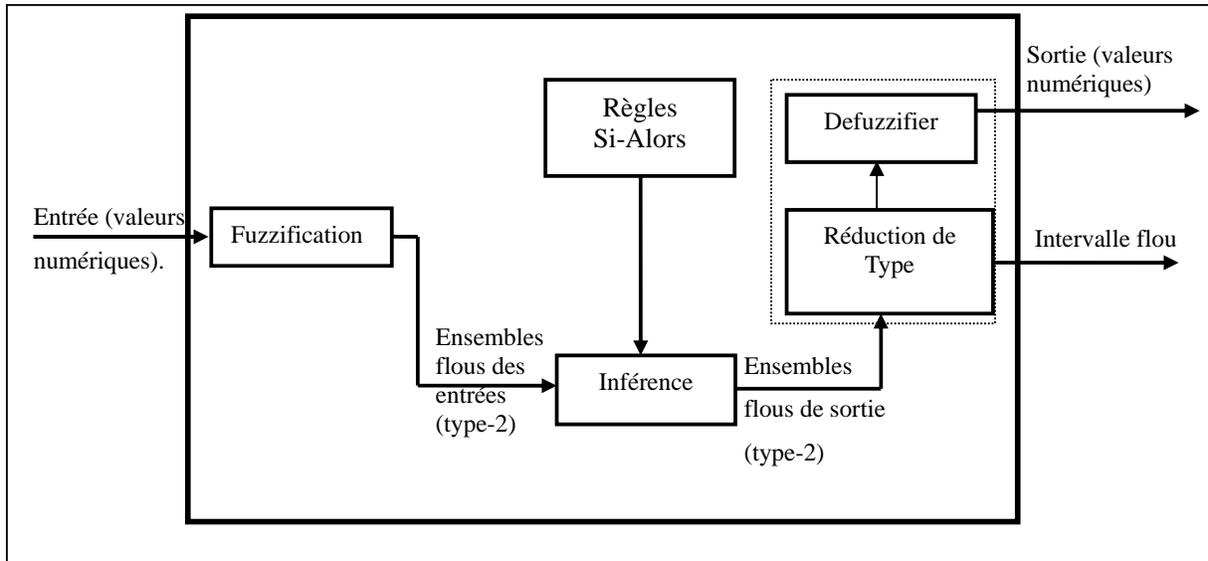


Figure III.15 Schéma du principe d'un SIF de type-2

On voit bien que la forme générale d'un contrôleur flou de type-2 n'a pas changé d'une manière radicale par rapport au contrôleur flou de type-1, à l'exception du bloc de réduction de type qui permet de passer des ensembles flous type-2 vers des ensembles flous type-1.

III.10.1 Fuzzification : Cette étape est la même que celle dans le cas de la logique floue de type-1, sauf que dans le cas de la logique floue de type-2, le degré d'appartenance est défini par un intervalle de valeurs comprises entre $[\underline{f}, \overline{f}]$ où \underline{f} représente le degré d'appartenance de la fonction d'appartenance inférieure (Lower MF) et \overline{f} est le degré d'appartenance de la fonction d'appartenance supérieure (Upper MF).

III.10.2 Tables des règles : La structure des règles dans le cas d'un contrôleur flou de type-2 est la même que dans le cas d'un contrôleur flou de type-1, la seule différence réside dans les prémisses et/ou les conséquences représentées au minimum par une valeur floue de type-2 [11].

$$R^i : IF x_1 \text{ is } \tilde{F}_1^i \text{ and } \dots \text{ and } x_p \text{ is } \tilde{F}_p^i \text{ THEN } u \text{ is } \tilde{G}^i, \quad i = 1, \dots, M. \quad (\text{III.12})$$

III.10.3 Inférence : Dans les systèmes d'inférence floue de type-2, comme dans le cas des systèmes type-1, le moteur d'inférence combine les règles actives pour donner un ensemble flou type-2 à sa sortie. Le moteur d'inférence combine les règles et définit la relation entre les variables floues des entrées et celles des sorties du système flou de type-2. Les multiples entrées actives sont connectées

par un opérateur d'intersection. Les degrés d'appartenance dans les ensembles d'entrée sont combinés avec ceux des ensembles de sortie en utilisant l'opérateur d'union. À cet effet, les règles floues peuvent être écrites comme suit:

$$R^i : F_1^i \times \dots \times F_p^i \rightarrow \tilde{G}^i = \tilde{A}^i \longrightarrow \tilde{G}^i \quad i = 1, \dots, M \quad (\text{III.13})$$

Le système (III.13) peut être réécrit de la façon suivante :

$$u_{R^i}(x, u) = \bigcap_{a=1}^p \mu_{\tilde{F}_a^i}(x_a) \bigcap \mu_{\tilde{G}^i}(u) \quad (\text{III.14})$$

Où \bigcap représente l'opérateur d'intersection, p est le nombre de variables. $\bigcap_{a=1}^p \mu_{\tilde{F}_a^i}(x_a) = \tilde{F}(x)$ qui est le résultat d'un ensemble flou intervalle est décrit par l'équation suivante [30] :

$$\tilde{F}(e) = \left[\underline{f}^i(x'), \overline{f}^i(x') \right] = \left[\underline{f}^i, \overline{f}^i \right] \quad (\text{III.15})$$

Où \underline{f} et \overline{f} peuvent être représentés par :

$$\begin{aligned} \overline{f}^i(x') &= \overline{\mu}_{\tilde{F}_1^i}(x'_1) * \dots * \overline{\mu}_{\tilde{F}_p^i}(x'_p) \\ \underline{f}^i(x') &= \underline{\mu}_{\tilde{F}_1^i}(x'_1) * \dots * \underline{\mu}_{\tilde{F}_p^i}(x'_p) \end{aligned} \quad (\text{III.16})$$

Où $*$ est un opérateur de produit mathématique.

L'exemple suivant montre la méthode d'inférence Max-Min souvent utilisée :

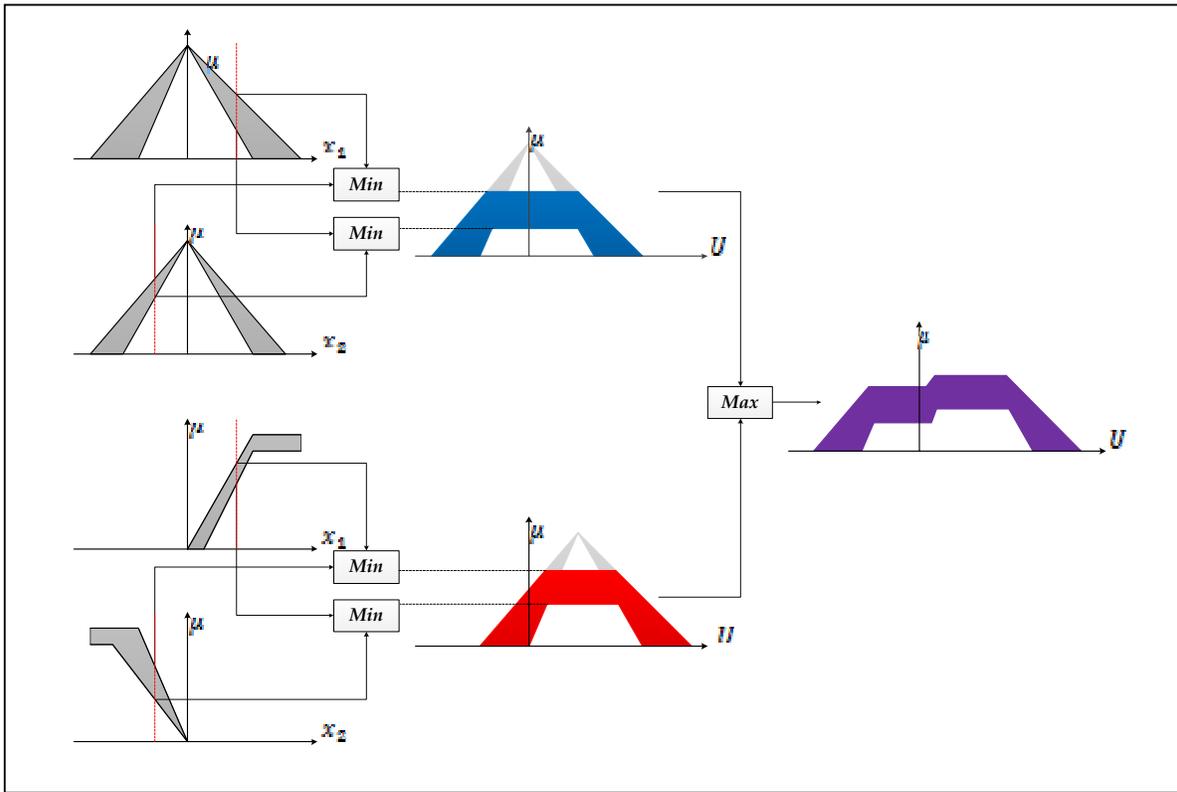


Figure III.16 Méthode d'inférence type-2 Max-Min

III.10.4 Réduction de type : L'implication et l'agrégation des règles floues de type-2 donnent à la sortie un sous ensemble flou type-2 sachant que ce sous-ensemble ne pourra pas être directement appliqué au système à commander. A cet effet, pour le cas type-2, il est nécessaire d'ajouter un bloc appelé « Réduction de type » qui permet de transformer l'ensemble flou type-2 de sortie vers un autre sous-ensemble flou qui sera cette fois un intervalle. La défuzzification s'en suivra normalement comme dans le cas classique, où elle consiste cette fois à calculer le centre de l'intervalle issu de la réduction de type. Il existe 5 méthodes de réduction de type qui sont : la méthode du centre de l'ensemble (Centroïde), la méthode de la hauteur, la méthode de la hauteur modifiée, la méthode de la valeur maximum et la méthode de la moyenne des centres. La méthode la plus utilisée est celle du Centroïde qui s'exprime par la relation générale suivante :

$$\begin{aligned}
 U_{\text{cos}} = [y_l, y_r] = & \int_{y^1 \in [y_l^M, y_r^M]} \cdots \int_{y^M \in [y_l^M, y_r^M]} \int_{f^1 \in [f^1, \bar{f}^1]} \cdots \\
 & \cdots \int_{f^M \in [f^M, \bar{f}^M]} \frac{1}{\frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i}} \quad \text{(III.17)}
 \end{aligned}$$

L'équation (III.17) est calculée à l'aide de l'algorithme de *Karnik-Mendel*. L'idée principale de cette

méthode itérative est de discrétiser l'ensemble de sortie type-2 à N points, puis de trouver le point de commutation pour y_l et y_r . Prenons y_l par exemple qui est le minimum de U_{cos} . Puisque y_l^i croit de la gauche vers la droite le long de l'axe horizontal de la figure III.17, on prend généralement le degré d'appartenance supérieur pour y_l^i gauche et le degré d'appartenance inférieur y_l^i droit. Lorsque le point de commutation est trouvé pour $y \leq k$, le degré d'appartenance supérieur est utilisé pour calculer y_l et pour $k > i$ c'est le degré d'appartenance inférieur qui est utilisé comme montré dans la figure III.17.

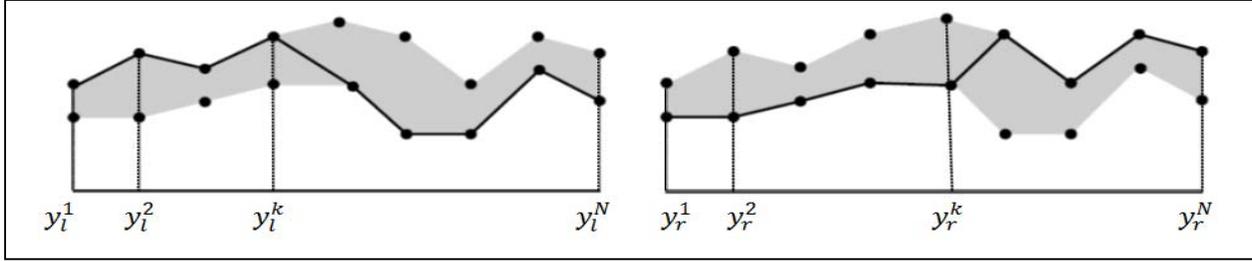


Figure III.17. Exemple montrant la méthode Karnik-Mendel pour la réduction de type

L'équation générale de la technique du centre de gravité (III.17), qui correspond au centre de gravité de l'ensemble flou de type-2 intervalle $G_{\tilde{G}^i}$ [5] peut être simplifié comme suit :

$$G_{\tilde{G}^i} = \int_{\theta_1 \in J_{y_1} \dots} \int_{\theta_N \in J_{y_N}} 1 / \frac{\sum_{i=1}^N y_i \theta_i}{\sum_{i=1}^N \theta_i} = [y_l^i, y_r^i] \tag{III.18}$$

avec:

$$y_l = \frac{\sum_{i=1}^M f_l^i y_l^i}{\sum_{i=1}^M f_l^i} \tag{III.19}$$

$$y_r = \frac{\sum_{i=1}^M f_r^i y_r^i}{\sum_{i=1}^M f_r^i}$$

Les algorithmes donnés dans les semi programmes suivants expriment d'une façon détaillée les calculs de y_l et y_r .

Start:

//pour calculer y_l

Initialization f^i by: $f^i = \frac{f_l^i + f_r^i}{2}$

Compute \dot{y} by: $\dot{y} = \frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i}$

do

$\dot{y} = \ddot{y}$

Compute k where: $y_k^i \leq \dot{y} \leq y_{k+1}^i$

Calculate by $\ddot{y} = \frac{\sum_{i=1}^k f_r^i y^i + \sum_{i=k+1}^M f_l^i y^i}{\sum_{i=1}^M f^i + \sum_{i=1}^M f^i}$

While $y_l \neq \dot{y}$

$y_l = \dot{y}$

Initialization f^i by: $f^i = \frac{f_l^i + f_r^i}{2}$

Compute \dot{y} by: $\dot{y} = \frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i}$

do

$\dot{y} = \ddot{y}$

Compute k where: $y_k^i \leq \dot{y} \leq y_{k+1}^i$

Calculate by $\ddot{y} = \frac{\sum_{i=1}^k f_l^i y^i + \sum_{i=k+1}^M f_r^i y^i}{\sum_{i=1}^M f^i + \sum_{i=1}^M f^i}$

While $y_r \neq \dot{y}$

$y_r = \dot{y}$

III.10.5. La défuzzification : la défuzzification donne une sortie numérique applicable au système à commander, elle se calcule comme suit [8] :

$$y = \frac{y_l + y_r}{2} \quad (\text{III.20})$$

III.11 CONCLUSION

La logique floue dite de type-1 est largement appliquée dans un nombre très important de domaines de recherche et de développement, notamment dans le domaine de la commande des systèmes. Ce dernier est peut être le domaine le plus sollicité par la logique floue. Par ailleurs, ce type de logique est apparu comme une solution dépassée, surtout pour les systèmes qui présentent une forte non-linéarité avec des incertitudes paramétriques et/ou des perturbations internes ou externes. A cet effet, la communauté scientifique qui travaille dans ce domaine pense à de nouvelles solutions qui permettent de pallier les nouveaux problèmes apparus avec l'arrivée de ces nouveaux systèmes très

complexes très difficiles à commander. Le fruit de ces recherches a engendré l'apparition de la logique floue de type-2.

Ce chapitre a été consacré aux notions de base de la logique floue de type-1 et de type-2, nous avons constaté qu'il existe deux différences essentielles du contrôleur flou de type-2 par rapport à celui de type-1, telles que les formes des fonctions d'appartenance et le processus de sortie. Dans le cas de type-1, les fonctions d'appartenance sont certaines, cependant, pour le type-2, les fonctions d'appartenance sont incertaines. La sortie du contrôleur classique de type-1 nécessite uniquement la défuzzification, par contre, le processus de sortie type-2 comporte une étape supplémentaire qui est la réduction de type.

CHAPITRE IV. *Limites Des Commandes Par Mode De Glissement, Et Par Mode De Glissement Flou.*

IV.1 INTRODUCTION

Dans ce chapitre, on traite les cas classiques de la commande par mode de glissement et son hybridation avec un contrôleur flou. L'objectif est de simplifier l'assimilation des problèmes de ce domaine. Nous proposons dans ce cas l'usage d'un modèle issu de l'identification des axes du robot pour la synthèse de la commande par mode de glissement. Nous avons ensuite proposé une commande hybride afin de surmonter le problème de broutement. Les résultats ont été obtenus en simulation pour un cas de poursuite d'un profil de vitesse trapézoïdale.

IV.2 SYNTHESE D'UNE COMMANDE PAR MODE GLISSANT FLOU BASEE SUR UN MODELE IDENTIFIE

La mise en œuvre d'une commande par mode glissant passe par deux étapes : le choix de la surface de glissement et l'établissement de la loi de commande.

La surface de glissement est choisie par la formule (IV.1) :

$$S(t) = \lambda \cdot e_1 + e_2 \quad (\lambda > 0) \quad (\text{IV.1})$$

Où le paramètre λ est déterminé d'une manière adaptative par l'équation (IV.2) :

$$\lambda = \frac{\lambda_0}{(1 + |e_1 \Phi|)} \quad (\text{IV.2})$$

Avec λ_0 est une valeur constante positive,

Φ : est la bonde de glissement, e_1 : est l'erreur.

Le modèle simplifié d'un axe du robot manipulateur après identification s'écrit pour chaque axe par l'équation (IV.3) :

$$\ddot{q} + a_1 \cdot \dot{q} + a_0 \cdot q = b_0 \cdot U \quad (\text{IV.3})$$

Avec une marge d'incertitude sur les paramètres dynamiques présentés comme suit :

$$\begin{aligned} a_0^* - \Delta a_0 < a_0 < a_0^* + \Delta a_0 \\ a_1^* - \Delta a_1 < a_1 < a_1^* + \Delta a_1 \\ b_0^* - \Delta b_0 < b_0 < b_0^* + \Delta b_0 \end{aligned} \quad (\text{IV.4})$$

a_0^* , a_1^* et b_0^* sont des paramètres estimés avec une marge de l'ordre Δa_0 , Δa_1 et Δb_0 .

L'objectif est de synthétiser une loi de commande $u(t)$ pour amener le plus vite possible le vecteur d'état de l'erreur $e(t) = [e_1, e_2]^T$ sur une variété de commutation $S(t) = 0$ et l'y maintenir jusqu'à l'état désiré.

Lorsque le régime glissant se produit sur la surface $S = 0$, la commande équivalente s'écrit :

$$u_{eq} = b_0^{-1} \cdot [a_0 \cdot e_1 + (a_1 - \lambda_1) \cdot e_2 + a_0 \cdot q_R + a_1 \cdot \dot{q}_R + \ddot{q}_R] \quad (\text{IV.5})$$

À cause des erreurs de modélisation, la commande équivalente u_{eq} est estimée comme suit :

$$u_{eq}^* = u_{eq} + \Delta u_{eq} \quad (\text{IV.6})$$

Avec :

$$u_{eq}^* = b_0^{*-1} \cdot [a_0^* \cdot e_1 + (a_1^* - \lambda_1) \cdot e_2 + a_0^* \cdot q_R + a_1^* \cdot \dot{q}_R + \ddot{q}_R]$$

A partir de (IV.5) et (IV.6) on obtient :

$$\Delta u_{eq} = u_{eq}^* - u_{eq} = d_3 \cdot e_1 + d_4 \cdot e_2 + d_3 \cdot q_R + d_1 \cdot \dot{q}_R + d_2 \cdot \ddot{q}_R \quad (\text{IV.7})$$

Avec :

$$d_1 = a_1^* \cdot b_0^{*-1} - a_1 \cdot b_0^{-1},$$

$$d_2 = b_0^{*-1} - b_0^{-1}$$

$$d_3 = a_0^* \cdot b_0^{*-1} - a_0 \cdot b_0^{-1},$$

$$d_4 = d_1 - \lambda_1 \cdot d_2$$

IV.2.1 La commande discontinue

La commande discontinue utilisée dans ce travail est la plus connue dans la littérature. Elle est donnée par l'équation (IV.8) :

$$\Delta u = -K \cdot \text{sign}(S) \quad (\text{IV.7})$$

La condition de glissement que la commande doit satisfaire est $S \cdot \dot{S} < 0$

La commande par mode glissant finale est donnée par l'expression (IV.9) :

$$u = u_{eq} + \Delta u_{eq} + \Delta u \quad (\text{IV.9})$$

IV.2.2. La condition d'existence d'une surface de glissement

Elle peut être déduite de la fonction énergétique de Lyapunov donnée par la relation (IV.10) :

$$V = \frac{1}{2} \cdot S^2 \quad (\text{IV.10})$$

Une condition suffisante pour que le système à commander soit stable est :

$$\dot{V} = \frac{1}{2} \cdot \frac{\partial}{\partial t} \cdot (S^2) \leq -\eta \cdot |S| \quad (\text{IV.11})$$

Où η est une constante positive.

IV.2.3 Résultats de simulation

Pour effectuer l'opération de poursuite des trajectoires de référence imposée au système, c'est-à-dire que les actionneurs doivent suivre la trajectoire désirée, point par point, à des vitesses et des accélérations bien déterminées. La loi de commande par mode glissant a été appliquée au système en mode de poursuite de trajectoire. Le réglage des paramètres de la loi de commande glissante et de la surface de glissement sont réalisés de manière à garantir une convergence exacte et rapide. Après avoir réalisé notre loi de commande, nous allons procéder à une simulation pour exposer les différents résultats obtenus avec ce type de commande, ceci pour le mode de commande poursuite de trajectoire.

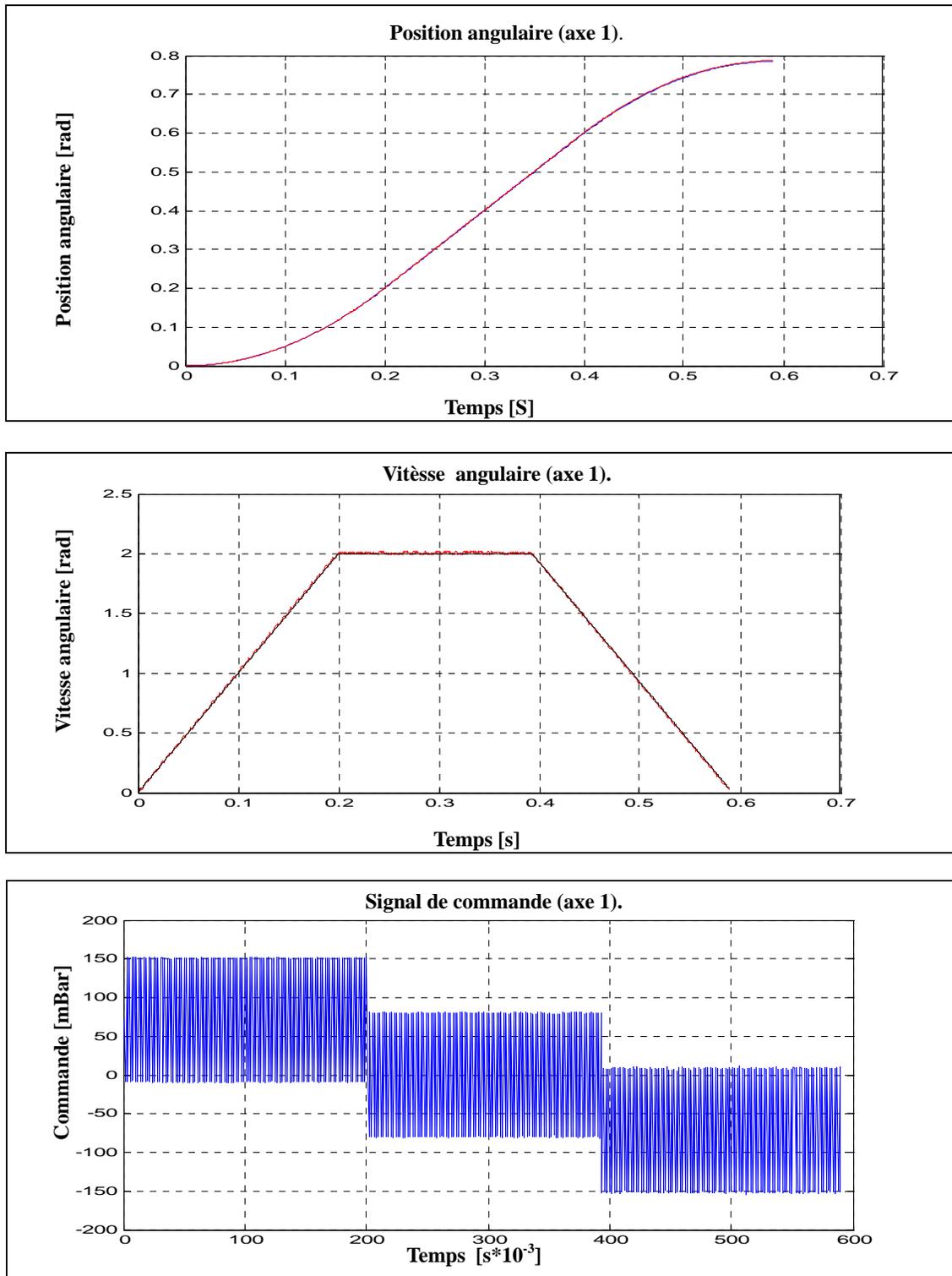


Figure IV.1. Commande par mode glissant classique de l'axe 1.

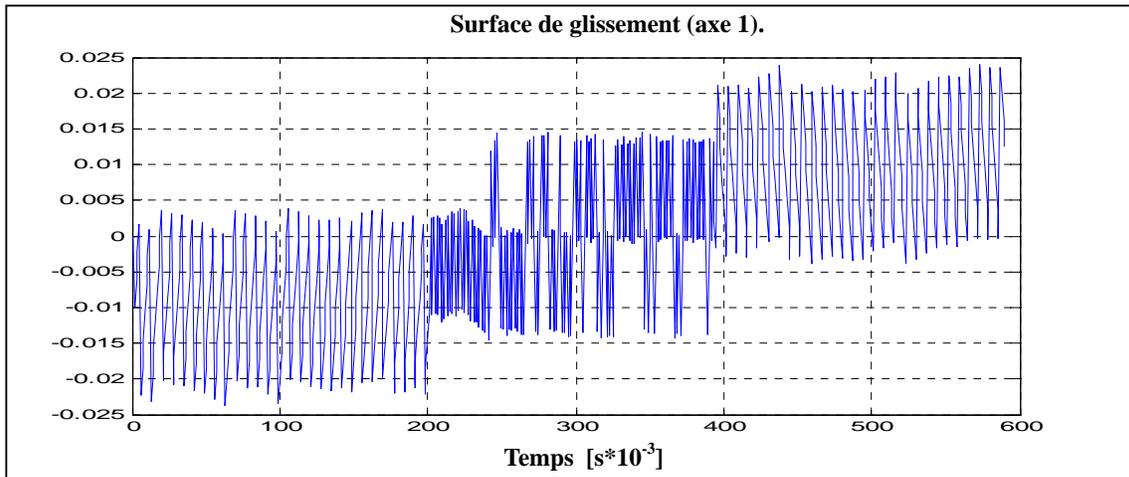


Figure IV.1. Commande par mode glissant classique de l'axe 1.

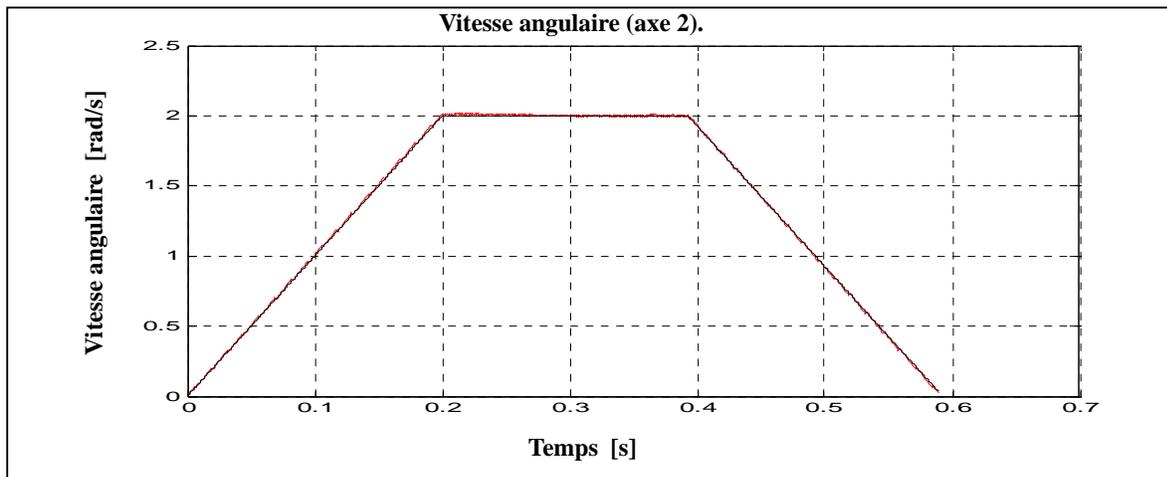
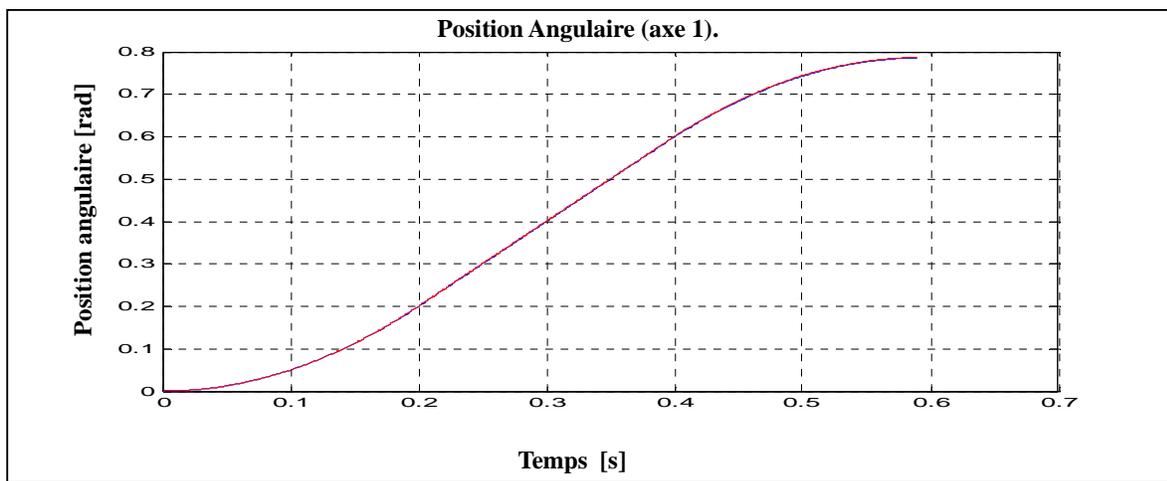


Figure IV.2 Commande par mode glissant classique de l'axe 2.

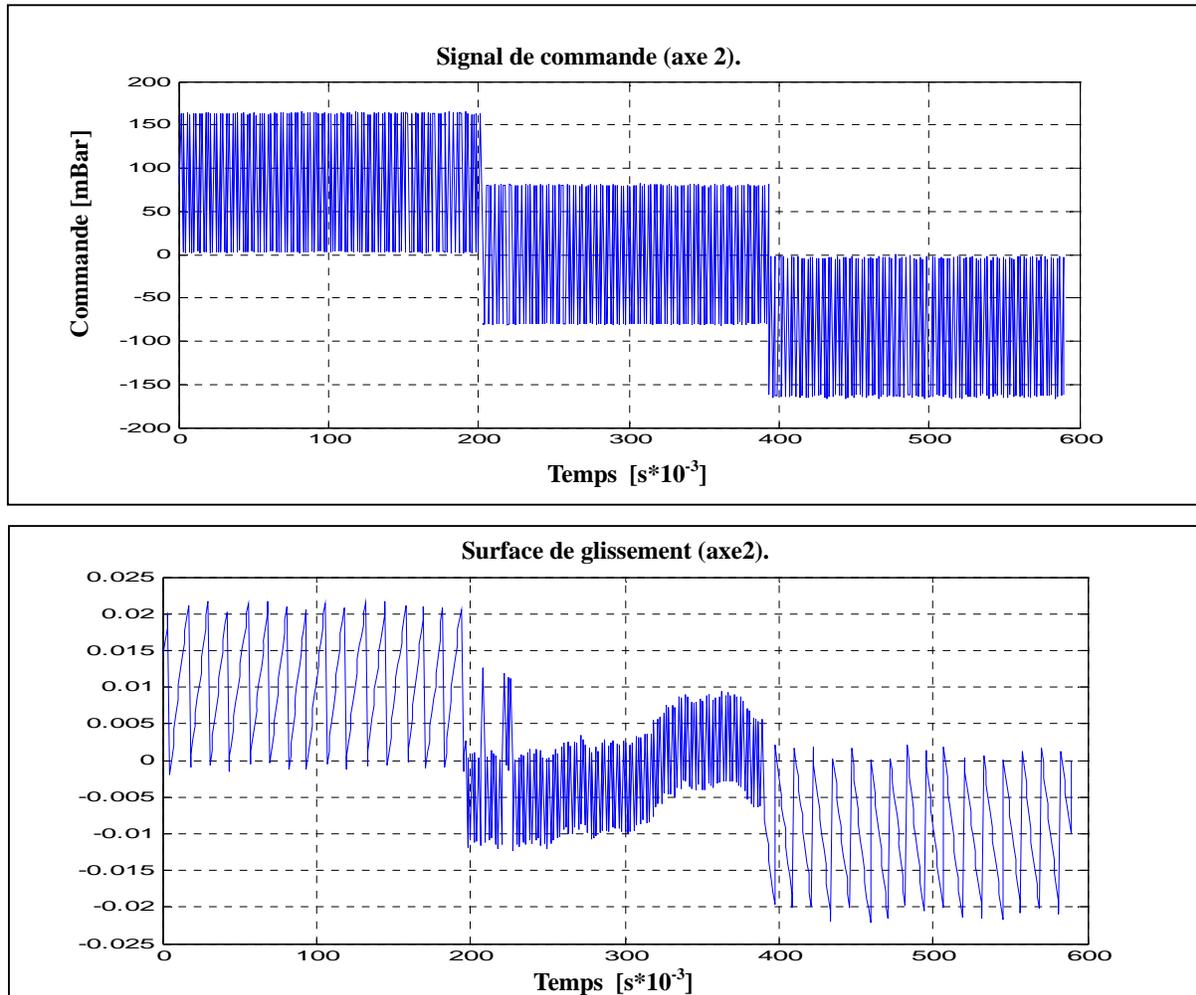


Figure IV.2 Commande par mode glissant classique de l'axe 2.

IV.2.4 Interprétation des résultats

Le bras du robot à muscles artificiels est approximé par un modèle linéaire de second ordre dont les paramètres sont déterminés par identification, la commande à structure variable par mode glissant développée a été appliquée sur chaque axe du robot. Afin d'effectuer le mode de poursuite de trajectoire, nous remarquons, dans les résultats, la poursuite convenable de trajectoire de position et de vitesse pour chaque axe, mais avec l'existence du phénomène de broutement qui apparaît dans les signaux de commande. Pour remédier à ce problème, plusieurs méthodes ont été développées. Une de ces méthodes est la commande par mode glissant flou qui sera développée dans la deuxième partie de ce chapitre.

IV.3 COMMANDE FLOUE-GLISSANTE

Pour remédier au problème de broutement de la commande par mode glissant d'une part, et au problème d'absence d'une méthode pour l'analyse de la stabilité dans la commande par la logique floue d'autre part, la commande "Floue-glissante" qui est apparue dans plusieurs travaux de recherche [17][35][42][44][52] combine les avantages de chacune de ces deux méthodes. Cette commande a montré ses qualités pour l'atténuation du broutement. Dans cette partie, nous proposons une méthode simple et élégante de conception d'une commande floue glissante. La logique floue est un formalisme permettant de construire une transformation entre l'espace d'entrée et l'espace de sortie à l'aide d'une certaine connaissance exprimée sous forme des règles floues. Cependant, il n'existe pas de principe général permettant de déterminer ces règles en fonction du problème. L'objectif de ce contrôleur flou est de remplacer le gain de commutation K par un gain issu d'un contrôleur flou nommé K_{fuzzy} .

IV.3.1 Définition des fonctions d'appartenance des variables d'entrée/ sortie

Les univers de discours des variables d'entrée (l'erreur e et sa variation de) du contrôleur étudié sont décomposés en sept (07) sous-ensembles flous pour chacune des deux (voir figure IV.3 et IV.4). Sur une partition dite « floue forte », cette dernière est simple et permet d'obtenir une base de règles faciles à interpréter. La seule variable floue de sortie est le gain d'hybridation K_{fuzzy} ; le nombre de ces fonctions d'appartenance est limité à trois (03). Les labels utilisés sont résumés dans le tableau (IV.1).

Erreur.	Variation de l'erreur.	Le gain d'hybridation(sortie).
NB : negative big ; NM : negative middle ; NS : negative small ; AZ : zero ; PS : positive small; PM : positive middle ; NB : positive big.	nb : negative big ; nm : negative middle ; ns : negative small ; ze : zero ; ps : positive small; pm : positive middle ; pb : positive big.	P : positive ; Z : zero; N : negative.

Tableau IV.1. les labels de K_{fuzzy} .

Les distributions des valeurs floues dans les différents univers de discours des entrées et de sortie sont données dans les figures IV.3, IV.4 et IV.5 :

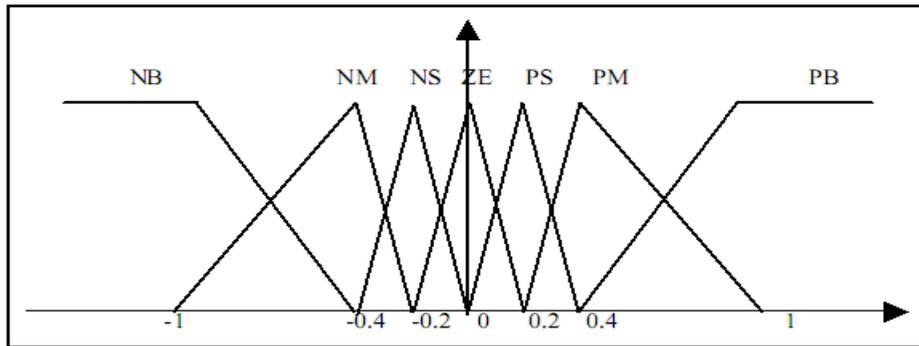


Figure IV.3 Les fonctions d'appartenance de l'erreur.

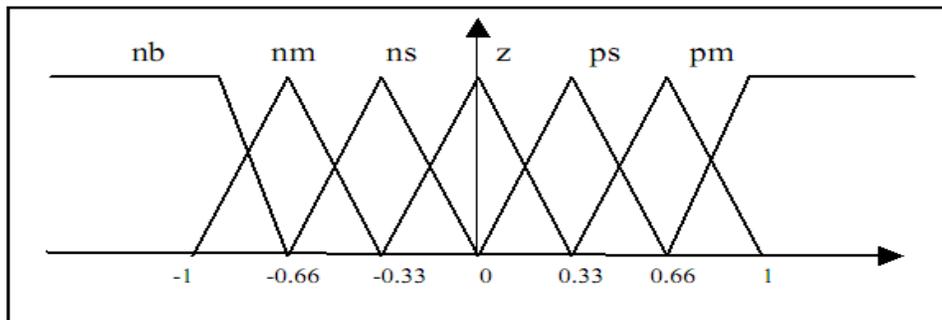


Figure IV.4 Les fonctions d'appartenance de la variation de l'erreur.

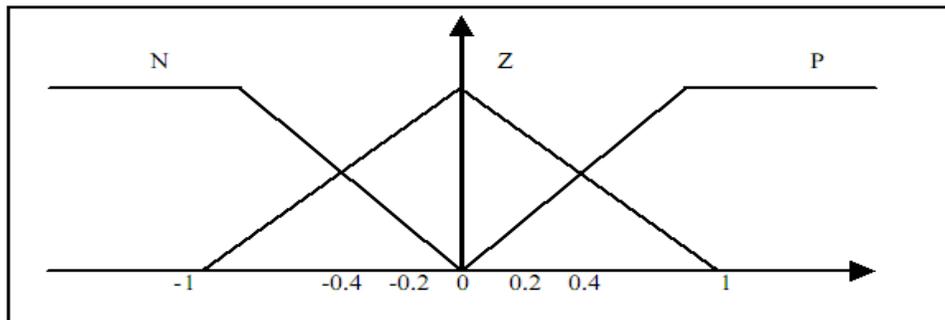


Figure IV.5 Les fonctions d'appartenance de la sortie k_{fuzzy} .

IV.3.2 Inférence floue

Nous utilisons un contrôleur flou de type « Mamdani ». L'étape d'inférence floue fait appel à des règles floues représentant un lien entre les différentes variables d'entrée (l'erreur e et sa dérivé de) et de la sortie k_{fuzzy} . Ces règles floues sont de la forme suivante :

if e is NM and de is PB then k_{fuzzy} is P .
 if e is NM and de is PS then k_{fuzzy} is N .

L'ensemble des combinaisons possibles des variables floues d'entrée est représenté dans le tableau d'inférence IV.2 :

$e \backslash de$	NB	NM	NS	ZE	PS	PM	PB
PB	P						
PM	N	P	P	P	P	P	P
PS	N	N	P	P	P	P	P
ZA	N	N	N	Z	P	P	P
NS	N	N	N	N	P	P	P
NM	N	N	N	N	N	P	P
NB	N	N	N	N	N	N	P

Tableau IV.2. Base des règles floues pour la détermination du gain K_{fuzzy} .

Finalement, la loi de commande par mode glissant-flou est donnée par l'expression (IV.11) :

$$u = u_{eq} + \Delta u_{eq} + \Delta u_{fuzzy} \quad (IV.11)$$

Avec $\Delta u_{fuzzy} = K_{fuzzy} \cdot sat(S)$

Avec K_{fuzzy} est le gain de commutation de la sortie du contrôleur flou proposé dans cette section.

IV.3.3 Résultats de simulation

La loi de commande par mode glissant flou a été appliquée au système en mode de poursuite de trajectoire, les paramètres du contrôleur flou et de la commande par mode glissant ont été choisis de manière à garantir une convergence rapide et exacte. Après avoir réalisé notre contrôleur flou-glissant, nous allons procéder à une simulation pour exposer les différents résultats obtenus avec ce type de contrôleur.

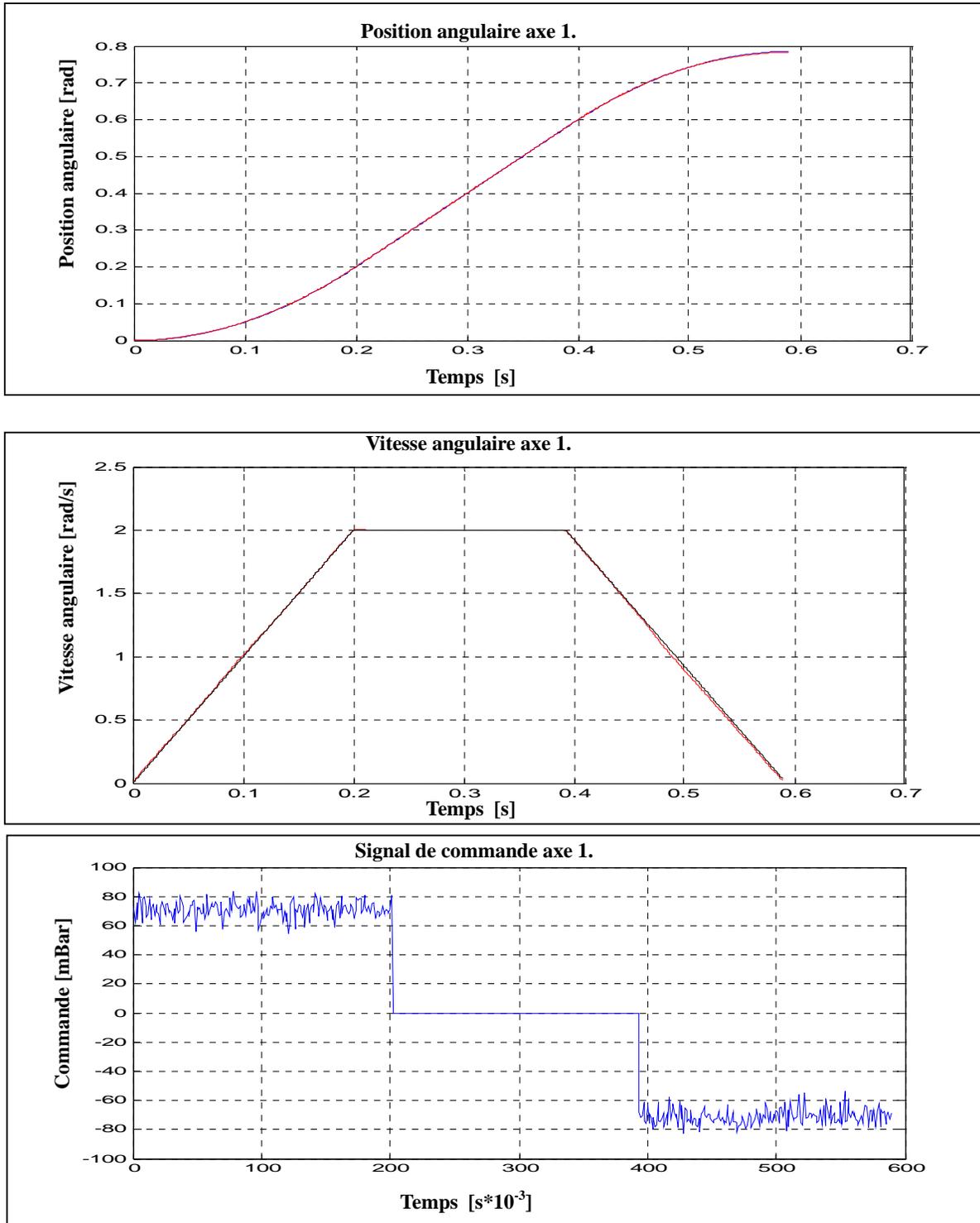


Figure IV.6 Commande par mode glissant flou appliquée sur les deux axes 1 et 2 du robot,

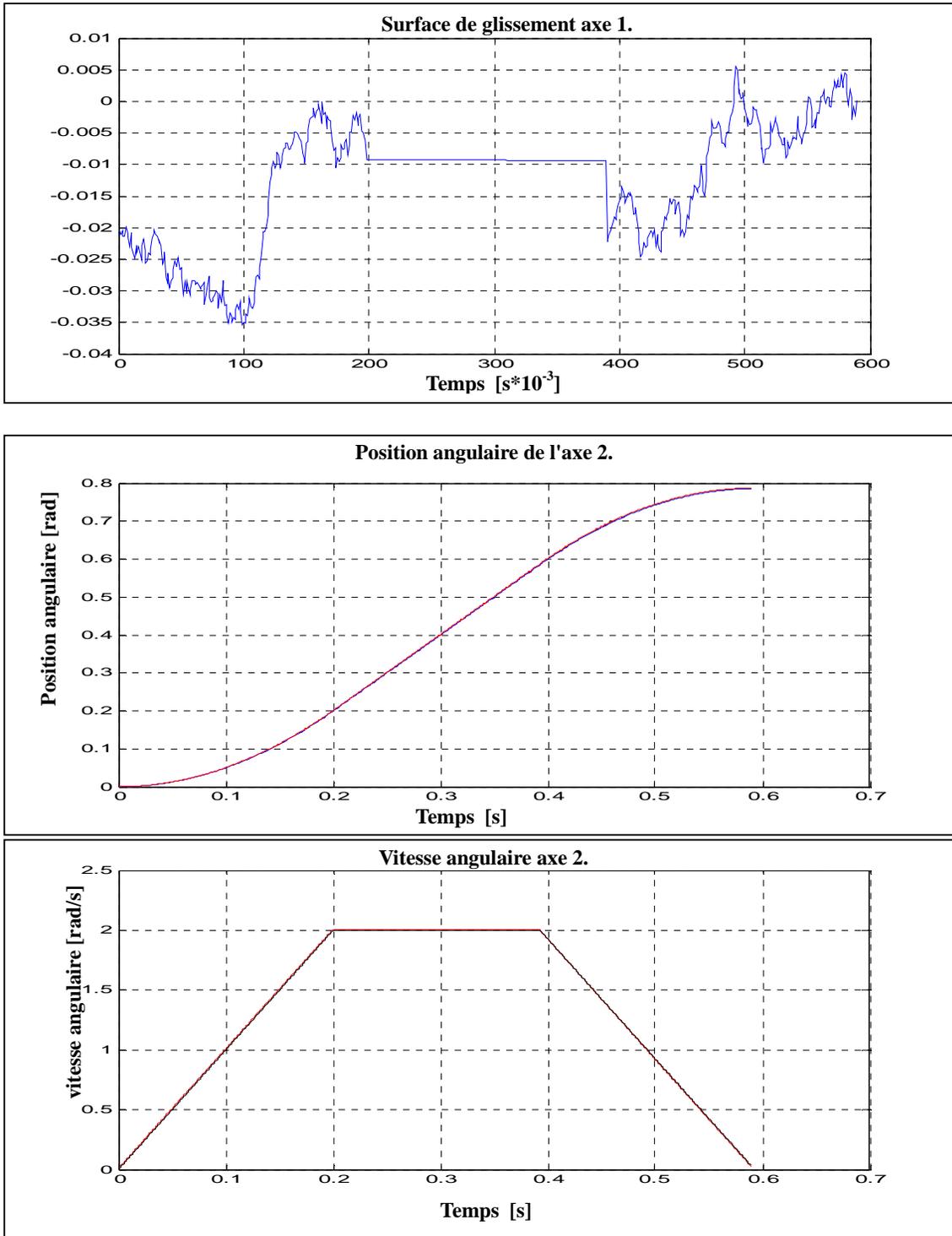


Figure IV.6 Commande par mode glissant flou appliquée sur les deux axes 1 et 2 du robot,

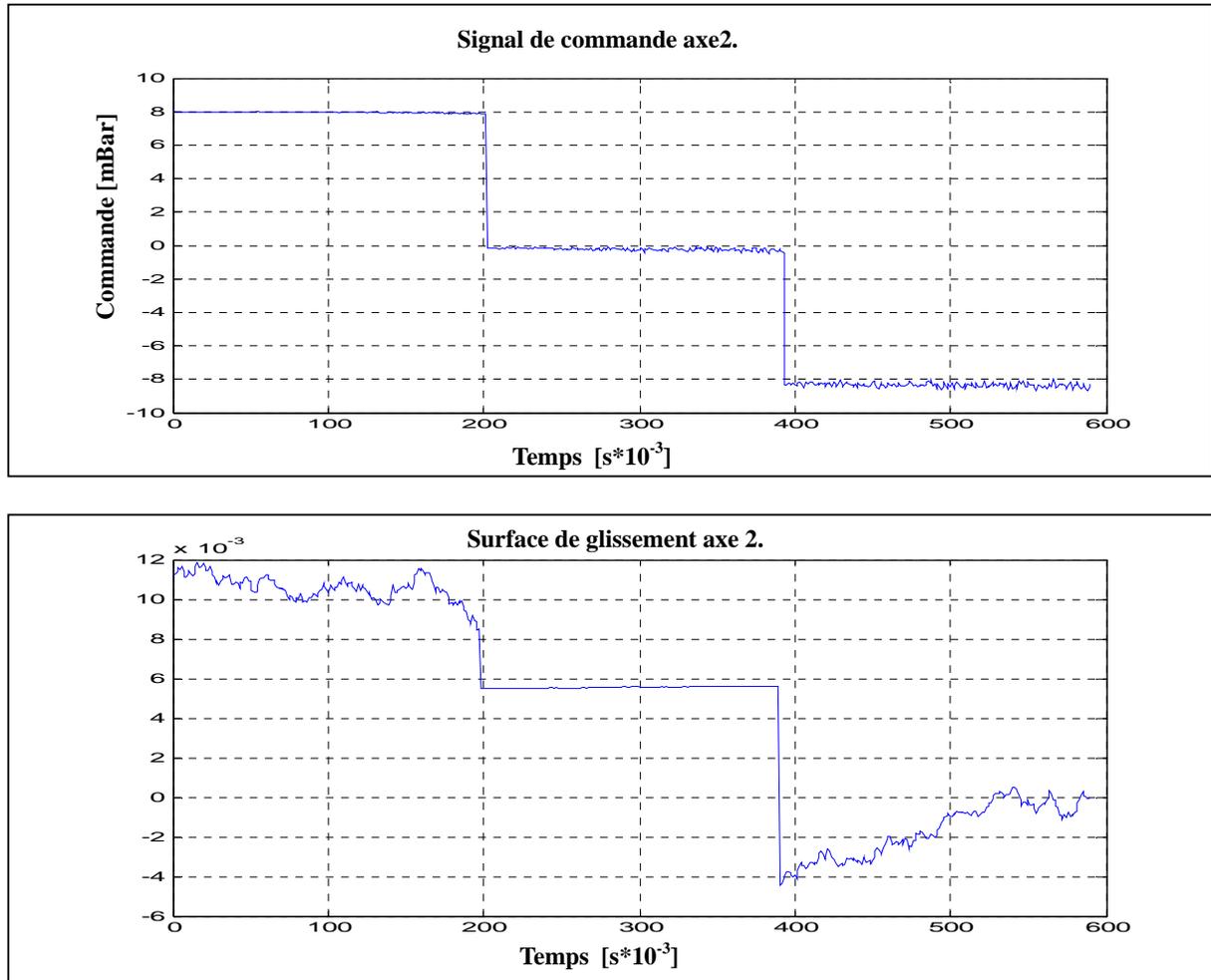


Figure IV.6 Commande par mode glissant flou appliquée sur les deux axes 1 et 2 du robot,

IV.3.4 Interprétation

Pour remédier au problème de broutement lié à la commande par mode de glissement classique, nous avons proposé une nouvelle commande basée sur l'hybridation des deux techniques de commande quantitative (mode glissant) et qualitative. Les résultats obtenus dans la commande floue prouvent l'efficacité de la solution proposée.

IV.4 CONCLUSION

Le travail exposé dans ce chapitre montre l'intérêt d'utiliser un contrôleur flou associé à la commande en régime glissant pour la réduction du broutement. L'efficacité d'un tel contrôleur réside dans son aptitude à atténuer le phénomène de broutement.

Les commandes proposées dans ce chapitre ont donné de bonnes performances. Cependant, elles présentent beaucoup de difficultés et d'inconvénients tels que :

- La commande par mode de glissement classique dépend du modèle du système à commander. Dans notre cas, le modèle du robot à MAPs est difficile à obtenir avec précision et le modèle issu de l'identification est insuffisant pour la commande du robot.
- La commande par mode de glissement classique ne garantit aucune convergence en temps fini.
- Dans le cas du contrôleur flou pour l'atténuation de broutement, nous avons utilisé un SIF de type Mamdani. Ce contrôleur donne plus de précision par rapport à celui de Takagi Sugéno d'ordre zéro. Cependant, ce contrôleur nécessite un temps de traitement relativement grand, ce qui est un inconvénient majeur pour la commande en temps réel.
- la commande par logique floue de type-1 est utilisée dans ce chapitre assure moins de performances comparée à la logique floue de type-2. une comparaison entre ces deux types de commande est résumée dans la section III.7.

Pour toutes ces raisons, dans les chapitres suivants nous proposerons plusieurs nouvelles stratégies de commande par logique floue de type-2, par terminal sliding mode flou, par réseau de neurones, etc.

CHAPITRE V. *Commande Floue De Type-2 Intervalle D'un Robot Manipulateur à Muscles Artificiels Pneumatiques.*

V.1 INTRODUCTION

Depuis sa première application en 1974 pour la commande d'un système à vapeur, la logique floue a connu un succès remarquable, notamment pour la commande des systèmes robotisés. Beaucoup d'applications ont montré la supériorité des résultats apportés par la commande floue de type-1. Cependant, cette technique est incapable de traiter la majorité des incertitudes existantes dans les systèmes commandés, ce qui a amené à la proposition de la logique floue de type-2. Dans ce chapitre une commande floue de type-2 adaptative a été proposée au début. Par la suite, une commande floue glissante basée sur la condition d'attractivité de mode de glissement a été introduite également.

V.2 COMMANDE FLOUE DE TYPE-2 INTERVALLE D'UN ROBOT MANIPULATEUR A MAPs

Dans cette partie, nous allons présenter un contrôleur de type-2 avec deux entrées et une sortie, les entrées sont l'erreur en position angulaire et sa dérivée pour chaque axe. Ces deux informations fournissent une description suffisamment détaillée sur la dynamique des états du système à commander. Dans ce cas, les règles floues associées à la précision et la vitesse de convergence sont formulées. La sortie consiste à la différence de pression qui doit être injectée aux MAPs de chaque axe. Les variables d'entrée / sortie sont définies par onze (11) valeurs floues dénommées: $\{Pe$ (Positive), Ze (Zero), et Ne (Négative)) pour l'erreur et $\{P\dot{e}$ (Positive), $Z\dot{e}$ (Zéro), et $N\dot{e}$ (Négative)) pour la dérivée de l'erreur. Les fonctions d'appartenance utilisées ainsi que leurs distributions sur les univers de discours sont présentées dans les figures V.1 et V.2 suivantes :

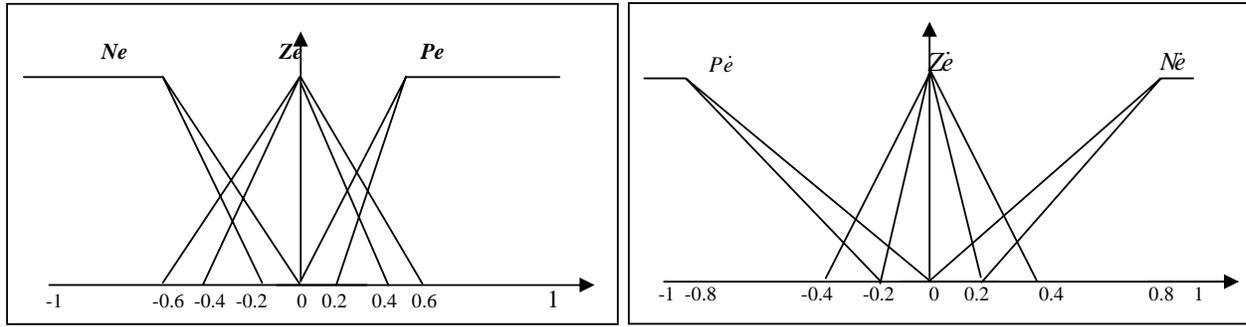


Figure .V.1. Les variables d'entrées (l'erreur et la variation de l'erreur sur les univers de discours).

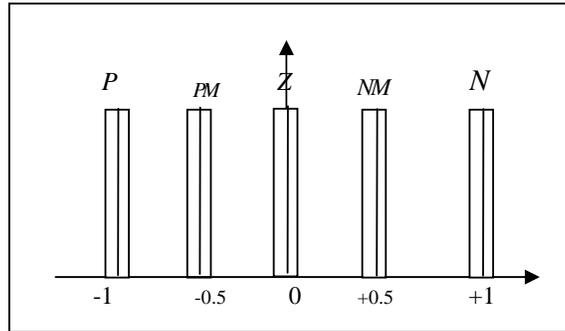


Figure .V.2. La distribution des fonctions d'appartenances de la sortie sur l'univers de discours.

La pression à injecter aux actionneurs du robot a été modélisée par des fonctions d'appartenance de type-1. L'objectif de l'utilisation de ces fonctions d'appartenance est de minimiser le temps de calcul lors de l'expérimentation. Cinq (05) labels sont utilisés et nommés : $\{P$ (Positive), PM (Positive moyen) Z (Zero), NM (Négative moyen), N (Négative)}. Le tableau V.1 représente les règles d'inférence utilisées:

e	N	Z	P
N	N	NM	Z
Z	NM	Z	PM
P	Z	PM	P

Tableau V.1: table de règles.

La réduction de type est l'étape essentielle dans les Systèmes d'inférence Floue de Type-2 (SIFT2), elle consiste à transformer les données de type-2 issues de l'inférence à des données facile à défuzzifier. Pour effectuer l'opération de la réduction de type, plusieurs solutions ont été proposées

dans la littérature [14][1][2]. La méthode la plus utilisée est celle de centre des ensembles. L'ensemble réduit est un intervalle caractérisé par ces deux extrémités y_l et y_r . Dans ce travail, nous avons utilisé la méthode du centroïde pour déterminer y_l et y_r .

V.3 COMMANDE ADAPTATIVE FLOUE DE TYPE-2 INTERVALLE POUR LE ROBOT MANIPULATEUR A MAPs

Dans la majorité des études réalisées dans la logique floue, le gain dit de « dénormalisation de la sortie de contrôleur flou » est choisi constant et de petite valeur afin d'éviter le risque de rapprochement à l'instabilité du système commandé. Le choix d'une petite valeur évite à l'utilisateur le problème d'instabilité, mais provoque en revanche une dégradation des performances, notamment sur le temps de réponse du système commandé [34]. Afin de résoudre ce problème et d'avoir un meilleur gain de sortie, un nombre assez réduit de travaux ont abordé le sujet. Dans [5], les auteurs ont utilisé un gain adaptatif basé sur la minimisation d'un critère quadratique. Ce dernier représente le carré de la surface de glissement terminal. L'objectif de la méthode est à la fois d'atténuer l'effet de broutement provoqué par la partie discontinue de la commande par terminal sliding mode et d'améliorer les performances de la commande par terminal sliding mode flou statique. Récemment, le gain de sortie d'un contrôleur flou de type-2 a été ajusté en ligne par un système d'inférence flou de type-1. L'objectif est d'avoir les meilleures performances possibles pour le contrôle d'un Bioréacteur. Les performances obtenues dépendent explicitement de la base des règles du contrôleur d'ajustement [25]. Dans ce travail, nous proposons une méthode d'adaptation des gains des sorties du contrôleur flou de type-2 pour chaque axe du robot. Ceci est réalisé par la recherche des meilleurs gains de sorties, par l'usage d'une recherche basée sur la méthode de descente du gradient. Cet algorithme d'optimisation a été utilisé dans plusieurs travaux dans le domaine de la commande floue. Ce que ces méthodes et la notre ont en commun, c'est qu'elles minimisent toutes un critère quadratique appelé l'erreur quadratique moyenne [34] dont l'équation est donnée comme suit :

$$E = \frac{1}{2} \varepsilon H \varepsilon^T = \frac{1}{2} \sum_{i=1}^2 e_i^2 \quad (\text{V.1}).$$

où $H = \text{diag}(1,1)$ est une matrice identité de dimension 2×2 .

$$\text{et } \varepsilon = Q_d - Q_m = \begin{bmatrix} q_{d1} - q_{m1} \\ q_{d2} - q_{m2} \end{bmatrix} = \begin{bmatrix} e1 \\ e2 \end{bmatrix},$$

Nous pouvons réécrire l'équation (I.7) comme suit :

$$\ddot{\varepsilon} + A_1 \cdot \dot{\varepsilon} + A_0 \cdot \varepsilon = BU \quad (\text{V.2})$$

La descente du gradient vise à minimiser la fonction (V.1) par la modification de ces paramètres via la loi suivante [5] :

$$d\alpha_i = -\eta_i \frac{dE}{d\alpha_i} \quad (\text{V.3}).$$

Où η_i est un gain à déterminer par tâtonnement au moment de l'expérimentation, i représente le nombre d'axes. Le gain de sortie adaptatif pour chaque axe est donné par le développement mathématique suivant :

$$\begin{aligned} d\alpha_i &= -\frac{1}{2} \eta_i \frac{dE}{d\alpha_i} \\ d\alpha_i &= -\frac{1}{2} \eta_i \frac{de_i^2}{d\alpha_i} \\ d\alpha_i &= -\frac{1}{2} \eta_i \frac{de_i^2}{de_i} \cdot \frac{de_i}{d\alpha_i} \\ d\alpha_i &= -\eta_i \cdot e_i \cdot \frac{d(q_{mi} - q_{di})}{d\alpha_i} \\ d\alpha_i &= -\eta_i \cdot e_i \cdot \frac{1}{a_{i0}} \left(\frac{d(-\ddot{e}_i - a_{i1}\ddot{e}_i + \alpha_i \cdot b_i \cdot u_i)}{d\alpha_i} \right) \\ d\alpha_i &= -\frac{1}{a_{i0}} \eta_i \cdot e_i \cdot b_i \cdot u_i \\ d\alpha_i &= -\eta_i' \cdot e_i \cdot u_i \end{aligned} \quad (\text{V.4})$$

Où $\eta_i' = \frac{1}{a_{i0}} \eta_i \cdot b_i$ and $i = 1, 2$

Le schéma synoptique de la commande développée est donné par la figure V.3 :

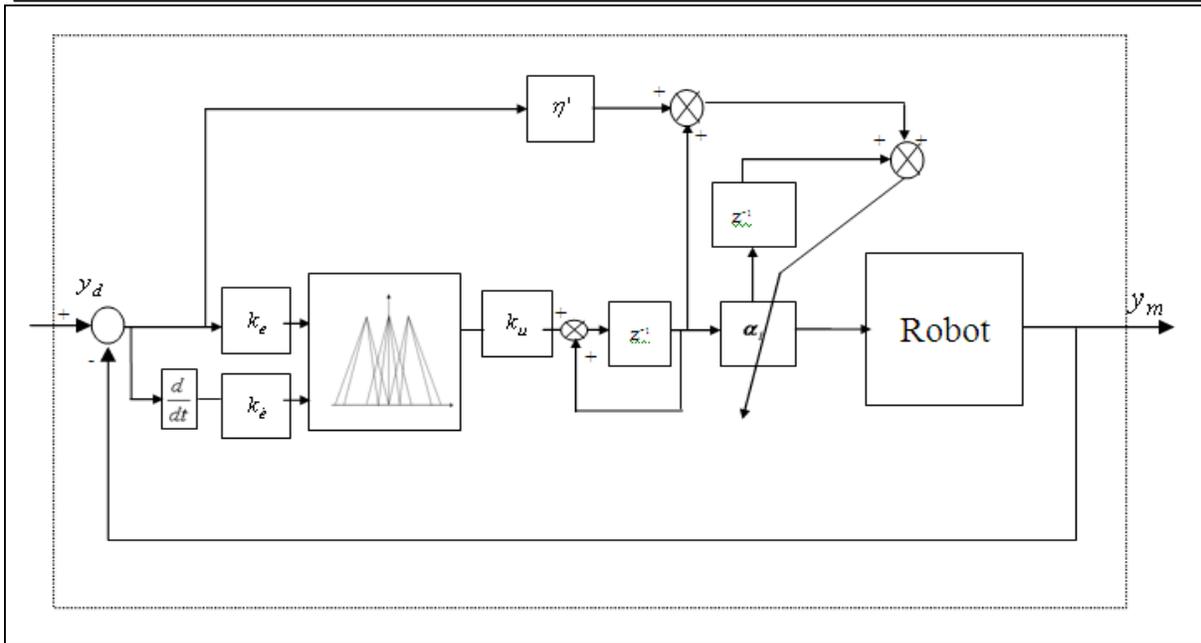


Figure .V.3. Commande floue de type-2 adaptative du robot manipulateur à MAPs.

V.4 RESULTATS EXPERIMENTAUX

En vue de tester l'efficacité de la commande floue de type-2 adaptative proposée, nous présentons dans cette section les résultats expérimentaux obtenus sur le robot à MAPs du CDTA. La commande a été examinée en mode régulation, où l'objectif est de déplacer le robot d'une position initiale verticale de $(q_1, q_2) = (0^\circ, 0^\circ)$ à une position désirée de $(q_{d1}, q_{d2}) = (14^\circ, 10^\circ)$. Le contrôleur a été appliqué avec une période d'échantillonnage de 10 ms. Nous avons choisi de représenter les courbes des réponses en position angulaire, les signaux de commande ainsi que l'évolution des gains adaptatifs pour les deux axes du robot. A titre de comparaison et afin de prouver l'amélioration apportée par l'approche adaptative proposée, nous l'avons comparé avec le cas classique, c'est-à-dire qu'on garde fixe le gain initial du sortie déterminé arbitrairement. Parce que la commande a été synthétisée en mode monovariante, deux expérimentations ont été réalisées. Nous avons testé les commandes en mode découplé c'est-à-dire que chaque axe du robot est commandé indépendamment de l'autre. Dans une seconde expérience, le robot a été commandé en mode multivariable où les deux axes sont commandés simultanément. La commande par logique floue dépend de l'expertise et ne nécessite aucune modélisation du robot à commander. À cet effet, nous avons effectué ces expériences en nous basant sur notre expérience sur le robot réel du CDTA. Il faut souligner que le robot est stable en boucle ouverte, ce qui nous a permis de faire des tests en boucle ouverte et d'acquérir une riche expertise sur le comportement du robot. Les paramètres utilisés pour les deux contrôleurs dans le cas mono et multivariable sont les mêmes pour les deux axes et ils sont résumés dans le tableau (V.2). Par la suite, une étude comparative qualitativement et quantitativement sera

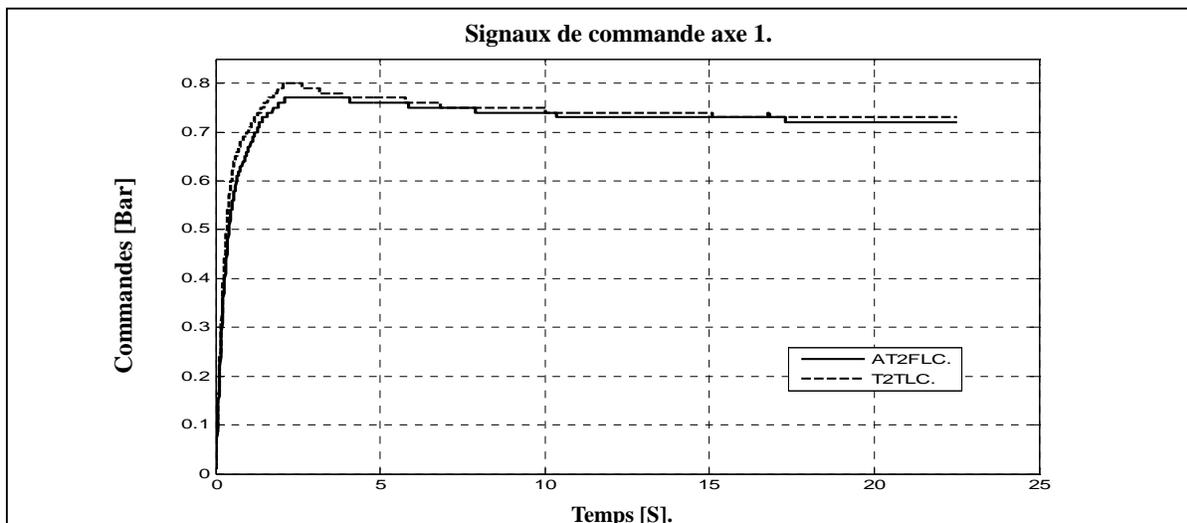
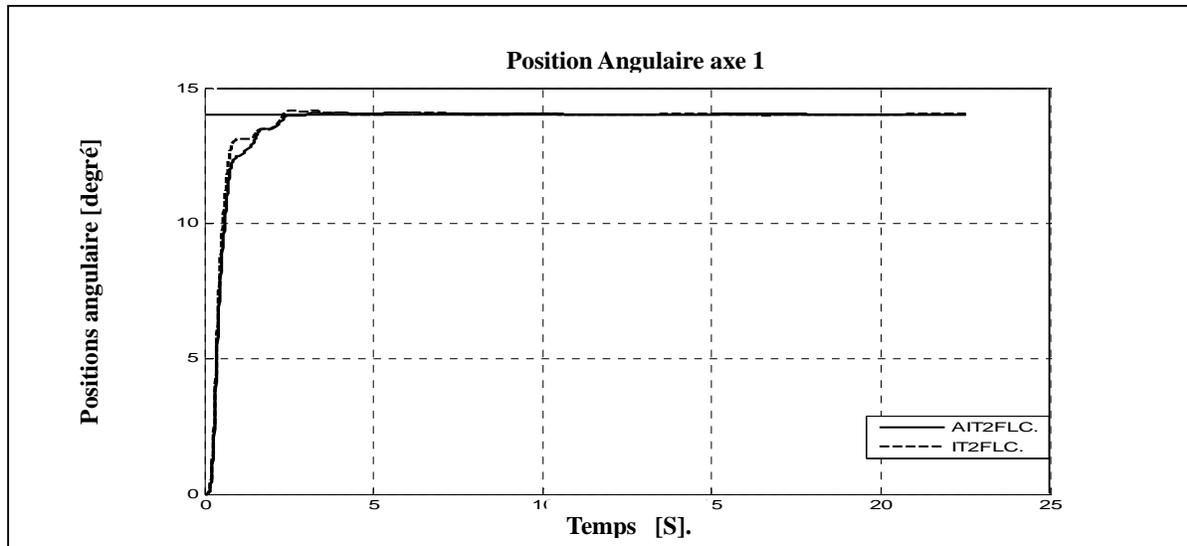
présentée pour les deux commandes floues de type-2 et floue de type-2 adaptative:

<i>Paramaters</i>	k_e	k^e	k_u	η_i'
Joint 1	0.1	0.03	30	8.944
Joint 2	0.02	0.002	55	3.571

Tableaux V.2 Paramètres d'expérimentation

V.4.1 Résultats Expérimentaux, Cas Découplé

Les figures (V.4) et (V.5) représentent les résultats expérimentaux obtenus après test sur le robot en mode découplé. Nous représentons la réponse en position angulaire, les signaux de commande ainsi que le gain de sortie de la commande floue après adaptation. Nous avons testé ces algorithmes pour un temps suffisamment important égal à 22.5 secondes dans toutes les expérimentations.



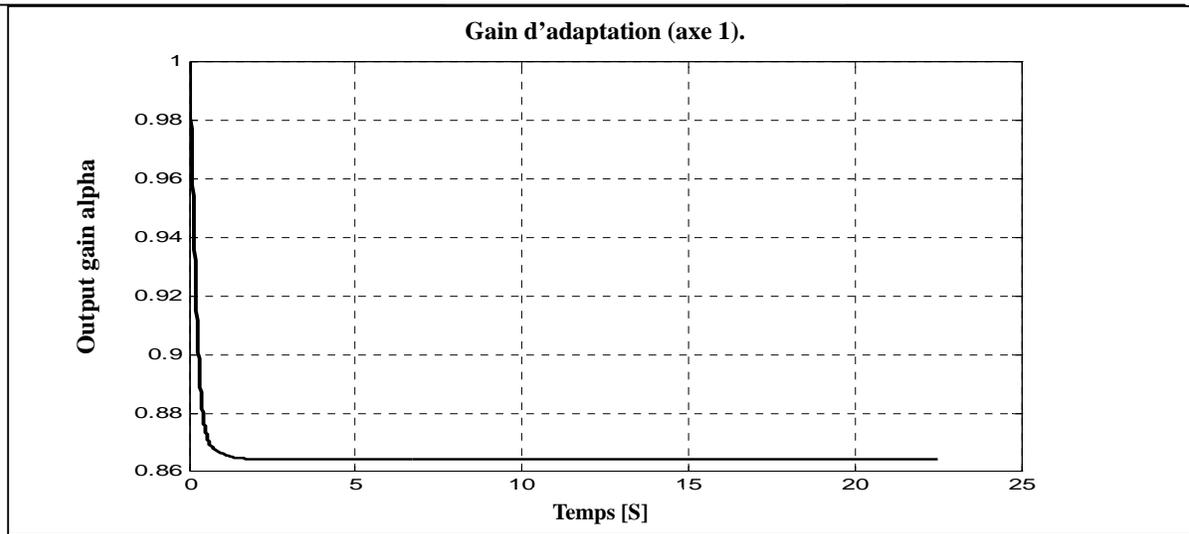


Figure .V.4. Résultats expérimentaux de la commande par logique floue statique et adaptative pour l'axe 1 (a) Position angulaire, (b) Signal de commande, (c) Gain adaptatif.

Comme dans le cas de l'axe 1, les figures V.5.a-c représentent les résultats obtenus après application de la commande floue de type-2 et celle adaptative :

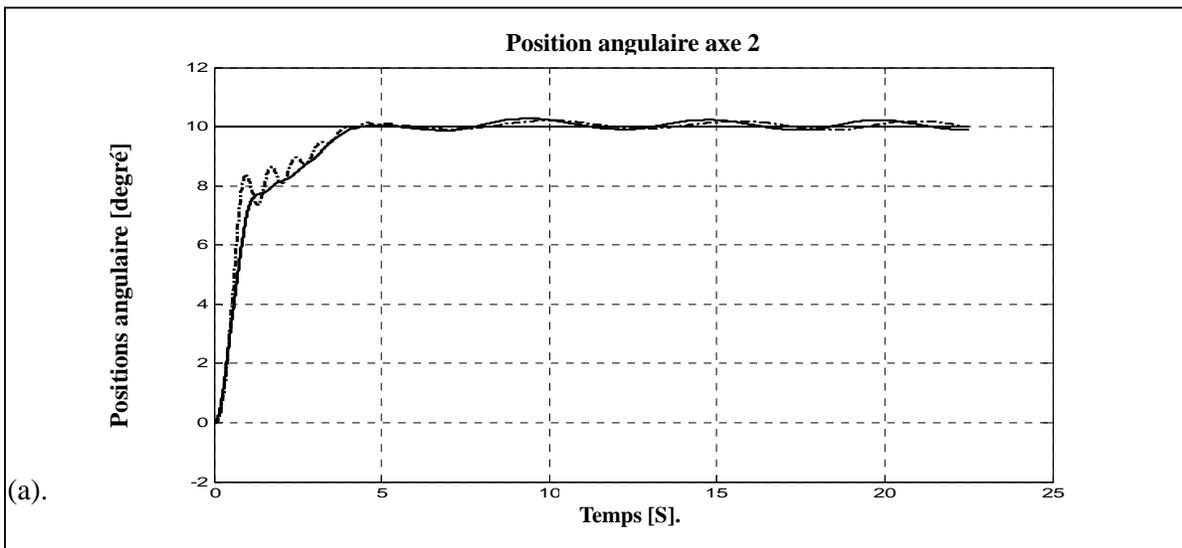


Figure .V.5. Résultats expérimentaux de la commande par logique floue statique et adaptative pour l'axe 2 (a) Position angulaire, (b) Signal de commande, (c) Gain adaptatif.

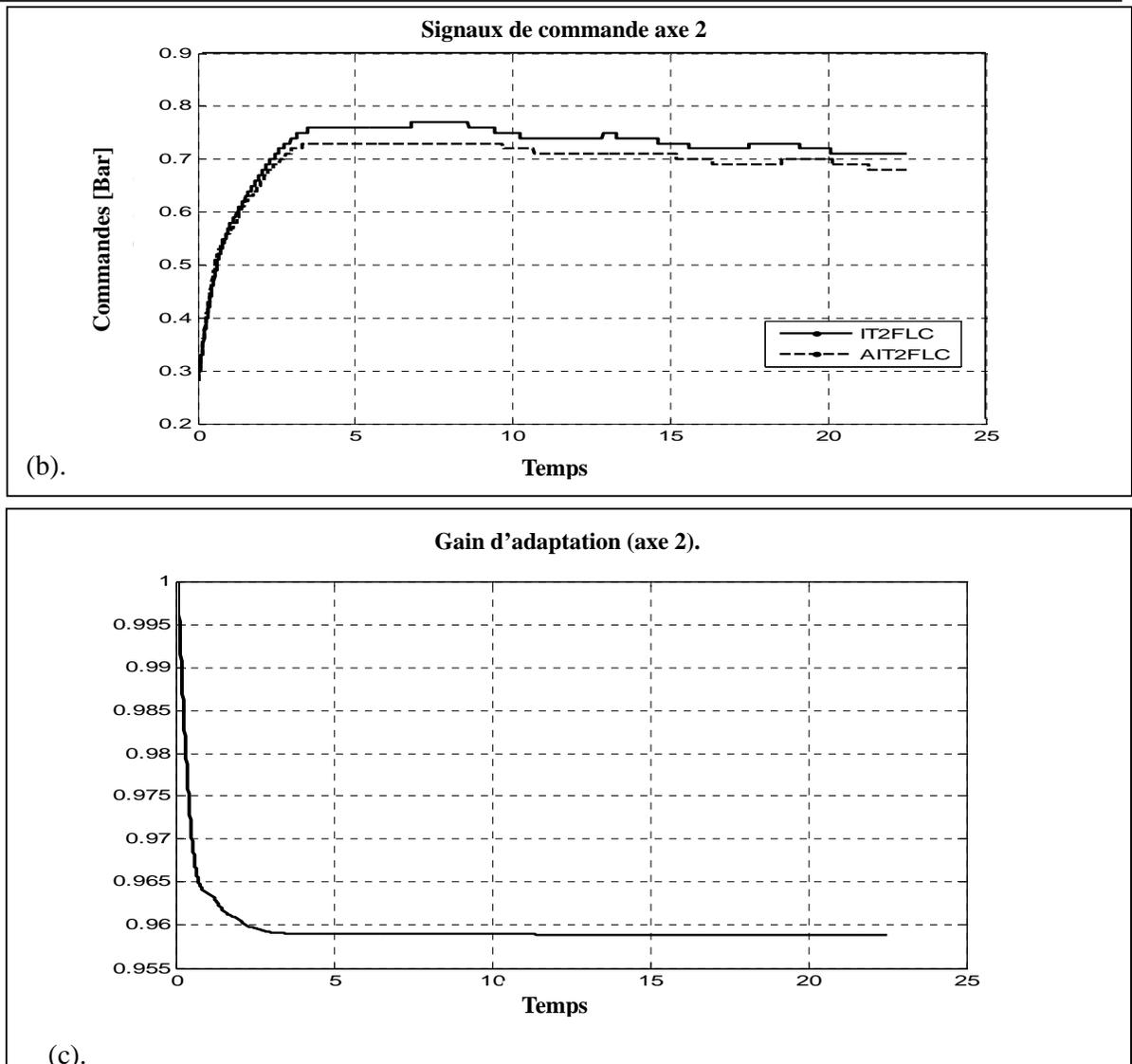


Figure .V.5. Résultats expérimentaux de la commande par logique floue statique et adaptative pour l'axe 2 (a) Position angulaire, (b) Signal de commande, (c) Gain adaptatif.

A partir des résultats expérimentaux exprimés pour les deux axes du robot (figure V.4 et V.5), nous pouvons souligner le suivi parfait de la consigne appliquée à une vitesse élevée pour l'axe 1. Le signal de commande est quasiment lisse. L'erreur en régime permanent est quasiment nulle (de l'ordre de 0.05°), ce qui montre la stabilisation parfaite de cet axe à son angle désiré. Comme nous l'avons mentionné, la commande de l'axe 2 est plus difficile comparée avec l'axe 1. Malgré ces difficultés, nous avons pu commander cet axe avec une vitesse relativement élevée, l'erreur en régime permanent est négligeable (de l'ordre de 0.3°). La commande adaptative a montré sa nette amélioration, notamment pour le cas de deuxième axe où nous remarquons l'élimination des oscillations en régime transitoire. Pour tous les résultats, le retard pur a été considérablement

diminué, voire éliminé pour l'axe 1. Le gain d'adaptation commence au début par une valeur égale à 1 et par la suite a été diminuée d'une manière exponentielle, il se stabilise à 0.85 pour l'axe1 et 0.958 pour l'axe 2, ce comportement est plutôt logique à cause de la relation inversé entre la réponse de système et la quantité de l'énergie à utiliser pour arriver au point désiré et s'y maintenir. De plus, la diminution du gain adaptatif permet de trouver le meilleur compromis entre le temps de montée et le dépassement vu la réponse de l'axe 1.

V.4.2 Commande Simultanée

Dans l'expérience précédente, nous avons présenté les résultats la commande du robot en mode axe par axe, en réalité, le robot est un système multivariable et couplé. Dans le but de tester la robustesse de la commande développée par rapport au problème de couplage considéré négligeable dans ce travail, une commande simultanée a été adoptée dans cette section. Dans ce cas, nous avons injecté les contrôleurs flous aux deux axes du robot simultanément. Afin d'optimiser nos ressources informatiques, nous avons choisi d'utiliser le même contrôleur flou pour les deux axes avec l'emploi des paramètres entrées/sortie adéquats pour chaque axe. Les figures (V.7), (V.8) et (V.9) représentent les résultats pour le cas de la commande floue de type-2 adaptative et son extension statique. Les gains pour les deux axes ont été choisis égaux à 1, l'expérience a été réalisée pour un temps égal à 22.5 secondes pour une période d'échantillonnage égale à 10ms.

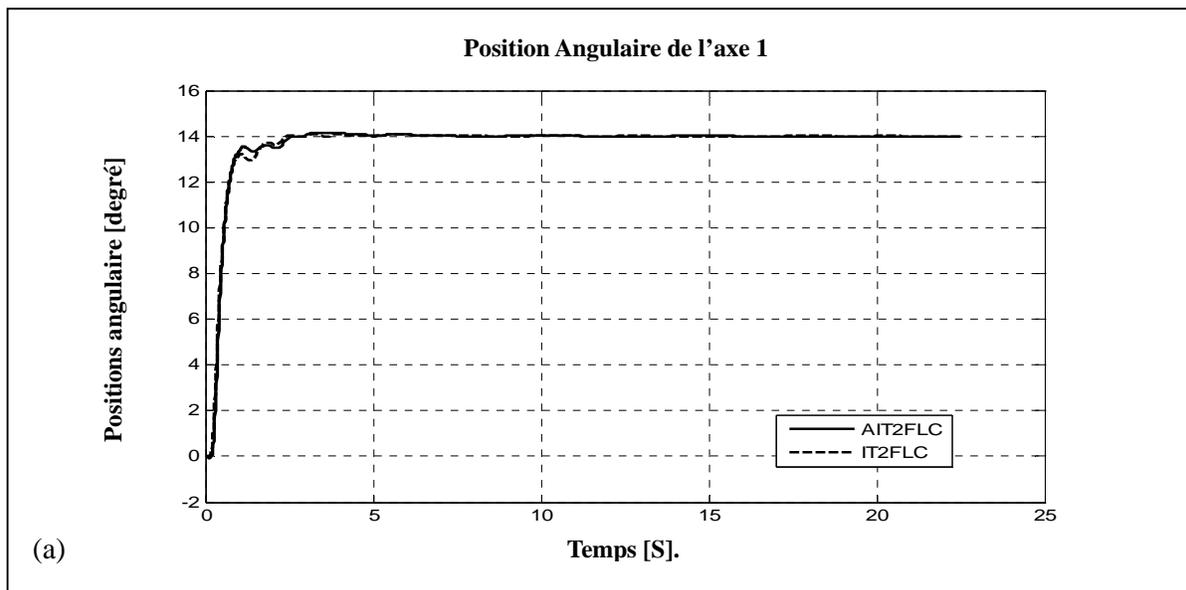


Figure .V.6. Réponses simultanées (a) axe 1 (b) axe 2.

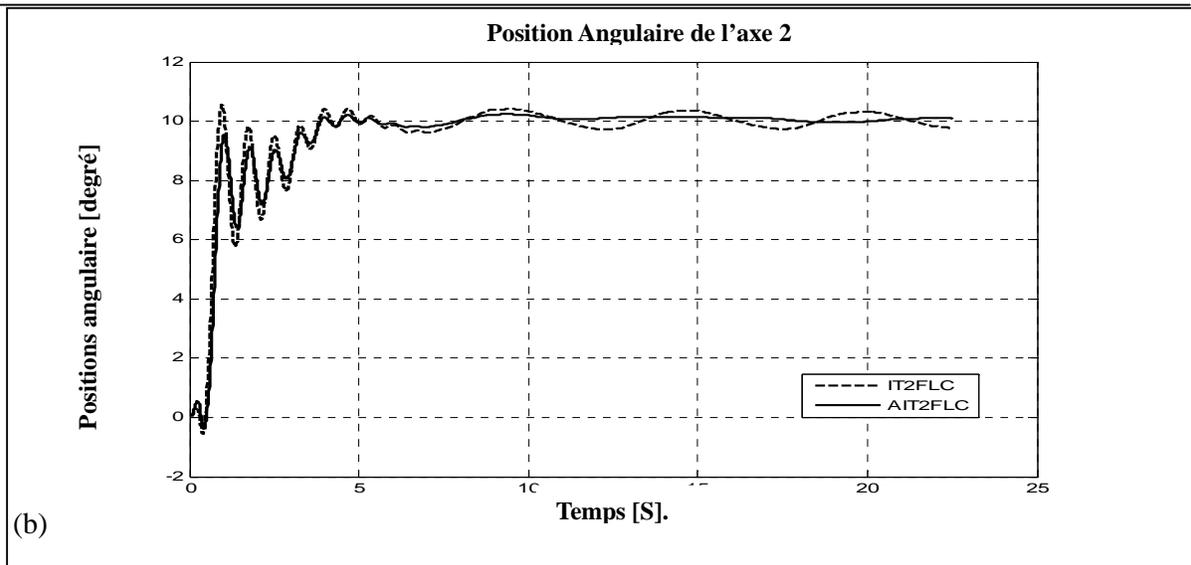


Figure .V.6. Réponses simultanées (a) axe 1 (b) axe 2.

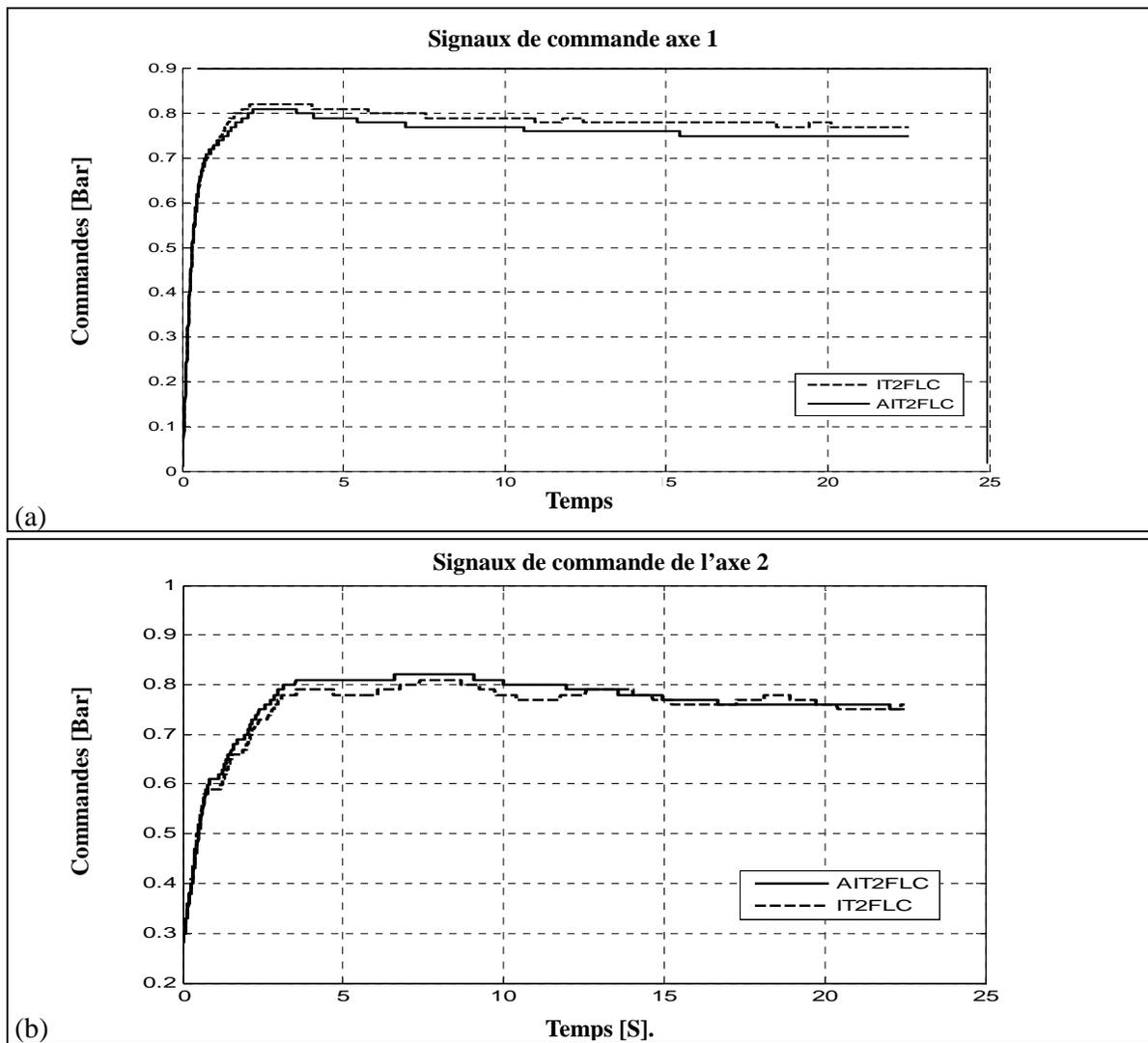


Figure .V.7. Signaux de commande cas simultané (a) axe 1 (b) axe 2.

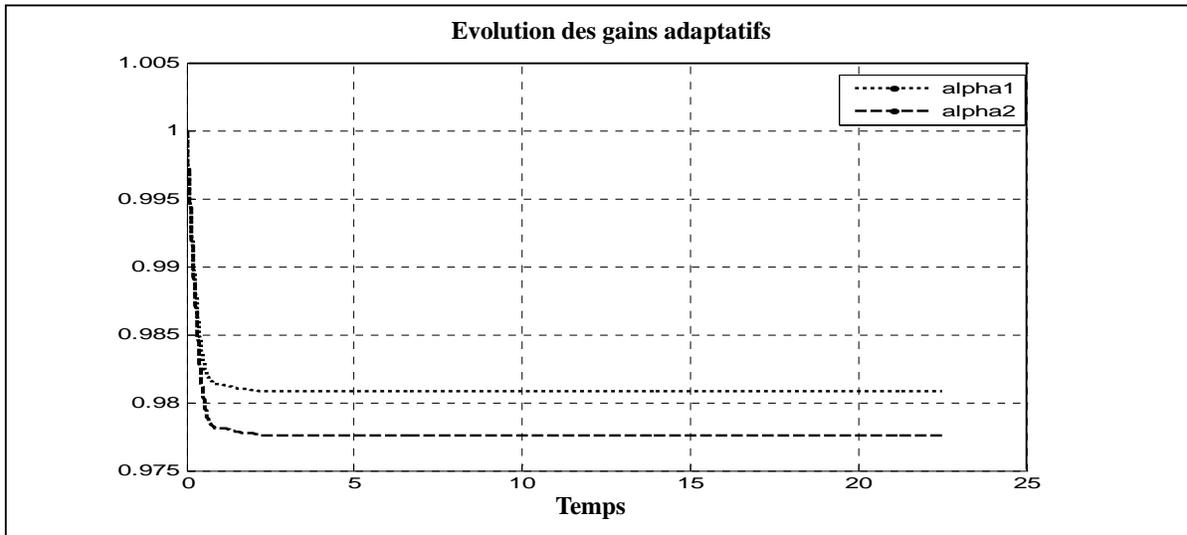


Figure V.8. Variations des gains d'ajustement de la commande AIT2FLC.

Les résultats de la commande simultanée des axes du robot ont confirmé que la commande par logique floue de type-2 adaptative apporte une nette amélioration de la réponse de robot. Cette amélioration est particulièrement visible dans la réponse de l'axe 2, où une commande lisse et une réponse stable ont été obtenues avec un minimum d'erreurs en régime dynamique et statique (enivrante 0.2°), le retard pur observé dans la réponse de l'axe 2 est causé par l'effet de couplage, le temps de montée a été amélioré dans le cas de la commande adaptative ainsi que pour le dépassement.

V.4.3 Analyse des Performances

On veut étudier quantitativement les performances des contrôleurs flou de type-2 et flou de type-2 adaptatif dans le cas de la commande axe par axe et celle simultanée, nous avons évalué plusieurs critères de performance tels que : l'Intégral Quadratique de la Commande (*IVS*). Ce critère a été choisi pour évaluer la quantité d'énergie consommée par chaque axe et pour chaque commande appliquée sur le robot. Le deuxième critère adopté est l'Intégrale de l'Erreur Absolue (*IAE*). Ce critère est pour but de quantifier et comparer les erreurs que ce soient en régime transitoire ou permanent pour chaque axe commandé. Le troisième critère est l'Intégrale de l'Erreur Absolue Multiplié par le Temps Instantané (*ITAE*). Ce dernier est pour voir beaucoup plus l'élimination de l'erreur statique (erreur en régime permanent). Ces critères sont donnés par les équations suivantes :

1/ l'intégrale quadratique de la commande (*IVS*):

$$IVS = \int_0^{\infty} u^2 dt \quad (V.5)$$

2/ l'intégrale de la valeur absolue de l'erreur (*IAE*):

$$IAE = \int_0^{\infty} |e| dt \quad (V.6)$$

3/ l'intégrale de la valeur absolue de l'erreur multipliée par le temps instantané (*ITAE*):

$$ITAE = \int_0^{\infty} t |e| dt \quad (V.7)$$

Les résultats obtenus après calcul de ces critères sont résumés dans le tableau V.3 :

	<i>Performences</i>		<i>ISV</i>	<i>IAE</i>	<i>ITAE</i>
<i>SISO</i> <i>Control</i>	<i>IT2FLC</i>	Joint	1192.9	801.41	815.97
	<i>AIT2FLC</i>	1	1229.7	859.024	768.49
	<i>IT2FLC</i>	Joint	1482.3	1264.5	6079.2
	<i>AIT2FLC</i>	2	1178.5	1325.5	4061.6
<i>MIMO</i> <i>Control</i>	<i>IT2FLC</i>	Joint	1367.6	806.07	794.87
	<i>AIT2FLC</i>	1	1295.5	817.47	677.31
	<i>IT2FLC</i>	Joint	1296.7	1516.8	6185.4
	<i>AIT2FLC</i>	2	1334.2	1388.7	3682.8

Tableau V.3 Résultat des performances quantitatives

Les performances qualitatives de ces deux contrôleurs sont données dans la table 3 dans le cas couplé et découplé, nous pouvons constater que le cas adaptatif a apporté des améliorations des performances dans le cas couplé et découplé, particulièrement dans le cas de (ITAE). La technique développée permet d'améliorer l'erreur en régime permanent.

V.5 COMMANDE PAR LOGIQUE FLOUE BASEE SUR LES MODES DE GLISSEMENT

Le bras de robot du CDTA peut être représenté dans l'espace d'état linéaire multivariable comme suit :

$$\dot{X}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -7.8573 & -5.6654 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -16.2075 & -2.9595 \end{bmatrix} X + \begin{bmatrix} 0 & 0 \\ 15.3217 & 0 \\ 0 & 0 \\ 0 & 2.0194 \end{bmatrix} U(t)$$

$$Y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} X \quad (V.8)$$

Il est très important de noter ici que le modèle de l'équation (V.8) représente partiellement le robot réel, en revanche, cette linéarisation est utilisée comme une étape transitoire pour, premièrement synthétiser le contrôleur souhaité, et par la suite, effectuer des simulations permettant de juger la qualité des performances réalisées et ainsi que le garent de la robustesse. Cette linéarisation est pour nous très utile, surtout avec l'utilisation de la commande par mode de glissement qui est une technique connue par sa robustesse vis-à-vis des erreurs de modélisation du système à commander

V.5.1 Conception De Contrôleur Flou De Type-2 Glissant

Nous avons divisé cette section en deux parties. Dans la première, nous allons présenter l'approche de la commande par mode de glissement d'un système linéaire multivariable. La deuxième partie sera consacrée à la présentation de la technique de commande par logique floue de type-2 basée sur les conditions d'attractivité de mode de glissement. Cette technique sera appliquée sur le robot à muscle du CDTA, et comparée au cas de la commande par logique floue de type-2.

V.5.2 Commande Par Mode De Glissement Cas Multi Variables Linéaire

Une régulation en mode glissant est un cas particulier très populaire de la commande à structure variable (VSC). Malgré que la commande par mode de glissement classique est une commande standard, sa synthèse dépende la dynamique du système à commander. Pour le cas du robot du CDTA, sous des hypothèses, nous pouvons le considérer comme un système MIMO décrit dans l'espace d'état par l'équation générale (V.9) [68] :

$$\dot{X}(t) = F(X) + BU(t) \quad (\text{V.9})$$

où $X(t)$ est un vecteur des variables d'état, $F(X)$ est un vecteur de fonctions linéaires et/ou nonlinéaires du système, B est la matrice des entrées et $U(t)$ est le vecteur de commande. Comme dans le cas SISO, la détermination de la commande par mode de glissement requiert deux principales étapes. Le choix de la surface de glissement et la synthèse de la loi de commande.

V.5.3 La surface de glissement dans le cas MIMO

La surface de glissement pour la commande du système MIMO (V.9) est peut être donnée par la forme générale (V.10) [68]:

$$S(X, t) = G(X^d(t) - X(t)) = \phi(t) - S_a(X) \quad (\text{V.10})$$

Où $\phi(t) = GX^d(t)$, $S_a(t) = GX(t)$. et G est des matrices diagonales.

La surface de glissement linéaire multivariable est donnée par l'hyperplan suivant :

$$S(X, t) = \begin{bmatrix} S_1 \\ \vdots \\ S_n \end{bmatrix} = \begin{bmatrix} e_1 + \lambda_1 \cdot \dot{e}_1 \\ \vdots \\ e_n + \lambda_n \cdot \dot{e}_n \end{bmatrix} \quad (\text{V.11})$$

Où λ_i sont des constantes positives et $i = 1 \dots n$.

Une fois la fonction de glissement est désignée, le problème de poursuite nécessite la conception d'une loi de commande dont l'objectif est de maintenir le vecteur d'état $X(t)$ sur la surface de glissement. Dans le cas $S(X, t) = 0$ pour seulement $t \geq 0$ une commande souhaitable U sera conçue avec l'objet d'atténuer au maximum l'erreur entre la dynamique du système et la dynamique de la surface de glissement. Pour achever cette commande, la fonction candidate de Lyapunov $V \geq 0$ est définie comme suite :

$$V(s) = \frac{S^T S}{2} \quad (\text{V.12})$$

La condition suffisante pour la stabilité de système est donnée par :

$$\frac{dV(s)}{dt} = -S^T D \text{sign}(S) \leq 0 \quad (\text{V.13})$$

Où $D = \text{diag}(d_{ii})$ est une matrice diagonale de gains définie positive.

V.5.4 Conception de la loi de commande

La commande par mode de glissement multivariable se compose par deux termes tels que la commande équivalente et la commande discontinue :

$$U(t) = U_{eq}(t) + U_s(t) \quad (\text{V.14})$$

$U_{eq}(t)$ est le terme équivalent calculé à partir du modèle du système quand $\dot{S} = 0$ tel que :

$$U_{eq}(t) = -(GB)^{-1} \left(GF(X) - \frac{d\phi}{dt} \right) \quad (\text{V.15})$$

$U_s(t)$ décrit la commande discontinue donnée comme suit :

$$U_s = (GB)^{-1} D \cdot \text{sign}(s) = K \text{sign}(S) \quad (\text{V.16})$$

V.6 COMMANDE PAR LOGIQUE FLOUE DE TYPE-2 INTERVALLE BASEE SUR LES MODE DE GLISSEMENT

Dans cette section, nous avons choisi de ne pas utiliser la commande équivalente pour commander le robot à MAPs du CDTA, à cause de l'insuffisance de modèle identifié dans le cas pratique. Nous avons opté pour l'utilisation seule de partie discontinue, la partie commande équivalente sera compensée par le contrôleur flou de type-2 l'intervalle. La commande qu'on cherche à développer dans cette section est basée sur la condition d'attractivité du mode de glissement. Dans ce cadre, les règles d'inférence sont déduites systématiquement à partir de cette condition.

Pour réaliser cette commande robuste floue de type-2 par mode de glissement, nous avons choisi un schéma décentralisé de type-2 floue avec deux entrées et une sortie pour chaque axe du robot. Les

entrées du contrôleur sont la surface de glissement et sa dérivée ($S_i = e_{i1} + \lambda_{i1} \cdot \dot{e}_{i1}$ et \dot{S}_i), la sortie est la loi de commande (u_{fuzzy}) qui doit être appliquée sur les muscles. Les fonctions d'appartenance supérieures et inférieures des variables d'entrées floues sont choisies pour être complètement chevauchés, triangulaires, trapézoïdales et symétriques, celles-ci sont présentées dans les figures suivantes :

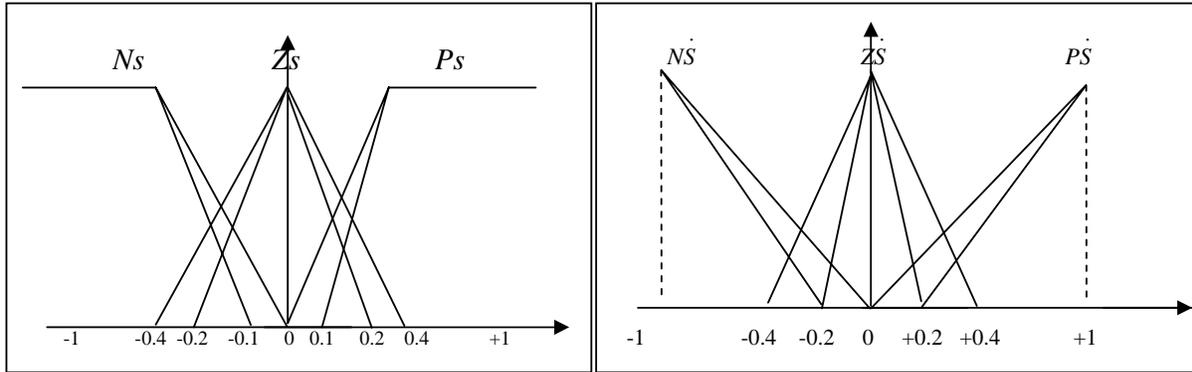


Figure V.9. Les entrées de contrôleur IT2FSMC (a) surface de glissement (b) la dérivée de la surface de glissement.

La figure V.10 présente la répartition des fonctions d'appartenance sur l'univers de discours. Pour la sortie, afin de minimiser le temps d'exécution du contrôleur flou, nous avons choisi trois fonctions d'appartenance de type singleton dénommées (N Négatif, Z Zéro, et P Positif) :

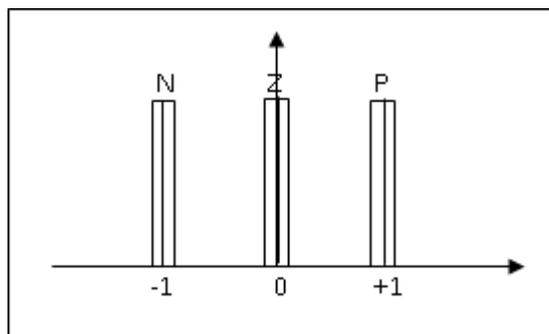


Figure V.10. Les valeurs floues de la sortie pour la commande IT2FSMC.

La V.11 montre le schéma bloc de contrôleur développé :

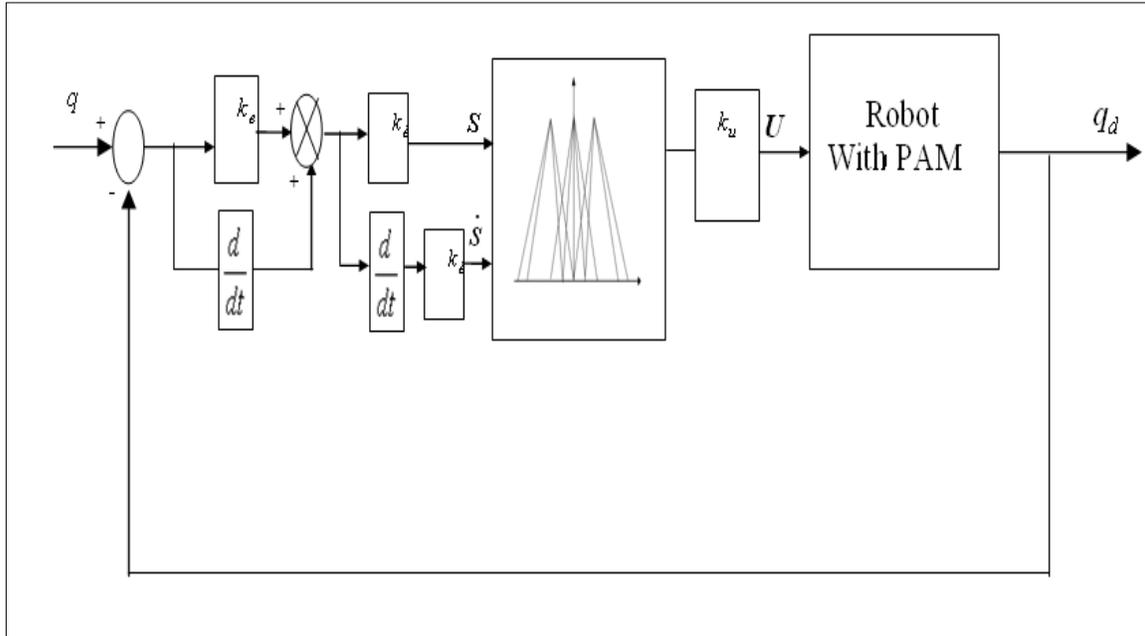


Figure V.11. Schéma bloc de la commande IT2 FSMC du robot à MAP.

Sur la base de la condition d'attractivité (l'équation V.12) [35], nous pouvons être amenés à la représentation diagonale des règles de raisonnement si-alors, où, à travers ce raisonnement nous sommes capables d'assurer la stabilité globale du robot en boucle fermée, le tableau 4 suivant résume ce raisonnement:

	\dot{s}	<i>N</i>	<i>Z</i>	<i>P</i>
<i>S</i>				
<i>P</i>		<i>Z</i>	<i>PM</i>	<i>P</i>
<i>Z</i>		<i>NM</i>	<i>Z</i>	<i>PM</i>
<i>N</i>		<i>N</i>	<i>NM</i>	<i>Z</i>

Tableau V.4 Table des règles.

Le moteur d'inférence est le noyau du système flou. Pour sa réalisation, nous avons utilisé l'équation générale (III.16). La méthode de centre de l'ensemble a été utilisée pour la réduction de type équation (III.17). A partir de l'étape réduction de type, on obtient un intervalle nommé U_{cos} intervalle qui doit être defuzzifié. La défuzzification est la dernière étape dans la commande floue de type-2, sa sortie est une valeur numérique. Dans notre cas, c'est une pression qui doit être injectée aux actionneurs du bras de robot. L'équation (III.20) est utilisée pour le calcul de cette valeur.

V.7 RESULTATS EXPERIMENTAUX ET DISCUSSION

Pour examiner la faisabilité et la validité de la commande floue glissante de type-2 proposée, des tests expérimentaux ont été réalisés. Les programmes informatiques d'expérimentation ont été réalisés à l'aide du langage C sur un Pentium 4. Afin de gérer efficacement une telle commande, le système logiciel d'implémentation est sous Linux avec un module RTAI et Syndex, plus de détails sont donnés dans [49].

Le mode de régulation a été adopté afin de tester la capacité du contrôleur proposé à maintenir les performances et la robustesse envisagées. Nous voulons que le robot *2-dll* arrive à la position angulaire désirée de 14 degrés pour l'axe 1 et 10 degrés pour l'axe 2, la position initiale est de zéro degré pour les deux axes. Nous avons choisi de présenter les positions angulaires des axes, les signaux de commande et les surfaces de glissement pour afficher l'atténuation ou non de l'effet de broutement. La figure V.12 représente les résultats expérimentaux pour le contrôle de l'axe 1 et l'axe 2. A titre de comparaison avec le contrôleur proposé, un régulateur flou de type-2 intervalle a été implémenté. Les entrées de ce dernier sont l'erreur en position angulaire et sa dérivée, la sortie est la pression qui doit être injectée dans les axes du robot. Les fonctions d'appartenance de la commande floue de type-2 sont similaires à celle de la commande floue glissante.

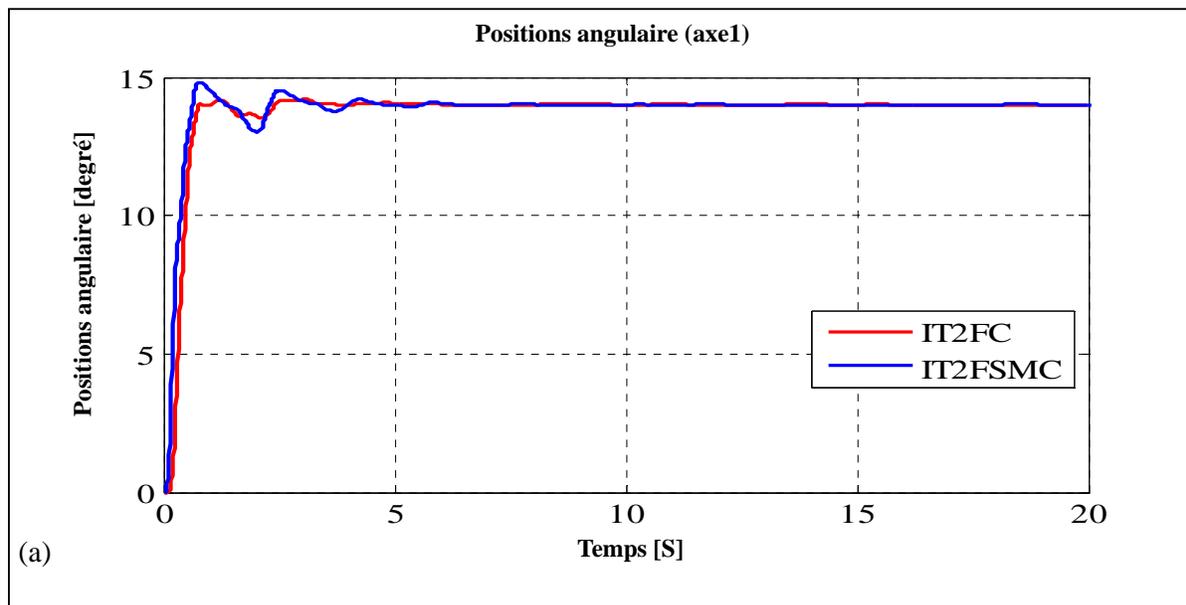


Figure V.12 Résultats expérimentaux de la commande IT2FSMC sur l'axe 1 (a) repense en position angulaire IT2FSMC (bleu) and IT2FC (rouge), (b) signaux de commande IT2FSMC (bleu) and IT2FC (rouge) (c) surface de glissement.

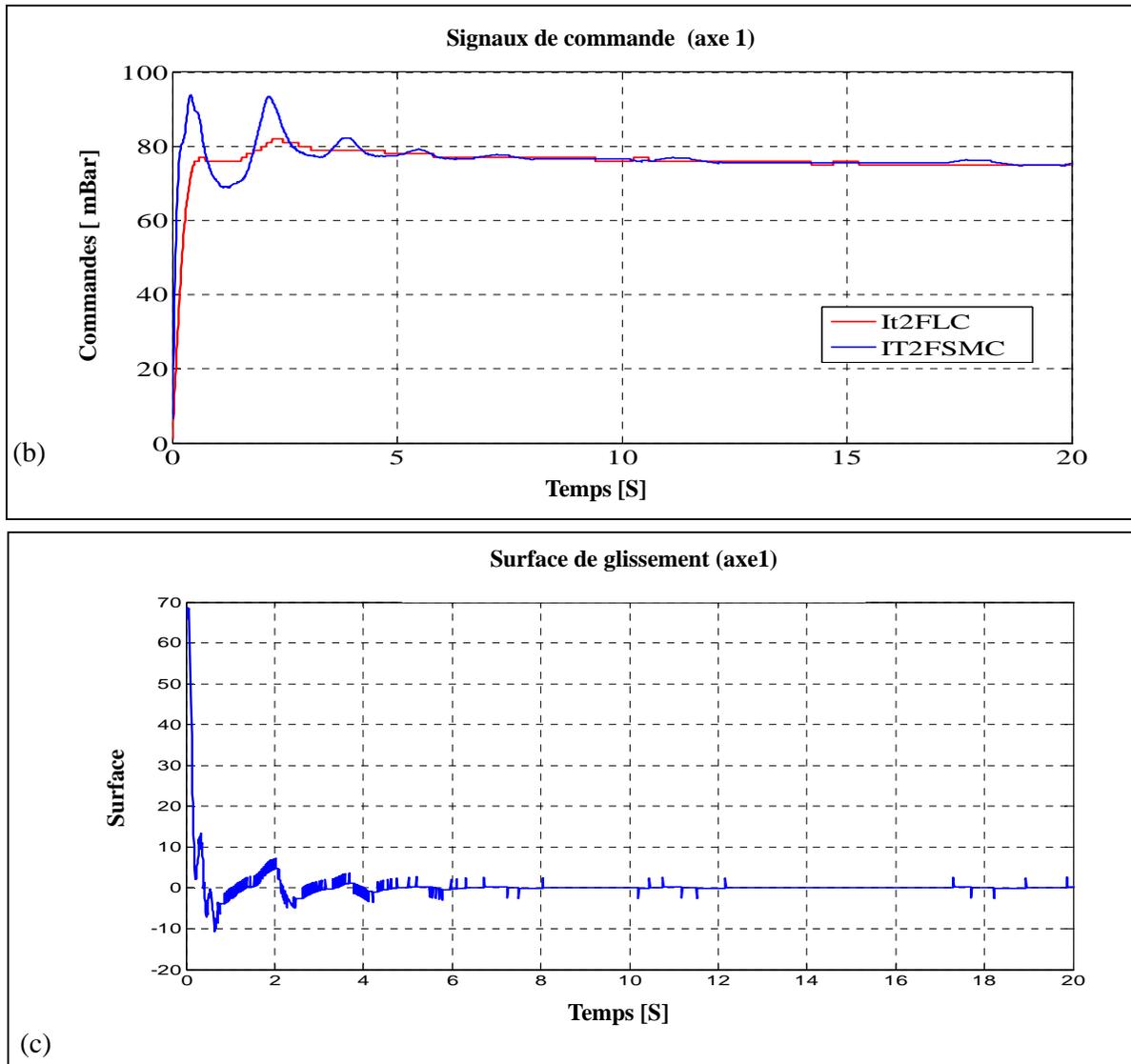


Figure V.12 Résultats expérimentaux de la commande IT2FSMC sur l'axe 1 (a) réponse en position angulaire IT2FSMC (bleu) et IT2FC (rouge), (b) signaux de commande IT2FSMC (bleu) et IT2FC (rouge) (c) surface de glissement.

La figure V.13 représente les résultats obtenus après expérimentation sur l'axe 2:

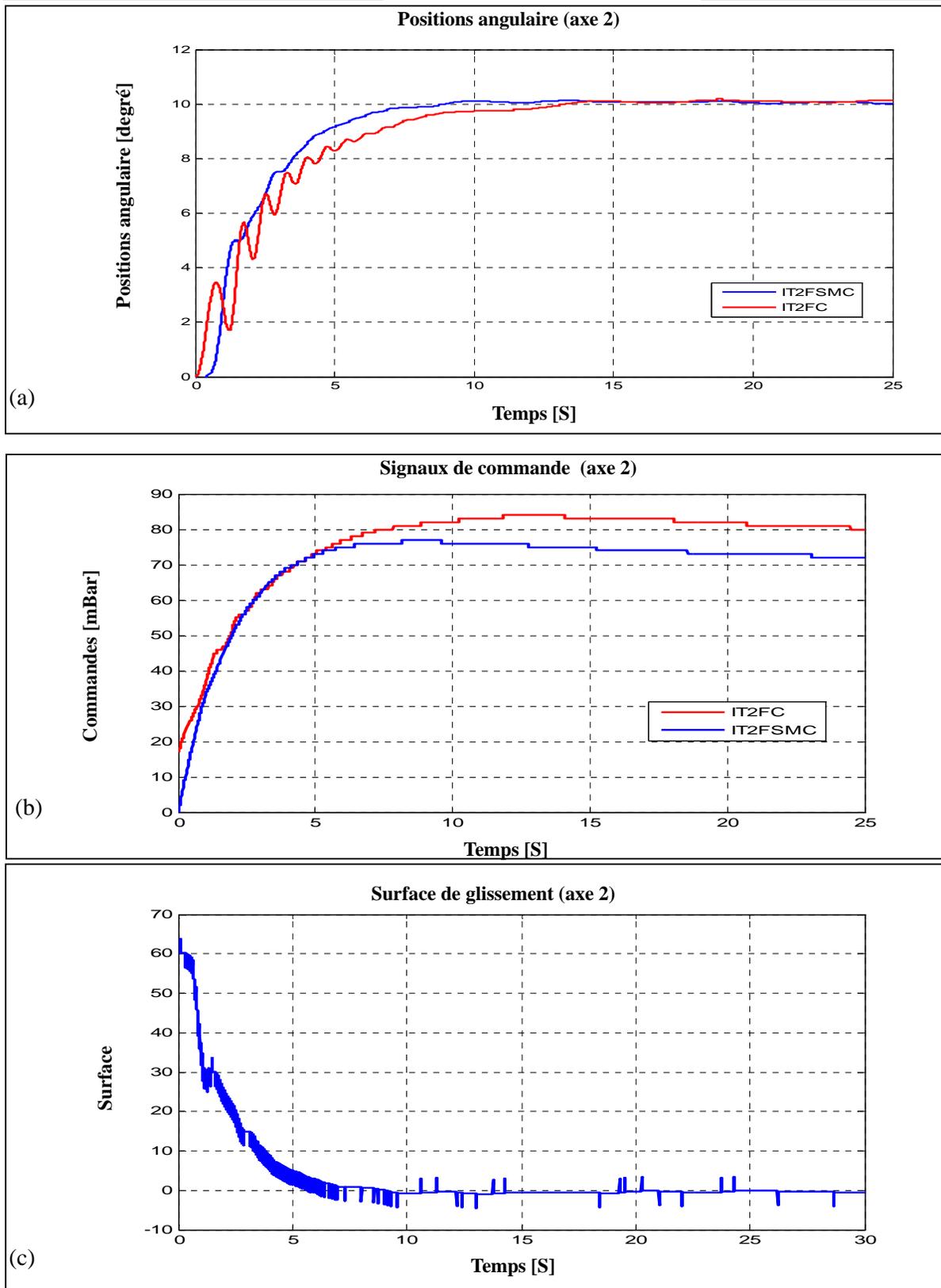


Figure V.13 Résultats expérimentaux de la commande IT2FSMC sur l'axe 2 (a) réponse en position angulaire IT2FSMC (bleu) et IT2FC (rouge), (b) signaux de commande IT2FSMC (bleu) et IT2FC (rouge) (c) surface de glissement.

Pour la commande IT2FSMC, on observe à partir de la réponse en position angulaire que, l'axe 1 a suivi d'une manière adéquate l'angle de référence imposé, avec l'existence d'une erreur statique nulle (0°) et dynamique quasiment faible. Nous avons pu observer à partir des figures V.12-b et V.13-b le lissage des signaux de la commande ainsi que ceux de la surface de glissement, ce qui signifie une atténuation de l'effet de broutement. Malgré que l'IT2FSMC et l'IT2FC aient présentés de bons résultats, l'IT2FSMC a apporté une amélioration, que ce soit par rapport à la réponse de l'axe du robot ou bien la quantité d'énergie consommée (commande). A cause des difficultés existantes dans le robot du CDTA (mentionnées dans le chapitre 1), il faut noter ici que la commande de l'axe 2 est très difficile à réaliser par rapport à celle de l'axe 1. On observe à partir de la réponse en position angulaire que la commande IT2FSMC a suivi de façon adéquate l'angle de référence imposé, avec l'existence d'erreurs statiques (environ 0.1°) de et dynamiques relativement faibles. Un petit retard (0.5 Second) a été remarqué sur l'axe 2, ce dernier est causé par l'effet gravitationnel, les caractéristiques des muscles ainsi que l'effet de couplage. Nous pouvons voir d'après les figures du signal de commande comme celle de la surface de glissement le lissage de ces signaux, ce qui montre l'atténuation de l'effet de broutement. Globalement, nous pouvons dire que l'IT2FSMC a donné de meilleures performances par rapport à la commande IT2FC, cette amélioration a été fortement manifestée dans l'axe 2.

V.8 CONCLUSION

Dans ce chapitre, nous avons proposé un contrôleur flou de type-2 intervalle adaptatif basé sur l'algorithme de descente du gradient pour le contrôle de la position angulaire d'un bras d'un robot actionné par des MAPs. Une étude expérimentale dont le but de contrôler la position angulaire en mode de régulation de deux degrés de liberté a été réalisée afin de montrer l'efficacité de la méthode de contrôle (AIT2FL) proposée par rapport au cas classique (IT2FLC). Dans la deuxième partie de ce chapitre, un contrôleur flou de type-2 intervalle basé sur la condition d'attractivité de la commande par mode de glissement a été proposé. Ce contrôleur a été mis en œuvre en temps réel sur le robot du CDTA. Les résultats expérimentaux montrent que non seulement la performance de bon suivi a été obtenue, mais aussi la stabilité et la robustesse ont été garanties avec un effet de broutement quasiment inexistant.

CHAPITRE VI. *Commande Par Mode De Glissement Basée Sur Un Réseau De Neurones Et Un Contrôleur Flou De Type-2.*

VI.1 INTRODUCTION

La synthèse d'une commande par mode de glissement pour un système non linéaire nécessite d'avoir suffisamment de connaissances sur son modèle mathématique. Il est bien connu que le modèle mathématique d'un robot à MAPs est difficile à obtenir. D'autre part, la commande équivalente de la commande par mode de glissement est une combinaison linéaire et/ou non linéaire des variables d'entrées/sorties ainsi que du modèle du robot. A cet effet, cette commande est difficile à obtenir. Afin de résoudre ce problème, la commande équivalente a été conçue par l'usage d'un réseau de neurones à fonctions de bases radiales, le choix de ce réseau de neurones est justifié par la clarté de son architecture ainsi que par sa rapidité en termes de temps de traitement. Pour la partie discontinue, un contrôleur flou de type-2 a été sollicité. Ce dernier nous permet non seulement de résoudre le problème de broutement, mais aussi d'améliorer les performances et la robustesse du système commandé.

VI.2 COMMANDE PAR MODE DE GLISSEMENT D'UN ROBOT MANIPULATEUR A N-DDL

L'objectif de cette partie est d'utiliser l'approche de la commande par mode glissant pour la synthèse d'une commande robuste qui permet de forcer les axes d'un robot manipulateur à n -ddl décrit par l'équation (I.1) à suivre des trajectoires désirées. La commande développée doit être basée sur le modèle (I.1) et doit être stable au sens de Lyapunov. La synthèse de la commande repose sur deux principales étapes telles que : le choix de la surface de glissement et la synthèse de la loi de commande. La surface de glissement dans ce cas est choisie de la forme multi-variable classique suivante :

$$S = e_1 + \lambda \cdot e_2 = \Lambda e \tag{V.1}$$

Où $S \in R^{n \times 1}$ est l'hyper-surface de glissement,

$\Lambda = \text{diag}(1, \lambda_1, \dots, \lambda_n) \in R^{n \times n}$ est une matrice diagonale à valeurs positives,

$e = q - q_d$ et $\dot{e} = \dot{q} - \dot{q}_d$ sont le vecteur d'erreur et sa dérivée. Les vecteurs de vitesse et l'accélération ont été choisis de la forme suivante [43] :

$$\dot{q}_r = \dot{q}_d - \Lambda.e, \quad \ddot{q}_r = \ddot{q}_d - \Lambda.\dot{e} \quad (\text{VI.2})$$

A partir des équations (VI.1) et (VI.2), nous pouvons écrire le vecteur de la surface et sa dérivée de la manière suivante :

$$S = \dot{q} - \dot{q}_r, \quad \dot{S} = \ddot{q} - \ddot{q}_r \quad (\text{VI.3})$$

La forme générale du contrôleur par mode de glissement comporte deux termes, qui sont la commande équivalente et la commande discontinue telle que :

$$\tau = \hat{\tau} - k \text{sign}(S) \quad (\text{VI.4})$$

$k = \text{diag}(k_{ii}) \in R^{n \times n}$, dans ce cas, la commande équivalente notée par $\hat{\tau}$ est donnée par :

$$\hat{\tau} = M\ddot{q}_r + C\dot{q} + G \quad (\text{VI.5})$$

VI.2.1. Analyse de stabilité

Soit la fonction candidate de Lyapunov :

$$V = \frac{1}{2} S^T M S \quad (\text{VI.6})$$

La dérivation de la fonction (VI.6) permet d'obtenir l'équation suivante :

$$\dot{V} = S^T M \dot{S} + \frac{1}{2} S^T \dot{M} S \quad (\text{VI.7})$$

Propriété 1 : pour le modèle du robot donné par l'équation (I.1), la propriété dite d'asymétrie suivante est valable [16] :

$$S^T (\dot{M} - 2C) S = 0 \quad (\text{VI.8})$$

Avec la substitution de (VI.6) et (VI.5) dans (VI.7) et l'utilisation de la propriété d'asymétrie 1, la

dérivée de la fonction de Lyapunov devient :

$$\dot{V} = S^T (\tau - M\ddot{q}_r - C\dot{q}_r - G - T_d) \quad (\text{VI.9})$$

La substitution de (VI.4) et (VI.5) dans (VI.9) donne l'équation (VI.10) suivante :

$$\dot{V} = S^T (CS - k \operatorname{sgn}(S)) \quad (\text{VI.10})$$

Finalement, l'équation (VI.10) peut être écrite sur la forme suivante :

$$\dot{V} = S^T CS - \sum_{i=1}^n k_i |S_i| \quad (\text{VI.11})$$

Avec $i=1 \dots n$ est le nombre d'axes du robot.

Si nous choisissons k_i comme suit :

$$k_i \geq \|CS\|_i + \eta_i \quad (\text{VI.12})$$

Où η_i sont des constantes positives. A partir de l'équation (VI.11) et avec la prise en considération de la contrainte (VI.12), la stabilité du robot manipulateur donné par le modèle (I.1) est garantie, et la forme finale de la dérivée de la fonction de Lyapunov est donnée par l'équation (VI.13) suivante :

$$\dot{V} \leq -\sum_{i=1}^n \eta_i |S_i| \quad (\text{VI.13})$$

La loi de commande donnée par l'équation (VI.4) et (VI.5) avec la condition (VI.12) permettent de conduire le robot à converger au point d'équilibre choisi asymptotiquement avec garant de la stabilité. Par ailleurs, l'utilisation de la commande par mode de glissement nécessite un certain minimum de connaissances sur le modèle de robot à commander, ceci notamment pour la synthèse de la commande équivalente. Les différentes composantes du modèle dynamique du robot sont des fonctions très complexes à déterminer car elles sont en fonction de la configuration instantanée du robot. De plus, le vecteur de force gravitationnelle est un terme indésirable qui requiert une puissance supplémentaire et qui au point de vue commande est une source d'instabilité. Le modèle du robot présenté par l'équation I.1 est extrêmement difficile à obtenir avec précision. Les performances réalisées par ces techniques de commande basées sur le modèle du robot ont une relation étroite avec l'exactitude de la modélisation. A cause des inconvénients des MAPs, ce problème de modélisation est plus délicat. Pour surmonter ce problème, nous avons sollicité les

techniques de commande qui présentent une robustesse naturelle telle que la commande à structure variable par mode de glissement. En revanche, cette commande nécessite un certain minimum de connaissances sur le système à commander qui permettent de synthétiser la commande équivalente. Cette dernière est liée explicitement au modèle du robot, ce qui rend leur calcul inexact. De plus, cette commande souffre du phénomène de broutement causé par sa partie discontinue. Pour toutes ces raisons, nous avons opté pour l'utilisation des techniques de l'intelligence artificielle pour résoudre les problèmes de la commande par mode de glissement, telles que les réseaux de neurones et les systèmes d'inférence floue.

VI.3 COMMANDE PAR MODE DE GLISSEMENT D'UN SYSTEME MONO ENTREE MONO SORTIE

Hypothèse 1: Dans l'objectif de simplifier l'établissement de la loi de commande proposée, les interactions entre les axes du robot sont considérées comme des perturbations permanentes qui seront compensées par la commande par mode de glissement. En fait, l'équation du modèle dynamique du robot donnée dans (I.7) peut être considérée comme un ensemble de systèmes mono-entrée mono-sortie et s'écrit dans l'espace d'états comme suit:

$$\begin{cases} \dot{e}_{1i} = e_{2i} \\ \dot{e}_{2i} = \tilde{f}_i(e, t) + \tilde{b}_i(e, t) \cdot u_i + \xi_i \end{cases} \quad (\text{VI.14})$$

Avec : $i=1 \dots n$ est le nombre des axes qui constituent le robot, $x_{1i} = q_i$, $\dot{x}_{1i} = x_{2i} = \dot{q}_i$, avec $e_i = x_{1i} - x_{1di}$, $e_{2i} = x_{2i} - x_{2di}$,

$\tilde{f}_i(e, t)$ est une fonction non-linéaire continue et bornée par $f_i(e, t)_{\min} < \tilde{f}_i(e, t) < f_i(e, t)_{\max}$.

$\tilde{b}_i(e, t)$ est une fonction également non-linéaire bornée par $0 < b_i(e, t)_{\min} < \tilde{b}_i(e, t)$.

ξ_i présente une perturbation bornée par $|\xi_i| < \xi_{\max i}$.

u_i est la commande pour chaque axe i .

Cette hypothèse permet d'éviter le calcul de la matrice inverse d'inertie du robot, de plus, la capacité des MAPs à absorber les chocs provoqués par l'effet de couplage entre les axes du robot valide

notre choix.

La surface de glissement dans ce cas a été choisie comme suit :

$$S_i = e_{1i} + \lambda_i \cdot e_{2i} = \Lambda_i e_i \quad (\text{VI.15})$$

Où: $\Lambda_i = [1, \lambda_i]$ est un vecteur de valeurs réelles positives et $e_i = [e_{1i}, e_{2i}]^T$ sont les erreurs dans l'espace d'états.

La loi de commande par mode de glissement est composée de deux parties connues par la commande équivalente et la commande discontinue. La première permet d'assurer le suivi idéal de la référence imposée et la deuxième permet de garantir la robustesse du système en boucle fermée telle que:

$$u_i = u_{eqi} + u_{hi} \quad (\text{VI.16})$$

THEOREME 1: Pour les systèmes qui possèdent des incertitudes, la commande équivalente peut être écrite de la façon suivante [67] :

$$u_{eqi} = u_{eqi}^* + \Delta u_{eqi} \quad (\text{VI.17})$$

avec: $u_{eqi}^* = -[\nabla S_i b_i]^{-1} [f_i]$, $\nabla u_{eqi} = -[\nabla S_i b_i]^{-1} [\nabla S_i \xi_i]$ et $\nabla S_i = \frac{\partial S_i}{\partial q_i}$.

La commande équivalente est calculée par l'usage du modèle du système à commander à partir de la condition de glissement $\dot{S}_i = 0$. Dans le cas du système (VI.14), cette loi est écrite comme suit :

$$u_{eqi} = \tilde{b}_i^{-1} (\Lambda_{mi}^T e_i - \tilde{f}_i(e, t) - \xi_i) \quad (\text{VI.18})$$

Dans ce cas \dot{S}_i est donné par :

$$\dot{S}_i = \Lambda_i \dot{e}_i = \Lambda_{mi}^T e_i - \tilde{f}_i(e, t) - \tilde{b}_i(e, t) \cdot u_i - \xi_i \quad (\text{VI.19})$$

Avec $\Lambda_i = [1, \lambda_i]$ et $\Lambda_{mi} = [0, \lambda_i]$

La partie u_{hi} dans (VI.16) représente la commande discontinue et est donnée par l'équation (VI.20)

:

$$u_{hi} = -\tilde{b}_i^{-1} k_i \text{sign}(S_i) \quad (\text{VI.20})$$

Avec $\tilde{b}_i = \sqrt{b_{\max} b_{\min i}} = \text{constante}$ [32].

À cause des non linéarités des robots à MAPs, et parce que la commande équivalente est basée sur le modèle du robot, sa synthèse avec exactitude sera difficile. De plus, la commande équivalente (VI.18) contient une partie incertaine. Pour cette raison, nous avons sollicité les techniques de l'intelligence artificielle pour son estimation, notamment un réseau de neurones à fonctions de bases radiales (RBFNN). Avant de présenter l'approche développée dans la section VI.4, nous citons les avantages de l'hybridation entre la commande par mode de glissement et les réseaux de neurones artificiels. D'après Ertugrul et Kaynak [68], un contrôleur Neuro-glissant a grossièrement les avantages suivants :

- 1) L'apprentissage et la mise en œuvre de la commande se fait simultanément.
- 2) Amélioration de l'erreur en poursuite.
- 3) L'atténuation du broutement.
- 4) Cette méthode ne nécessite pas le calcul de la matrice d'inertie et de son inverse au moment de la synthèse de la commande.
- 5) Le nombre de couches, le nombre de neurones et la connexion entre eux est plus simple par l'usage de la commande par mode de glissement, car le mode de glissement donne une idée sur la partie réseau de neurones.
- 6) Les poids synaptiques du réseau de neurones ne sont pas définis d'une manière aléatoire. Les performances désirées à partir de la commande par mode de glissement aident à l'initialisation de ces poids.
- 7) Le contrôleur neuronal obtenu est robuste compte tenu de la robustesse des modes de glissement.

VI.4 ESTIMATION DE LA COMMANDE EQUIVALENTE PAR RBFNN

VI.4.1. L'architecture du RBFNN

L'utilisation de perceptron multicouche pour la commande des systèmes souffre de deux principaux inconvénients; le problème des minimas locaux et celle du temps d'apprentissage trop lent qui demande un matériel de traitement des données relativement lourd [34][42]. Le réseau de neurones

à fonctions de bases radiales présente une alternative très prometteuse pour résoudre les limites des réseaux de neurones conventionnels. Grâce à de son avantage principal comme un approximateur universel, le RBFNN est sollicité pour estimer des fonctions non-linéaires, pour modéliser et commander des systèmes [27,32,42,44]. Le RBFNN utilisé est constitué de trois couches : la couche d'entrée, la couche cachée et la couche de sortie. La couche d'entrée est composée de l'erreur en position angulaire et sa dérivée, la couche cachée est constituée de m neurones activés par des fonctions Gaussiennes, la sortie du réseau donne la commande équivalente estimée. L'architecture du réseau utilisé permet d'estimer la commande équivalente pour un seul axe indépendamment des autres. Il faut noter ici qu'il est possible d'utiliser un réseau Multi-Variable développé et simulé pour un robot manipulateur [27]. Bien que cette architecture prenne en considération l'effet du couplage, elle souffre toutefois de plusieurs difficultés. Premièrement, la complexité et la non maîtrise de la dynamique de l'architecture du réseau augmente de plus en plus avec l'augmentation du nombre d'axes à commander. Deuxièmement, le temps de calcul des sorties estimées dans le cas MIMO deviendrait important. Enfin, l'ajustement des paramètres du réseau sera plus complexe dans le cas de la commande du système MIMO.

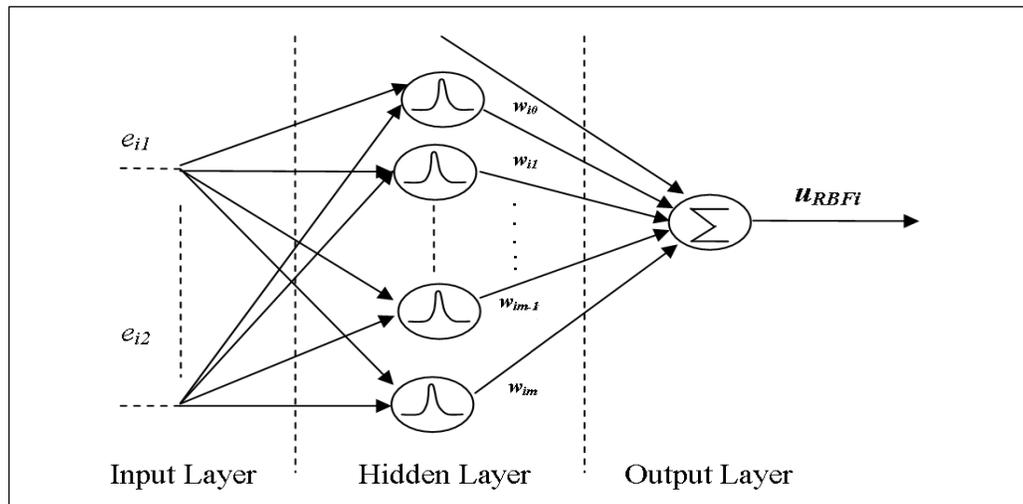


Figure VI.1. Structure d'un réseau de neurones de fonctions à bases radiales.

Comme nous l'avons évoqué, les entrées du RBFNN utilisées sont l'erreur en position angulaire et sa dérivée pour chaque axe du robot: $e_i = [e_{i1}, e_{i2}]^T$. Chaque neurone de la couche cachée est caractérisé par une Fonction d'Activation Gaussienne (FAG) donnée par sa forme générale suivante :

$$h_{ji}(x) = e^{-\left(\frac{\|e_i - c_{ji}\|}{2 \cdot \sigma_{ji}^2}\right)} \quad (\text{VI.21})$$

Où c_{ji} est le centre de la FAG, $j=1\dots m$ est le nombre de neurones pour chaque axe i , σ_{ji} est la variance de la FAG. Mathématiquement, la sortie du réseau de neurones est calculée comme suit :

$$u_{RBFi} = \sum_{j=1}^n w_{ji} \cdot h_{ji} = W_i^T \cdot H_i \quad (\text{VI.22})$$

Où $W_i = [w_{0i}, w_{1i}, w_{2i} \dots w_{ni}]$ avec w_{ji} est le poids associé à la sortie de chaque neurone j pour l'axe i . $H_i = [h_{0i}, h_{1i}, h_{2i} \dots h_{ni}]$ est le vecteur de sortie avec h_{ji} la sortie de neurones pour des entrées données, qui est une fonction de base radiale (équation VI.21). Cette fonction représente la différence entre le centre de chaque neurone c_{ji} et la variable d'entrée e_{ji} [32].

$$\|e_i - c_{ji}\| = (e_{i1} - c_{j1})^2 + (e_{i2} - c_{j2})^2 \quad (\text{VI.23})$$

VI.4.2 Estimation de la commande équivalente par le réseau de neurones à fonctions de bases radiales

Plusieurs techniques d'adaptation des paramètres du réseau de neurones à fonctions de bases radiales ont été développées telles que : la méthode de descente du gradient [32], la méthode k-main [16], etc. Dans ce travail, nous avons choisi d'ajuster seulement les poids synaptiques de chaque neurones avec l'objectif de minimiser le temps de calcul au moment de l'expérimentation. Ces paramètres seront adaptés d'une manière spécifique, déduits à partir de l'analyse de stabilité, qui sera présentée dans une prochaine section. Le RBFNN dispose d'une fonctionnalité similaire au système flou de type-1. A cet effet, nous avons la possibilité d'utiliser le théorème suivant donné dans [32].

THEOREME 2: soient $x = [e, \dot{e}]^T$ et $T \in R^2$ [32], il existe un vecteur $W = W^* = [w_0, w_1, w_2 \dots w_n]$ qui satisfait la condition suivante :

$$\sup_{T \in R^2} |u_{eq} - u_{BRF}^*| \leq \varepsilon_0 \quad (\text{VI.24})$$

Avec $\varepsilon_0 \geq 0$ [32], dans notre cas, nous avons choisi :

$$u_{eqi} - u_{BRFi}^* = \rho_i \quad (\text{VI.25})$$

Avec: $u_{BRFi}^* = W_i^{*T} H_i$, u_{BRFi}^* est l'approximation optimale de u_{eqi} et $\rho_i \geq \varepsilon_{0i}$

Par la prise en considération de ce théorème, la loi de commande (VI.16) devient :

$$\mathbf{u}_i = \mathbf{u}_{RBFi} + \mathbf{u}_{hi} \quad (\text{VI.26})$$

D'autre part, l'équation de perturbation peut être obtenue à partir de l'équation (VI.18), comme suit :

$$\xi_i = \Lambda_{mi}^T e_i - \tilde{f}_i(e, t) - b_i \cdot u_{eqi} \quad (\text{VI.27})$$

A partir de la substitution des équations (VI.26) et (VI.27) dans (VI.14), nous obtenons :

$$\begin{cases} \dot{e}_{1i} = e_{2i} \\ \dot{e}_{2i} = \tilde{b}_i \cdot (u_{eqi} - u_{RBFi} - u_{hi}) - \Lambda_{mi}^T e_i \end{cases} \quad (\text{VI.28})$$

Nous pouvons réécrire l'équation (VI.28) de la manière suivante :

$$\dot{e}_i = A_i \cdot e_i + B_i \cdot u_i \quad (\text{VI.29})$$

Avec: $\dot{e}_i = [\dot{e}_{1i}, \dot{e}_{2i}]^T$, $A_i = \begin{bmatrix} 0 & 1 \\ 0 & -\lambda_i \end{bmatrix}$, $B_i = \begin{bmatrix} 0 \\ b_i \end{bmatrix}$ u_i devient :

$$\mathbf{u}_i = \mathbf{u}_{eqi} - \mathbf{u}_{RBFi} - \mathbf{u}_{hi} \quad (\text{VI.30})$$

THEOREME 3: u_{eqi} représente la meilleure approximation quand $\rho_i = u_{eqi} - u_{RBFi}^*$ [33], avec :

$$u_{RBFi} - u_{RBFi}^* = (W_i - W_i^*)^T \cdot H_i = \tilde{W}_i^T \cdot H_i \quad (\text{VI.31})$$

Avec $\tilde{W}_i^T = (W_i - W_i^*)^T$.

VI.4.3. Analyse de stabilité et adaptation des paramètres de la commande RBFNN

Soit la fonction candidate de Lyapunov pour chaque axe donnée par:

$$V_i = \frac{1}{2} S_i^2 + \frac{b_i}{\delta_i} \tilde{W}_i^T \tilde{W}_i \quad (\text{VI.32})$$

Où $\tilde{W}_i = (W_i - W_i^*)$ et δ_i est une constante positive.

Le développement mathématique de la dérivation de cette fonction de Lyapunov V_i pour garantir la

stabilité du système en boucle fermée est donné comme suit :

$$\begin{aligned}
\dot{V}_i &= S_i \dot{S}_i + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\
&= S_i \Lambda_i^T \dot{e}_i + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\
&= S_i \Lambda_i^T (A_i \cdot e_i + B_i \cdot u_{eqi} - u_{RBFi} - u_{hi}) + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\
&= S_i \Lambda_i^T A_i \cdot e_i + S_i \Lambda_i^T B_i \cdot (u_{eqi} - u_{RBFi} - u_{hi}) + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\
&= S_i \Lambda_i^T A_i \cdot e_i + S_i \Lambda_i^T B_i \cdot (u_{RBFi}^* - u_{RBFi} - u_{hi} - \rho_i) + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\
&= -\frac{b_i}{\delta_i} \tilde{W}_i^T (\delta_i S_i H_i - \dot{\tilde{W}}_i) - b_i S_i \rho_i - b_i S_i u_{hi}
\end{aligned} \tag{VI.33}$$

En choisissant la loi adaptative suivante :

$$\dot{\tilde{W}}_i = \delta_i S_i H_i \tag{VI.34}$$

Et en utilisant l'équation (VI.34), l'équation (VI.33) devient :

$$\dot{V}_i = -b_i S_i \varepsilon_i - b_i S_i u_{hi} \tag{VI.35}$$

Dans l'objectif de maintenir la stabilité en boucle fermée, nous avons choisi la commande discontinue u_{hi} avec compensateur de la forme suivante [42] :

$$u_{hi} = k_{0i} S_i + k_{1i} \text{sign}(S_i) \tag{VI.36}$$

Par l'usage de (IV.21) dans (IV.20) nous obtenons :

$$\begin{aligned}
\dot{V}_i &= -b_i S_i \rho_i - b_i k_{0i} S_i^2 - b_i k_{1i} S_i \text{sign}(S_i) \\
&= -b_i S_i \rho_i - b_i k_{0i} S_i^2 - b_i k_{1i} |S_i| \leq 0
\end{aligned} \tag{VI.37}$$

Avec : $k_{1i} > |\rho_i|$

De l'équation (VI.36) et (VI.26), la loi de commande est écrite dans sa forme finale comme suit :

$$u_i = u_{RBFi} + k_{0i} S_i + k_{1i} \text{sign}(S_i) \quad (\text{VI.38})$$

À partir des équations (VI.34) et (VI.37), les poids du RBFNN sont ajustés et la stabilité du système en boucle fermée est garantie par l'usage de la loi de commande donnée par l'équation (VI.38). Cependant, l'usage de l'équation (VI.38) pour la commande du robot souffre de l'effet de broutement. Pour résoudre ce problème, on a fait appel à la commande floue de type-2 afin de remplacer la commande discontinue, l'usage de cette commande permet non seulement l'atténuation de l'effet de broutement, mais aussi d'améliorer les performances et la robustesse du robot en boucle fermée.

VI.5 LE CONTROLEUR FLOU DE TYPE-2 POUR L'AMELIORATION DES PERFORMANCES ET L'ATTENUATION DE BROUITEMENT

L'objectif de cette partie est d'atténuer l'effet de broutement et d'améliorer les performances de la commande appliquée sur le robot à MAPs du CDTA. Pour ce faire, nous avons choisi un système d'inférence floue de type-2 avec une seule entrée et une seule sortie, l'entrée est la surface de glissement ($S_i = e_i + \lambda_i \cdot e_{2i}$) et la sortie remplace la partie discontinue de la commande par mode de glissement ($k_i \text{sign}(S_i)$). Après calcul de la sortie, elle sera substituée dans la commande globale donnée par l'équation (VI.38). Les labels utilisés pour la variable floue surface sont: {surface négative (Ns), surface zéro (Zs) et surface positive (Ps)}. Nous avons choisi trois fonctions d'appartenance de type-1 pour la variable de sortie. Ce choix a pour objectif de minimiser le temps de traitement des données par le système flou de type-2, les labels utilisés dans ce cas sont (Nb Négative, Z Zéro, et Pb Positive). La figure VI.2 représente la distribution des valeurs floues sur les univers de discours pour les deux variables floues d'entrée et de sortie, le tableau VI.1 présente le jeu de règles utilisé.

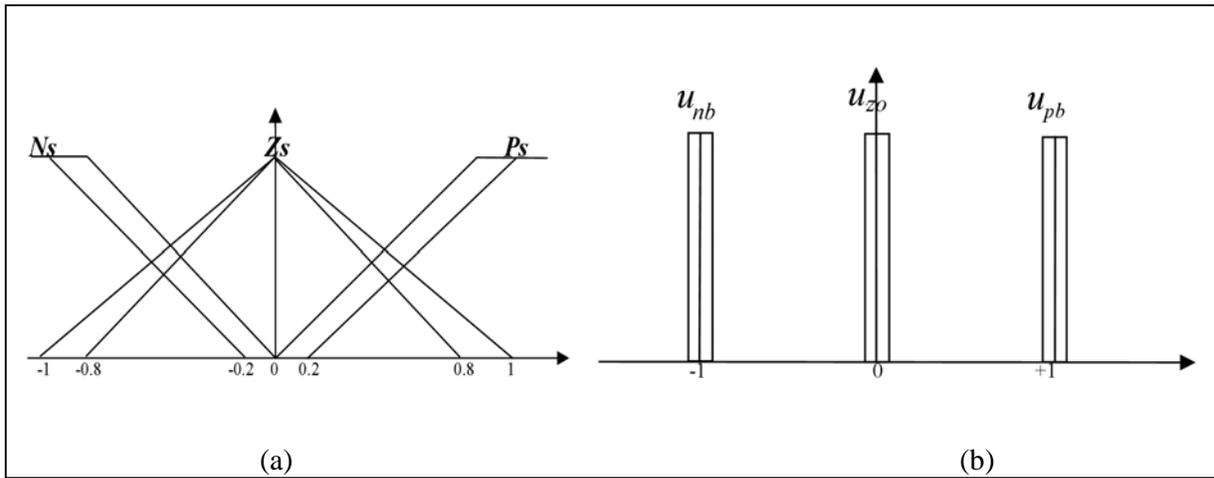


Figure VI.2. (a) L'entrée, (b) la sortie du contrôleur flou de type-2 utilisé pour remplacer la partie discontinue de la commande par mode de glissement.

Tableau VI. 1. Table de règles.

S_i	Ns	Zs	Ps
$k_i \text{sign}(S_i)$	u_{nb}	u_{z0}	u_{pb}

L'étape d'inférence est le cœur du système flou, l'équation (III.16) permet le calcul de l'inférence de ce contrôleur. Pour la réduction de type nous avons utilisé la méthode de centre de l'ensemble donnée par l'équation (III.17), la défuzzification représente le centre de l'intervalle issu de la réduction de type.

VI.6 APPLICATION DE LA COMMANDE RBFT2FSMC SUR LE ROBOT A MAPs DU CDTA

Dans l'objectif de mettre en évidence la commande proposée, des simulations et des expérimentations ont été effectuées pour le robot à MAPs du CDTA (voir figure I.3). la forte non linéarité du robot fait de lui un très bon candidat pour le test de cette loi de commande. La figure VI.3 montre un schéma bloc du contrôleur développé.

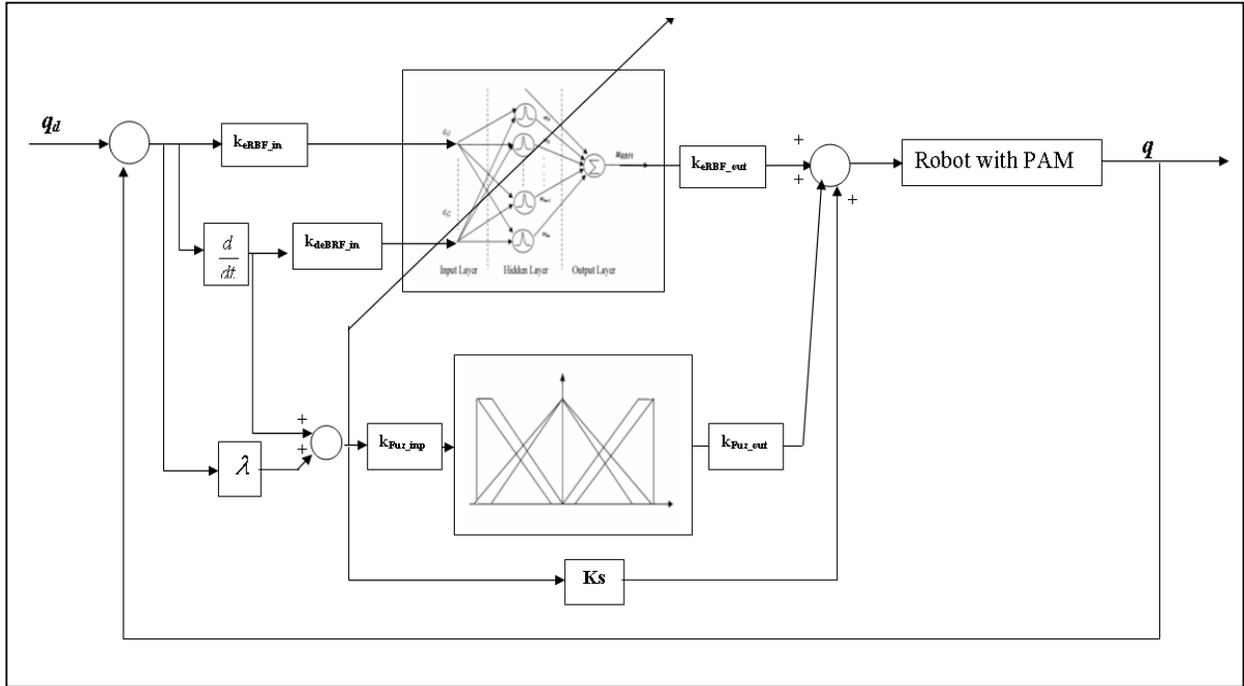


Figure VI.3. Schéma bloc de la commande RBFT2FSMC du robot à MAPs du CDTA.

VI.5.1 Simulation de la commande RBFT2FSMC

Comme dans [1], [5], et plusieurs autres travaux, pour réaliser la simulation, nous avons adopté le modèle donné par l'équation I.6. Les paramètres de ce modèle ont été identifiés par l'usage d'une méthode d'estimation basée sur le moindre carré simple, les résultats de cette identification ont été donnés dans le chapitre I. Le modèle utilisé pour la simulation est donné par l'équation (VI.39) dont les paramètres ont été identifiés dans le chapitre I.

$$\begin{cases} e_{1i} = e_{2i} \\ e_{2i} = -\tilde{a}_{0i}e_{1i} - \tilde{a}_{1i}e_{2i} - \tilde{b}_i u_i \end{cases} \quad (VI.39)$$

Avec:

$$\begin{aligned} A_0 &= \text{diag}(a_{01}, a_{02}) = \text{diag}(7.8573, 16.2075) \\ A_1 &= \text{diag}(a_{11}, a_{12}) = \text{diag}(5.6654, 29595) \\ B &= \text{diag}(b_1, b_2) = \text{diag}(15.3217, 2.0194) \end{aligned} \quad (VI.40)$$

Afin de nous rapprocher du modèle réel, nous avons considéré des incertitudes sur les paramètres du modèle estimé. Dans ce cas, l'équation (VI.40) devient :

$$\begin{aligned}
\tilde{A}_0 &= \text{diag}(\tilde{a}_{01}, \tilde{a}_{02}) = A_0 + \Delta A_0 \\
\tilde{A}_1 &= \text{diag}(\tilde{a}_{11}, \tilde{a}_{12}) = A_1 + \Delta A_1 \\
\tilde{B} &= \text{diag}(\tilde{b}_1, \tilde{b}_2) = B + \Delta B
\end{aligned} \tag{VI.41}$$

Mathématiquement, ces incertitudes ont été formulées comme suit :

$$\Delta A_0 \leq |0.2 \cdot A_0 \cdot \text{randn}|,$$

$$\Delta A_1 \leq |0.2 \cdot A_1 \cdot \text{randn}|$$

$$\text{et } \Delta B \leq |0.2 \cdot B \cdot \text{randn}|$$

Les figures (VI.4) et (VI.5) présentent les résultats de simulation des commandes RBFT1FSMC et RBFT2FSMC. Pour chaque axe du robot, nous avons choisi de représenter la position angulaire, le signal de commande, l'évolution de la surface de glissement et enfin l'incertitude paramétrique incorporée dans le modèle de chaque axe. Les résultats de l'application de la commande par RBFT2FSMC sont représentés par des lignes continues rouges, pour le cas de la commande RBFT1FSMC, ils sont représentés par des lignes noires. En position de repos, les axes du robot sont à (0,0) (position verticale), l'objectif est de les contrôler afin d'atteindre la position de 14 degrés pour l'axe 1 et 10 degré pour l'axe 2. La période d'échantillonnage est choisie égale à 10 ms et la simulation est effectuée pour une durée de 10 secondes. Les paramètres utilisés dans la simulation sont résumés dans le tableau VI.2.

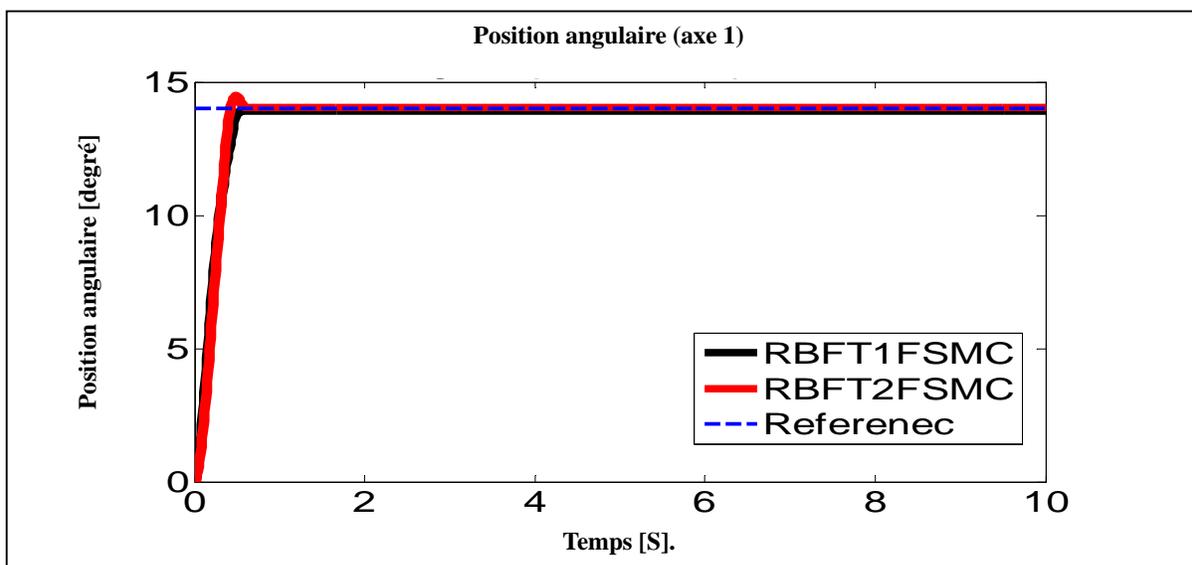


Figure VI.4. Résultats de simulation de l'application des commandes RBFT2FSMC et RBFT1FSMC sur l'axe 1.

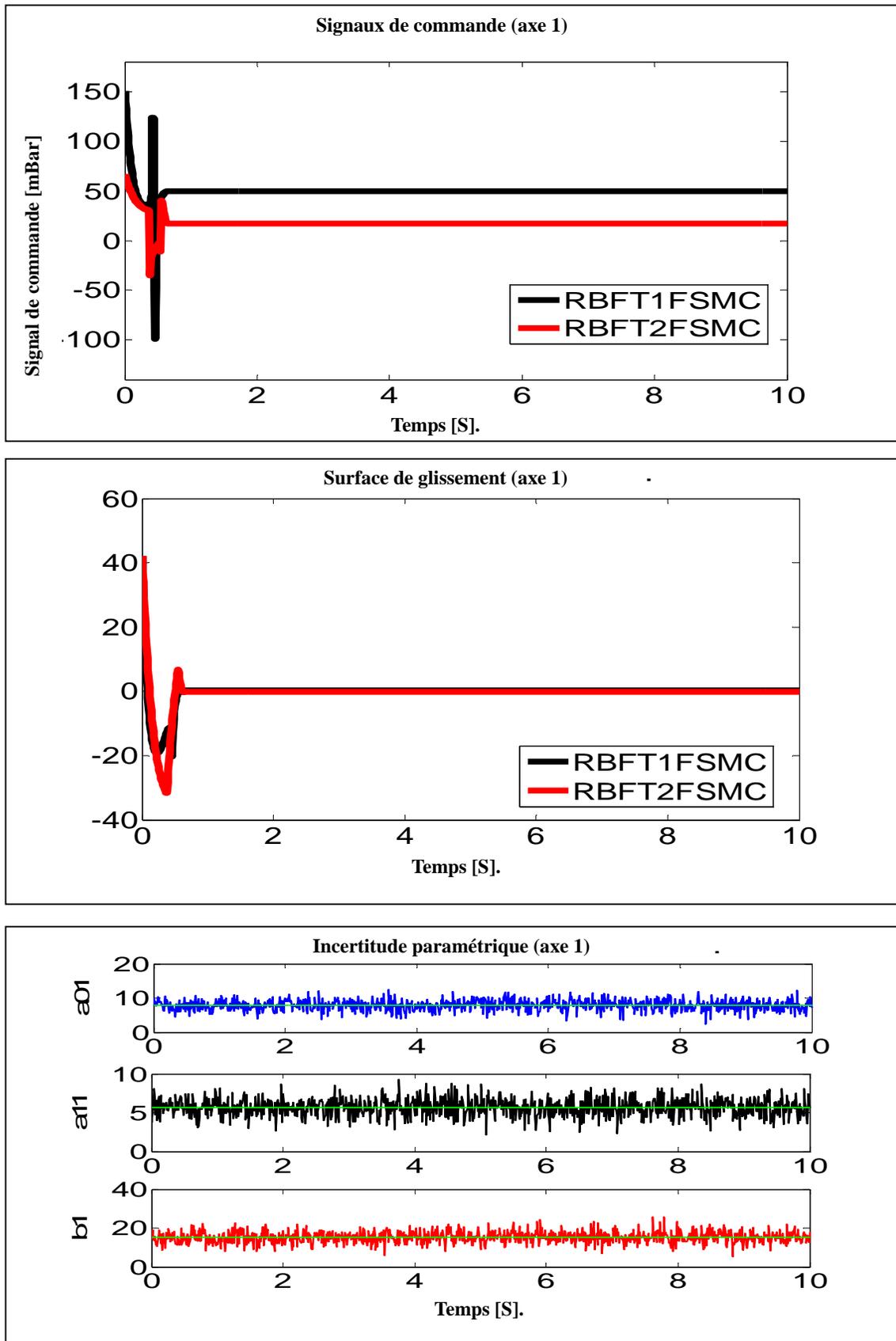


Figure VI.4. Résultats de simulation de l'application des commandes RBFT2FSMC et

RBFT1FSMC sur l'axe 1.

Méthode	RBFT1FSMC				
Paramètres de Simulation	λ	K_{0i}	K_{Fuz_in}	K_{Fuz_out}	m
Axe 1	3	0.7	0.1	40	6
Axe 2	2.1	2	0.8	30	6
	RBFT2FSMC				
Paramètres de Simulation	λ	K_{0i}	K_{Fuz_in}	K_{Fuz_out}	m
Axe 1	3	0.5	0.1	25	6
Axe 2	2	2	0.5	30	6

Tableau VI. 2. Paramètre des simulations.

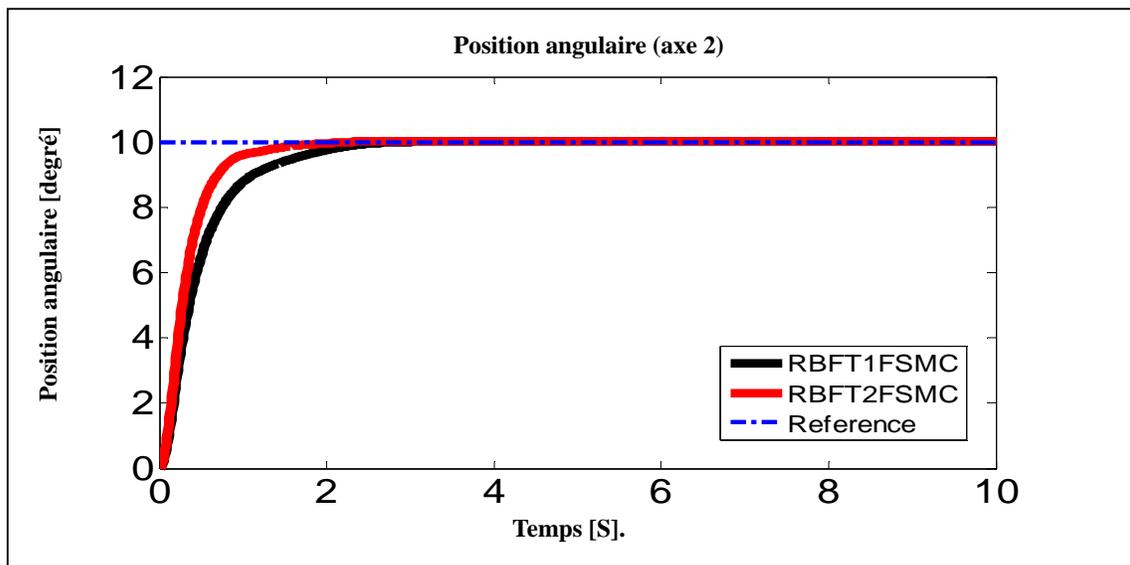


Figure VI.5. Résultats de simulation de l'application de la commande RBFT1FSMC et RBFT1FSMC sur l'axe 2.

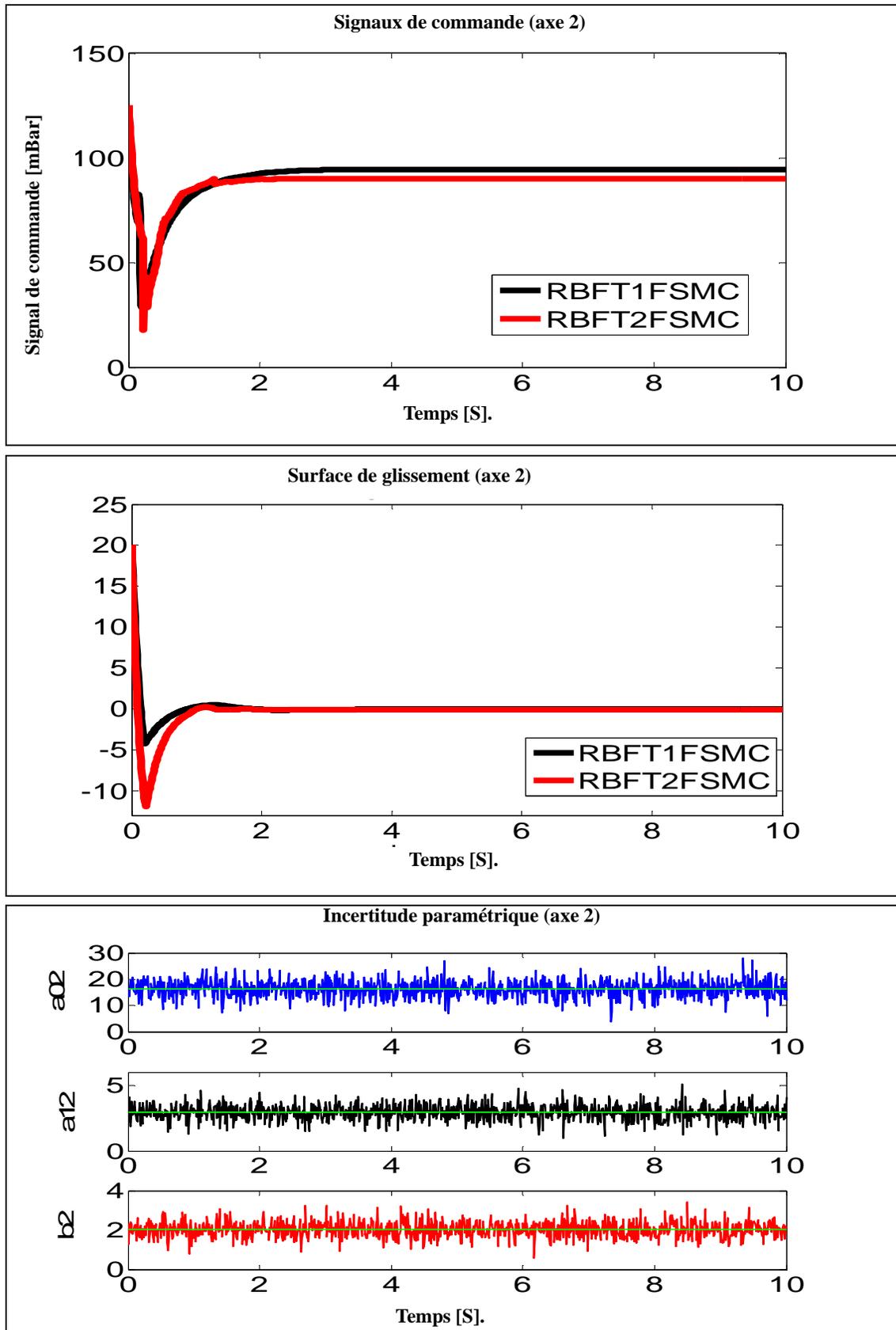


Figure VI.5. Résultats de simulation de l'application de la commande RBFT1FSMC et RBFT2FSMC sur l'axe 2.

Pour les deux axes du robot (axe 1 et 2), les résultats en RBFT1FSMC et en RBFT2FSMC ont montrés une bonne stabilisation des axes du robot. La commande RBFT2FSMC a donné de meilleures performances comparées avec les résultats en RBFT1FSMC. On peut constater qu'en position angulaire, particulièrement le temps de réponse et l'erreur en régime transitoire et permanent sont améliorés. Le signal de commande est relativement important dans le cas de la commande RBFT1FSMC comparé avec celle du RBFT2FSMC. Nous avons obtenu un lissage parfait de la commande et de la surface de glissement et ceci pour les deux commandes appliquées. Ce lissage est dû à l'effet de la commande par logique floue, que ce soit pour le type-1 ou 2, ce qui montre que l'hybridation de la commande par mode de glissement et la commande par réseau de neurones et floue résolvent les problèmes, notamment d'estimation de la commande équivalente de ce type de robots, mais aussi de ces commandes entre elles. Les deux commandes sont appliquées sur un système avec des variations paramétriques assez importantes, ce qui a montré la robustesse de la technique proposée vis-à-vis de ce phénomène qui est apparu naturellement sur ce type de systèmes.

5.2 Résultats expérimentaux

Comme la commande RBFT2FSMC a donné une amélioration remarquable comparée avec la RBFT1FSMC, nous avons opté pour le test uniquement de la commande RBFT2FSMC sur le robot du CDTA. Dans le cas de cette expérimentation, nous avons choisi de représenter les réponses en position angulaire des deux axes du robot, les signaux de commande ainsi que l'évolution de la surface de glissement en mode régulation. Comme dans le cas de la simulation, nous voulons que le robot se déplace d'une position verticale d'origine $(q_1, q_2) = (0^\circ, 0^\circ)$ à une position désirée égale à $(q_{d1}, q_{d2}) = (14^\circ, 10^\circ)$. Pour des raisons de sécurité la commande a été saturée à 200 mBar. Les paramètres de l'expérimentation sont résumés dans le tableau VI.3. La période d'échantillonnage est de 10 ms et la réponse des axes du robot a été testée pendant 25 secondes.

Méthode	RBFT2FSMC				
Paramètres	A	K_{0i}	K_{Fuz_in}	K_{Fuz_out}	m
Joint1	7	0.5	0.1	28	6
Joint2	8.5	1.5	0.5	30	6

Tableau VI.3. Paramètres expérimentaux.

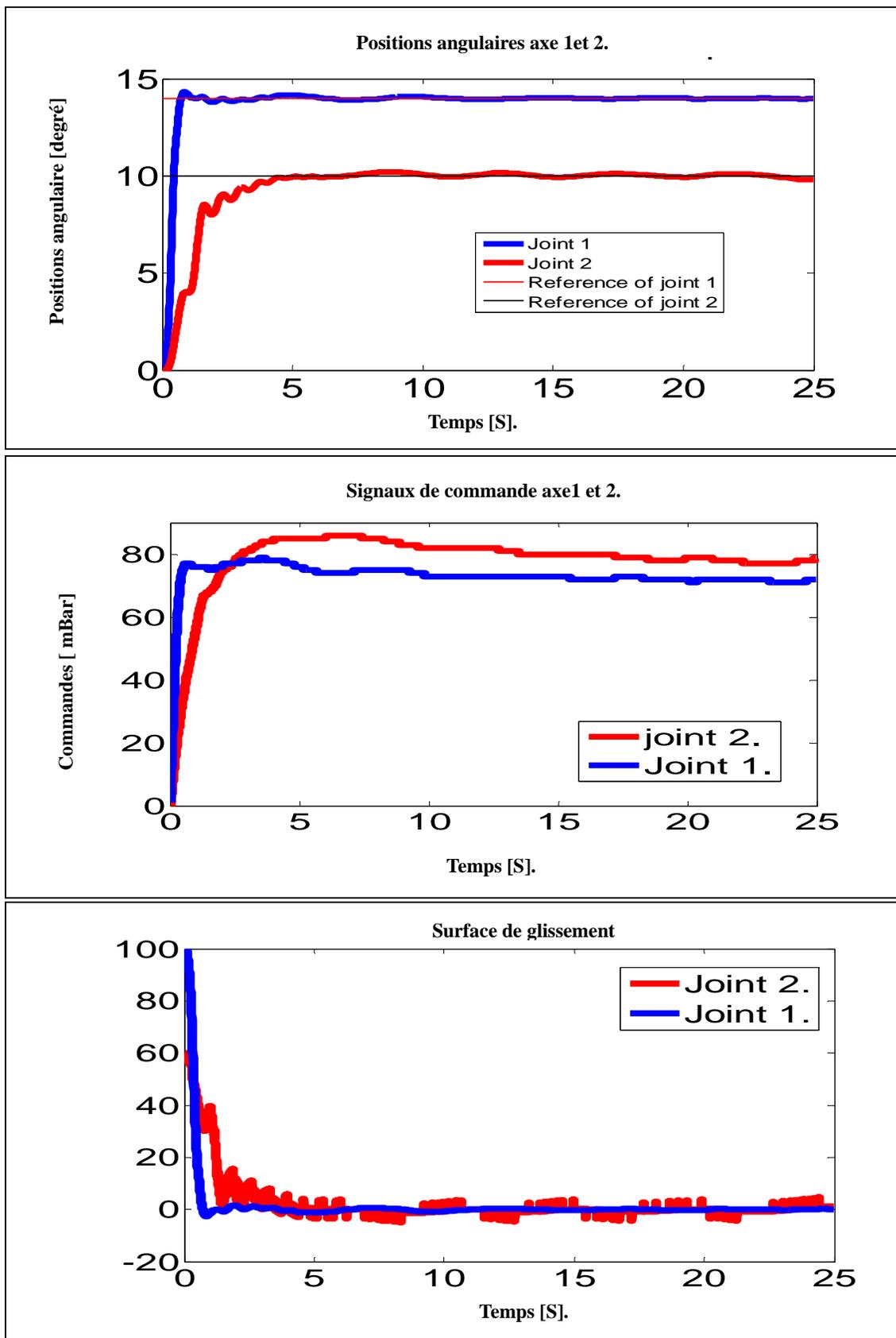


Figure VI.6. Résultats expérimentaux de l'application de la commande RBFT2FSMC sur l'axe 1 et 2 du robot du CDTA.

Nous observons à partir des courbes de la réponse en position de l'axe 1 qu'il suit la consigne désirée d'une manière adéquate, avec une erreur statique négligeable (autour de 0.1 degré), de même que pour l'erreur en régime transitoire. Le signal de commande donné par la figure (11.b) et celui de la surface de glissement montrent l'élimination du broutement. A cause de l'effet de la force gravitationnelle, du couplage entre les deux axes du robot ainsi que les caractéristiques des muscles, le contrôle de l'axe 2 est délicat comparé avec l'axe 1. Dans la réponse de cet axe, on remarque l'existence d'un temps de retard relativement faible dû aux phénomènes physiques cités auparavant. En régime permanent, l'axe 2 oscille lentement autour de la position désirée avec une erreur autour de 0.3° qui est une erreur acceptable dans notre cas d'application. Pour cet axe, les courbes du signal de commande et celle de la surface de glissement montrent une atténuation considérable de l'effet de broutement. Les paramètres de la simulation et les paramètres de l'expérimentation ne sont pas les mêmes, ceci justifie les phénomènes physiques qui n'étaient pas pris en considération dans l'étape de la simulation cela justifie aussi l'existence de cette commande et sa capacité à compenser ces erreurs de modélisation. En comparant les signaux de commande des deux axes, nous avons pu remarquer que l'énergie nécessaire pour tenir l'axe 2 à la position désirée est plus grande que celle consommée par l'axe 1, ceci est dû au même temps à l'effet de couplage et à la force gravitationnelle. Les résultats expérimentaux indiquent que la commande RBFT2FSMC proposée fonctionne efficacement, d'une manière satisfaisante et en temps réel, la technique est suffisamment robuste vis-à-vis de l'effet de couplage, les variations paramétriques vis-à-vis de la modélisation du robot.

VI.6 CONCLUSION

Dans ce chapitre, une nouvelle approche de commande robuste appelée commande par réseau de neurones de fonctions à bases radiales floue de type-2 glissante (RBFT2FSMC) a été proposée et appliquée sur un robot manipulateur à MAPs. La commande par mode de glissement a été utilisée pour assurer la robustesse du système en boucle fermée. Le réseau de neurones à fonctions de bases radiales a été sollicité pour estimer la commande équivalente de la commande par mode de glissement, le système d'inférence floue de type-2 a remplacé la commande discontinue de la commande par mode de glissement, ceci a pour objet d'atténuer le phénomène de broutement et en même temps d'améliorer les performances des commandes classiques.

Chapitre VII. Commande Par Terminal Sliding Mode Flou Adaptatif D'un Robot Manipulateur Anthropomorphe A Muscles Artificiels Pneumatiques.

VII.1 INTROCUCTION

Le Terminal Sliding Mode Control (TSMC) est un cas particulier de la commande par mode de glissement d'ordre supérieur [69]. Cette méthode a été développée pour améliorer les performances de la commande par mode de glissement classique tout en conservant la robustesse. Le TSMC a été développé essentiellement pour assurer la convergence en temps fini du système commandé. Dans la littérature, on trouve un nombre assez limité de solutions par TSM pour le contrôle de robots manipulateurs [22][23][38][41]. Les performances réalisées par ces méthodes dépendent explicitement de l'exactitude de la modélisation. Sachant que nous ne disposons pas du modèle pour le robot à MAPs, dans ce travail, nous avons opté pour l'approche par TSM qui est la plus adéquate pour le robot manipulateur à muscles de l'INSA. La méthode utilisée n'exige aucune connaissance *a priori* sur le modèle mathématique du robot, mais elle utilise cette dernière qui sera présentée en détails dans la section suivante. La première partie de ce travail consiste à une mise en œuvre par simulation de la commande par terminal sliding mode flou de type-1 et de type-2 dans le but de commander les axes du robot en position et en vitesse avec une consigne à vitesse trapézoïdale pour le robot du CDTA. En deuxième partie, une commande par terminal sliding mode flou de type-1 adaptative est appliquée sur le robot de l'INSA de Toulouse.

VII.2 LA SYNTHÈSE DE LA COMMANDE PAR TERMINAL SLIDING MODE NONSINGULIÈRE

En simulation, la commande du robot manipulateur à muscles artificiels du CDTA revient à la commande de son modèle identifié dans le chapitre 1. La surface de glissement terminal non singulière pour chaque axe est donnée comme suit [23] :

$$S = e_i + \beta \cdot \dot{e}_i^{p/q} \quad (\text{VII.1})$$

i représente les axes du robot.

La dérivée de la surface de glissement pour chaque axe i est décrite par l'équation (VII.2) :

$$\dot{s}_i = \dot{e}_i + \beta \frac{p}{q} \ddot{e}_i \cdot \dot{e}_i^{(p/q)-1} \quad (\text{VII.2})$$

Où $i=1,2$, $e_i = x_i - xd_i$. et $\dot{e}_i = \dot{x}_i - \dot{x}d_i$.

A partir de (VII.2), nous pouvons formuler l'égalité suivante :

$$\dot{s}_i = \dot{e}_i + \beta \frac{p}{q} \ddot{e}_i \cdot \dot{e}_i^{(p/q)-1} = -\eta \text{sign}(s_i) \quad (\text{VII.3})$$

Par la substitution de l'équation (VII.1) dans (VII.3), la loi de commande par terminal sliding mode non singulière pour le robot à MAP du CDTA est donné par l'équation (VII.4) :

$$u_{NTSMC} = -\frac{1}{b_i} (a_{i1} \cdot e_{i2} + a_{i0} \cdot e_{i1} + \beta \cdot \left(\frac{5}{7}\right) \cdot e_{i2}^{(3/5)} - \eta \cdot \text{sign}(s_i)) \quad (\text{VII.4})$$

où $q=7$, $p=5$, $f(x) = a_{i1} \cdot e_{i2} + a_{i0} \cdot e_{i1}$, $i=1,2$. $e_{i1} = x_{i1} - xd_{i1}$. et $e_{i2} = \dot{x}_i - \dot{x}d_i$.

VII.3 SYNTHÈSE DU CONTRÔLEUR PAR MODE DE GLISSEMENT TERMINAL FLOU [23]

Dans le but de remédier au problème de broutement causé par la partie discontinue dans u_{NTSMC} , nous avons sollicité un système d'inférence floue de type-1 pour remplacer η . L'architecture du contrôleur est composée de deux entrées et une sortie. Pour les variables d'entrée, nous avons choisi l'erreur en position angulaire et sa variation. Ces deux variables permettent d'apporter une description précise de la dynamique du système commandé, les règles associées à la précision et la vitesse sont formulées. Le raisonnement dans ce cas est tributaire du comportement de la fonction sign en fonction de la surface de glissement. L'utilisation de ces entrées permet de maîtriser le

comportement du système au moment où nous cherchons à stabiliser la position et la vitesse à la fois. La sortie qui représente le gain de commutation engendre un lissage de la commande par l'ajustement de ce paramètre à chaque itération du système en boucle fermée. Le contrôleur utilisé est composé de trois étapes telles que :

Étape 1: Fuzzification : nous définissons la première variable d'entrée par trois labels tel que {Pe (erreur Positive), Ze (erreur Zero) et Ne (erreur Negative)}. Nous avons utilisé des fonctions d'appartenance symétriques par rapport à l'axe des coordonnées. La deuxième variable floue est décrite par trois labels tel que : {P \dot{e} (variation d'erreur Positive), Z \dot{e} (variation d'erreur Zero) et N \dot{e} (variation d'erreur Negative) }. La distribution des valeurs floues sur les univers de discours normalisés entre [-1, +1] sont montrés dans la Figure VII.1 :

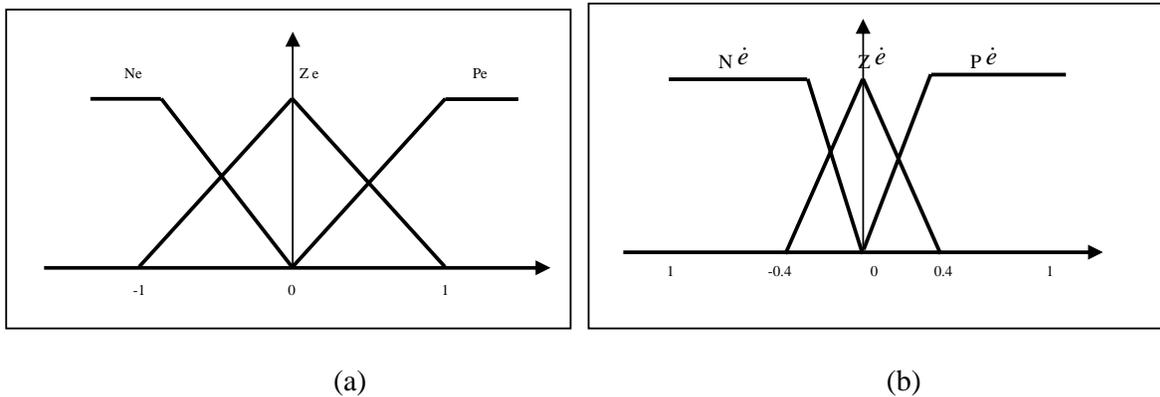


Figure.VII.1. (a) Les variables des entrées floues (a) l'erreur (e), (b) variation de l'erreur (\dot{e}).

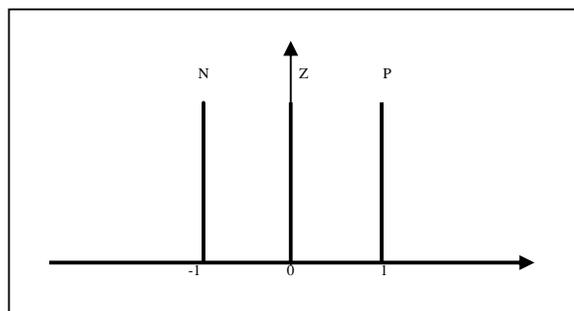


Figure. VII.2. Variable floue de la sortie η_{fuzzy} .

Les valeurs floues de la sortie sont au nombre de trois (03) de type singleton {P (Positive), Z (Zero), et N (Négative)}. Ces fonctions d'appartenance ont été normalisées entre [-1, +1].

Étape 2: Inférence: L'inférence est définie comme le noyau de la commande floue. Par l'usage de nos connaissances sur le système et sur la logique floue, nous avons déterminé la table d'inférence

la plus adéquate pour la commande de notre robot. A cet effet, le raisonnement flou utilisé est résumé dans le tableau VII.1.

e	Ne	Ze	Pe
\dot{e}			
$P\dot{e}$	Z	P	P
$Z\dot{e}$	N	Z	P
$N\dot{e}$	N	N	Z

Tableau VII.1 Table de règles floues.

Étape 3: Défuzzification : la défuzzification transforme l'information floue en une décision applicable sur le système physique. La méthode de défuzzification utilisée est le Centre de Gravité [72].

La relation entre les variables d'entrées/ sorties est donnée par la surface floue montrée dans la figure VII.3 :

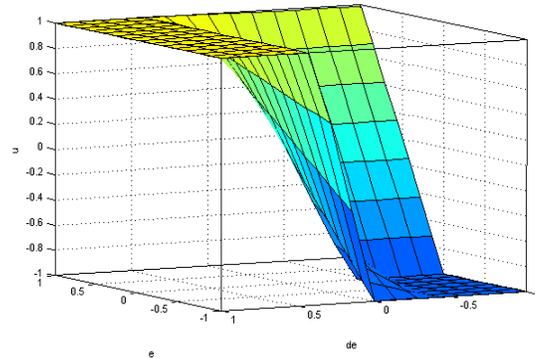


Figure VII.3. Surface floue.

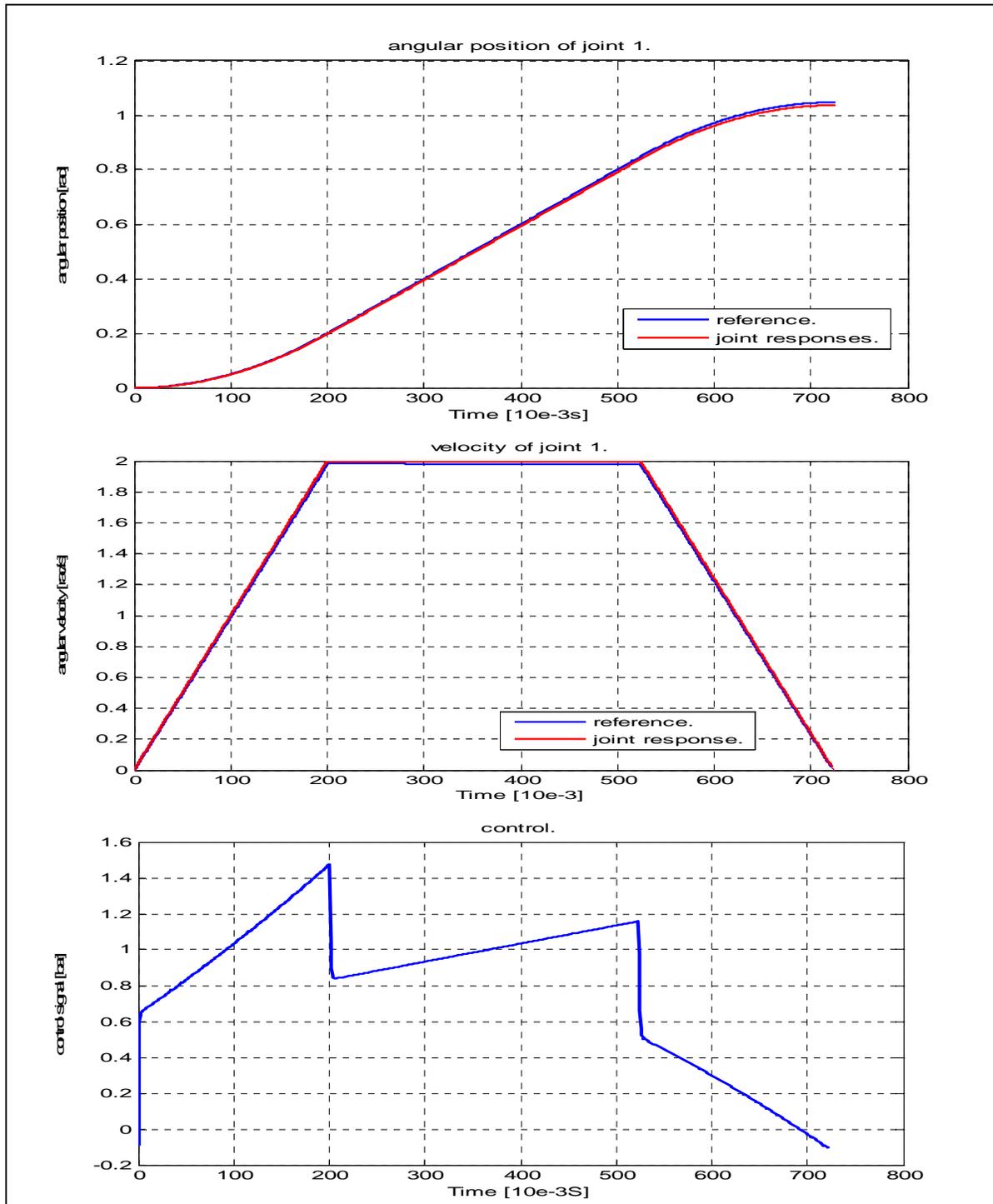
La combinaison de la commande par (TSMC) et le contrôleur flou est donnée par l'équation (VII.5) de la commande finale :

$$u_{FNTSMC} = -\frac{1}{b_i} (a_{i1} \cdot e_{i2} + a_{i0} \cdot e_{i1} + \beta \cdot \left(\frac{5}{7}\right) \cdot e_{i2}^{(3/5)} - \eta_{fuzzy} \cdot \text{sign}(s_i)) \quad (\text{VII.5})$$

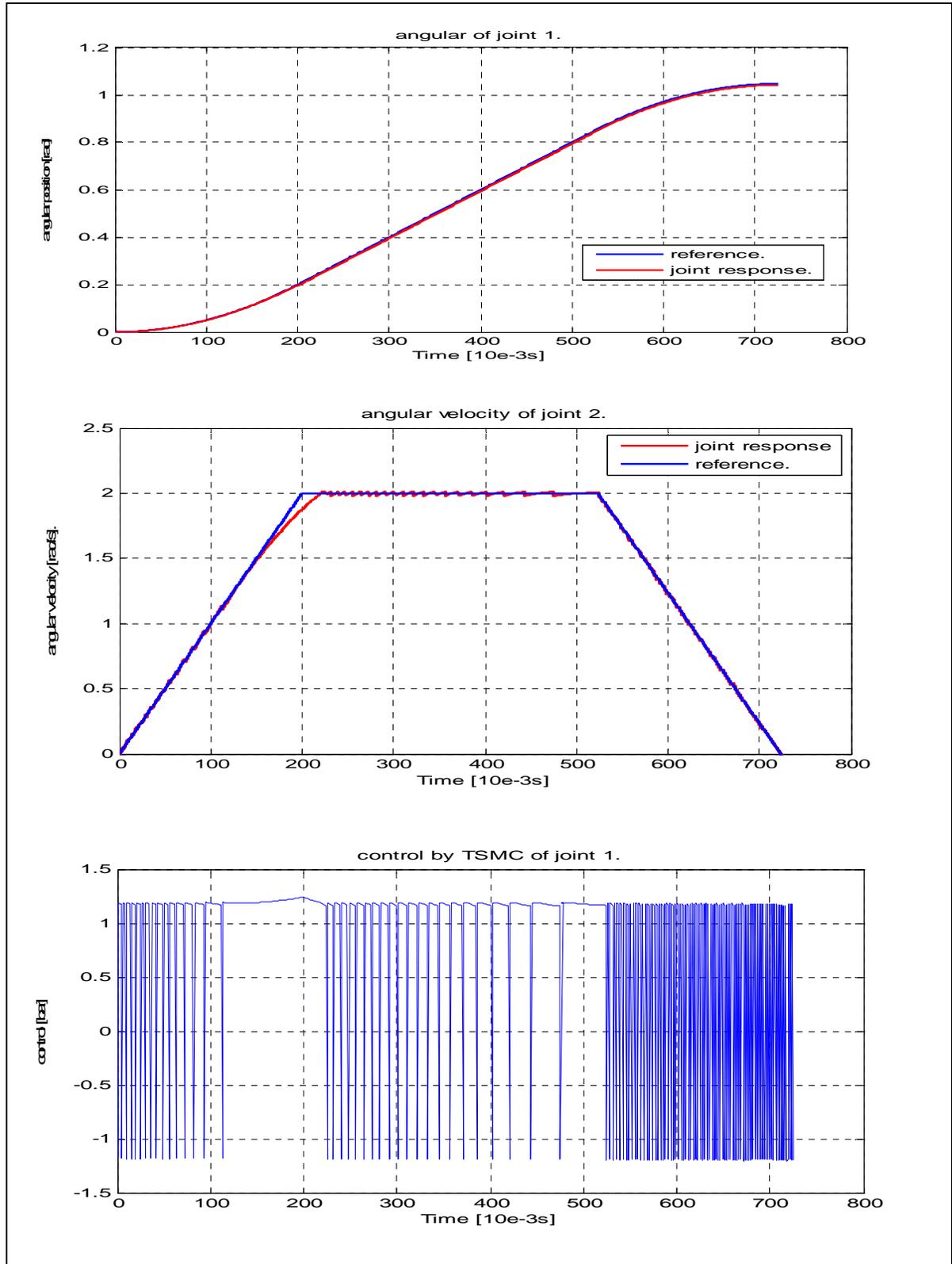
VII.4 RESULTATS DE SIMULATION ET DISCUSSION

La section suivante montre les résultats de simulation de l'application de la commande par FTSMC et TSMC sur le modèle de simulation du robot du CDTA. L'objectif de la simulation est de suivre

un polynôme de deuxième ordre dont la vitesse est imposée sous forme d'un trapèze. Les conditions initiales pour le robot sont $q_1=0$ et $q_2=0$ et l'objectif est d'arriver au point $q_{d1}=1.0472$ rad pour l'axe 1 et $q_{d2}=0.785$ rad pour l'axe 2. Dans tous les cas, la vitesse a été choisie égale 2 rad/s. Les résultats de simulation pour la commande FTSMC sont présentés dans les figures (VII.4.a et VII.5.a). Afin de prouver la supériorité de la commande proposée, le cas classique est montré dans les figures (VII.4.b et VII.5.b).

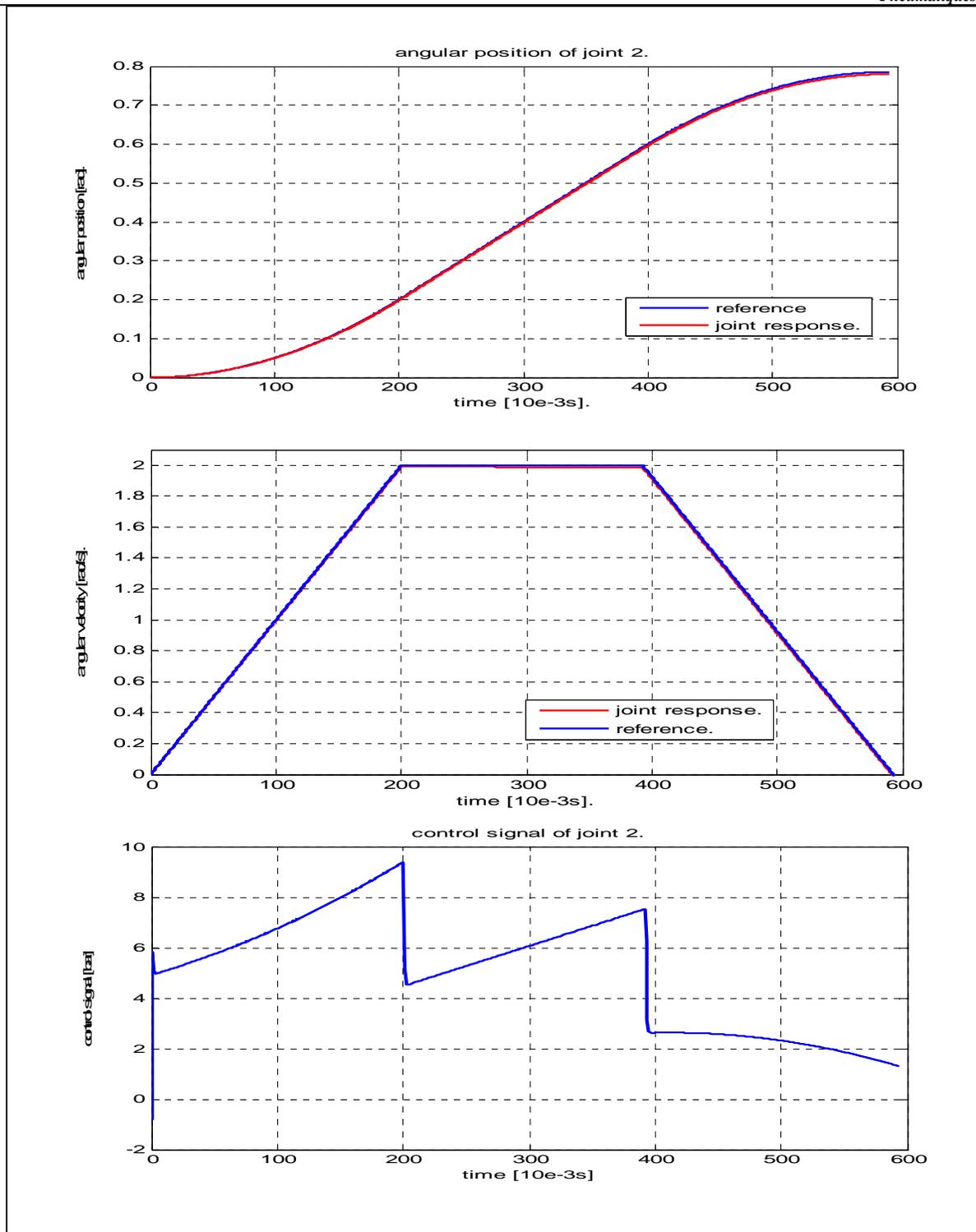


(a)



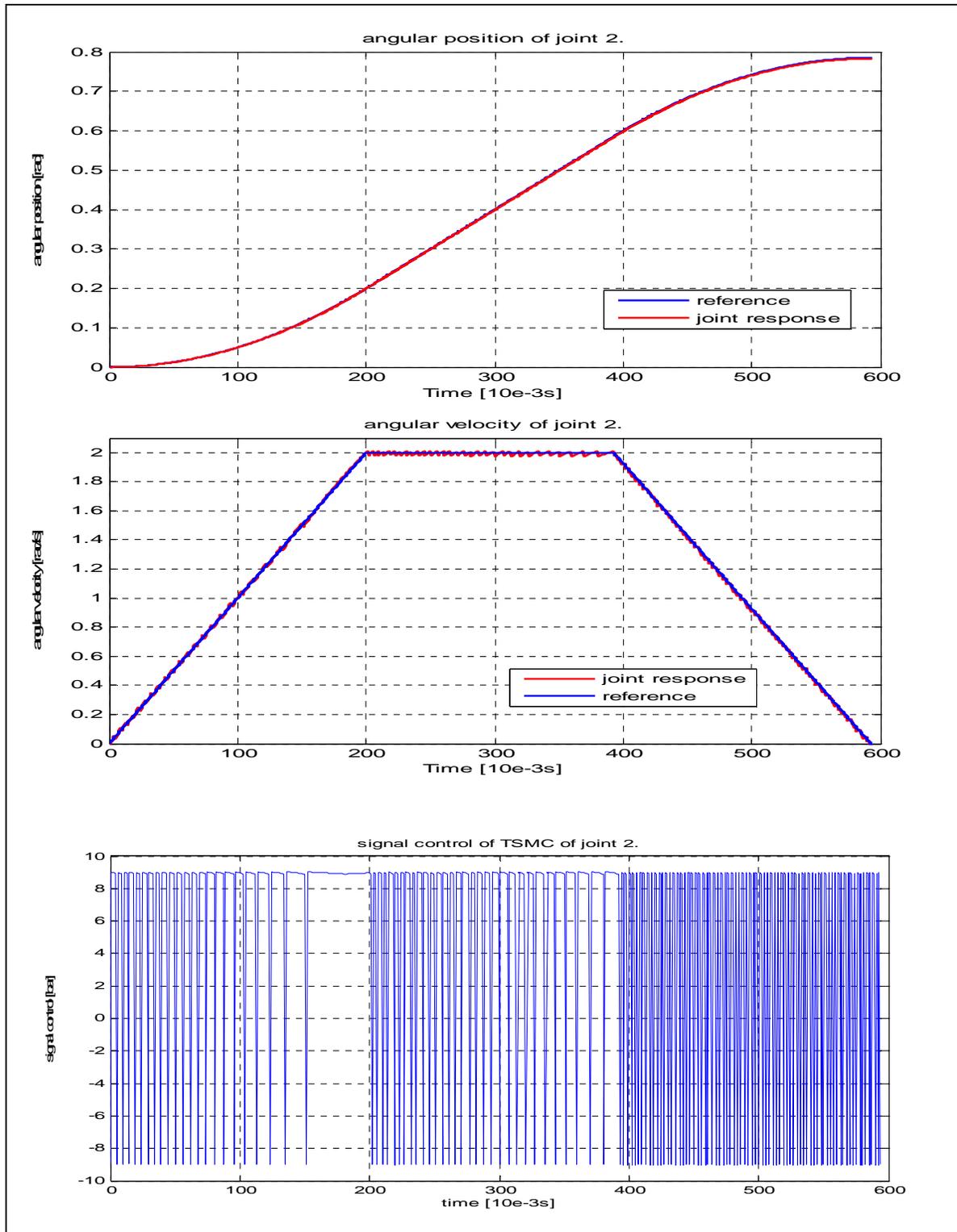
(b)

Figure. VII.4. Commande de l'axe 1 par (a) Terminal Sliding Mode Flou, (b) Terminal sliding mode.



(a)

Figure. VII.5. Commande de l'axe 2 par (a) Terminal Sliding Mode Flou, (b) Terminal sliding mode.



(b)

Figure. VII.5. Commande de l'axe 2 par (a) Terminal Sliding Mode Flou, (b) Terminal sliding mode.

Nous observons un signal de commande lisse dans le cas de la commande par (FTSMC) comparé avec (FTSM). Le phénomène de broutement est complètement éliminé. Nous avons obtenu une réponse parfaite avec une grande vitesse dans les deux cas de commande et pour les deux axes du robot. Cependant, la commande utilisée ici est basée sur un modèle identifié, ce dernier ne permet pas d'avoir une commande équivalente assez proche du cas réel, ce qui ne va pas donner par conséquent de bons résultats dans un cas pratique. Pour résoudre ce problème, nous avons fait appel à une technique basée sur la méthode d'estimation à temps de retard [38]. La technique ainsi que son application en simulation et en expérimentation seront présentées dans la section VII.5.

VII.5 COMMANDE PAR TERMINAL SLIDING MODE FLOU ADAPTATIF DU ROBOT ANTROPOMORPHE DE L'INSA DE TOULOUSE

L'objectif de notre étude est d'utiliser l'approche par TSM pour la synthèse d'une commande robuste vis-à-vis de la modélisation du robot. Cette commande permettant de forcer les articulations du robot à suivre les trajectoires désirées pour un robot manipulateur à n -ddl décrit par le modèle dynamique donné par l'équation (I.7). La commande développée doit prendre en considération la dynamique du robot, mais sans l'utiliser et doit être stable au sens de Lyapunov. La synthèse de la commande par TSM repose sur deux principales étapes, le choix de la variable de glissement et la synthèse de la loi de commande. La commande par TSM choisie est basée sur une notion connue dans la littérature sous le nom d'estimation par temps de retard (Delay-Time Estimation (TDE) en Anglais). La TDE a été introduite par les travaux de K. Youcef Toumi et al. [39] dont l'idée principale était d'estimer la dynamique inconnue et les perturbations par l'utilisation des informations intentionnelles précédentes [38]. Pour assurer une fonction TDE correcte, le temps L -retard doit être suffisamment petit. La plupart du temps, L est choisi pour être le temps d'échantillonnage. Il est intéressant de noter que la TDE a été proposée dans la fin des années 1980, quand il a été rendu possible d'obtenir un échantillonnage assez petit grâce à la fréquence du développement important des microprocesseurs et des périphériques informatiques. L'efficacité de la méthode TDE a été démontrée dans [38] et [39] avec une erreur dynamique linéaire. Dans le cas du robot manipulateur, cette démarche est donnée comme suit :

VII.6 COMMANDE PAR TERMINAL SLIDING MODE DU ROBOT DE L'INSA

Soit la matrice diagonale de valeurs constantes réelles $\bar{M} = \text{diag}(m_i)$ et $i=1 \dots n$, par l'usage de \bar{M} , l'équation du robot (I.7) peut être réécrite sous la forme suivante :

$$M\ddot{q} + N(q, \dot{q}, \ddot{q}) = \tau \quad (\text{VII.6})$$

Avec :
$$N(q, \dot{q}, \ddot{q}) = \left[M(q) - \bar{M} \right] \ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(q, \dot{q}) + \tau_d$$

La variable de glissement terminal non singulière pour un robot manipulateur à n -ddl est donnée par [21,41,56]:

$$S = e + Ke^{\dot{e}^{p/q}} \quad (\text{VII.7})$$

$K = k_{ii}$ est une matrice diagonale de valeurs positives,

p et q sont des valeurs entières positives et $1 < p/q < 2$ pour chaque élément de la variable de glissement.

e est le vecteur d'erreur en position et \dot{e} son vecteur de vitesse.

Par l'utilisation de (VII.6) et (VII.7), la commande par TSM peut être écrite comme suit :

$$\tau = \bar{M}u + \hat{N}(q, \dot{q}, \ddot{q}) \quad (\text{VII.8})$$

Où:

$$u = \ddot{q}_d + \frac{q}{p} K^{-1} e^{2-q/p} + K_{sw} \text{Sgn}(S) \quad (\text{VII.9})$$

K_{sw} est une matrice diagonale avec des éléments positifs,

$\hat{N}(q, \dot{q}, \ddot{q})$ est déterminée par la méthode d'estimation par temps de retard TDE [38-39] telle que:

$$\hat{N}(q, \dot{q}, \ddot{q}) = N(q, \dot{q}, \ddot{q})_{t-L} \quad (\text{VII.10})$$

$\hat{N}(q, \dot{q}, \ddot{q})$ représente l'estimation de $N(q, \dot{q}, \ddot{q})$, \bullet_{t-L} désigne la valeur de \bullet à l'instant $t-L$, et L est la période d'échantillonnage. Cette hypothèse est valide seulement avec une fréquence d'échantillonnage suffisamment rapide, car un système de contrôle numérique peut être considéré comme un système continu lorsque la cadence d'échantillonnage est 30 fois plus rapide que la bande passante du système [40]. De (VII.10) et (VII.8), nous pouvons obtenir :

$$N(q, \dot{q}, \ddot{q})_{t-L} = \tau_{t-L} - \bar{M}\ddot{q}_{t-L} \quad (\text{VII.11})$$

Ainsi, avec la combinaison de (VII.8) ET (VII.11), la loi de commande est exprimée par :

$$\tau = \tau_{t-L} - \bar{M}\ddot{q}_{t-L} + \bar{M} \left[\ddot{q}_d + \frac{q}{p} K^{-1} \dot{e}^{2-q/p} + K_{sw} \text{Sgn}(S) \right] \quad (\text{VII.12})$$

VII.7 PROBLEME DE BROUTEMENT

Un régime glissant idéal requiert une commande pouvant commuter à une fréquence infinie. Évidemment, pour une utilisation pratique, seule une commutation à une fréquence finie est possible. Ainsi, durant le régime glissant, les discontinuités appliquées à la commande peuvent entraîner un phénomène de broutement, appelé réticence ou "chattering" en anglais. Celui-ci se caractérise par de fortes oscillations des trajectoires du système autour de la surface de glissement [57]. La commande donnée par l'équation VII.12 souffre de problème de broutement causé par la partie discontinue de ce contrôleur. De nombreuses études ont été effectuées dans le but de réduire, voire d'éliminer ce phénomène. L'une des solutions proposées consiste à remplacer la fonction sign par une fonction de saturation. Dans ce cas, la commande (VII.12) devient :

$$\tau = \tau_{t-L} - \bar{M}\ddot{q}_{t-L} + \bar{M} \left[\ddot{q}_d + \frac{q}{p} K^{-1} \dot{e}^{2-q/p} + K_{sw} \text{Sat}(S, \Phi) \right] \quad (\text{VII.13})$$

Où

$$\text{Sat}(S) = \begin{cases} \frac{S_i}{|S_i|} & \text{if } |S_i| \geq \Phi_i \\ \frac{S_i}{\Phi_i} & \text{if } |S_i| \leq \Phi_i \end{cases}$$

Il est bien utile ici de mentionner qu'il existe une alternative à la commande par TSM d'un robot manipulateur qui a été donnée dans [41]. Cette approche est basée sur le modèle mathématique du robot, ainsi, les paramètres des matrices d'Inertie, de Coriolis et de l'effet Gravitationnel sont nécessaires. Cependant, ces paramètres ne sont pas disponibles dans le cas du robot de l'INSA. On remarque aussi que la complexité de ce contrôleur engendre des difficultés au moment de l'implémentation :

$$\tau = M_0(q)\ddot{q}_d + C_0(q, \dot{q})\dot{q} + G_0(q) + F_0(q, \dot{q}) + u_a + u_b \quad (\text{VII.14})$$

avec

$$u_a = -M_0(q) \frac{q}{p} K^{-1} \dot{e}^{2-q/p}$$

$$u_b = -\frac{q}{p} \frac{\left[S^T K^{-1} \text{diag} \left(\dot{e}^{2-q/p} \right) M_0^{-1}(q) \right]}{\left\| S^T K^{-1} \text{diag} \left(\dot{e}^{2-q/p} \right) M_0^{-1}(q) \right\|} \times \|S\| \times \left\| S^T K^{-1} \text{diag} \left(\dot{e}^{2-q/p} \right) M_0^{-1}(q) \right\| \times (b_0 + b_1 \|q\| + b_2 \|\dot{q}\|)$$

Le remplacement de la partie discontinue par un SIF est l'objet de la section VII.8. Deux hybridations ont été conçues. La première est classique, elle consiste en une hybridation entre le TSM et un SIF de type-1. Pour la deuxième commande, la position des fonctions d'appartenance de la sortie précédant SIF seront adaptées à l'aide d'un algorithme de descente du gradient.

VII.8 CONTROLEUR FLOU POUR L'ATTENUATION DU BROUITEMENT [43]:

L'objectif de cette partie est d'atténuer l'effet de brouitement et d'améliorer les performances de la commande du robot à MAPs. Pour ce faire, nous avons choisi un système d'inférence floue de type-1 avec une seule entrée et une seule sortie. L'entrée est la surface de glissement terminal ($S = e + K\dot{e}^{p/q}$) et la sortie remplace la partie discontinue de la commande par mode de glissement ($k\text{sign}(S)$). Après calcul, la sortie sera substituée dans la commande globale donnée par l'équation (VII.12). Les labels utilisés pour la variable floue surface sont : {surface négative (N), surface zéro (Z) et surface positive (P)}. Nous avons choisi trois fonctions d'appartenance de type singleton pour la variable de sortie. Les labels utilisés dans ce cas sont (Nb Négative, Z Zéro, et Pb Positive). La figure (VII.6) représente la distribution des valeurs floues sur les univers de discours :

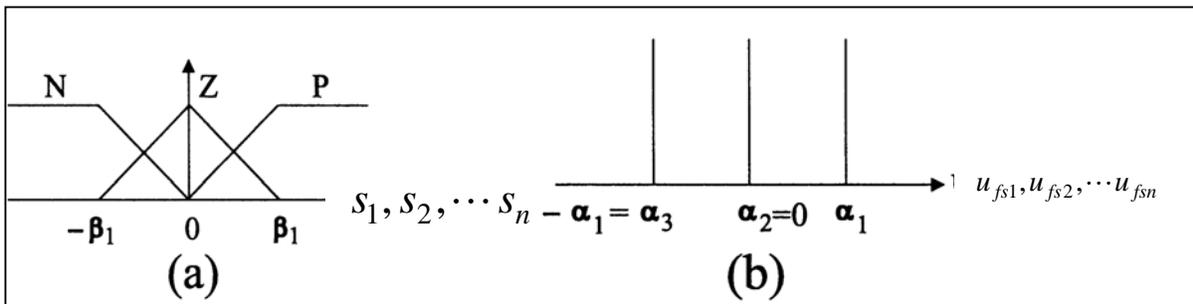


Figure VII.6. Les entrées/ sortie floues de type-1.

VII.9 COMMANDE ADAPTATIVE FLOUE DU ROBOT A MUSCLES ARTIFICIELS PNEUMATIQUES

Le schéma du contrôleur adaptatif par terminal glissement mode flou est montré dans la figure

VII.7. Les fonctions d'appartenance utilisées sont données dans la figure VII.6.

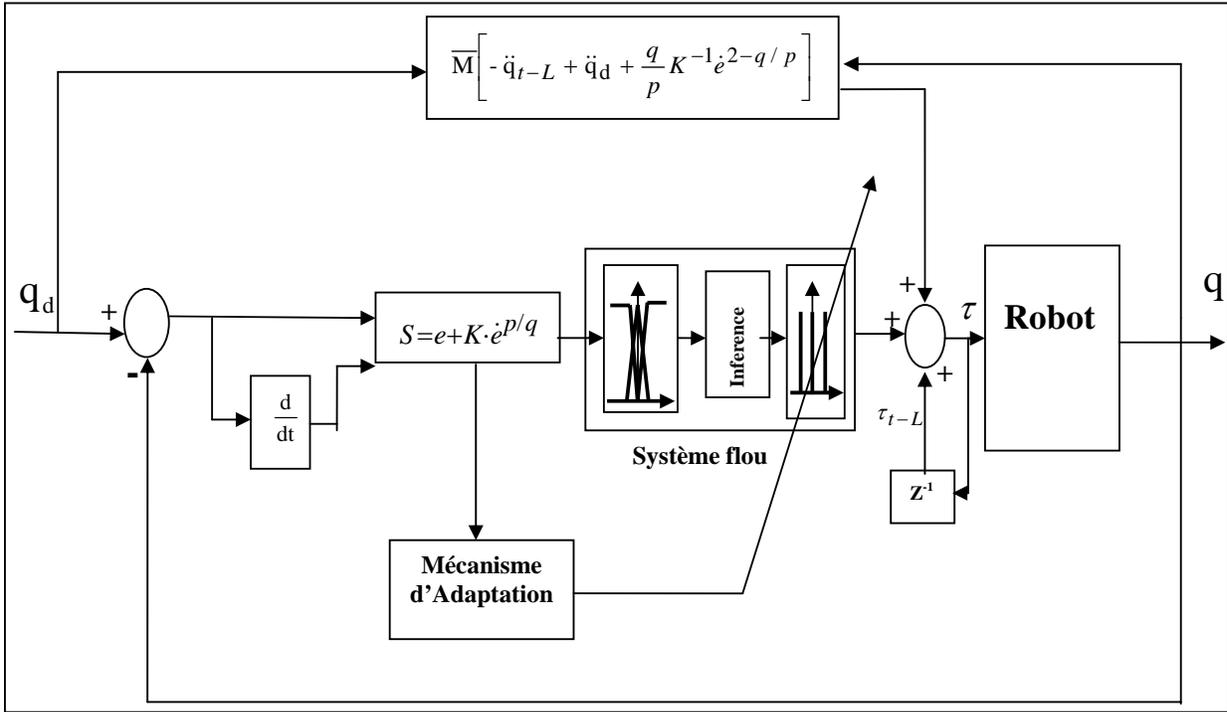


Figure VII.7. Schéma bloc de la commande par terminal sliding mode flou adaptatif.

Les vecteurs d'entrée/ sortie des variables floues sont données par les deux équations suivantes:

$$S = [s_1, s_2, \dots, s_n]$$

$$u_{fs} = [u_{fs1}, u_{fs2}, \dots, u_{fsn}] \quad (\text{VII.15})$$

Les règles floues si-alors du contrôleur de commutation sont conçues comme suit :

- 1) *If is s_j , then is u_{fsj} .*
- 2) *If is s_j , then is u_{fsj} .*
- 3) *If is s_j , then is u_{fsj} .*

Afin de simplifier le calcul mathématique, les fonctions d'appartenance d'entrée/sortie sont conçues pour être symétriques par rapport à l'axe de coordonnées. La sortie du contrôleur flou de commutation après défuzzification est donnée par l'équation générale (VII.16):

$$u_{fs} = \sum_{i=1}^3 g_i \left(\text{diag} \left(\sum_{i=1}^3 L_i(S) \right) \right)^{-1} L_i(S) \quad (\text{VII.16})$$

Avec:

$$L_i(S) = \begin{bmatrix} l_{fi}(S_1) \\ \vdots \\ l_{fi}(S_m) \end{bmatrix} \quad (\text{VII.17})$$

Où L_i est un vecteur colonne des fonctions d'appartenance floue. La distribution des fonctions d'appartenance pour l'entrée et la sortie permet l'activation d'une seule fonction d'appartenance de l'entrée et de la sortie à la fois. Dans ce cas, le terme est égal à $\text{diag} \left(\sum_{i=1}^3 L_i(S) \right) = 1$. A cet effet, l'équation VII.16 deviendra comme suit :

$$u_{fs} = \sum_{i=1}^3 g_i L_i(S) \quad (\text{VII.18})$$

VII.9.1 Mécanisme d'Adaptation en Ligne

Afin de simplifier la conception du contrôleur flou de commutation, les valeurs de la sortie floue sont spécifiées pour être symétriques à zéro (Figure VII.6.b.). Cette configuration permet d'avoir une commande de commutation nulle lorsque le système est en mode de glissement idéal. De plus, si la valeur d'appartenance de la fonction d'appartenance n'est pas nulle, alors la valeur d'appartenance de la fonction de symétrie est zéro. Ainsi, la sortie de la commande floue peut être simplifiée pour être comme suit :

$$\begin{aligned} u_{fs} &= \sum_{i=1}^3 g_i L_i(S) \\ &= \alpha_1 \text{diag}(\text{sign}(S)) L(S) \end{aligned} \quad (\text{VII.19})$$

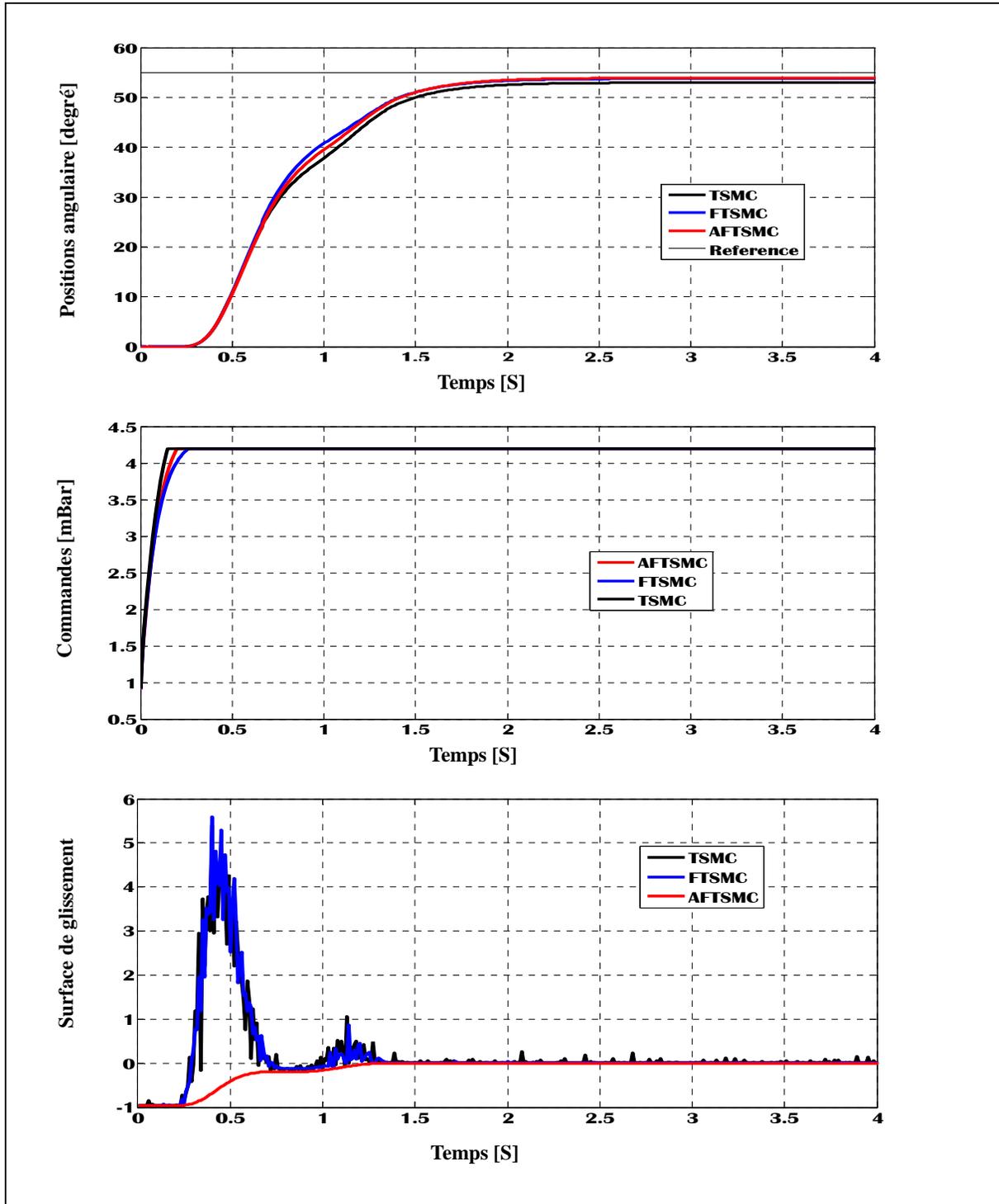
Les paramètres du contrôleur de commutation flou sont ajustés en ligne, et afin d'avoir une commande optimale, les gains α_1 de la commande floue (VII.18) ont été ajustés de la manière suivante :

$$\begin{aligned}\delta\alpha_1 &= -\eta_1 \frac{\partial S^T S}{\partial \alpha_1} \\ &= -\eta_1 S^T \text{diag}(\text{sign}(S))L(S)\end{aligned}\tag{VII.20}$$

Où η_1 est un gain d'apprentissage positif à déterminer par tâtonnement au moment de l'expérimentation.

VII.9.2 Résultats Expérimentaux

Plusieurs expériences ont été réalisées, car nous avons appliqué sur les deux axes du robot (épaule et coude) la commande par TSM, son hybridation par un SIF de type-1 ainsi que le cas adaptatif présenté auparavant. L'objectif des commandes proposées est d'étudier le comportement du robot face aux problèmes de non linéarité des actionneurs, du couplage des axes, de l'effet gravitationnel et à la robustesse des commandes vis-à-vis d'une variation de charge. En mode de régulation, l'objectif est d'atteindre la position 55° et 100° , pour l'épaule et le coude respectivement. La période d'échantillonnage est 10 ms et la charge utilisée pour les tests de robustesse est de 200 g. En mode poursuite de trajectoire, nous avons réalisé des expériences seulement sur le coude à cause des problèmes physiques liés à l'épaule (présentés dans le chapitre 1). Les résultats en régulation sont représentés dans les figures VII.8 :

Figure VII.8. Réponse du robot à une consigne égale à $(55^\circ, 100^\circ)$

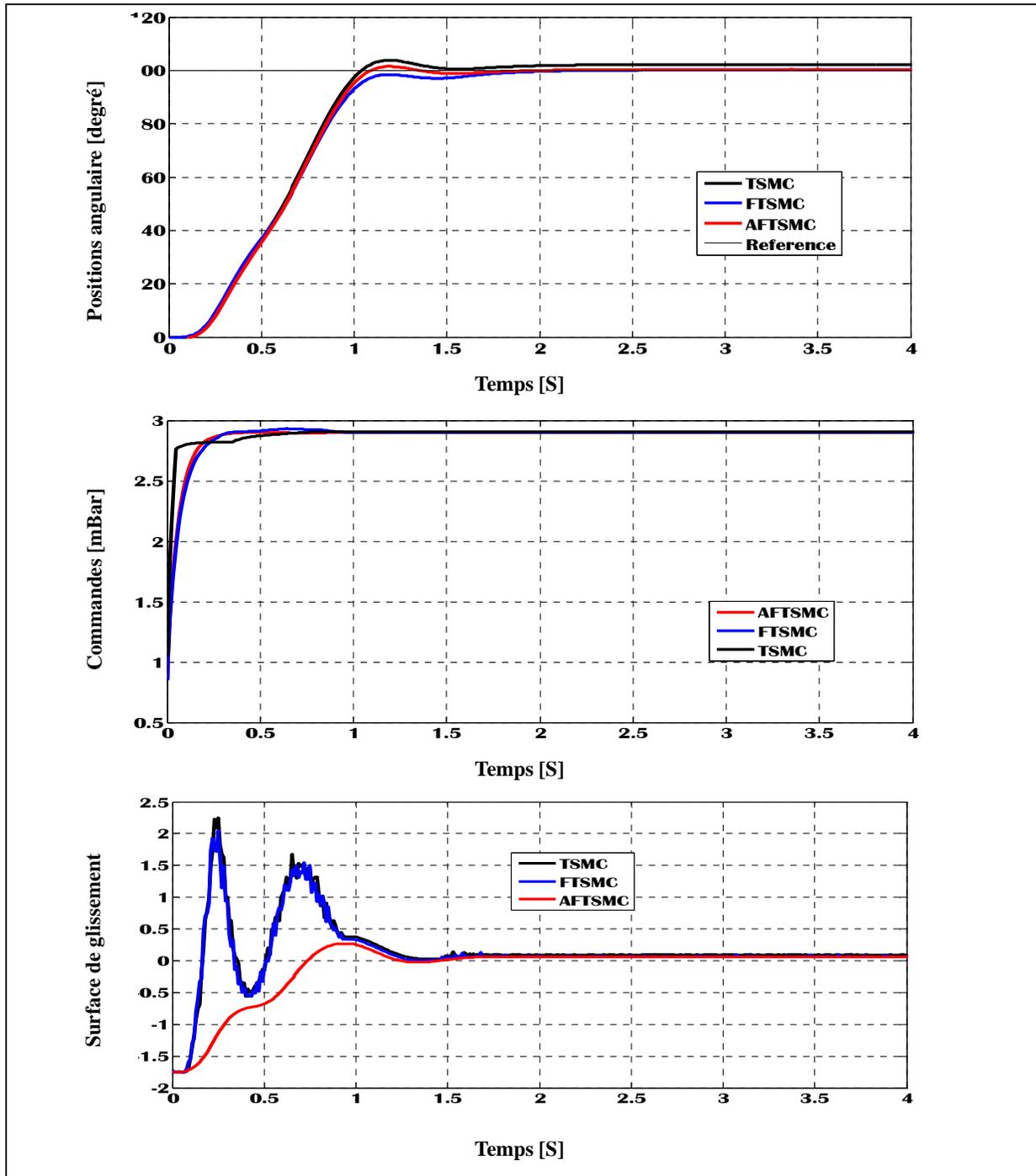


Figure VII.8. Réponse du robot à une consigne égale à $(55^\circ, 100^\circ)$

Poursuite de trajectoire : pour ce cas, nous voulons que l'épaule suive la trajectoire $q_{d2}(t) = 50 \cdot \sin(2 \cdot \pi \cdot f \cdot t) + 50$, avec une fréquence $f = 0.3 \text{ ms}$, les résultats obtenus sont donnés par la figure VII.9 :

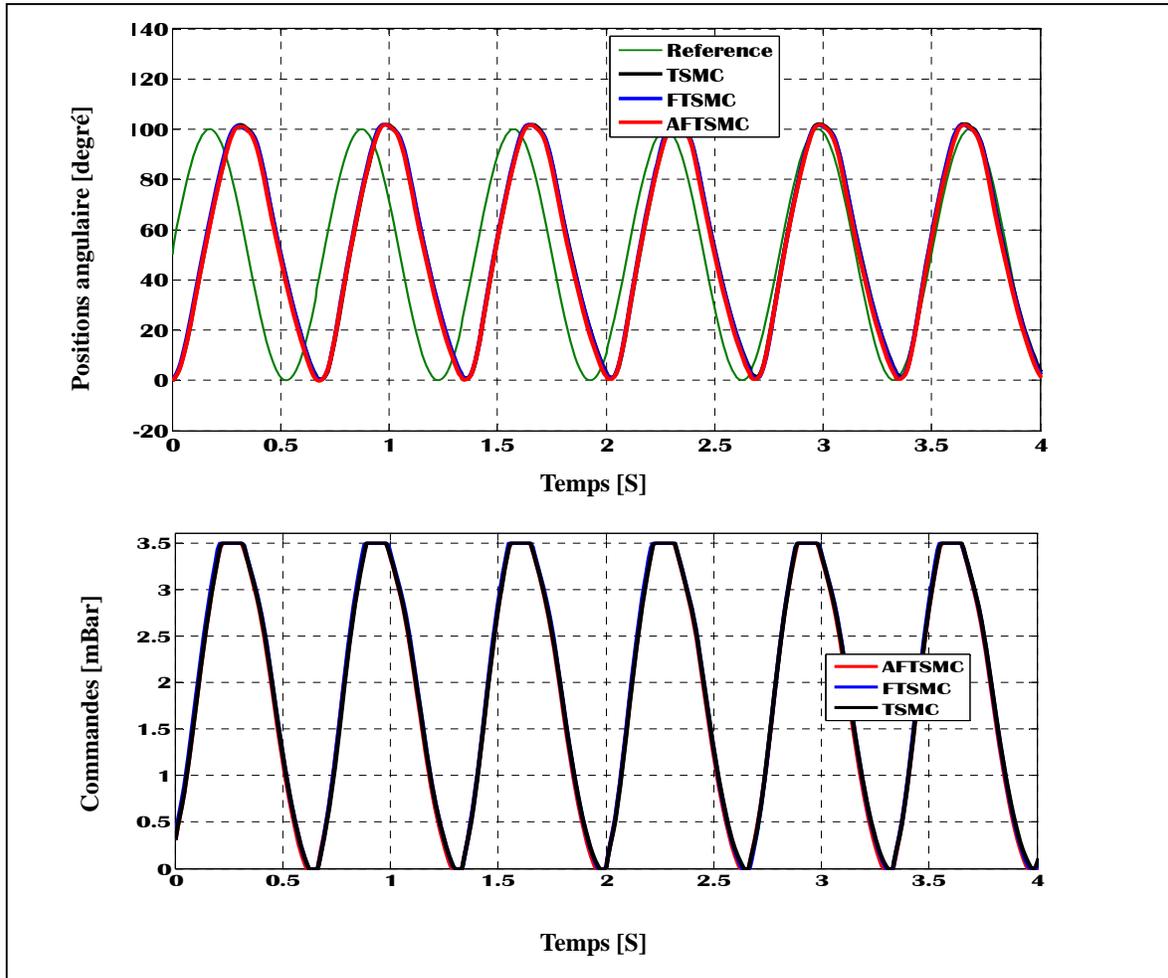


Figure VII.9. Poursuite de trajectoire du coude ($q_d = 100^\circ$).

Le bras du robot à MAPs à 7-*ddl* a été simplifié à un robot à deux axes, la commande à structure variable par terminal sliding mode et celle hybride par un système d'inférence floue adaptative et non adaptative ont été appliquées en expérimentation pour le contrôle en position du robot manipulateur à MAP de l'INSA. Nous remarquons dans les résultats un suivi convenable de la trajectoire de position imposée sur les deux axes. Une nette amélioration a été obtenue avec la commande adaptative, ceci est observé dans les résultats des réponses en position des deux axes. Une nette amélioration en précision et aussi par rapport à l'atténuation de broutement ont été notées. Par contre, en mode poursuite nous n'avons pas obtenu des résultats remarquables. Pour cette raison, un test de robustesse par rapport à une perturbation permanente (une charge) est peut être intéressant, ce test fera l'objet de la section suivante.

VII.10 TESTS DE ROBUSTESSE

VII.10.1 Mode de régulation : une charge de 200g a été utilisée pour tester la robustesse de la commande proposée, les résultats obtenus sont montrés dans les figures VII.10, VII.11 en mode régulation et poursuite de trajectoire, respectivement :

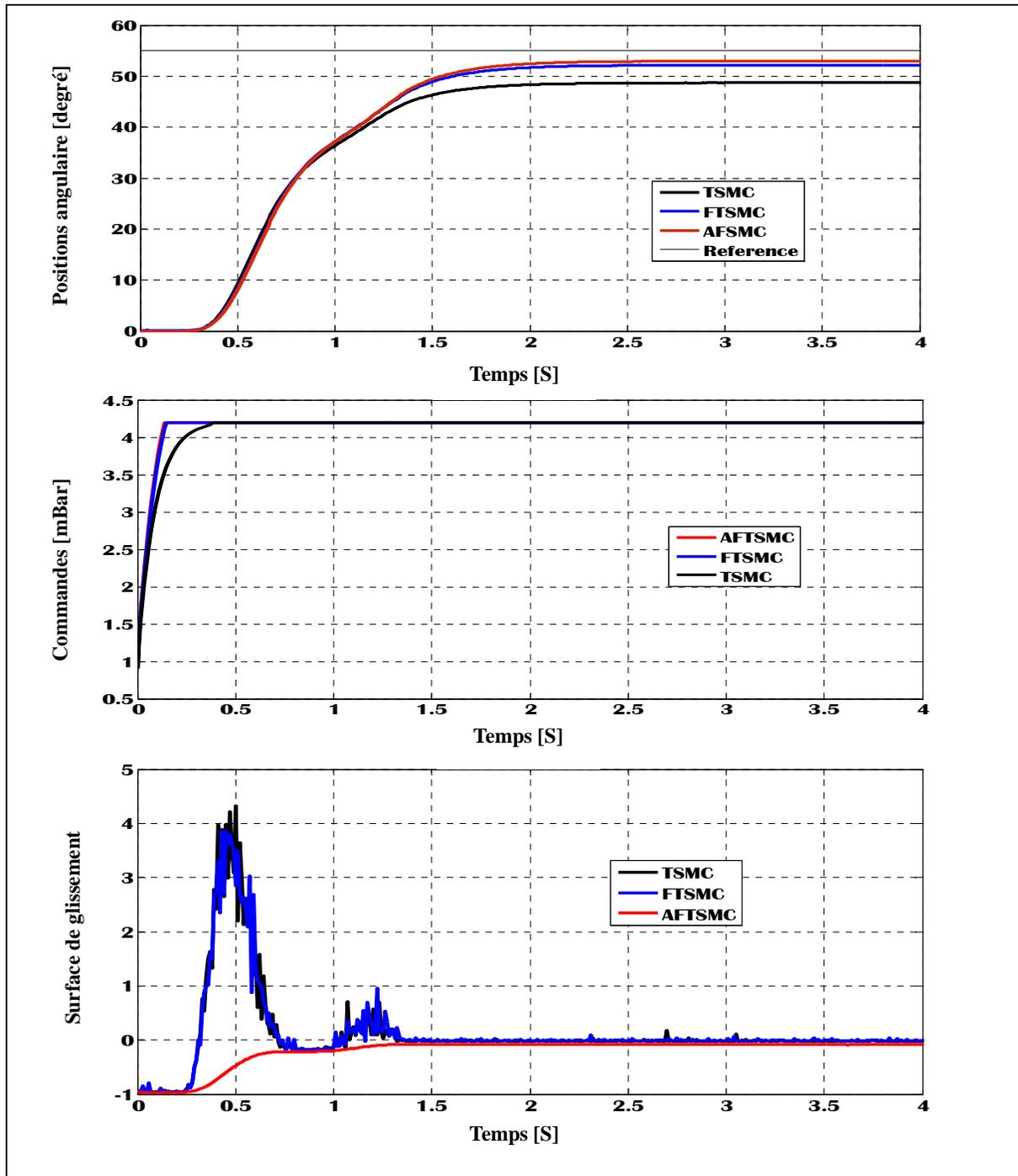


Figure VII.10. Robustesse du robot vis-à-vis d'une charge de 200 g en mode régulation.

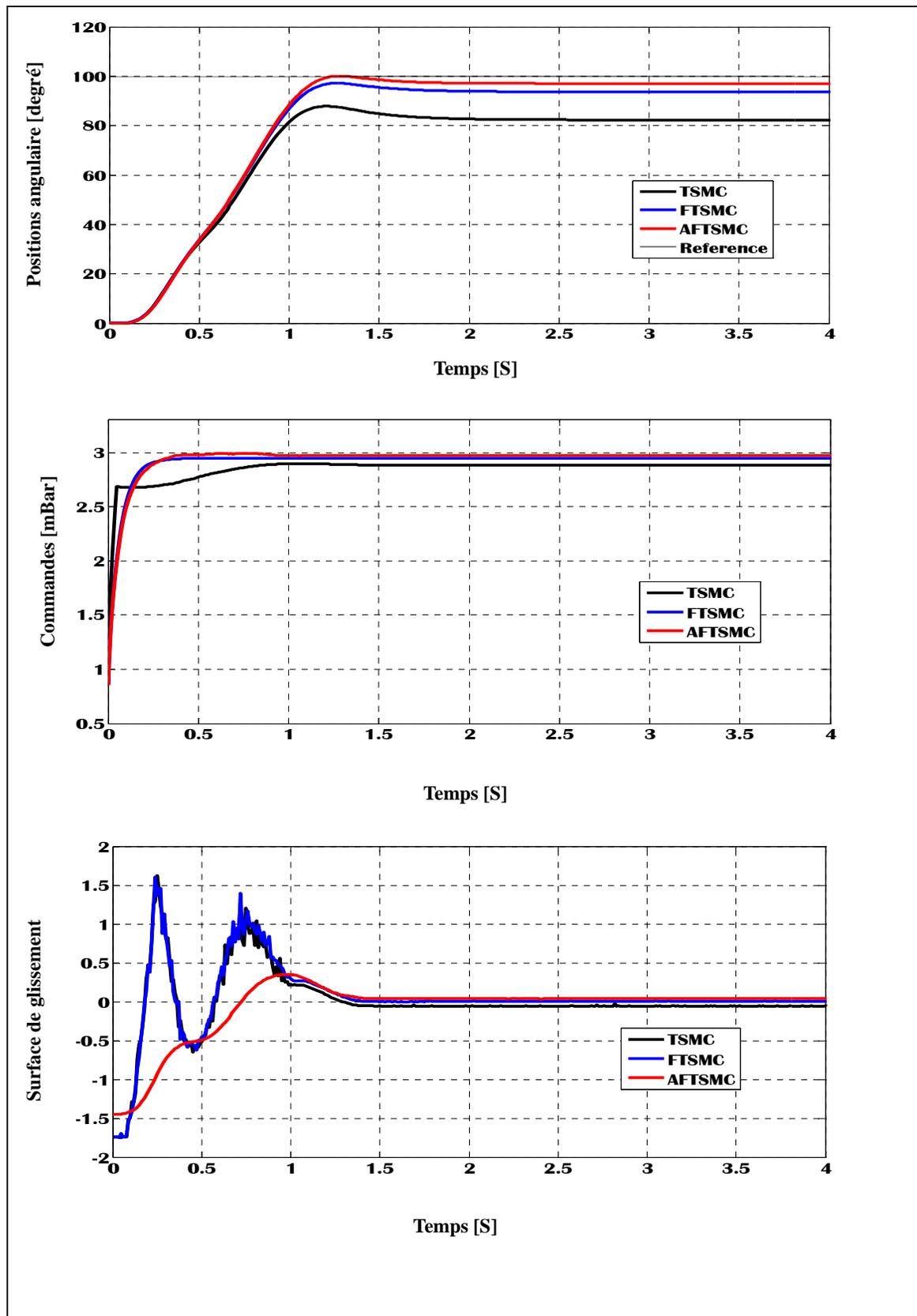


Figure VII.10. Robustesse du robot vis-à-vis d'une charge de 200 g en mode régulation.

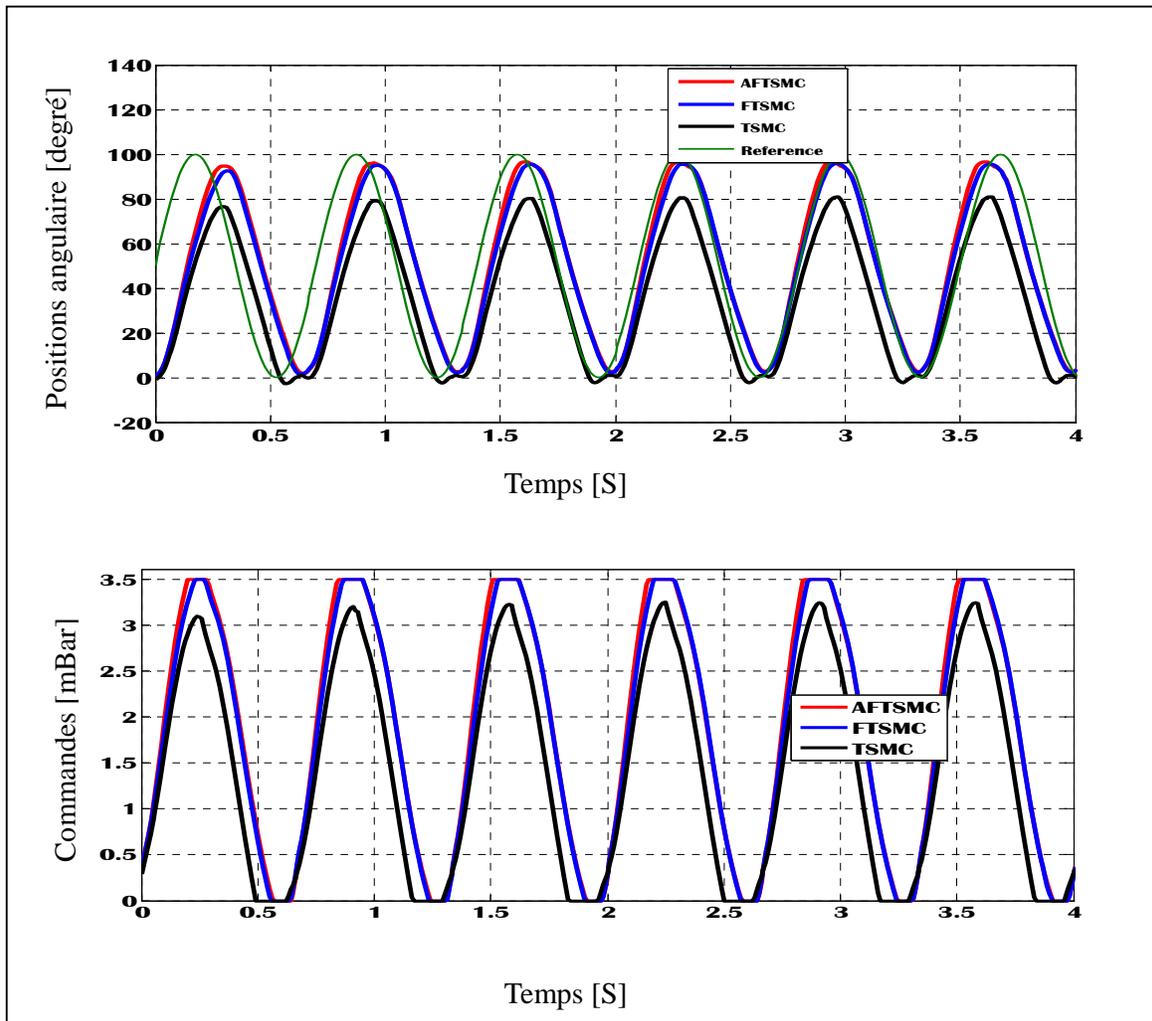


Figure VII.11. Robustesse du robot vis-à-vis d'une charge de 200 g en mode poursuite de trajectoire.

La commande par terminal sliding mode flou adaptatif a montré une meilleure robustesse comparée avec les deux autres cas (commande par terminal sliding mode et commande par mode de glissement flou statique). Que ce soit en mode de régulation ou en poursuite de trajectoire, la commande adaptative FTSM adaptatif a pu maintenir un bon suivi de trajectoire en présence de la charge.

6.7. CONCLUSION

Dans ce travail, nous avons proposé une combinaison de la commande par TSM et un système d'inférence flou de type-1 statique et adaptatif. Cette approche a été proposée afin d'avoir

les meilleures performances possibles et aussi pour améliorer la robustesse du système en boucle fermée face à la variation de charge et atténuer le phénomène de broutement. Les objectifs de l'étude sont de contrôler les positions angulaires d'un bras de robot à 2-DOF actionné entièrement par des MAPs. L'approche proposée a fournit une meilleure précision et un niveau de robustesse très élevé.

Conclusion Générale

Les travaux menés dans cette thèse ont conduit à la proposition, la simulation et l'expérimentation de plusieurs lois de commande pour des robots manipulateurs à muscles artificiels pneumatiques. Au début, nous avons présenté les MAPs afin de donner une idée précise des MAPs les plus répandus, leurs avantages et inconvénients, leur principe de fonctionnement, les applications les plus connues, ainsi que les modèles théoriques et les commandes déjà appliquées. Du point de vue modélisation des systèmes étudiés, dans la chapitre I, nous avons proposé deux modèles de simulation pour les deux robots du CDTA et celui de l'INSA-LAAS de Toulouse. Le premier modèle concerne le robot du CDTA, qui est de forme linéaire découplée, il a été obtenu après une identification par l'usage de Signaux Binaires Pseudo Aléatoires. Le modèle issu de l'identification était de type multi-variable. A travers une étude du découplage par l'emploi de la méthode appelée « Gain Array Methodology », nous avons montré que le système dans sa configuration peut être considéré comme découplé. Pour le cas du robot à 7-*ddl* de l'INSA, nous avons choisi de le simplifier à un robot à 2-*ddl*, ceci est justifié par le fait que le modèle de muscle est difficile à élaborer à cause des non linéarités du robot, les variations paramétriques et la dynamique du muscle. Ces phénomènes ont été considérés comme des perturbations qui doivent être compensés par une ou plusieurs commandes robustes.

Afin de bien se situer par rapport à la théorie de la logique floue et de mode de glissement, les chapitres II et III ont été consacrés à leurs fondements théoriques. La commande par « Terminal Sliding Mode Control » a été étudiée dans le chapitre II, où nous avons essayé de présenter l'essentiel de cette technique de commande. Dans le chapitre III, nous avons présenté une brève comparaison entre la logique floue classique dite de type-1 et celle de type-2.

Le chapitre IV a été consacré à la présentation d'un cas classique d'hybridation entre la commande par mode de glissement et un système d'inférence floue type-1, ceci afin de montrer les limites de ces techniques.

La mise en œuvre expérimentale de la commande par logique floue de type-2 a fait l'objet du chapitre V, une amélioration de cette commande par l'incorporation d'un gain adaptatif de la sortie a été proposée en se basant sur la méthode de descente du gradient. Par la suite, nous avons réalisé une commande floue de type-2. Cette fois-ci, la base de règles a été déduite à partir de la condition d'attractivité de la commande par mode de glissement, ce qui a permis de garantir la stabilité et la

robustesse de la commande implémentée, les résultats obtenus ont montré une nette amélioration de la commande par logique floue de type-2.

La commande par mode de glissement se compose d'une manière générale d'une commande équivalente et d'une commande discontinue. La synthèse de la première nécessite des connaissances assez précises sur la dynamique du système à commander. En ce qui concerne la commande du robot manipulateur à MAPs, cette connaissance est difficile à obtenir. A cet effet, nous avons opté pour son estimation par l'usage d'un « réseau de neurones dit réseau de neurones de fonctions à bases radiales ». Le choix de ce réseau est dicté par plusieurs avantages telle qu'une architecture claire et simple, la facilité de mise en œuvre, le temps d'exécution rapide car cette méthode ne souffre pas de la lourdeur de l'algorithme de rétro-propagation, etc. L'emploi de réseau de neurones pour l'estimation de la commande équivalente a amélioré les performances obtenues. Néanmoins, l'effet de broutement reste encore un problème à résoudre. Afin de remédier à ce dernier, nous nous sommes intéressés à l'exploitation de la théorie des systèmes d'inférences floues, où deux contrôleurs flous ont été appliqués : un contrôleur flou de type-1 et un autre de type-2. La commande finale développée a été dénommée RBFT2FSMC où elle a montré une nette amélioration des performances comparée à la commande RBFT1FSMC. Parce que la commande développée a le profil d'une commande par mode de glissement, la stabilité et la robustesse ont été garanties *a priori*.

Les schémas de commande proposés dans les chapitres précédents ont été basés sur la commande par mode de glissement classique. Cette dernière ne garantit aucune convergence en temps fini des trajectoires du système commandé à ses points d'équilibre. La commande dite « Terminal Sliding Mode » est une nouvelle approche qui a été développée afin d'aboutir à une convergence en temps fini. Pour le cas du robot manipulateur, plusieurs algorithmes ont été proposés dans la littérature, leur principe est identique au cas classique, avec l'exception d'une variable de glissement non linéaire qui permet une convergence en temps fini, la commande terminal sliding mode nécessite d'avoir des connaissances sur la dynamique du système à commander. Comme nous l'avons évoqué le long de cette thèse, l'obtention de modèles pour les robots à MAPs est extrêmement délicate vu les phénomènes physiques qui les régissent. Afin d'appliquer la commande par terminal sliding mode sur le robot de l'INSA-LAAS de Toulouse, nous nous sommes penchés sur l'approche de commande basée sur la méthode dite « Méthode D'Estimation à Temps De Retard ». Cette approche contourne l'exigence de la modélisation du robot, et afin de surmonter le problème de broutement, nous avons testé un contrôleur flou de type-1 avec un ajustement des positions de fonctions d'appartenance de sortie afin d'avoir les meilleures performances possibles.

Comme perspectives nous nous proposons dans le futur l'ajout des observateurs pour l'estimation des informations qui sont obtenues dans ce travail par des dérivations numériques, notamment la variation de l'erreur, et donc par conséquent la surface de glissement. La variation de la surface de glissement est aussi est une variable très intéressante à estimer par un observateur. L'adaptation de robot du CDTA pour une application médicale ou pour qu'il soit une partie d'un robot humanoïde est aussi un objectif intéressant.

BIBLIOGRAPHIE

- [1] F. Daerden, D. Lefeber, "Pneumatic Artificial Muscles: actuators for robotics and automation", *European journal of mechanical and environmental engineering*, Vol. 47, No. 1, 2002, pp. 11-21.
- [2] C. P. Chou, B. H. Naford, "Measurement and Modeling of McKibben Pneumatic Artificial muscles", *IEEE Trans. Robotics Automation*, 12, 1996, pp. 90-102.
- [3] T. D. C. Thanh, K. K. Ahn, "Nonlinear PID control to improve the control performance of 2 axes pneumatic artificial muscle manipulator using neural network", *Mechatronics* (16), 2006, pp. 577-587.
- [4] K. K. Ahn, H. T. C. Nguyen, "Intelligent switching control of a pneumatic muscle robot arm using learning vector quantization neural network", *Mechatronics* 17, 2007, pp. 255-262.
- [5] M. Chettouh, R. Toumi, M. Hamerlain, "High-Order Sliding Modes for a Robot Driven by Pneumatic Artificial Rubber Muscles", *Advanced Robotics*, (22), 2008, pp. 689-704.
- [6] B. Tondu, K. Braikia, M. Chettouh, S. Ippolito, "Second Order Sliding Mode Control for an Anthropomorphic Robot-Arm driven with Pneumatic Artificial Muscles", 9th IEEEERAS International Conference on Humanoid Robots, Paris, 2009, pp. 47-54.
- [7] B. Vanderborght, B. Verrelst, R. Van Ham, J. Naudet, J. Vermeulen, D. Lefeber & F. Daerden, "Lucy, a Bipedal Walking Robot with Pneumatic Artificial Muscles", *Int. IEEE Conf. Mechatronics & Robotics*, Aachen, Germany, 2004.
- [8] J. Schroder, K. Kawamura, T. Gockell, R. Dillmann, "Improved control of a humanoid arm driven by pneumatic actuators", in *Proc. of Humanoids*, 2003.
- [9] S. Diaz Zagal, *Conception et Développement d'un Mini-Actionneur à Muscle Artificiel. Application à la Robotique Médicale*, Thèse de doctorat Présentée à L'Institut National des Sciences Appliquées de Toulouse, 2007.
- [10] R. Lezama Morales, *Modélisation et programmation d'un robot anthropomorphe à 7 degrés de liberté action par muscles artificiels pneumatiques*, thèse présentée à l'institut national des sciences appliqués de Toulouse, avril 2008.
- [11] B. Tondu, & Lopez, P. Modeling and control of McKibben artificial muscle robot actuators. *IEEE Control Systems Magazine*, 20(2), 2000, pp. 15-38.
- [12] N. Tsagarakis, & Caldwell, D. G. "Improved modelling and assessment of pneumatic muscle actuators" In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2000, pp. 3641-3646.
- [13] D. B. Reynolds, Repperger, D. W., Phillips, C. A., & Bandry, G. "Modeling the dynamic characteristics of pneumatic muscle", *Annals of Biomedical Engineering*, 31(3), 2003, pp. 310-317.

- [14] K.Kawashima, Sasaki, T., Ohkubo, A., Miyata, T., & Kagawa, T. "Application of robot arm using fiber knitted type pneumatic artificial rubber muscles", In Proc. of the IEEE Int. Conf. on Robotics and Automation, 2004, pp. 4937-4942.
- [15] M. Bertetto, A. & Ruggiu, M." characterization and modeling of air muscles". Mechanics Research Communications, 31 (2), 2004, pp. 185-194.
- [16] T. D. C. Thanh, K. K. Ahn, "Intelligent phase plane switching control of pneumatic artificial muscle manipulators with magneto-rheological brake", Mechatronics (16), 2006, pp. 85-95.
- [17] Chang M. K., "An adaptive self-organizing fuzzy sliding mode controller for a 2-DOF rehabilitation robot actuated by pneumatic muscle actuators", Journal of Control Engineering Practice, (18), 2010, pp. 13-22.
- [19] A. Pedro, A. Sala, "Multivariable Control Systems, an Engineering Approach Series: Advanced Textbooks in Control and Signal Processing", Edit Springer, 2004.
- [20] P. Lopez, A. S. Nouri, "Théorie élémentaire et pratique de la commande par les régimes glissants", Edit Springer (in French), 2006.
- [21]S. Li Tzuu-Hseng and H. Yun-Cheng, MIMO adaptive fuzzy terminal sliding-mode controller for robotic manipulators, Information Sciences, Vol. 180 2010, pp 4641–4660.
- [22] Y. S. Huang, C.C. Sung "Implementation of a Fast Terminal Sliding Mode Controller for Direct Thrust Control Systems" IEEE Int. Conf. on Industrial Technology, ICIT 2009, pp 1-6.
- [23] **A. Rezoug, S. Mahjoub, M. Hamerlain, and M. Tadjine**, "Fuzzy Terminal Sliding Mode Controller for Robot Arm Actuated By Pneumatic Artificial Muscles", Int. Multi-Conf. on Systems, Signals & Devices, Conference on Systems, Analysis & Automatic Control, SSD 2011, Sousse, Tunis, Mars 22 – 25, 2011.
- [24] T. K. Kang, H. Zhang, G. T. Park, "Stereo-Vision Based Motion Stabilization of a Humanoid Robot for the Environment Recognition by Type-2 Fuzzy Logic", 17th Mediterranean Conference on Control & Automation, Greece, 2009, pp. 772-777.
- [25] M. Galluzzo, B. Cosenza, "Adaptive Type-2 fuzzy logic control of a bioreactor", journal of Chemical Engineering Science, Vol. 65, 2010, pp. 4208-4221.
- [26] L. Leottau, M. Melgarejo, "A Simple Approach for Designing a Type-2 Fuzzy Controller for a Mobile Robot Application", Annual Meeting of the North American Fuzzy Information Processing Society, 2010, pp. 1-6.
- [27] Lu, H.C., Tsai, C.H., Chang, M.H. "Radial basis function neural network with sliding mode control for robotic manipulators" Proc. IEEE Int. Conf. on Systems Man and Cybernetics, Turkey, 2010, pp. 1209 – 1215.

- [28] J. H. Lilly, L. Yang, "Sliding Mode Tracking for Pneumatic Muscle Actuators in Opposing Pair Configuration", *IEEE Trans. On Control Systems Technology*, Vol. 13, No. 4, 2005, pp. 550-558.
- [29] Castillo, O., P. Melin, "A review on the design and optimization of interval type-2 fuzzy controllers", *Applied Soft Computing* Vol. 12, 2012, pp. 1267–1278.
- [30] Liang, Q., and Mendel, J. M., "Interval Type-2 Fuzzy Logic Systems: Theory and Design", *IEEE Trans. on Fuzzy Systems*, 8(5), 2000, pp. 535–550.
- [31] Lenz A., Anderson R S., Pipe A. G., Melhuish C., Dean P., and Porrill J., "Cerebellar-Inspired Adaptive Control of a Robot Eye Actuated by Pneumatic Artificial Muscles", *IEEE Trans. On Systems, Man, and Cybernetics—Part B: Cybernetics*, 39(6), pp. 1420–1433 (2009).
- [32] Tai, N. T., Ahn, K. K., "A RBF Neural Network Sliding Mode Controller for SMA Actuator", *Int. Journal of Control, Automation, and Systems* 8(6), 2010, pp. 1296-1305.
- [33] Chen, J.Y., "Expert SMC-based fuzzy control with genetic algorithms", *Journal of the Franklin Institute*. 336, 1999, pp. 589–610.
- [34] **A. Rezoug , M. Hamerlain , M. Tadjine**, "Adaptive Interval Type-2 Fuzzy Controller For Robot Arm Actuated By Pneumatic Artificial Rubber Muscles", *The Mediterranean Journal of Measurement and Control*, 2011, Vol. 7, No. 3, pp. 260-272.
- [35] **A. Rezoug, M. Hamerlain, B. Tondou and M. Tadjine**, Interval Type-2 Fuzzy Sliding Mode Controller for Robot Arm Driven By Artificial Muscles, 8th Int. Conf. on informatics in control, automation and robotics, ICINCO 2011, Noordwijkerhout, the Netherlands, July 28– 31, 2011.
- [36] R. Morgan and U. Ozguner, "A decentralized variable structure control algorithm for robotic manipulators," *IEEE Trans. Robot. Autom.*, vol. RA-1, no. 1, 1985, pp. 57–65.
- [37] T. C. Hsia, T. A. Lasky, and Z. Guo, "Robust independent joint controller design for industrial robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 38, no. 1, 1991, pp. 21–25.
- [38] J. Maolin, L. Jinh, H. C. Pyung, and C. Chintae, "Practical Nonsingular Terminal Sliding-Mode Control of Robot Manipulators for High-Accuracy Tracking Control", *IEEE Trans On Industrial Electronics*, Vol. 56, No. 9, 2009.
- [39] K. Youcef-Toumi and O. Ito, "A time delay controller design for systems with unknown dynamics", *Trans. ASME, J. Dyn. Syst. Meas. Control*, vol. 112, no. 1, 1990, pp. 133–142.
- [40] G. F. Franklin, J. Powell, and M. Workman, "Digital Control of Dynamic Systems", Reading, MA: Addison-Wesley, 1998.
- [41] Yong Fenga, Xinghuo Yub, Zhihong Manc, Non-singular terminal sliding mode control of rigid manipulators, *automatica*, 38 (2002) , pp. 2159 – 2167
- [42] **A. Rezoug, M. Hamerlain, and M. Tadjine**, "Decentralized RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Manipulator Driven by Artificial Muscles", *International Journal of*

Advanced Robotic Systems, Vol. 09, 2012, pp 1-12.

[43] C. W. Tao, J. S. Taur, and Mei-Lang Chan, "Adaptive Fuzzy Terminal Sliding Mode Controller for Linear Systems With Mismatched Time-Varying Uncertainties" IEEE Trans. ON Systems, Man, and Cybernetics—Part B, Vol. 34, No. 1, 2004, pp. 255-262.

[44] **A. Rezoug, M. Hamerlain, M. Tadjine**, "Adaptive RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Arm with Pneumatic Muscles", IEEE International Conference on Robotics and Biomimetics (ROBIO 2012), Guangzhou, china, December 2012, pp. 1287-1292.

[45] **A. Rezoug, A. Meddahi, K. Baizid , M. Hamerlain, and M. Tadjine**, Fuzzy Logic Controller for a Pneumatic Artificial Muscle Robot based on Sliding Mode Control" *14th IASTED International Conference on Robotics and Applications, ~RA 2009~* , Cambridge, Massachusetts, USA, November 2 – 4, 2009.

[46] O. Castillo, P. Melin, "Type-2 Fuzzy Logic: Theory and Applications", Edit Springer, 2008.

[47] H. Chaoui, W. Gueaieb, "Type-2 Fuzzy Logic Control of a Flexible-Joint Manipulator", Journal of Intelligence Robot Systems (51), 2008, pp. 159-186.

[48] O. Castillo, L. Aguilar, N. Cazarez, S. Cardenas, "Systematic design of a stable type-2 fuzzy logic controller", Applied Soft Computing, 2008, pp. 1274-1279.

[49] P. Pomier, "Modular robot arm based on pneumatic artificial rubber muscles (PARM)", in: Proc. CLAWAR, Catania, 2003, pp. 879-886.

[50] P. Pomiers, A. Semerano, "SynDEX for Real-Time Applications Implementations", Proceedings of the 4th International Conference on Climbing and Walking Robots, Germany, 2001.

[51] D. Wu, "design and analysis of type-2 fuzzy logic systems", a thesis submitted for the degree of Master of engineering department of electrical and computer engineering, National University of Singapore, 2005.

[52] M. Y. Hsiao, T. H. S. Li, J. Z. Lee, C. H. Chao, S. H. Tsai, "Design of interval type-2 fuzzy sliding-mode controller", Journal Information Sciences, Vol. 178, No. 6, 2008.

[53] N. N. Karnik, J. M. Mendel, Q. Liang, "Type-2 Fuzzy Logic Systems", IEEE Trans. on Fuzzy Systems, Vol. 7, No. 6 December, 1999, PP. 643-653.

[54] Braikia, K., Tondu, B., Chettouh, M., Acco, P.: Improved control strategy of 2-sliding controls applied to a flexible robot arm, Advanced Robotics. 25(11), pp. 1515–1538 (2011).

[55] Jin, Y., Decentralized adaptive fuzzy control of robot manipulators, IEEE Trans. On **Sys, Man, and Cyb**—Part B: Cybernetics, 28(1), pp. 47–57 (1998).

[56] C. K. Lin "Nonsingular Terminal Sliding Mode Control of Robot Manipulators Using Fuzzy Wavelet Networks" IEEE Trans. on Fuzzy Systems, (2006) Volume: 14, pp 849-859.

[57] S. Riachy, Contribution à l'estimation et à la commande de systèmes mécaniques sous-actionnés, Docteur de l'Ecole Centrale de Lille, 2008

- [58] V. Boitier. Implementation and control of a SCARA robot, actuated by McKibben pneumatic artificial muscles. PhD thesis, INSA – Université de Toulouse, 1996.
- [59] K. Braikia, asservissement des systèmes incertains par des commandes à mode glissant application à un robot flexible, institut national des sciences appliquées de Toulouse, février 2011.
- [60] Z. Q. Mei, Yun-can Xue, “Inverse pseudo-random binary sequence design and its application in identifying of the SCARA-type manipulator”, IEEE Int. Conf. on Systems, Man, and Cybernetics, Taipei, 2006, pp. 4459-4464.
- [61] Z. Luo, S. Yang, D. Xue, L. Hao and H. Liu, Advances in Research on Artificial Muscles Technology and Its Control Algorithm, 2nd International Conference on Advanced Computer Control (ICACC), 2010, pp. 48-51.
- [62] M. Chettouh, commande d’un robot à muscles artificiels par modes glissants et réduction du broutement, thèse de doctorat présentée à USTHB, Décembre 2008.
- [63] Schmitt J., Grabert F. and Raatz A., Design of a Hyper-Flexible Assembly Robot Using Artificial Muscles, *Proc. Of IEEE int. conf. on robotics and biomimrtics, Tianjin, china, PP. 897-902.* 2010
- [64] ayman hussain, contribution à la commande adaptative robuste par modes glissants, thèse de doctorat présentée à l’Université de Reims Champagne Ardenne, décembre 2009.
- [65] Oscar Castillo, Patricia Melin, Janusz Kacprzyk, Witold Pedrycz, Type-2 Fuzzy Logic: Theory and Applications, 2007 IEEE International Conference on Granular Computing.
- [66] N. Ezziani, commande adaptative floue backstepping d’une machine asynchrone avec et sans capteur mecanique, thèse de doctorat présentée à l’Université de Reims Champagne Ardenne, Avril 2010.
- [67] F. Harashima, H. Hashimoto, K. Maruyama, practical robust control of robot arm using variable structure system, IEEE 1986.
- [68] Yu X., Kaynak, O.: Sliding-Mode Control With Soft Computing: A Survey, IEEE Tran. On Industrial Electronics, 56(9), 3275–3285 (2009).
- [69] S. Laghrouch, ”commande par mode glissant d’ordre supérieur : théorie et applications”, thèse de doctorat de l’Ecole Centrale de Nantes, décembre 2004.
- [70] T. benabdallah, "les techniques de découplage explicite et implicite des systèmes multivariables. Application sur le modèle d'une colonne de distillation binaire " mémoire de magister, université de boumerdes, 2011.
- [71] I. Déo, L. Dau, “commande des systèmes, conception, identification et mise en œuvre”, Edition Lavoisier, juin 2002.
- [72] L. Foulloy, S. Galichet et A. Title, “commande floue 1 de la stabilisation à la supervision”, édition Lavoisier, 2003.

- [73] F. Hamerlain, stratégies de commande par mode de glissement d'ordre supérieur appliquées à des robots à roues, thèse de doctorat de l'école centrale de lille, 2007.
- [74] A. Girin, Contribution à la Commande Non Lineaire d'un Système Electropneumatique pour Une Utilisation Aéronautique : Application sur un Benchmark Dédié, Thèse de Doctorat, L'École Centrale de Nantes et l'Université de Nantes, 2007.
- [75] M. Hamerlain, " Commande hiérarchisée à modèle de référence et à structure variable d'un robot manipulateur à muscles artificiels ", Thèse, N° ordre 223, GARI /DGE /INSA, Toulouse, Jan. 1993.
- [76] A. Levant, "Sliding order and sliding accuracy in sliding mode control", Int. Journal of Control, vol.58, n°6, pp.1247-1263, 1993.
- [77] J. Liu, F. Sun, "Chattering free adaptive fuzzy terminal sliding mode control for second order nonlinear system, Journal of Control Theory and Applications 4 (2006) 385–391
- [78] S.V. Emelyanov, Korovin S.K. and Levantovsky A., « A drift algorithm in control of uncertain processes », Problems of control an Information Theory, 6, pp. 425-438, 1986.
- [79] A. Rezoug, "poursuite de trajectoire par asservissement visual d'un robot mobile" mémoire de magister, ecole nationale polytechnique, mars 2008.
- [80] Z. Qiu, B. Ma, X. Zhang, End Edge Feedback and RBF Neural Network Based Vibration Control of Flexible Manipulator, ieee int. conf of Robotics and Biomimetics, 2012, 1680-1685.