

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique

Laboratoire de Commande des Processus



Mémoire de projet de Fin d'Etudes

En vue de l'obtention du diplôme d'Ingénieur d'État

en Automatique

Thème

Commande Multimodèle d'un bras de robot

Réalisé par :

M. DJIDA Aymen

M. MOUCHOU Larbi

Proposé et encadré par :

Dr ILLOUL R.

Soutenu le : 20/06/2016

Devant le jury composé de :

Président : Dr. ACHOUR H. Ecole Nationale Polytechnique

Rapporteur : Dr. ILLOUL R. Ecole Nationale Polytechnique

Examineur : Pr. BOUKHETALA D. Ecole Nationale Polytechnique

Juin 2016

Dédicaces

*Je dédie ce mémoire à ma famille
et mes amis qui m'ont soutenu tout
au long de mon parcours.*

Larbi MOUCHOU

*Je dédie ce travail à toute
personne qui m'a soutenu, mes
parents, ma famille et mes amis.*

Aymen DJIDA

Remerciement

Le travail présenté dans ce mémoire a été réalisé au sein du Laboratoire de Commande des Processus de l'École Nationale Polytechnique sous la direction de M. Rachid ILLOUL. Le bon déroulement de ce travail a été possible grâce à lui, nous lui sommes reconnaissant du soutien qu'il nous a apporté. Nous lui exprimons également notre profonde reconnaissance pour ses précieux conseils et la disponibilité dont il a toujours fait preuve à notre égard et sa qualité humaine.

Nos remerciements vont aussi à Messieurs les membres du jury pour avoir accepté d'évaluer notre travail. Nous adressons également nos remerciements à l'ensemble des enseignants de génie électrique et en particulier à ceux du Département d'Automatique.

Enfin, nous voudrions réserver nos derniers remerciements à nos familles respectives ainsi qu'à tous nos camarades et à tous ceux qui nous ont soutenus ou qui, d'une manière ou d'une autre, ont contribué à la réalisation de ce travail.

ملخص:

تتطرق هذه المذكرة إلى دراسة التحكم الضبابي المتكيف المباشر مع النموذج المرجعي للأنظمة اللاخطية التي تمثل بنيتها النموذج الضبابي لتاكاغي-سوغينو. اعتبر قانون التكيف تردد خطأ التقريب لنظام المنطق الضبابي بحيث يوفر متابعة مقارنة للإشارة المرجعية للأنظمة مع التغييرات الزمنية. يضمن قانون التحكم المتكيف المطور تحديد إشارات الأنظمة ذات الدائرة المغلقة، إضافة لذلك، حالة النظام تتبع نفس حالة النموذج المرجعي مقارنة مع تغير الزمن بالنسبة لكل إشارة داخلية نموذجية محدودة.

كلمات مفاحية: الأنظمة اللاخطية، النموذج تاكاغي-سوغينو، متعدد النماذج، التحكم المتكيف متعدد النماذج، التحكم المتكيف المباشر، التحكم الضبابي، النظم الغير الخطية متعدد المتغيرات

Abstract:

In this thesis, we introduce a direct model reference adaptive fuzzy control (MRAFC) for MIMO nonlinear systems whose structure is represented by the Takagi-Sugeno fuzzy model. The adaptive law of the MRAFC estimates the approximation error of the fuzzy logic system so that it provides asymptotic tracking of the reference signal for the systems with uncertain or slowly time-varying parameters. The developed control law and adaptive law guarantee the boundedness of all signals in the closed-loop system. In addition, the plant state tracks the state of the reference model asymptotically with time for any bounded reference input signal.

Keywords: nonlinear system, Takagi-Sugeno model, Multi-Model, PDC, Multi-Model adaptive control, Adaptive fuzzy control, State feedback linearization, direct adaptive control, Fuzzy control, nonlinear systems multivariable, Stability.

Résumé :

Dans ce mémoire, nous présentons une commande floue adaptative directe avec modèle de référence (MRAFC) pour les systèmes non linéaires MIMO dont la structure est représentée par le modèle flou de Takagi-Sugeno. La loi d'adaptation du MRAFC estime l'erreur d'approximation du système de logique floue afin qu'il assure le suivi asymptotique du signal de référence pour les systèmes avec des paramètres variables dans le temps. La loi de commande adaptative développée garantit que tous les signaux dans le système en boucle fermée sont bornés. En outre, l'état de système suit l'état du modèle de référence asymptotiquement avec le temps pour tout signal d'entrée de référence borné.

Mots-clés : Système non linéaire, Modèles Takagi-Sugeno, multimodèle, PDC, Commande adaptative multimodèle, Linéarisation par retour d'état, Commande adaptative directe, Commande floue, Systèmes non linéaires multivariables, Stabilité.

Table des matières

Liste des tableaux	7
Liste des figures	8
Liste des abréviations	10
Introduction générale	11
Chapitre 1 : Modélisation du bras de robot	14
1.1. Modèle géométrique :	15
1.1.1. Modèle géométrique direct :	15
1.1.2. Modèle géométrique inverse :	17
1.2. Modèle dynamique :	19
1.2.1. Formalisme de Lagrange-Euler :	20
1.3. Application et Simulation :	27
Conclusion :	29
Chapitre 2 : Multimodèles et modèles flous de Takagi-Sugeno	30
2.1. Représentation d'état et systèmes non linéaires :	31
2.2. Présentation des modèles flous de type Takagi-Sugeno (T-S)	31
Règles floues :	33
Variables de prémisses :	33
Fonctions d'appartenance :	33
Les zones de fonctionnement :	33
2.3. Les différents types de multimodèles :	34
2.4. Obtention des modèles flous de type Takagi-Sugeno (T-S)	36
2.4.1. Approche par Secteur non-linéaire :	37
2.4.2. Approximation locale dans les espaces de partition floue :	38
2.5. Stabilité et stabilisation des modèles T-S standards	38
2.5.1. Stabilité des modèles T-S :	38
2.5.2. Stabilisation par retour d'état des modèles T-S standards.....	40
2.5.2.1. Lois de commande PDC (Parallel distributed compensation)	40
2.6. Commande PDC	41
2.7. Application et simulation :	42
2.7.1. Multimodèle avec 12 sous-modèles flous:	44
2.7.2. Multimodèle avec 27 sous-modèles flous:	45
2.7.3. Commande de multimodèle T-S (27 modèles) avec PD:	47
2.7.4. Commande multimodèle PDC :	48

Conclusion	50
Chapitre 3 : Commande adaptative multimodèle floue directe avec modèle de référence	42
3.1. Loi d'adaptation.....	52
3.2. Stabilité de la commande MRAFC :.....	56
3.3. Application et analyse :.....	57
Simulation :	58
Conclusion :	65
Chapitre 4 : Etude comparative des performances de la commande MRAFC avec d'autres méthodes de commande.....	66
4.1. Présentation des différents contrôleurs :	67
4.2. Etude comparative sur une trajectoire sinusoïdale :	68
4.3. Etude comparative avec plusieurs cas de variations paramétriques :	71
Conclusion :	86
Bibliographie	88
Annexe A	91
A.1 Matrices A_i et B_i pour les 12 modèles locaux :	91
A.2 Matrices A_i et B_i pour les 27 modèles locaux :	92

Liste des tableaux :

Tableau 1. 1 – les valeurs des paramètres du bras de robot	16
Tableau 1. 2 – transformation entre chaque articulation	16
Tableau 2. 1 – Multimodèles couplés et découplés	34

Liste des figures :

Figure 1. 1 – bras de robot a deux (2) articulations rotoides	16
Figure 1. 2 – Position des centres de masse des membrures	21
Figure 1. 3 – L’angle q_1 pour une commande PD	28
Figure 1. 4 – L’angle q_2 pour une commande PD	28
Figure 1. 5 – Signale de commande pour une commande PD.....	28
Figure 2. 1 – Structure et implémentation d’un modèle T-S Standard.....	34
Figure 2. 2 – Multimodèles couplés	35
Figure 2. 3 – – Multimodèles découplés	35
Figure 2. 4 – Différentes approches possibles	35
Figure 2. 5 a – Secteur non linéaire global Figure 2. 5b – Secteur non linéaire local	37
Figure 2. 6 – Fonctions d'appartenance du multimodèle avec 12 modèles locaux.....	44
Figure 2. 7 – performance du multimodèle avec 12 modèles locaux.....	45
Figure 2. 8 – Fonctions d'appartenance du multimodèle avec 27 modèles locaux.....	46
Figure 2. 9 – Stabilité du multimodèle avec 27 modèles locaux.....	46
Figure 2. 10 – L’angle q_1 du Multimodèle pour une commande PD	47
Figure 2. 11 – L’angle q_2 du Multimodèle pour une commande PD	48
Figure 2. 12 – Signale de commande du Multimodèle pour une commande PD	48
Figure 2. 13 – L’angle q_1 avec commande PDC	49
Figure 2. 14 – L’angle q_2 avec commande PDC	49
Figure 3. 1 – Schéma bloc de la commande adaptative directe MRAFC.....	55
Figure 3. 2 – Commande du système pour une trajectoire linéaire dans l’espace cartésien	59
Figure 3. 3 – Réponses de système sur une trajectoire linéaire dans l’espace cartésien	59
Figure 3. 4 – L’angle q_1 d’une trajectoire linéaire	60
Figure 3. 5 – L’angle q_2 d’une trajectoire linéaire	60
Figure 3. 6 – L’erreur de l’angle q_1 sur une trajectoire.....	61
Figure 3. 7 – L’erreur de l’angle q_2 sur une trajectoire.....	61
Figure 3. 8 – Commande du système pour une trajectoire circulaire.....	62
Figure 3. 9 – Réponses de système sur une trajectoire circulaire.....	62
Figure 3. 10 – L’angle q_1 d’une trajectoire.....	63
Figure 3. 11 – L’angle q_2 d’une trajectoire	63
Figure 3. 12 – L’erreur de l’angle q_1 sur une trajectoire circulaire.....	64
Figure 3. 13 – L’erreur de l’angle q_2 sur une trajectoire circulaire.....	64
Figure 4. 1– Structure de la commande adaptative glissante [Su et al, 1993].....	67
Figure 4. 2 – Trajectoire sinusoïdale	68
Figure 4. 3 – Signale de commande pour trajectoire sinusoïdale.....	69
Figure 4. 4 – L’angle q_1 d’une trajectoire sinusoïdale	69
Figure 4. 5 – L’angle q_2 d’une trajectoire sinusoïdale	70
Figure 4. 6 – L’erreur de l’angle q_1 d’une trajectoire sinusoïdal avec des petites variations paramétriques.....	70
Figure 4. 7 – L’erreur de l’angle q_2 d’une trajectoire sinusoïdal avec des petites variations paramétriques.....	71
Figure 4. 8 – Signale de commande pour trajectoire droite avec des petites variations paramétriques	72
Figure 4. 9 – Réponses de système sur une trajectoire linéaire dans l’espace cartésien avec des petites variations paramétriques	73

Figure 4. 10 – L’angle q_1 d’une trajectoire linéaire avec des petites variations paramétriques	73
Figure 4. 11 – L’angle q_2 d’une trajectoire linéaire avec des petites variations paramétriques.....	74
Figure 4. 12 – L’erreur de l’angle q_1 sur une trajectoire linéaire avec des petites variations paramétriques	74
Figure 4. 13 – L’erreur de l’angle q_2 sur une trajectoire linéaire avec des petites variations paramétriques.....	75
Figure 4. 14 – Signale de commande pour une trajectoire circulaire avec des petites variations paramétriques	75
Figure 4. 15 – Réponses de système sur une trajectoire circulaire dans l’espace cartésien avec des petites variations paramétriques	76
Figure 4. 16 – L’angle q_1 d’une trajectoire circulaire avec des petites variations paramétriques	76
Figure 4. 17 – L’angle q_2 d’une trajectoire circulaire avec des petites variations paramétriques	77
Figure 4. 18 – L’erreur de l’angle q_1 sur une trajectoire circulaire avec des petites variations paramétriques.....	77
Figure 4. 19 – L’erreur de l’angle q_2 sur une trajectoire circulaire avec des petites variations paramétriques	78
Figure 4. 20 – Grande variations paramétriques de M_1	79
Figure 4. 21 – Grande variations paramétriques de M_2	79
Figure 4. 22 – signale de commande pour une trajectoire linéaire avec des grandes variations paramétriques.....	80
Figure 4. 23 – Réponses de système sur une trajectoire linéaire dans l’espace cartésien avec des grandes variations paramétriques	80
Figure 4. 24 – L’angle q_1 d’une trajectoire linéaire avec des grandes variations paramétriques	81
Figure 4. 25 – L’angle q_2 d’une trajectoire linéaire avec des grandes variations paramétriques	81
Figure 4. 26 – L’erreur de l’angle q_1 sur une trajectoire linéaire avec des grandes variations paramétriques.....	82
Figure 4. 27 – L’erreur de l’angle q_2 sur une trajectoire linéaire avec des grandes variations paramétriques.....	82
Figure 4. 28 – signale de commande pour une trajectoire circulaire avec des grandes variations paramétriques.....	83
Figure 4. 29 – Réponses de système sur une trajectoire circulaire dans l’espace cartésien avec des grandes variations paramétriques	83
Figure 4. 30 – L’angle q_1 d’une trajectoire circulaire avec des grandes variations paramétriques	84
Figure 4. 31 – L’angle q_2 d’une trajectoire circulaire avec des grandes variations paramétriques	84
Figure 4. 32 – L’erreur de l’angle q_1 sur une trajectoire circulaire avec des grandes variations paramétriques.....	85
Figure 4. 33 – L’erreur de l’angle q_2 sur une trajectoire circulaire avec des grandes variations paramétriques.....	85

Liste des abréviations

LMI Linear Matrix Inequality

LPV systèmes linéaires à paramètres variant dans le temps

LTI Linéaire à Temps Invariant

MFC Modèle Flou Continu

MGD Modèle Géométrique Direct

MGI Modèle Géométrique Indirect

MIMO Multi-Input Multi-Output

MRAFC commande floue adaptative directe avec modèle de référence

MTSS Modèle de Tagaki-Sugeno Standard

ODE Equations Différentielles Ordinaires

PD Proportionnel Dérivé

PDC, Parallel Distributed Compensation

PID Proportionnel Intégrale Dérivé

PPDC Proportionel Parallel Distributed Compensation

T-S Tagaki-Sugeno

Introduction générale

La notion de système joue un rôle primordial en automatique. Un système physique peut être considéré comme un ensemble de phénomènes liés entre eux et évoluant au cours du temps, modélisables théoriquement via des équations dynamiques et conduisant à un modèle de connaissance mathématique supposé représenter au mieux la réalité. Ainsi, la grande majorité des travaux utilisant une représentation d'état offrant la possibilité de décrire les relations entre les variables du système par le biais d'Equations Différentielles Ordinaires (ODE). Ces dernières permettent de relier entre elles les grandeurs de sorties et d'entrées par le biais d'un vecteur d'état décrivant l'évolution du système.

L'un des enjeux de l'automatique est de proposer un contrôleur adapté au système à piloter garantissant la réalisation de la tâche souhaitée. De nombreuses approches, pour les systèmes continus, ont été proposées pour la synthèse de lois de commande et peuvent être classées en deux catégories dites « linéaires » ou « non linéaire ».

- L'automatique linéaire considère le fonctionnement du système autour d'un point de fonctionnement donnée. La synthèse d'une loi de commande linéaire se base alors sur une théorie bien maîtrisée mais ne garantit pas la stabilité du système sur tout l'espace d'état [Delarminat, 1993] [Kuo, 1995].
- L'automatique non linéaire repose quant à elle sur des outils plus complexes et qui nécessitent parfois de nouveaux développements théoriques. L'objectif est de garantir la stabilité des systèmes pour lesquels le cas linéaire n'est pas adapté, notamment lorsqu'il s'agit de traiter des problèmes de stabilité ou d'asservissement sur une plus large région de l'espace d'état [Khalil, 1996] [Sastry, 1999].

Parmi les approches non linéaires, on distingue deux catégories :

Les approches non formelles, qui ne nécessitent pas la connaissance d'un modèle formalisé, telles que la commande floue de type Mamdani. Ces approches fournissent des résultats souvent satisfaisants d'un point de vue performances mais leurs principaux inconvénients résident dans le fait qu'il s'avère difficile, dans certains cas, de garantir (ou de prouver) la stabilité mais aussi de tenir compte des capacités physiques du système à commander.

Les approches formelles, qui nécessitent la connaissance a priori d'un modèle du système à commander, permettent quant à elles de garantir dans certains cas la stabilité du système sur tout ou une région compacte de l'espace d'état. Malheureusement, cette garantie s'avère difficile à obtenir à mesure que la taille du système augmente (nombre de variables d'état à contrôler) et/ou que la dynamique du système est fortement non linéaire (commutations franches, grand nombre de termes non linéaires...) [Khalil, 1996] [Liberzon, 2003]. En d'autres termes, ces approches sont le plus souvent applicables aux seuls systèmes de petites tailles et faiblement non linéaires.

Une manière élégante de représenter un système non linéaire repose sur une représentation multi-modèle. Celle-ci consiste en une collection de systèmes linéaires combinés par des fonctions non linéaires. Plusieurs catégories de multimodèles existent dans la littérature, notamment les systèmes linéaires à paramètres variant dans le temps (LPV) [Henrion et Garulli, 2004][Zerar et al., 2009] ou les systèmes quasi LPV, encore appelés systèmes Takagi-Sugeno (T-S) [Takagi et Sugeno, 1985]. Notons que ces derniers possèdent une propriété d'approximation universelle des systèmes affins en la commande et présentent l'avantage de pouvoir représenter de manière exacte un modèle de connaissance non linéaire sur un compact de l'espace d'état [Tanaka et Wang, 2001]. Ainsi, l'intérêt majeur de ce type d'approche est qu'elle permet d'étendre de nombreux concepts théoriques de l'automatique linéaire au cas des systèmes non linéaires.

Lors de l'étape de modélisation, le défi de l'automaticien est de proposer un modèle de connaissance capable de représenter fidèlement les phénomènes physiques rencontrés. Notons que cette modélisation suppose dans la majeure partie des cas des approximations susceptibles de dégrader les performances désirées pour le système réel [Dubuisson, 1990][Oustaloup et Mathieu, 1999]. Dans ce contexte, il peut être intéressant de tenir compte de ces imprécisions pour pouvoir assurer les performances désirées par le biais de la synthèse de lois de commande robustes [Zhou et Doyle, 1998]. Notons par ailleurs qu'un modèle trop simplifié est « faux » alors qu'un modèle trop détaillé peut être inexploitable avec les outils actuels d'analyse des systèmes. Ainsi, la prise en compte d'incertitudes pour la stabilisation robuste des systèmes non linéaires constitue un compromis entre complexité et validité de la solution et conduit à une alternative intéressante pour la commande des systèmes complexes.

C'est dans ce contexte que s'inscrivent les travaux de recherche de ce mémoire visant aussi bien à présenter qu'à étendre des méthodologies de synthèse de la lois de commande pour la stabilisation robuste des multimodèle T-S incertains et perturbés.

Ce mémoire, composé de quatre chapitres, est organisé de la façon suivante :

Le premier chapitre est constitué de deux parties qui ont pour but de modéliser un bras manipulateur. Au cours de la première partie, nous introduisons les différents niveaux de modélisation, à savoir les modèles géométriques et les modèles dynamique. La seconde partie est consacrée à la modélisation d'un bras manipulateur à deux articulations rotoïdes.

Le deuxième chapitre a pour but d'introduire les concepts élémentaires de la commande des systèmes T-S. Nous introduisons les différentes représentations d'état des systèmes non linéaires complexes. La classe des modèles T-S standards ainsi que leurs méthodes d'obtention sont alors présentées. Puis, nous faisons un tour d'horizon sur le concept de compensation parallèle distribuée (PDC), ainsi que quelques résultats fondamentaux concernant la stabilité et la stabilisation par retour d'état.

Le troisième chapitre présente une commande floue adaptative directe avec modèle de référence (MRAFC) pour les systèmes non linéaires MIMO dont la structure est représentée par le modèle flou Takagi-Sugeno. Avec une vérification de la validité et l'efficacité de la commande MRAFC, appliquées à la commande de poursuite de robot manipulateur présenté dans le chapitre 1.

Le quatrième chapitre, l'objectif de ce chapitre est d'évaluer les performances de la commande MRAFC. D'un point de vue comparatif, trois contrôleurs ont été simulés pour la poursuite d'une trajectoire du bras manipulateur à deux degrés de liberté présentée lors de premier chapitre. Le premier est un contrôleur PD classique; le second est un contrôleur adaptatif glissant et la commande adaptatif présenté dans ce mémoire.

Enfin, une conclusion générale, suivie de la bibliographie et des annexes terminent ce manuscrit.

Chapitre 1

Modélisation du bras de robot

Un bras manipulateur est le bras d'un robot généralement programmable, avec des fonctions similaires à un bras humain. Les liens de ce manipulateur sont reliés par des axes permettant, soit du mouvement de rotation (comme dans un robot articulé) ou de translation (linéaire) de déplacement.

Il peut être autonome ou contrôlé manuellement et peut être utilisé pour effectuer une variété de tâches avec une grande précision. Les bras manipulateurs peuvent être fixes ou mobiles (c'est-à-dire à roues) et peuvent être conçus pour des applications industrielles.

Notre travail porte sur les bras articulés, un bras articulé est un bras avec des axes rotatifs. Il est alimenté par une variété de moyens, y compris les moteurs électriques.

Pour commander ou simuler le comportement d'un bras mécanique articulé, on doit disposer d'un modèle. Plusieurs niveaux de modélisation sont possibles selon les objectifs, les contraintes de la tâche et les performances recherchées. Les modèles mathématiques nécessaires sont :

- Les modèles géométriques directs et inverses qui expriment les coordonnées opérationnelles de l'organe terminal en fonction des variables articulaires et inversement.
- Les modèles cinématiques direct et inverse qui expriment les vitesses en fonction des positions et vitesses articulaires et inversement.
- Les modèles dynamiques définissant les équations du mouvement du robot qui permettent d'établir les relations entre les couples ou forces exercées par les actionneurs et les positions, vitesses, accélérations des articulations.

1.1. Modèle géométrique :

Le modèle géométrique donne la relation entre les coordonnées opérationnelles X représentant la position/orientation de l'organe terminal et les variables articulaires q qui représentent les translations et les rotations des liaisons articulaires du robot.

1.1.1. Modèle géométrique direct :

Le modèle géométrique direct d'un bras de robot permet de calculer les coordonnées opérationnelles donnant la situation de l'organe terminal en fonction des coordonnées articulaires [Dombre et Khalil 2007] :

$$X = f(q) \quad (1.1)$$

X est le vecteur des coordonnées opérationnelles :

$$X = [x_1 \quad x_2 \quad \dots \quad x_n]^T$$

Q est le vecteur des coordonnées articulaires :

$$q = [q_1 \quad q_2 \quad \dots \quad q_n]^T$$

Le calcul du MGD consiste donc à exprimer la position du point $n+1$ et l'orientation du repère R_n lié à l'organe terminal, en fonction de sa configuration. Il faut pour cela multiplier entre elles les matrices de passage homogènes successives reliant le repère R_0 lié au bâti au repère R_n lié à l'organe terminal :

$${}^0T_n = {}^0T_1 * {}^1T_2 * {}^2T_3 * \dots * {}^{n-1}T_n \quad (1.2)$$

$$\text{Avec :} \quad {}^{n-1}T_n = \text{Rot}(x, \alpha_j) \text{Trans}(x, d_j) \text{Rot}(z, \theta_j) \text{Trans}(z, r_j) \quad (1.3)$$

Sous forme matricielle :

$${}^{n-1}T_n = \begin{bmatrix} \cos \theta_j & -\sin \theta_j & 0 & d_j \\ \cos \alpha_j \cos \theta_j & \cos \alpha_j \sin \theta_j & -\sin \alpha_j & -r_j \sin \alpha_j \\ \sin \alpha_j \cos \theta_j & \sin \alpha_j \sin \theta_j & \cos \alpha_j & r_j \cos \alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.4)$$

La matrice 0T_n est sous la forme suivante :

$${}^0T_n = \begin{bmatrix} {}^0A_n & {}^0P_n \\ 0 & 1 \end{bmatrix} \quad (1.5)$$

$${}^0A_n = {}^0A_1 * {}^1A_2 * {}^2A_3 * \dots * {}^{n-1}A_n \quad (1.6)$$

$${}^{n-1}A_n = \text{Rot}(x, \alpha_j) \text{Rot}(z, \theta_j) \quad (1.7)$$

Application :

Bras de robot a 2 articulations rotoides :

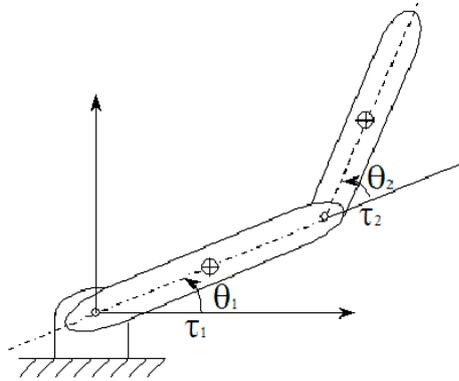


Figure 1. 1 – bras de robot a deux (2) articulations rotoides

Paramètre	Valeur	Unité
Masse du premier bras	1	Kg
Masse du deuxième bras	2	Kg
Longueur du premier bras	1	M
Longueur du deuxième bras	1.2	M
Position du centre de masse pour le premier bras	0.5	M
Position du centre de masse pour le deuxième bras	0.6	M
Inertie du premier bras	0.12	Kg.m ²
Inertie du deuxième bras	0.25	Kg.m ²

Tableau 1. 1 – les valeurs des paramètres du bras de robot[Y Cho 2007]

T	σ	α	d	θ	R
1	0	0	0	θ_1	0
2	0	0	L1	θ_2	0
Tool	0	0	L2	0	0

Tableau 1. 2 – transformation entre chaque articulation

La matrice de transformation 0T_1 (exprime R_1 dans R_0) :

$${}^0T_1 = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 & 0 \\ \sin \theta_1 & \cos \theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La matrice de transformation 1T_2 (exprime R_2 dans R_1) :

$${}^1T_2 = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & L_1 \\ \sin \theta_2 & \cos \theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La matrice de transformation ${}^2T_{Tool}$ (exprime R_{tool} dans R_2) :

$${}^2T_{Tool} = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La matrice de transformation ${}^0T_{Tool}$ (exprime R_{tool} dans R_0) :

$$\begin{aligned} {}^0T_{Tool} &= {}^0T_1 * {}^1T_2 * {}^2T_{Tool} \\ &= \begin{bmatrix} c1c2 - s1s2 & -c1s2 - s1c2 & 0 & L_1c1 + L_2(c1c2 - s1s2) \\ s1c2 + c1s2 & -s1s2 + c1c2 & 0 & L_1s1 + L_2(s1c2 + c1s2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ {}^0T_{Tool} &= \begin{bmatrix} \cos(\theta_1 + \theta_2) & \sin(\theta_1 + \theta_2) & 0 & L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) \\ \sin(\theta_1 + \theta_2) & -\cos(\theta_1 + \theta_2) & 0 & L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

1.1.2. Modèle géométrique inverse :

Le modèle géométrique inverse consiste à trouver le système d'équations qui exprime les coordonnées généralisées en fonction des coordonnées opérationnelles.

La résolution du MGI, c'est-à-dire l'existence d'un nombre fini de solutions, est fondamentale en matière de conception. Supposons que la situation P d'un bras manipulateur à n liaisons soit exprimée par un nombre m minimal de paramètres. Supposons par ailleurs que P soit une situation accessible par le bras manipulateur, c'est-à-dire que la situation appartient à l'espace de travail. Alors (dans la majorité des cas) :

- Si $n < m$, il n'existe pas de solution au modèle géométrique inverse (MGI) ;

- Si $n=m$, il existe un nombre fini de solutions en dehors de certaines configurations, appelées configurations singulières;
- Si $n > m$, il existe une infinité de solutions.

On sait que dans les cas où $n < 6$ les bras manipulateurs n'ont pas de singularité, c'est-à-dire qu'il existe une solution connue au MGI. C'est aussi le cas de la plupart des structures à six liaisons, notamment celles possédant un poignet sphérique (trois dernières liaisons rotoïdes concourantes).

Enfin, seul le calcul du MGI permet de connaître le nombre de solutions.

Application :

On a :

$$\begin{cases} x = L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) \\ y = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \end{cases} \quad (1.8)$$

En utilisant (1.8), on calcule $x^2 + y^2$:

$$x^2 + y^2 = L_1^2 + L_2^2 + 2L_1L_2(\cos\theta_1 * \cos(\theta_1 + \theta_2) + \sin\theta_1 * \sin(\theta_1 + \theta_2))$$

Avec : $\cos \theta_2 = \cos\theta_1 \cos(\theta_1 + \theta_2) + \sin\theta_1 \sin(\theta_1 + \theta_2)$

$$\cos \theta_2 = (x^2 + y^2 - L_1^2 - L_2^2)/2L_1L_2$$

$$\theta_2 = \arccos \left(\frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2} \right)$$

Maintenant on calcule l'angle θ_1 :

$$\begin{cases} x = L_1 \cos\theta_1 + L_2(\cos\theta_1 \cos\theta_2 - \sin\theta_1 \sin\theta_2) \\ y = L_1 \sin\theta_1 + L_2(\sin\theta_1 \cos\theta_2 + \cos\theta_1 \sin\theta_2) \end{cases}$$

$$\begin{cases} x = (L_1 + L_2 \cos\theta_2) \cos\theta_1 - L_2 \sin\theta_2 \sin\theta_1 \\ y = (L_1 + L_2 \cos\theta_2) \sin\theta_1 + L_2 \sin\theta_2 \cos\theta_1 \end{cases}$$

On remarque que les 2 équations sont sous la forme suivante :

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} L_1 + L_2 \cos\theta_2 & -L_2 \sin\theta_2 \\ L_2 \sin\theta_2 & (L_1 + L_2 \cos\theta_2) \end{bmatrix} \begin{bmatrix} \cos\theta_1 \\ \sin\theta_1 \end{bmatrix}$$

Donc :

$$\begin{bmatrix} \cos\theta_1 \\ \sin\theta_1 \end{bmatrix} = \begin{bmatrix} L_1 + L_2 \cos\theta_2 & -L_2 \sin\theta_2 \\ L_2 \sin\theta_2 & (L_1 + L_2 \cos\theta_2) \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\cos \theta_1 = \frac{y(L_1 + L_2 \cos \theta_2) - xL_2 \cos \theta_2}{(L_2 \sin \theta_2)^2 + (L_1 + L_2 \cos \theta_2)^2}$$

$$\sin \theta_1 = \frac{x(L_1 + L_2 \cos \theta_2) - yL_2 \cos \theta_2}{(L_2 \sin \theta_2)^2 + (L_1 + L_2 \cos \theta_2)^2}$$

$$\theta_1 = \operatorname{atan2}\left(\frac{x(L_1 + L_2 \cos \theta_2) - yL_2 \cos \theta_2}{(L_2 \sin \theta_2)^2 + (L_1 + L_2 \cos \theta_2)^2}, \cos \theta_1\right)$$

1.2. Modèle dynamique :

Le modèle dynamique est la relation entre les couples (et / ou forces) appliquées aux actionneurs et les positions communes, vitesses et accélérations. Le modèle dynamique est représenté par une relation de la forme :

$$\Gamma = f(q, \dot{q}, \ddot{q}, f_e) \quad (1.9)$$

Γ : Vecteur de couples / forces des actionneurs

q : Vecteur de position

\dot{q} : Vecteur de vitesse

\ddot{q} : Vecteur d'accélération

f_e : Vecteur représentant les forces extérieures et les moments que le robot exerce sur l'environnement

Il est commun d'appeler la relation précédente le modèle dynamique inverse, ou tout simplement le modèle dynamique.

Le modèle dynamique directe exprime les accélérations conjointes en termes de positions communes, des vitesses et des couples. Par conséquent, il est représenté par la relation :

$$\ddot{q} = g(q, \dot{q}, \Gamma, f_e) \quad (1.10)$$

Plusieurs formalismes ont été utilisés pour obtenir le modèle dynamique du robot. Les formalismes les plus utilisés sont :

- Le formalisme de Newton-Euler, il est mieux adapté pour le calcul du modèle dynamique inverse que l'on utilise en commande et le dimensionnement des parties mécaniques du robot.
- Le formalisme de Lagrange-Euler, il est mieux adapté pour le calcul du modèle dynamique direct qui est utile pour la simulation et la commande.

1.2.1. Formalisme de Lagrange-Euler :

Les équations dites de Lagrange permettent d'obtenir directement les relations entre les couples appliqués aux articulations et les mouvements des variables articulaires.

Il s'agit donc, pour un robot à (i) articulations de (i) équations différentielles du second ordre obtenues à partir de l'expression suivante :

$$\frac{d}{dt} \left(\frac{dL}{d\dot{q}_i} \right) - \frac{dL}{dq_i} = \Gamma_i \quad (1.11)$$

Avec :

$L = E - U$: Lagrangien du système

E : l'énergie cinétique du système

U : l'énergie potentielle du système

Γ_i : Couple appliquée sur l'articulation

Dans notre cas, le plan de travail du bras de robot est horizontal donc l'énergie potentielle est égale à zéro (U=0), alors l'équation devient :

$$\frac{d}{dt} \left(\frac{dE}{d\dot{q}_i} \right) - \frac{dE}{dq_i} = \Gamma_i \quad (1.12)$$

L'énergie cinétique E est sous la forme quadratique suivante :

$$E = \frac{1}{2} \dot{q}^T A(q) \dot{q} \quad (1.13)$$

Avec A une matrice symétrique définie positive appelée matrice d'inertie.

$$\Gamma = A(q) * \ddot{q} + C(q, \dot{q}) * \dot{q} \quad (1.14)$$

Avec :

$A(q)$: Matrice d'inertie

$C(q, \dot{q})$: matrice qui représente les forces de Coriolis

Modèle dynamique pour un bras de robot 2DDL :

On commence d'abord par calculer l'énergie cinétique pour chaque corps du bas de robot :

On a d'après (1.8) :

$$P = \begin{bmatrix} L1\cos\theta_1 + L2\cos(\theta_1 + \theta_2) \\ L1\sin\theta_1 + L2\sin(\theta_1 + \theta_2) \\ 0 \end{bmatrix} \quad (1.15)$$

pour obtenir le terme d'énergie cinétique nécessaire à la méthode de Lagrange, il faut connaître les vitesses de chacun des centres de masse des membrures obtenues par la dérivée de leur position respective. En faisant la supposition que les membrures sont de section uniforme et constante sur toute leur longueur, le centre de masse se trouve à mi-distance comme l'illustre la figure.

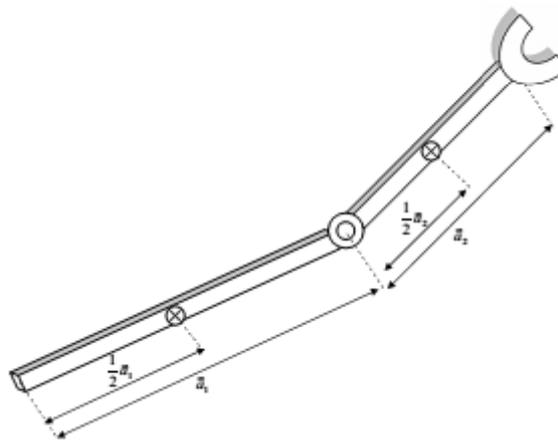


Figure 1. 2 – Position des centres de masse des membrures

La position du centre de masse pour le 1^{er} et le 2eme bras est :

$$P_{cm1} = \begin{bmatrix} x_{cm1} \\ y_{cm1} \\ z_{cm1} \end{bmatrix} = \frac{\tilde{a}_1}{2} \quad (1.16)$$

$$P_{cm2} = \begin{bmatrix} x_{cm2} \\ y_{cm2} \\ z_{cm2} \end{bmatrix} = \tilde{a}_1 + Q_1 \frac{\tilde{a}_2}{2} \quad (1.17)$$

\tilde{a}_i Vecteur lien entre l'origine du repère de la base avec celui de l'effecteur

Q_i Matrices de rotation

Alors

$$P_{cm1} = \begin{bmatrix} \frac{L1}{2} \cos \theta_1 \\ \frac{L1}{2} \sin \theta_1 \\ 0 \end{bmatrix} \quad (1.18)$$

$$P_{cm2} = \begin{bmatrix} L1\cos\theta_1 + \frac{L2}{2}\cos(\theta_1 + \theta_2) \\ L1\sin\theta_1 + \frac{L2}{2}\sin(\theta_1 + \theta_2) \\ 0 \end{bmatrix} \quad (1.19)$$

Maintenant pour avoir les vitesses, on dérive ces vecteurs :

$$\dot{P}_{cm1} = \begin{bmatrix} -\frac{L1}{2}\dot{\theta}_1 \sin\theta_1 \\ \frac{L1}{2}\dot{\theta}_1 \cos\theta_1 \\ 0 \end{bmatrix} \quad (1.20)$$

$$\dot{P}_{cm2} = \begin{bmatrix} -L1\dot{\theta}_1 \sin\theta_1 - \frac{L2}{2}(\dot{\theta}_1 + \dot{\theta}_2)\sin(\theta_1 + \theta_2) \\ L1\dot{\theta}_1 \cos\theta_1 + \frac{L2}{2}(\dot{\theta}_1 + \dot{\theta}_2)\cos(\theta_1 + \theta_2) \\ 0 \end{bmatrix} \quad (1.21)$$

Connaissant ces équations de vitesse, il faut maintenant les insérer dans celle de l'équation d'énergie cinétique en translation suivante :

$$E_{trans\ i} = \frac{1}{2} m v_i^2 \quad (1.22)$$

Avec :

$$v_i^2 = P^T_{cmi} * P_{cmi}$$

Alors :

$$\begin{cases} E_{trans1} = \frac{1}{2} * m_1 * P^T_{cm1} * P_{cm1} \\ E_{trans2} = \frac{1}{2} * m_2 * P^T_{cm2} * P_{cm2} \end{cases} \quad (1.23)$$

$$\begin{cases} E_{trans1} = \frac{1}{8} m_1 L_1^2 \dot{\theta}_1^2 \\ E_{trans2} = \frac{1}{2} * m_2 \left[\left(\frac{1}{4} L_2^2 + L_1^2 + L_1 L_2 \cos\theta_2 \right) \dot{\theta}_1 + \left(\frac{1}{2} L_2^2 + L_1 L_2 \cos\theta_2 \right) \dot{\theta}_1 \dot{\theta}_2 + \frac{1}{4} L_2^2 \dot{\theta}_2^2 \right] \end{cases} \quad (1.24)$$

Il ne s'agit là que de l'énergie cinétique en translation des centres de masse, mais puisqu'il s'agit de corps rigides, toute la masse ne se situe pas en ce point. Il est également important de considérer l'énergie cinétique en rotation du corps rigide qui se calcule avec l'équation :

$$E_{rot\ i} = \frac{1}{2} \bar{\Omega}_i^t [I_i] \bar{\Omega}_i \quad (1.25)$$

Où I est le tenseur d'inertie du corps et $\bar{\Omega}_i$ est le vecteur vitesse de rotation. Il est important pour que cette équation soit valide que le tenseur d'inertie I et les vecteurs de vitesse de rotation soient exprimés dans le même repère et pour des raisons pratiques c'est le repère inertiel attaché à la base qui est choisi.

Cependant, puisque les membrures changent d'orientation, les tenseurs d'inertie par rapport au repère attaché à la base changent également constamment avec les mouvements du manipulateur. Pour simplifier le problème, le tenseur d'inertie de chaque membrure est exprimé par une multiplication de deux termes.

$$[I_i] = Q_i I_i Q_i^T \quad (1.26)$$

Où l'indice 0 sur « [] » sert à indiquer que le terme entre crochets est exprimé par rapport au repère inertiel de la base, l'indice i dans le cas du tenseur d'inertie I_i qu'il s'agit de celui de la $i^{ème}$ membrure et sous la matrice Q , qu'il s'agit de la matrice de rotation qui amène du $i^{ème}$ repère à celui de la base O . De cette façon, c'est la matrice Q_i qui change tandis que le tenseur d'inertie reste inchangé puisqu'il est toujours exprimé par rapport à un repère attaché à la membrure et situé au centre de masse qui est également aligné avec les axes principaux d'inertie de la membrure de sorte que le tenseur soit diagonal de la forme suivante :

$$I = \begin{bmatrix} I_{i,x} & 0 & 0 \\ 0 & I_{i,y} & 0 \\ 0 & 0 & I_{i,z} \end{bmatrix} \quad (1.27)$$

Etant donné que les axes Z des différents repères sont tous parallèles et que le mouvement en rotation des articulations ne se fait que sur cet axe, les vecteurs vitesses de rotation subis par les membrures 1 et 2 restent les mêmes d'un repère à l'autre et sont respectivement :

$$\bar{\Omega}_1 = \begin{bmatrix} 0 \\ 0 \\ \theta_1 \end{bmatrix} \quad (1.28)$$

$$\bar{\Omega}_2 = \begin{bmatrix} 0 \\ 0 \\ \theta_1 + \theta_2 \end{bmatrix} \quad (1.29)$$

Connaissant $\bar{\Omega}_1$ et $\bar{\Omega}_2$, on peut calculer l'énergie cinétique de rotation en utilisant l'équation :

$$\begin{cases} E_{rot\ 1} = \frac{1}{2} \bar{\Omega}_1^t [Q_1 I_1 Q_1^T] \bar{\Omega}_1 \\ E_{rot\ 2} = \frac{1}{2} \bar{\Omega}_2^t [Q_2 I_2 Q_2^T] \bar{\Omega}_2 \end{cases} \quad (1.30)$$

$$\begin{cases} E_{rot\ 1} = \frac{1}{2} I_{1,z} \dot{\theta}_1^2 \\ E_{rot\ 2} = \frac{1}{2} I_{2,z} (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{cases} \quad (1.31)$$

Maintenant que tous les éléments d'énergie cinétique sont calculés, il est possible de les additionner pour obtenir le terme de l'énergie cinétique totale du manipulateur en fonction de ses coordonnées articulaires

$$\begin{aligned} E &= E_{rot\ 1} + E_{rot\ 2} + E_{trans\ 1} + E_{trans\ 2} \quad (1.32) \\ E &= \frac{1}{2} \left[\frac{1}{4} m_1 L_1^2 + m_2 \left(L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 + L_1^2 \right) + I_{1,z} + I_{2,z} \right] \dot{\theta}_1^2 \\ &\quad + \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 \right) + I_{2,z} \right] \dot{\theta}_1 \dot{\theta}_2 \\ &\quad + \left[\frac{1}{8} m_2 L_2^2 + \frac{1}{2} I_{2,z} \right] \dot{\theta}_2^2 \end{aligned}$$

On développe maintenant les équations différentielles couplées qui définiront la dynamique du manipulateur. En utilisant la méthode de Lagrange sur le manipulateur sériel plan à deux degrés de liberté, il y aura donc un système de deux équations et les coordonnées généralisées seront celles des articulations, soit θ_1 et θ_2 .

$$\frac{d}{dt} \left(\frac{dE}{d\dot{\theta}_1} \right) - \frac{dE}{d\theta_1} = \Gamma_1 \quad (1.33)$$

$$\frac{d}{dt} \left(\frac{dE}{d\dot{\theta}_2} \right) - \frac{dE}{d\theta_2} = \Gamma_2 \quad (1.34)$$

Avec l'énergie cinétique totale du système déjà calculé, il reste à l'insérer dans l'équation de Lagrange qui est définie par des dérivés du terme d'énergie cinétique. D'abord il faut trouver chacun des termes par rapport aux deux coordonnées généralisées. Les premières dérivées calculées sont celles par rapport aux vitesses :

$$\begin{aligned} \left(\frac{dE}{d\dot{\theta}_1} \right) &= \left[\frac{1}{4} m_1 L_1^2 + m_2 \left(L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 + L_1^2 \right) + I_{1,z} + I_{2,z} \right] \dot{\theta}_1 \\ &\quad + \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 \right) + I_{2,z} \right] \dot{\theta}_2 \end{aligned}$$

$$\left(\frac{dE}{d\dot{\theta}_2}\right) = \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2\right) + I_{2,z}\right] \dot{\theta}_1 + \left[\frac{1}{4} m_2 L_2^2 + I_{2,z}\right] \dot{\theta}_2$$

En dérivant ces deux dernières équations par rapport au temps on obtient le premier terme de l'équation de Lagrange :

$$\begin{aligned} \frac{d}{dt} \left(\frac{dE}{d\dot{\theta}_1}\right) &= -(m_2 L_1 L_2 \sin \theta_2) \dot{\theta}_1 \dot{\theta}_2 - \left(\frac{m_2 L_1 L_2 \sin \theta_2}{2}\right) \dot{\theta}_2^2 \\ &\quad + \left[\frac{1}{4} m_1 L_1^2 + m_2 \left(L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 + L_1^2\right) + I_{1,z} + I_{2,z}\right] \ddot{\theta}_1 \\ &\quad + \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2\right) + I_{2,z}\right] \ddot{\theta}_2 \end{aligned}$$

$$\begin{aligned} \frac{d}{dt} \left(\frac{dE}{d\dot{\theta}_2}\right) &= -\left(\frac{m_2 L_1 L_2 \sin \theta_2}{2}\right) \dot{\theta}_1 \dot{\theta}_2 + \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2\right) + I_{2,z}\right] \ddot{\theta}_1 \\ &\quad + \left[\frac{1}{4} m_1 L_2^2 + I_{2,z}\right] \ddot{\theta}_2 \end{aligned}$$

Maintenant pour le deuxième terme, le terme d'énergie cinétique est dérivé par rapport aux coordonnées généralisées :

$$\frac{dE}{d\theta_1} = 0$$

$$\frac{dE}{d\theta_2} = -\left(\frac{m_2 L_1 L_2 \sin \theta_2}{2}\right) \dot{\theta}_1^2 - \left(\frac{m_2 L_1 L_2 \sin \theta_2}{2}\right) \dot{\theta}_1 \dot{\theta}_2$$

Avec tous les termes calculés, ces équations sont insérées dans l'équation de Lagrange pour obtenir, après simplification, respectivement les deux équations définissant la dynamique du système pour chacune des coordonnées généralisées

$$\begin{aligned} \Gamma_1 &= \left[\frac{1}{4} m_1 L_1^2 + m_2 \left(L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 + L_1^2\right) + I_{1,z} + I_{2,z}\right] \ddot{\theta}_1 \\ &\quad + \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2\right) + I_{2,z}\right] \ddot{\theta}_2 \\ &\quad - (m_2 L_1 L_2 \sin \theta_2) \dot{\theta}_1 \dot{\theta}_2 - \left(\frac{m_2 L_1 L_2 \sin \theta_2}{2}\right) \dot{\theta}_2^2 \end{aligned} \quad (1.35)$$

$$\Gamma_2 = \left[m_2 \left(\frac{1}{2} L_1 L_2 \cos \theta_2 + \frac{1}{4} L_2^2 \right) + I_{2,z} \right] \ddot{\theta}_1 + \left[\frac{1}{4} m_2 L_2^2 + I_{2,z} \right] \ddot{\theta}_2 + \left(\frac{m_2 L_1 L_2 \sin \theta_2}{2} \right) \dot{\theta}_1^2 \quad (1.36)$$

Ces deux expressions différentielles peuvent être écrites sous la forme matricielle (1.14) avec la matrice $A(\theta)$ et le vecteur $C(\theta, \dot{\theta})$ qui sont sous les formes suivantes :

$$A(\theta) = \begin{bmatrix} a_1 + 2a_2 \cos \theta_2 & a_3 + a_2 \cos \theta_2 \\ a_3 + a_2 \cos \theta_2 & a_3 \end{bmatrix}$$

$$C(\theta, \dot{\theta}) = \begin{bmatrix} -a_2 \sin \theta_2 \dot{\theta}_2 & -a_2 \sin \theta_2 (\dot{\theta}_1 + \dot{\theta}_2) \\ a_2 \sin \theta_2 \dot{\theta}_1 & 0 \end{bmatrix}$$

Avec :

$$a_1 = \frac{1}{4} m_1 L_1^2 + \frac{1}{4} m_2 L_2^2 + m_2 L_1^2 + I_{1,z} + I_{2,z}$$

$$a_2 = \frac{1}{2} m_2 L_1 L_2$$

$$a_3 = \frac{1}{4} m_2 L_2^2 + I_{2,z}$$

Alors les deux équations différentielles sont :

$$\begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix} = \begin{bmatrix} a_1 + 2a_2 \cos \theta_2 & a_3 + a_2 \cos \theta_2 \\ a_3 + a_2 \cos \theta_2 & a_3 \end{bmatrix} * \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -a_2 \sin \theta_2 \dot{\theta}_2 & -a_2 \sin \theta_2 (\dot{\theta}_2 + \dot{\theta}_1) \\ a_2 \sin \theta_2 \dot{\theta}_1 & 0 \end{bmatrix} * \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

En effectuant un changement de paramètre pour simplifier

$$M * \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + N * \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix}$$

Avec :

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

$$N = \begin{bmatrix} -h\dot{\theta}_2 & -h(\dot{\theta}_1 + \dot{\theta}_2) \\ h\dot{\theta}_1 & 0 \end{bmatrix}$$

Et

$$M_{11} = a_1 + 2a_2 \cos \theta_2$$

$$M_{12} = M_{21} = a_3 + \frac{1}{2} a_2 \cos \theta_2$$

$$M_{22} = a_3$$

$$h = a_2 \sin \theta_2$$

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = -M^{-1}N \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + M^{-1} \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix}$$

$$\dot{x} = f(x) + g(x) * u$$

Avec:

$$f(x) = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ -M^{-1}N \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \end{bmatrix}$$

$$g(x) = \begin{bmatrix} 0 \\ \dot{0} \\ M^{-1} \end{bmatrix}$$

$$u = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix}, \text{ et } X = [q_1 \ q_2 \ \dot{q}_1 \ \dot{q}_2]^T$$

1.3. Application et Simulation :

Un contrôleur PD est obtenu par l'association de deux actions (proportionnelle, dérivée) et il remplit essentiellement les deux fonctions suivante :

- Il fournit un signal de commande en tenant compte de l'évolution du signal de sortie par rapport à la consigne.
- Il anticipe les variations de la sortie grâce au terme dérivateur.

La description mathématique du contrôleur PD appliqué à l'entrée de commande du bras manipulateur est décrite par :

$$\Gamma = -K_p.e - K_d.\dot{e}$$

Où les gains K_d et K_p ont les valeurs suivantes : $K_d = 100I$ et $K_p = 20 * K_d$.

Les références sont $X_{d1} = 1$ et $X_{d2} = -1$.

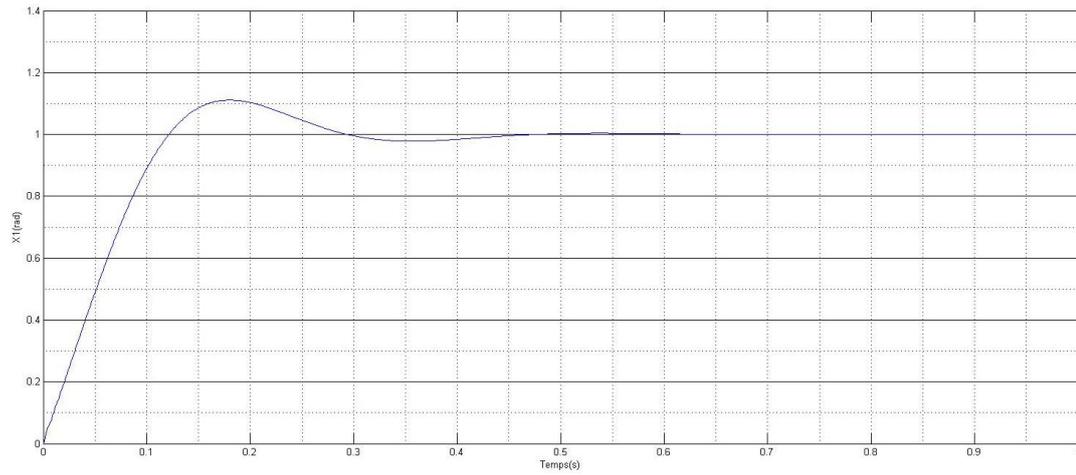


Figure 1. 3 – L'angle q_1 pour une commande PD

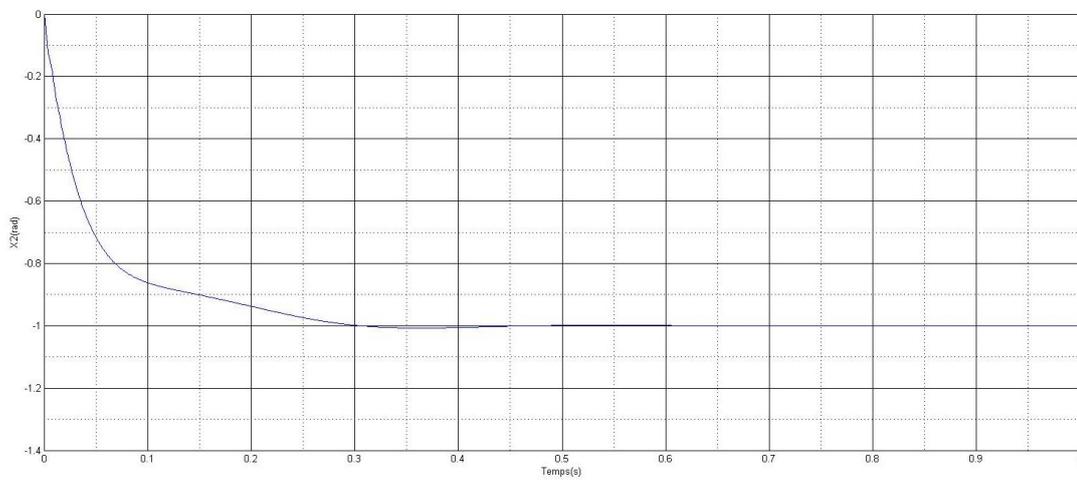


Figure 1. 4 – L'angle q_2 pour une commande PD

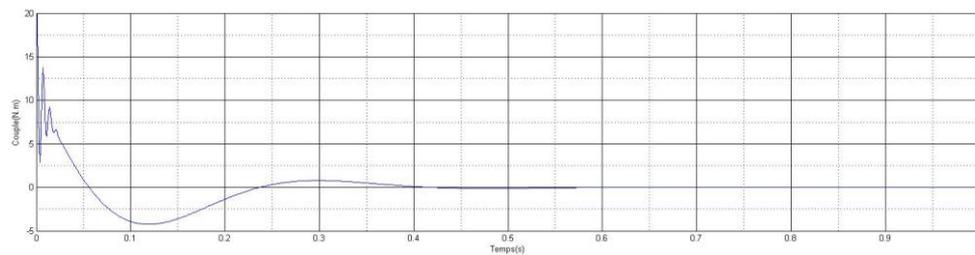
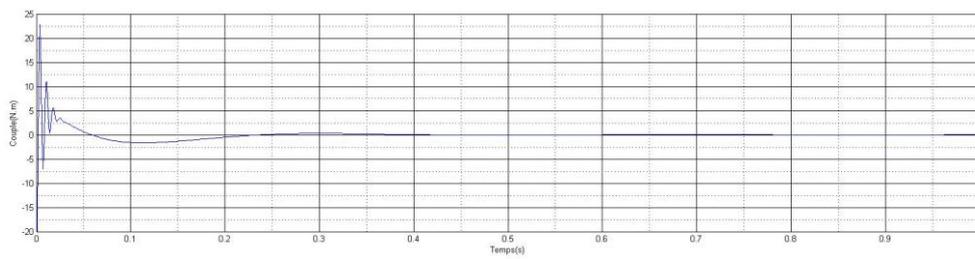


Figure 1. 5 – Signale de commande pour une commande PD

Conclusion :

Dans ce chapitre, une étude détaillée sur un robot manipulateur a deux articulations a été présentée, où nous avons parlé de la géométrie, de la constitution, et de la dynamique de ce type de robots.

Ce chapitre est une base pour le développement ultérieur du modèle dynamique du robot que nous allons utiliser pour la commande.

Chapitre 2

Multimodèles et modèles flous de Takagi-Sugeno

Ce chapitre a pour objet de présenter certains travaux sur la modélisation, la stabilité, et la stabilisation des modèles flous de type Takagi Sugeno (T-S). Nous présentons d'abord les différentes techniques d'obtention d'un modèle T-S. Ensuite, les notions de stabilité et de stabilisation de ce type de modèles rencontrées dans la littérature seront abordées. Enfin une application sur le bras manipulateur présenté dans le premier chapitre.

L'ensemble des notions présentées au cours de ce chapitre permettra alors de positionner notre travail dans le domaine et de poser les problématiques abordées dans ce mémoire.

2.1. Représentation d'état et systèmes non linéaires :

Tout système physique à évolution continue peut s'écrire sous la forme d'une représentation d'état. Celle-ci permet de décrire des relations d'entrées sorties d'un système par le biais d'une modélisation sous la forme d'équations différentielles ordinaires. La forme générale d'une représentation est donnée par :

$$\begin{cases} f(\dot{x}(t), x(t), u(t)) = 0 \\ y(t) = h(x(t), u(t)) \end{cases} \quad (2.1)$$

Où $x(t)$ est le vecteur d'état du système $u(t)$ le vecteur d'entrée et $y(t)$ le vecteur de sortie. La première équation est appelée « équation d'état » et la seconde, « équation de sortie ».

Notons que le système (2.1) est donné sous forme générale et inclut la classe des modèles écrit sous la forme d'une représentation d'état, dite « standard » affines en la commande donnée sous la forme :

$$\begin{cases} \dot{x}(t) = f(x(t)) + g(x(t))u(t) \\ y(t) = s(x(t)) + m(x(t))u(t) \end{cases} \quad (2.2)$$

Où $f(x(t))$ est la fonction d'état, $g(x(t))$ la fonction d'entrée, $s(x(t))$ la fonction de sortie et $m(x(t))$ est la matrice de couplage entrée-sortie.

Ce type de systèmes, couramment rencontrés en automatique, fera l'objet de la première partie de ce chapitre.

2.2. Présentation des modèles flous de type Takagi-Sugeno (T-S)

Les modèles flous de type Takagi-Sugeno sont représentés dans l'espace d'état par des règles floues de type « Si-Alors » [Takagi et Sugeno, 1985]. Les parties prémisses de ces règles floues sont représentatives de l'univers du discours sur lequel le modèle flou est valide et, les parties conclusions correspondent à des modèles locaux invariants dans le temps (représentations d'état linéaires). A titre d'exemple, ce type de modèle flou s'avère utile pour la représentation des systèmes non linéaires tels que les systèmes électriques, chaotiques, etc. La $i^{\text{ème}}$ règle floue d'un modèle T-S continu (en temps continu) (MFC) s'écrit alors sous la forme :

$$\begin{aligned} & \text{Si } z_1(t) \text{ est } M_1^i \text{ et } \dots \text{ et } z_p(t) \text{ est } M_p^i \\ \text{Alors } & \begin{cases} \dot{x}(t) = A_i x(t) + B_i u(t) \\ y(t) = C_i x(t) + D_i u(t) \end{cases} \quad i = 1, 2, \dots, r. \end{aligned} \quad (2.3)$$

Où R^i représente la $i^{\text{ème}}$ règle floue, M_p^i pour $i=1, \dots, r$ sont les sous-ensembles flous, r le nombre de règles floues, $z_p(t)$ sont les variables de prémisses qui dépendent de l'entrée et/ou de l'état du système, $x(t) \in \mathbb{R}^n$, $y(t) \in \mathbb{R}^q$ et $u(t) \in \mathbb{R}^m$ représentent respectivement le vecteur d'état, le vecteur de sortie et le vecteur de commande. $A_i \in \mathbb{R}^{n \times n}$, $B_i \in \mathbb{R}^{q \times n}$, $C_i \in \mathbb{R}^{q \times m}$ et $D_i \in \mathbb{R}^{q \times m}$ sont des matrices décrivant la dynamique du système.

Notons qu'une discrétisation de tels modèles est possible par une représentation d'état en temps discret (MFD). Le temps t est alors congru à k et le modèle est alors décrit par les équations de récurrence suivantes :

$$\begin{aligned} & \text{Si } z_1(t) \text{ est } M_1^i \text{ et ... et } z_p(t) \text{ est } M_p^i \\ \text{Alors } & \begin{cases} x_i(k+1) = A_i x(k) + B_i u(k) \\ y_i(k) = C_i x(k) + D_i u(k) \end{cases} \end{aligned} \quad (2.4)$$

A chaque règle R^i est attribué à un poids noté $\omega_i(z(t))$. Ce poids dépend du degré d'appartenance des variables de prémisses $z_j(t)$ aux sous-ensembles flous $M_p^i(z_j(t))$ et du connecteur « ET » reliant les prémisses choisi telles que :

$$\omega_i(z(t)) = \prod_{j=1}^p M_p^i(z_j(t)) \quad \text{Pour } i = 1, \dots, r \quad (2.5)$$

$M_p^i(z_j(t))$ Représente la valeur de la fonction d'appartenance $z_j(t)$ à l'ensemble flou M_p^i . On a alors les propriétés suivantes :

$$\begin{cases} \sum_{i=1}^r \omega_i(z(t)) > 0 \\ \omega_i(z(t)) \geq 0 \end{cases} \quad i = 1, 2, \dots, r \quad (2.6)$$

On pose :

$$h_i(z(t)) = \frac{\omega_i(z(t))}{\sum_{i=0}^r \omega_i(z(t))} \quad (2.7)$$

$h_i(z(t))$ représente donc la fonction d'activation de la $i^{\text{ème}}$ règle du modèle flou. Pour $i = 1, 2, \dots, r$, ces fonctions vérifient la propriété d'une somme convexe, c'est-à-dire :

$$\begin{cases} \sum_{i=1}^r h_i(z(t)) = 1 \\ h_i(z(t)) \geq 0 \end{cases} \quad i = 1, 2, \dots, r$$

Finalement, la défuzzification du modèle flou permet d'obtenir la représentation d'état d'un modèle non linéaire par la combinaison de modèles locaux invariants dans le temps par des fonctions d'activation non linéaires. On obtient alors :

$$\begin{cases} \dot{x}(t) = \sum_{i=1}^r h_i(z(t)) (A_i x(t) + B_i u(t)) \\ y(t) = \sum_{i=1}^r h_i(z(t)) (C_i x(t) + D_i u(t)) \end{cases} \quad i = 1, 2, \dots, r. \quad (2.8)$$

Dans le cadre de la modélisation par modèles Takagi-Sugeno, on rencontre souvent les termes: variables de prémisses, fonctions d'appartenance ou d'activation, zones de fonctionnement (sous espaces) et règles floues. Ceux-ci sont précisés par les définitions suivantes :

Règles floues :

Dénombrées par $r \in \mathbb{N}$ dans la représentation d'état d'un modèle flou T-S. Elles correspondent au nombre de modèles locaux LTI (Linéaire à Temps Invariant).

Variables de prémisses :

Notées $z(t) \in \mathbb{R}^j$. Grandeurs connues et accessibles permettent l'évaluation des fonctions d'appartenance. Elles dépendent éventuellement des variables d'état et de commande et de sortie.

Fonctions d'appartenance :

Notées $h_i(z(t)) : \mathbb{R}^j \rightarrow \mathbb{R}$, ce sont des fonctions non linéaires dépendant des variables de prémisses associées aux différentes zones de fonctionnement. Elles permettent de traduire la contribution d'un modèle local LTI correspondant à un point de fonctionnement par rapport à la zone de fonctionnement du système. Ainsi, elles assurent le passage progressif d'un modèle local LTI aux modèles locaux voisins.

Les zones de fonctionnement :

Représentées par des domaines l_i obtenus via la décomposition de l'espace de fonctionnement du système l , avec $l = \cup_i l_i$

La figure 2.1 illustre le schéma détaillé d'un modèle T-S standard. Notons que les modèles flous de type Takagi-Sugeno sont dotés d'une structure mathématique intéressante de point de vue de l'automatique. En effet, ils permettent de diminuer la complexité d'un problème non linéaire à traiter (stabilité, stabilisation, observation, diagnostic,...etc.) en le décomposant en un

ensemble de problèmes linéaires locaux. L'ensemble des solutions locales correspondantes à ces derniers constitue alors la solution globale du problème non linéaire initial.

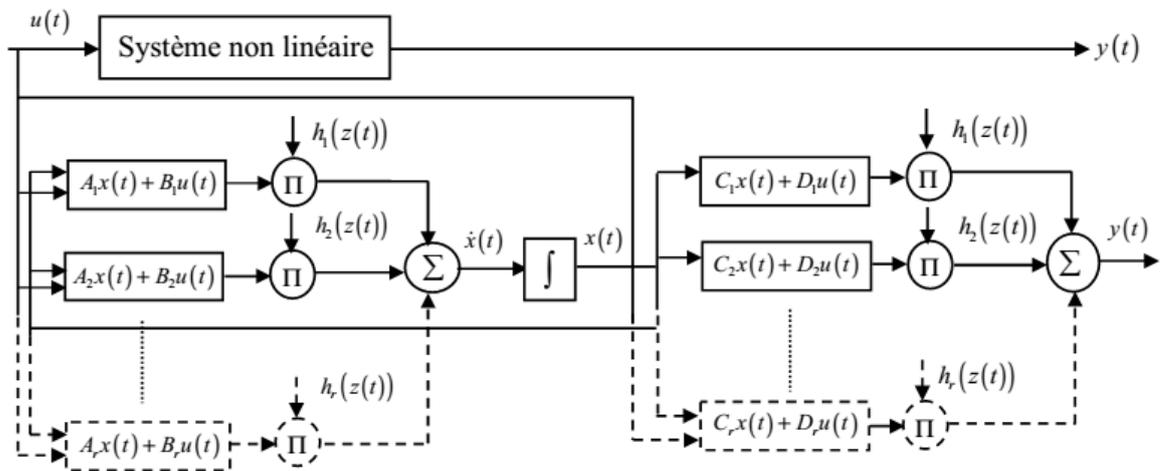


Figure 2. 1 – Structure et implémentation d'un modèle T-S Standard

2.3. Les différents types de multimodèles :

Les modèles T-S peuvent être regroupés en deux familles [Abonyi et al., 1999] distinctes ayant les structures indiquées par les figures suivantes [Bezzaoucha, 2013] :

Modèles T-S couplés	Modèles T-S découplés
L'état global x couple tous les états par le mélange des équations dynamiques des sous-modèles.	Chaque sous-modèle possède un espace d'état propre dans lequel il évolue indépendamment.
$\begin{cases} \dot{x}(t) = \sum_{i=1}^r \mu_i(z(t)) (A_i x(t) + B_i u(t)) \\ y(t) = \sum_{i=1}^r \mu_i(z(t)) C_i x(t) \end{cases}$	$\begin{cases} \dot{x}_i(t) = A_i x_i(t) + B_i u(t) \\ y_i(t) = C_i x_i(t) \\ y(t) = \sum_{i=1}^r \mu_i(z(t)) y_i(t) \end{cases}$

Tableau 2. 1 – Multimodèles couplés et découplés

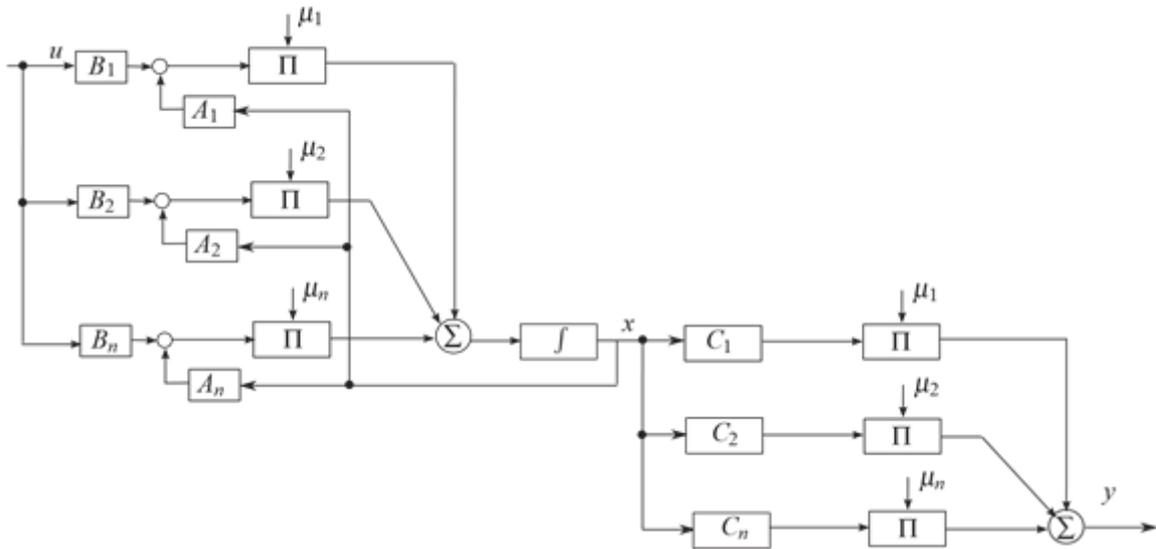


Figure 2. 2 – Multimodèles couplés

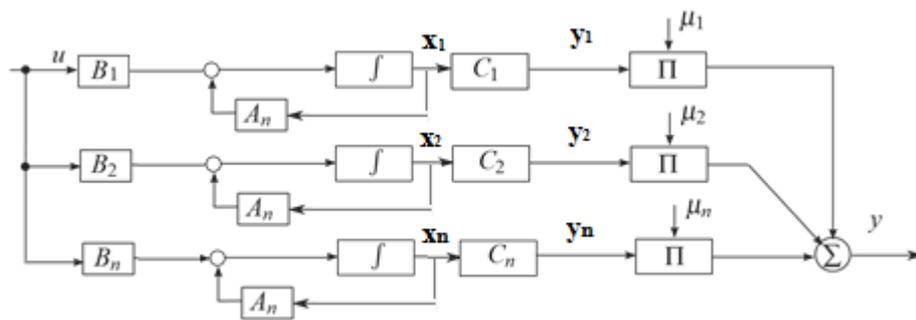


Figure 2. 3 – – Multimodèles découplés

Comme le montre la figure suivante :

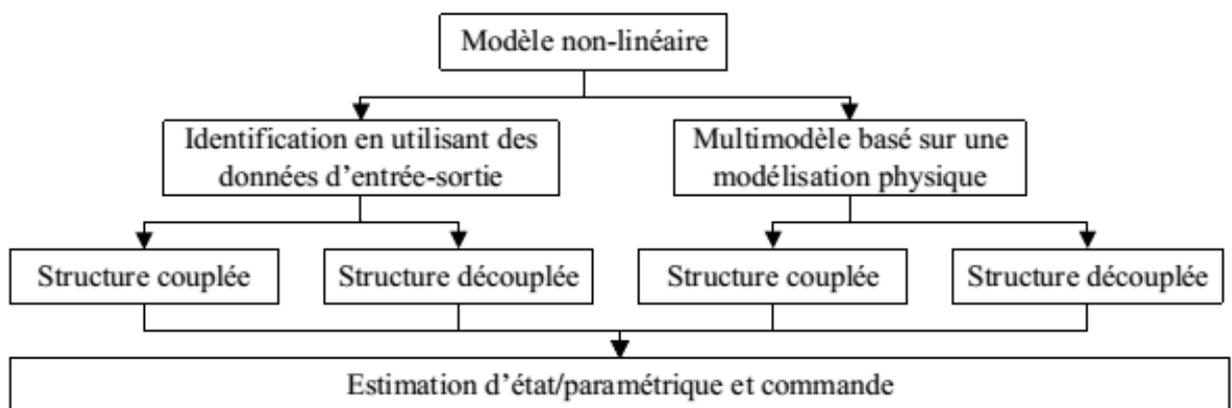


Figure 2. 4 – Différentes approches possibles

Il existe plusieurs approches possibles afin d'obtenir un multimodèle, nous allons en explorer quelques-unes dans ce mémoire.

2.4. Obtention des modèles flous de type Takagi-Sugeno (T-S)

Dans la littérature, il existe trois approches permettant le passage d'un modèle non linéaire affine en la commande à un modèle T-S. Ces approches visent à représenter les systèmes non linéaires complexes sur un large domaine de fonctionnement. Ces différentes approches sont :

- approche par identification [Gasso et al., 1999][Gasso et al., 2000]. Les mesures acquises sur les entrées et les sorties du système permettent l'identification des paramètres des modèles locaux autour des différents points de fonctionnement préalablement définis. Dans ce cas, le problème d'identification du modèle non linéaire se réduit à l'identification des modèles locaux (sous-modèles) LTI. Notons que, cette méthode est souvent utilisée dans le cas des systèmes dotés d'une dynamique difficile à décrire à l'aide d'un modèle analytique.
- approche par secteur non linéaire. Cette méthode a été initiée par [Kawamoto et al., 1992] et étendue par [Tanaka et Wang, 2001] et [Morère, 2001]. Le principe de celle-ci est basé sur une transformation polytopique convexe des termes non linéaires d'un système dynamique. Autrement dit, cette méthode consiste à trouver un secteur tel que $a_1x < f(x(t), u(t)) < a_2x$ avec $\dot{x}(t) = f(x(t), u(t))$ représente un système non linéaire.
- approche par linéarisation [Ma et al., 1998][Tanaka et Wang 2001]. Le principe de cette méthode consiste à linéariser le système non linéaire autour d'un ensemble fini de points de fonctionnement judicieusement choisis, conduisant à un nombre défini de modèles LTI. L'obtention d'un représentant T-S dans ce cas, est réalisé par l'interconnexion de ces modèles LTI à l'aide des fonctions d'appartenance non linéaires judicieusement choisies (gaussiennes, triangulaires, trapézoïdales, ...etc.).

Dans la suite de ce manuscrit, l'intérêt est porté sur la troisième méthode, puisqu'elle présente des avantages du point de vue précision et connaissance des fonctions d'appartenance assurant la combinaison des modèles locaux LTI. En effet, l'approche par linéarisation par rapport à l'approche par secteur non linéaire permet, d'une part, de minimiser l'erreur lors du passage du modèle analytique non linéaire au modèle T-S, d'autre part de choisir le nombre de modèles locaux. Il convient de souligner qu'il peut s'avérer difficile de trouver un secteur global pour un système non linéaire quelconque. Dans ce cas, il est nécessaire de considérer un

secteur non linéaire local. Les figures 2.5 représentent respectivement les secteurs non linéaires global et local.

2.4.1. Approche par Secteur non-linéaire :

Les secteurs non-linéaires sont basés sur l'idée suivante [Tanaka et Wang, 2001]. Considérons un système non linéaire simple $\dot{x}(t) = f(x)$, où $f(0) = 0$. Le but est de trouver le secteur global de telle sorte que $\dot{x}(t) = f(x) \in [a_1, a_2]x(t)$. La Figure 2.5a illustre l'approche de non-linéarité. Il est parfois difficile de trouver des secteurs globaux pour les systèmes non linéaires généraux. Dans ce cas, nous pouvons considérer la non-linéarité du secteur local. Cela est raisonnable comme les variables des systèmes physiques sont toujours bornées.

La Figure 2.5b montre la non-linéarité d'un secteur local, où deux lignes deviennent les secteurs locaux sous $-d < x(t) < d$.

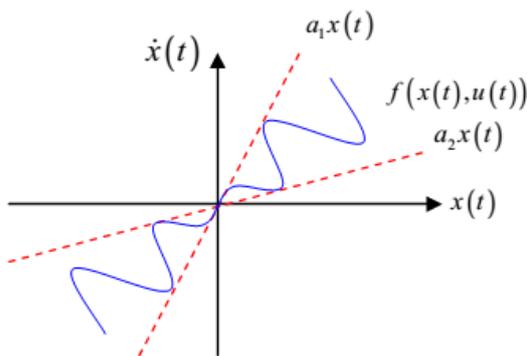


Figure 2. 5 a – Secteur non linéaire global

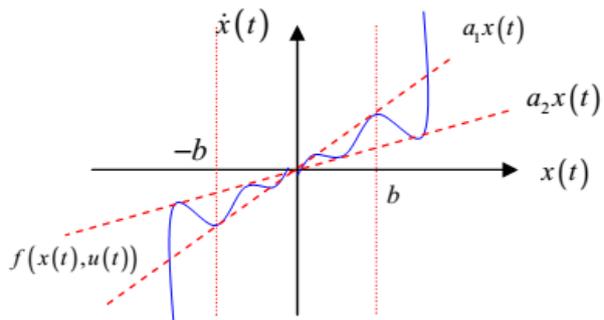


Figure 2. 5b – Secteur non linéaire local

Remarque 2.1 :

On considère le système non linéaire $\dot{x} = f(x(t))$, avec $f(0) = 0$. Selon les propriétés des termes non linéaires rencontrés dans le modèle mathématique non linéaire, nous distinguons deux types de représentant T-S, en effet :

- Si toutes les non-linéarités du système sont continues et bornées sur \mathbb{R}^n , alors le modèle T-S représente d'une manière exacte le système non linéaire sur l'intégralité de l'espace des variables d'état \mathbb{R}^n .
- Si toutes les non-linéarités du système sont uniquement continues, alors le modèle T-S représente de façon exacte le système non linéaire sur un sous-espace compact de l'espace des variables d'état \mathbb{R}^n .

Remarque 2.2 :

Les modèles T-S obtenus via une transformation polytopique convexe dépendent directement du nombre des non-linéarités à découper. Ainsi, lorsque l'on a nl termes non linéaires, alors le modèle T-S est constitué de 2^{nl} règles floues.

2.4.2. Approximation locale dans les espaces de partition floue :

Une autre approche pour obtenir des modèles flous T-S est l'approximation dite locale dans les espaces de partition floue. L'esprit de l'approche est de rapprocher les termes non linéaires par des termes linéaires judicieusement choisis. Cette procédure conduit à une réduction du nombre de modèles de règles.

Le nombre de règles est directement lié à la complexité de l'analyse et de la conception des conditions LMI. En effet, le nombre de règles pour le système de contrôle global est essentiellement la combinaison des règles du modèle et des règles de contrôle.

Comme indiqué ci-dessus, la technique d'approximation locale conduit à la réduction du nombre de règles floues pour les modèles. Cependant, la conception des lois de commande sur la base des modèles flous ne peuvent pas garantir la stabilité des systèmes non linéaires d'origine en vertu de ces lois de contrôle. L'une des approches pour atténuer le problème est d'introduire la conception de contrôleur robuste [Tanaka et Wang, 2001].

2.5. Stabilité et stabilisation des modèles T-S standards

L'étude de la stabilité et la synthèse des contrôleurs flous pour les Modèles T-S Standards (MTSS) (2.8) sont généralement basées sur la théorie de Lyapunov [Liapounoff, 1907]. Le principe de cette dernière est inspiré d'une réalité physique. En effet, si l'énergie d'un système est continûment dissipée, au final le système va atteindre un point d'équilibre. Dans la suite, sans perte de généralité, on suppose que le point d'équilibre est l'origine.

2.5.1. Stabilité des modèles T-S :

L'étude de la stabilité d'un MTSS autonomes (2.8) permet d'établir si sa dynamique est intrinsèquement stable lorsqu'il n'est soumis à aucune excitation externe $u(t) = 0$. Dans cette section, afin de permettre au lecteur d'appréhender les résultats proposés dans la suite de ce manuscrit, on présente les résultats significatifs, à l'origine des nombreux travaux sur la stabilité des MTSS. Ceux-ci sont donnés sous forme d'Inégalités Matricielles Linéaires (LMI) [Boyd et al., 1994]. Le résultat suivant traite la stabilité des MTSS décrits en temps continu:

Théorème 2.1 : [Tanaka et Sugeno, 1992]

Le MTSS continu autonome $u(t) = 0$ (2.8) est asymptotiquement stable s'il existe une matrice $P^T = P > 0$, telles que les LMI suivante est vérifiée pour $i=1, \dots, r$:

$$\text{Cas continu} \quad A_i^T P + P A_i < 0 \quad (2.9)$$

Preuve:

En considérant la fonction candidate quadratique de Lyapunov :

$$V(x(t)) = x^T(t) P x(t) \quad (2.10)$$

Le MTSS autonome (2.8) $u(t) = 0$ est stable si:

$$\dot{V}(x(t)) = \dot{x}^T(t) P x(t) + x^T(t) P \dot{x}(t) < 0 \quad (2.11)$$

C'est-à-dire si :

$$\dot{V}(x(t)) = \dot{x}^T \left(\sum_{i=1}^r h_i(z(t)) (A_i^T P + P A_i) \right) x(t) < 0 \quad (2.12)$$

Qui est vérifiée si les conditions suffisantes du théorème 2.1 le sont.

Remarque 2.3 :

Notons qu'aucune information sur les fonctions d'appartenance $h_i(z(t))$ n'est prise en compte dans le théorème 2.1. De ce fait, les conditions de stabilité LMI sont seulement suffisantes (et non nécessaires) et par conséquent conservatives. En effet, pour obtenir ces conditions, on cherche une solution à un problème d'optimisation en considérant que, pour qu'une somme soit négative, chacun des termes de cette somme doit être négatif. Or, il est évidemment possible de chercher des solutions aux conditions du théorème 2.1 qui autoriseraient certains termes de la somme (2.12) à être positifs alors que l'ensemble restera négatif. Dans ce cas, on obtient des conditions de stabilité dites « relâchées », c'est-à-dire moins conservatives.

Dans le cas où l'on peut établir que le système est stable, il peut être intéressant d'améliorer les performances. Dans le cas contraire, il s'avère nécessaire de synthétiser une loi de commande floue stabilisante pour pouvoir ramener le système vers le point d'équilibre.

2.5.2. Stabilisation par retour d'état des modèles T-S standards

Afin d'assurer la stabilité d'un MTSS en boucle fermée, on réalise la synthèse d'une loi de commande adéquate. Plusieurs lois de commande floues ont été proposées dans la littérature.

Les plus répandues se basent sur des lois de commande de type compensation parallèle distribuée (PDC, Parallel Distributed Compensation) [Wang et al., 1996][Tanaka et al., 1998].

Notons, par ailleurs que, suivant la classe des modèles T-S considérés, des variantes de ce type de loi de commande ont été également proposées dans la littérature, par exemple la PDC proportionnelle (PPDC) [Lin et Er, 2001] ou encore la loi de commande de type compensation et division pour modèles flous [Guerra et al., 1999].

2.5.2.1. Lois de commande PDC (Parallel distributed compensation)

La loi de commande PDC [Wang et al., 1996][Tanaka et al., 1998] a l'avantage de considérer les mêmes prémisses que les règles floues contenues dans le modèle T-S à stabiliser. De ce fait, cela revient à considérer que, à chaque modèle local correspond une commande par retour d'état linéaire que l'on peut interpoler par les mêmes fonctions d'activation $h_i(z(t))$ que celles du modèle T-S. Notons que dans ce cas, lorsque le modèle flou obtenu par découpage est exact, c'est-à-dire par transformation polytopique convexe, alors cette loi de commande est valable sur tout l'espace compact des variables d'état sur lequel a été effectué le découpage. Ainsi, cette loi de commande est donnée par :

$$u(t) = - \frac{\sum_{i=1}^r \omega_i(z(t)) \{K_i x(t)\}}{\sum_{i=1}^r \omega_i(z(t))} = - \sum_{i=1}^r h_i(z(t)) \{K_i x(t)\} \quad (2.13)$$

Avec $K_i \in \mathbb{R}$ représentent les matrices des gains de retour d'état.

La synthèse d'un correcteur flou consiste alors à déterminer les matrices de gain de retour d'état notées K_i .

Stabilisation des MTSS en boucle fermée continus :

Dans le cas continu, pour obtenir l'expression du modèle flou T-S en boucle fermée, on substitue la loi de commande (2.13) dans (2.8), on obtient alors :

$$\dot{x}(t) = \sum_{i=1}^r \sum_{j=1}^r h_i(z(t)) h_j(z(t)) (A_i - B_i K_j) x(t) \quad (2.14)$$

Des conditions suffisantes de stabilisation sont présentées dans le théorème suivant.

Théorème 2.2 : [Tanaka et Wang, 2001]

L'équilibre d'un système de commande floue (2.14) est globalement asymptotiquement stable s'il existe une matrice positive commune telle que les deux conditions suivantes sont satisfaites:

$$\text{Cas continu} \quad G_{ij}^T P - P G_{ij} = -Q \quad (2.15)$$

Pour tous, $i, j = 1, 2, \dots, l$, où $G_{ij} = A_i - B_i K_j$ et Q_{ij} est une matrice définie positive, un P commun doit exister pour tous les sous-systèmes.

Le problème de conception du contrôleur floue basée sur un modèle T-S est de sélectionner K_i tels que les conditions de stabilité dans le Théorème 2.2 sont satisfaites. Il est reconnu depuis longtemps qu'il y a un manque de procédures systématiques de trouver une matrice définie positive commune P pour vérifier la stabilité du système de commande floue. Le problème de P commune a été résolu de manière efficace par des techniques d'optimisation convexe LMI (Linéaire Matrix Inequality) [Tanaka et Wang, 2001][Elkhalil 2011]. Cependant, la commande floue (2.13) ne garantit pas la stabilité du système en présence d'incertitudes ou variations des paramètres [Chekireb et al., 2003]. En outre, la synthèse des paramètres de contrôle n'est pas possible pour les systèmes dont les paramètres sont inconnus. Pour pallier ces inconvénients, le prochain chapitre présente un système de commande adaptative pour les systèmes flous de type T-S dont les paramètres sont inconnus ou variables dans le temps.

2.6. Commande PDC

Considérons de nouveau le système non linéaire représenté par le modèle (2.3), où l'état $x \in \mathbf{R}^n$ est mesurable, $A_i \in \mathbf{R}^{n \times n}$ et $B_i \in \mathbf{R}^{n \times m}$, sont des matrices constantes inconnues et (A_i, B_i) sont contrôlables. L'objectif est de choisir le vecteur d'entrée $u \in \mathbf{R}^m$ de telle sorte que tous les signaux dans le système en boucle fermée sont bornés et l'état x suit l'état $x_m \in \mathbf{R}^n$ du modèle de référence spécifiée par le système suivant :

$$\dot{x}_m(t) = \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) \{ (A_m)_{ij} x_m(t) + (B_m)_{ij} r \}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} \quad (2.16)$$

Où $(A_m)_{ij} \in \mathbf{R}^{n \times n}$ satisfait la stabilité donnée en Théorème 2.2, C'est-à-dire., il existe une matrice symétrique définie positive commune $P = P^T > 0$ tel que $(A_m)_{ij}^T P + P (A_m)_{ij} < -Q_{ij}$ pour tous $Q_{ij} = Q_{ij}^T > 0$, $(B_m)_{ij} \in \mathbf{R}^{n \times m}$ et $r \in \mathbf{R}^m$ est une référence bornée. Le modèle de

référence et la consigne r sont choisis de sorte que x_m représente la trajectoire désirée que doit suivre x .

Si les matrices, A_i , B_i étaient connus, nous pourrions appliquer la loi de commande :

$$u(t) = \frac{\sum_{j=1}^r \mu_j(x(t)) \{-K_j^* x(t) + L_j^* r(t)\}}{\sum_{j=1}^r \mu_j(x(t))} \quad (2.17)$$

Où $\mu_j(x(t)) = \omega_j(x(t))$, et on obtient le système en boucle fermé :

$$\dot{x}(t) = \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) \{(A_i - B_i K_j^*) x(t) + B_i L_j^* r(t)\}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} \quad (2.18)$$

$K_j^* \in \mathbf{R}^{m \times n}$ et $L_j^* \in \mathbf{R}^{m \times m}$ Sont choisis pour satisfaire les équations algébriques suivantes:

$$A_i - B_i K_j^* = (A_m)_{ij} \quad B_i L_j^* = (B_m)_{ij} \quad (2.19)$$

De tel sorte que les matrices de transferts du système en boucle fermée est la même que celle du modèle de référence, Alors $x \longrightarrow X_m$ de façon asymptotiquement rapide pour tout signal d'entrée bornée.

2.7. Application et simulation :

Dans cette section, nous représentons le modèle de Takagi-Sugeno du robot manipulateur, et une description de la conception du contrôleur PDC.

Considérons le robot manipulateur de chapitre 1, dont la dynamique (1.38) et (1.39) peut être écrite explicitement :

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = -M^{-1} N \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + M^{-1} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (2.20)$$

Avec :

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \quad N = \begin{bmatrix} -h\dot{\theta}_2 & -h(\dot{\theta}_1 - \dot{\theta}_2) \\ h\dot{\theta}_1 & 0 \end{bmatrix}$$

Nous avons les équations d'états non linéaires suivantes:

$$\dot{x} = f(x) + g(x) * u \quad (2.21)$$

$$\text{Avec: } f(x) = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ -M^{-1} N \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ \dot{\theta}_1 \\ M^{-1} \end{bmatrix}, \quad u = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix}, \quad \text{et } X[q1 \ q2 \ \dot{q}1 \ \dot{q}2]^T$$

Puis, en appliquant la méthode de linéarisation de Lyapunov aux points de fonctionnement (x_i, u_i) , on obtient la description du modèle linéaire affine pour le robot manipulateur à deux degrés de liberté autour du point de fonctionnement comme suit :

$$\dot{x} = A_{X_i}x + B_{X_i}u + f_{X_i} \quad (2.22)$$

Où A_{X_i} et B_{X_i} désignent les matrices jacobéennes de $f(x)$ et $g(x)$ évalués aux points de fonctionnement (x_i, u_i) .

$$A_{X_i} = \left. \frac{\partial f(X)}{\partial X} \right|_{(x_i, u_i)} + \left. \frac{\partial g(X)}{\partial X} U \right|_{(x_i, u_i)}$$

$$B_{X_i} = g(X)|_{(x_i, u_i)}$$

$$f_{X_i} = f(x_i) - A_{X_i}x_i$$

Pour appliquer la commande PDC, on a besoin d'un modèle de référence pour la poursuite de l'état de système x . Dans cette simulation, les valeurs propres en boucle fermée pour chaque sous-système on était choisies pour être les même que celles du modèle de référence :

$$\dot{x}_m = A_m x_m + B_m u \quad (2.23)$$

$$\dot{x}_m = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -84.39 & 0 & -350 & 0 \\ 0 & -84.39 & 0 & -250 \end{bmatrix} x_m + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} u \quad (2.24)$$

La loi de commande finale u est composée de contrôleur flou PDC combiné avec le compensateur pour éliminer les termes biaisés des modèles locaux (2.22).

$$u = \frac{\sum_{i=1}^r \omega_i(x(t)) \{u_{fi} + u_{ai}\}}{\sum_{i=1}^r \omega_i(x(t))}$$

Les actions de contrôleur PDC sont de la forme suivante:

$$\text{Si } z_1(t) \text{ est } M_1^i \text{ et ... et } z_p(t) \text{ est } M_p^i$$

$$\text{Alors } u_{fi} = -k_i \cdot X + L_i \cdot r \quad i = 1, 2, \dots, r.$$

2.7.1. Multimodèle avec 12 sous-modèles flous:

Pour notre première simulation l'espace entier d'états formé par le vecteur d'état des équations non linéaires de bras manipulateur présenté dans le chapitre 1, est divisé en $3*2*2=12$ sous-espaces flous différents dont le centre est situé aux centres des fonctions d'appartenance représentées sur les figures, et sous la forme mathématique suivante :

$$M_1^1(x_2(t)) = e^{-\left(\frac{x+\frac{\pi}{2}}{\frac{\pi}{4}}\right)^2} \quad M_1^2(x_2(t)) = e^{-\left(\frac{x}{\frac{\pi}{4}}\right)^2} \quad M_1^3(x_2(t)) = e^{-\left(\frac{x-\frac{\pi}{2}}{\frac{\pi}{4}}\right)^2}$$

$$M_2^1(x_3(t)) = e^{-\left(\frac{x+2\pi}{\pi}\right)^2} \quad M_2^2(x_3(t)) = e^{-\left(\frac{x-2\pi}{\pi}\right)^2}$$

$$M_3^1(x_4(t)) = e^{-\left(\frac{x+2\pi}{\pi}\right)^2} \quad M_3^2(x_4(t)) = e^{-\left(\frac{x-2\pi}{\pi}\right)^2}$$

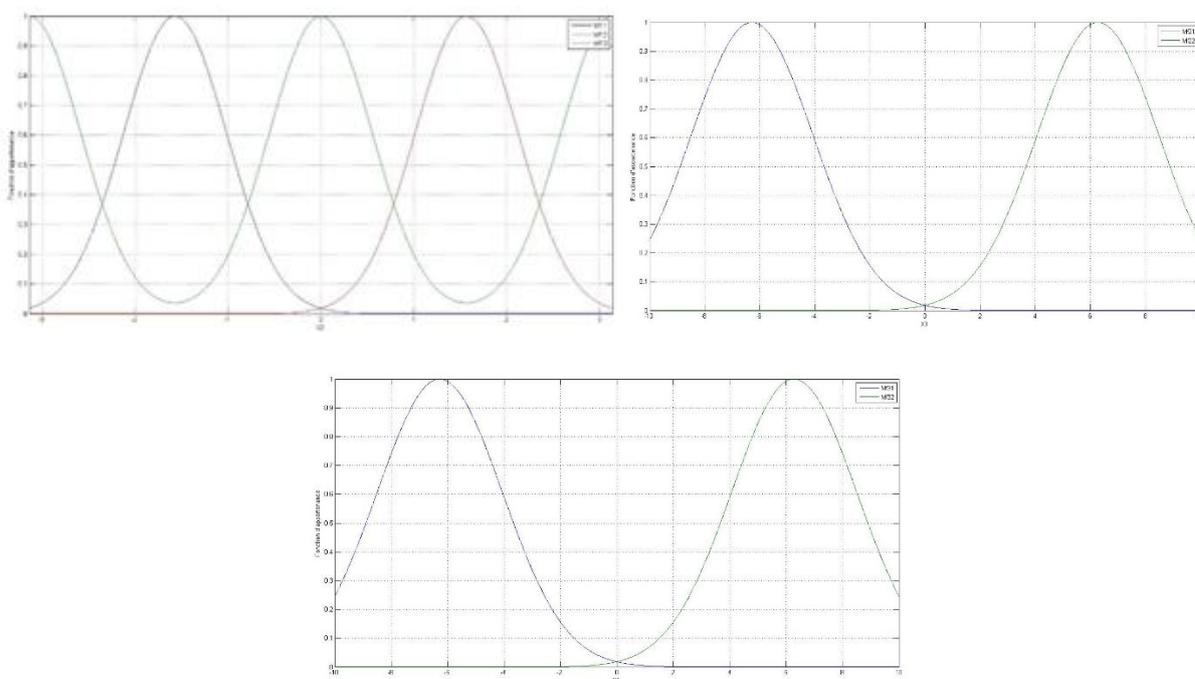


Figure 2. 6 – Fonctions d'appartenance du multimodèle avec 12 modèles locaux

Les 12 modèles linéaires correspondant au 12 points de fonctionnement sont présentés dans l'annexe A.1.

Pour vérifier que le multimodèle est équivalent au système non-linéaire, on va simuler le multimodèle avec une entrée nulle et l'état initial est à l'origine, c'est-à-dire $X= [0 \ 0 \ 0 \ 0]^T$, la figure ci-dessous représente la réponse de notre multimodèle :

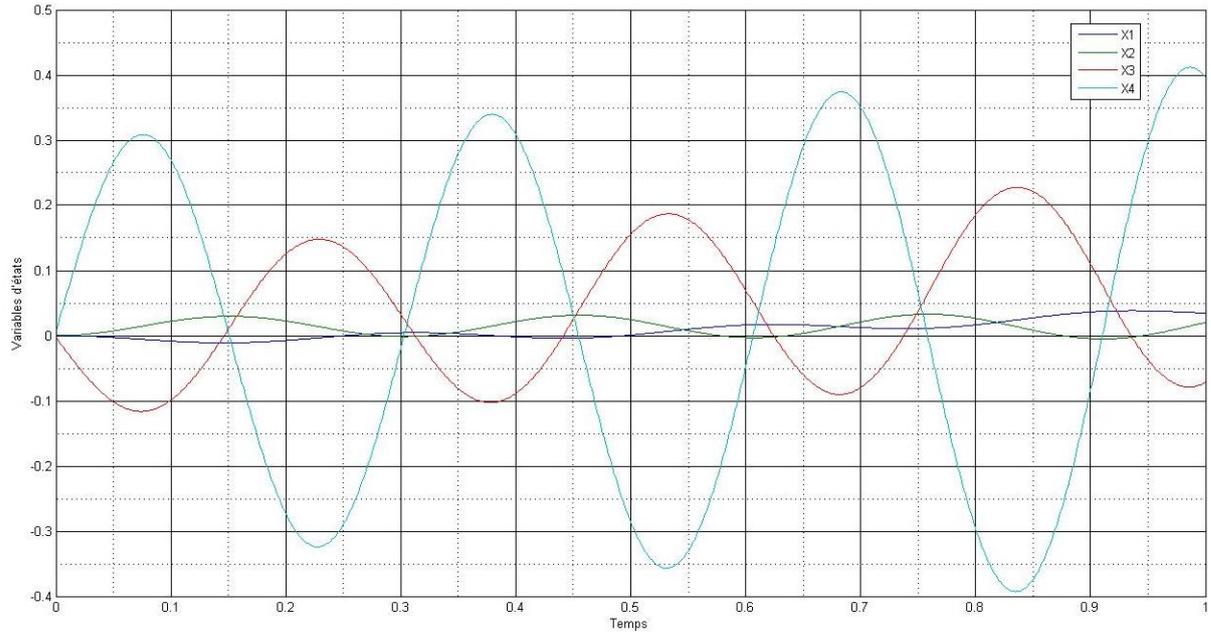


Figure 2. 7 – performance du multimodèle avec 12 modèles locaux

Si un multimodèle est parfaitement équivalent au système non linéaire on aurait eu un résultat à cette simulation avec une sortie nulle et stable (la réponse du système non linéaire), mais dans notre cas on a eu une réponse instable avec une grande erreur, Il est clair que le multimodèle avec 12 sous-espaces flous ne représente pas fidèlement le système non linéaire. Pour résoudre ce problème et avoir plus de précision, une des solutions les plus évidentes est d'augmenter le nombre de sous-modèle.

2.7.2. Multimodèle avec 27 sous-modèles flous:

Pour avoir plus de précision l'espace entier d'états est divisé en $3*3*3=27$ sous-espaces flous différents dont le centre est situé au centre de fonctions d'appartenance représentées sur la figure 2.8, et sous la forme mathématique suivante :

$$M_1^1(x_2(t)) = e^{-\left(\frac{x+\frac{\pi}{2}}{\frac{\pi}{4}}\right)^2}; M_1^2(x_2(t)) = e^{-\left(\frac{x}{\frac{\pi}{4}}\right)^2}; M_1^3(x_2(t)) = e^{-\left(\frac{x-\frac{\pi}{2}}{\frac{\pi}{4}}\right)^2}$$

$$M_2^1(x_3(t)) = e^{-\left(\frac{x+2\pi}{\pi}\right)^2}; M_2^2(x_3(t)) = e^{-\left(\frac{x}{\pi}\right)^2}; M_2^3(x_3(t)) = e^{-\left(\frac{x-2\pi}{\pi}\right)^2}$$

$$M_3^1(x_4(t)) = e^{-\left(\frac{x+2\pi}{\pi}\right)^2}; M_3^2(x_4(t)) = e^{-\left(\frac{x}{\pi}\right)^2}; M_3^3(x_4(t)) = e^{-\left(\frac{x-2\pi}{\pi}\right)^2}$$

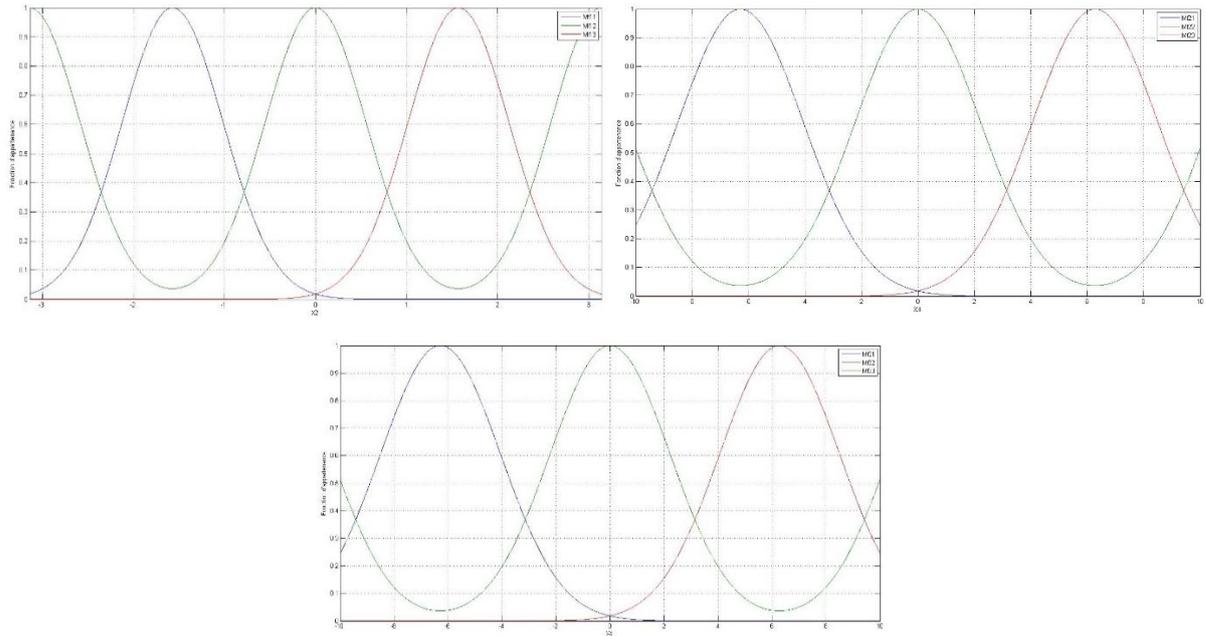


Figure 2. 8 – Fonctions d'appartenance du multimodèle avec 27 modèles locaux

Les 27 sous-modèles linéaires correspondant aux 27 points de fonctionnement qui sont présentés dans l'annexe A.2.

En suivant la même approche que précédemment pour vérifier que le multimodèle est équivalent au système non-linéaire, on va simuler le multimodèle avec une entrée nulle et l'état initial est à l'origine, c'est-à-dire $X = [0 \ 0 \ 0 \ 0]^T$, la figure ci-dessous représente la réponse de notre multimodèle :

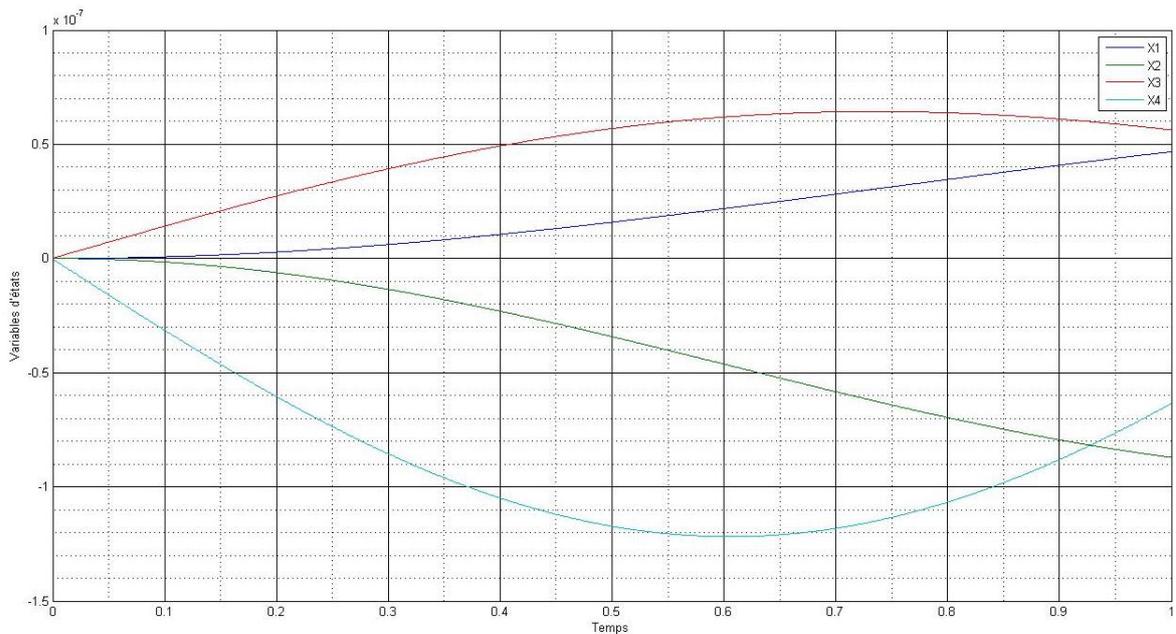


Figure 2. 9 – Stabilité du multimodèle avec 27 modèles locaux

Avec une erreur à l'ordre de 10^{-7} , on estime que 27 sous-modèles linéaires sont largement suffisants, ce qui va nous permettre par la suite d'établir notre commande adaptative finale. A savoir si on augmente encore plus le nombre de sous-modèle, le contrôle ou la stabilisation de notre système sera plus compliquer, alors il est plus judicieux de choisir de continuer notre travaille avec 27 sous-modèle.

2.7.3. Commande de multimodèle T-S (27 modèles) avec PD:

Le contrôleur proportionnel dérivé (PD) proposé dans le chapitre 1, est utilisé comme premier commande avec la forme générale suivante:

$$\tau = -Kp.e - Kd.\dot{e}$$

Où les gains Kp et Kd ont les valeurs suivantes : $Kd = 100I$ et $Kp=20*Kd$.

Les références sont $X_{d1}=1$ et $X_{d2}=-1$.

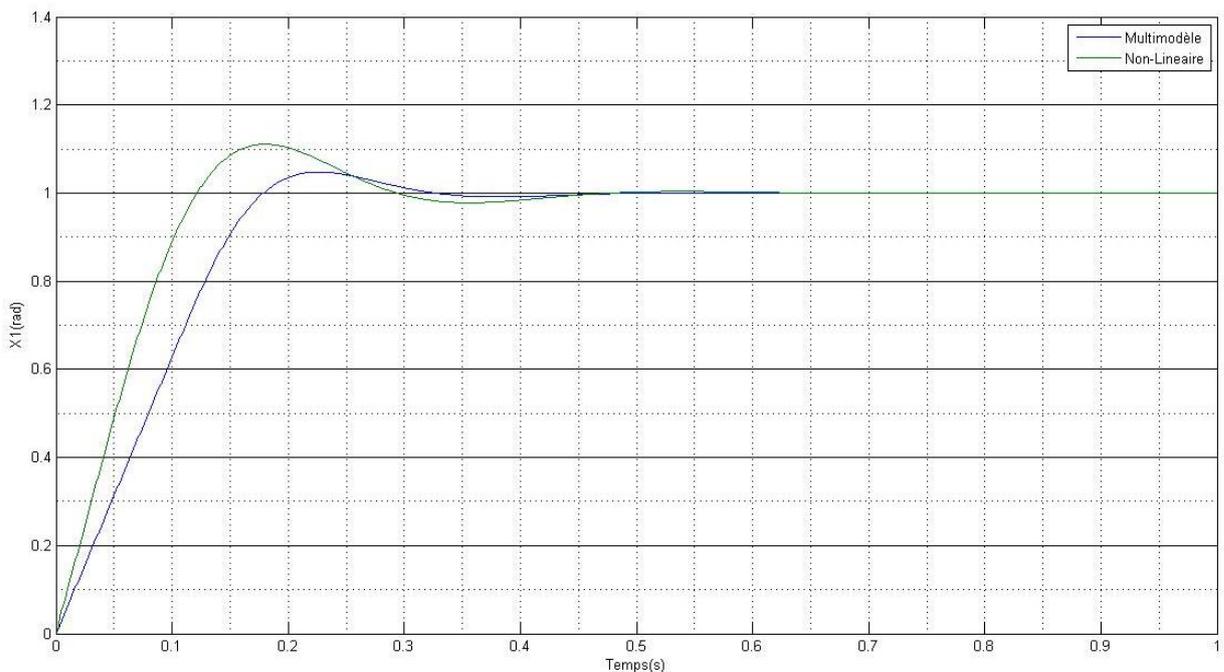


Figure 2. 10 – L'angle q_1 du Multimodèle pour une commande PD

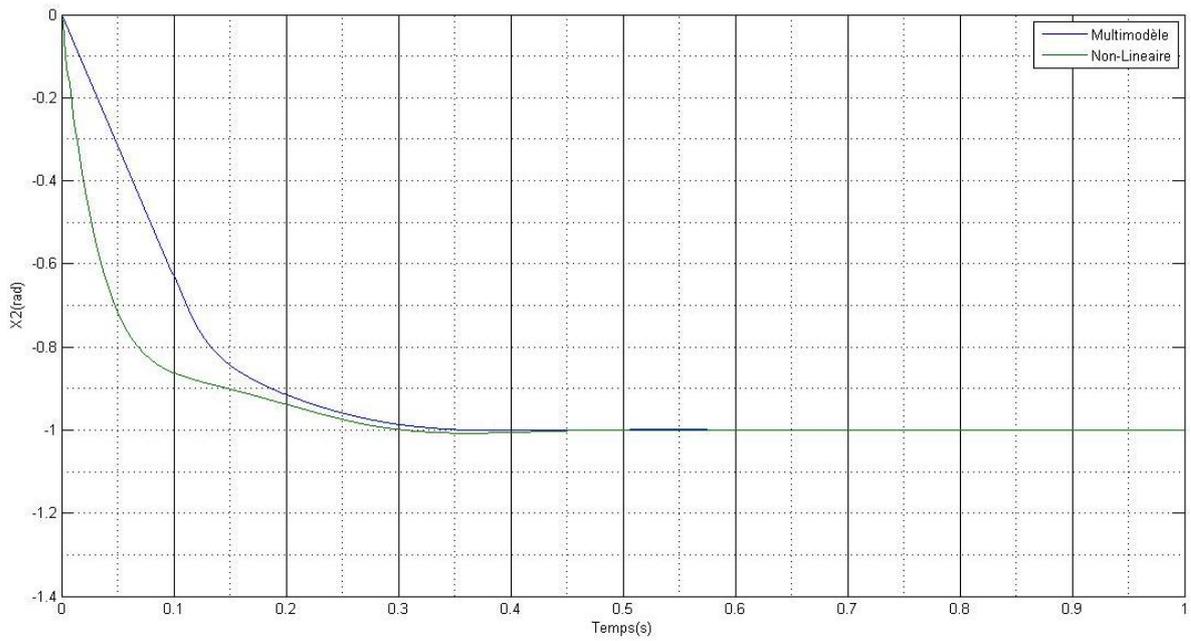


Figure 2. 11 – L'angle q_2 du Multimodèle pour une commande PD

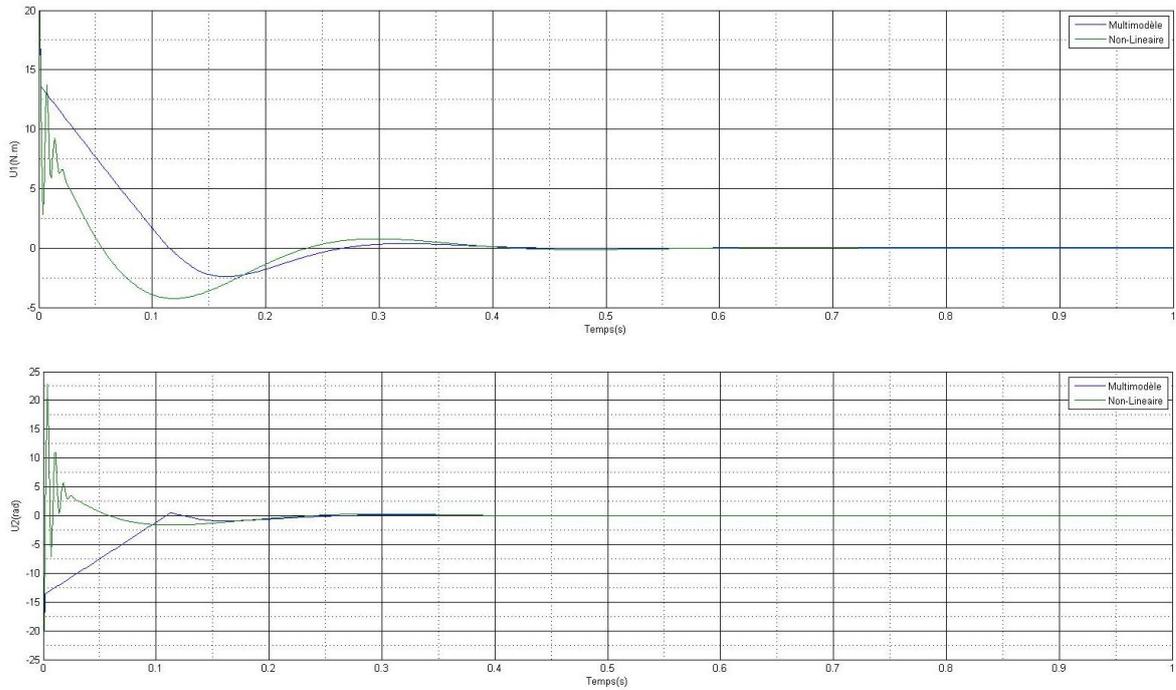


Figure 2. 12 – Signale de commande du Multimodèle pour une commande PD

D'après les résultats de simulation on déduit que la réponse de multimodèle avec 27 sous-espaces flous représente fidèlement le système non linéaire avec un temps de réponse inférieur à 1 seconde.

2.7.4. Commande multimodèle PDC :

Dans le cas de la commande PDC, la loi de commande est sous la forme suivante :

$$u(t) = \frac{\sum_{j=1}^r \mu_j(x(t)) \{-K_j^* x(t) + L_j^* r(t)\}}{\sum_{j=1}^r \mu_j(x(t))}$$

En utilisant (2.25), les gains K_i et L_i sont calculés pour satisfaire les équations algébriques suivantes :

$$A_i - B_i K_i^* = A_m \quad B_i L_i^* = B_m \quad \text{Pour } i = 1, \dots, 27$$

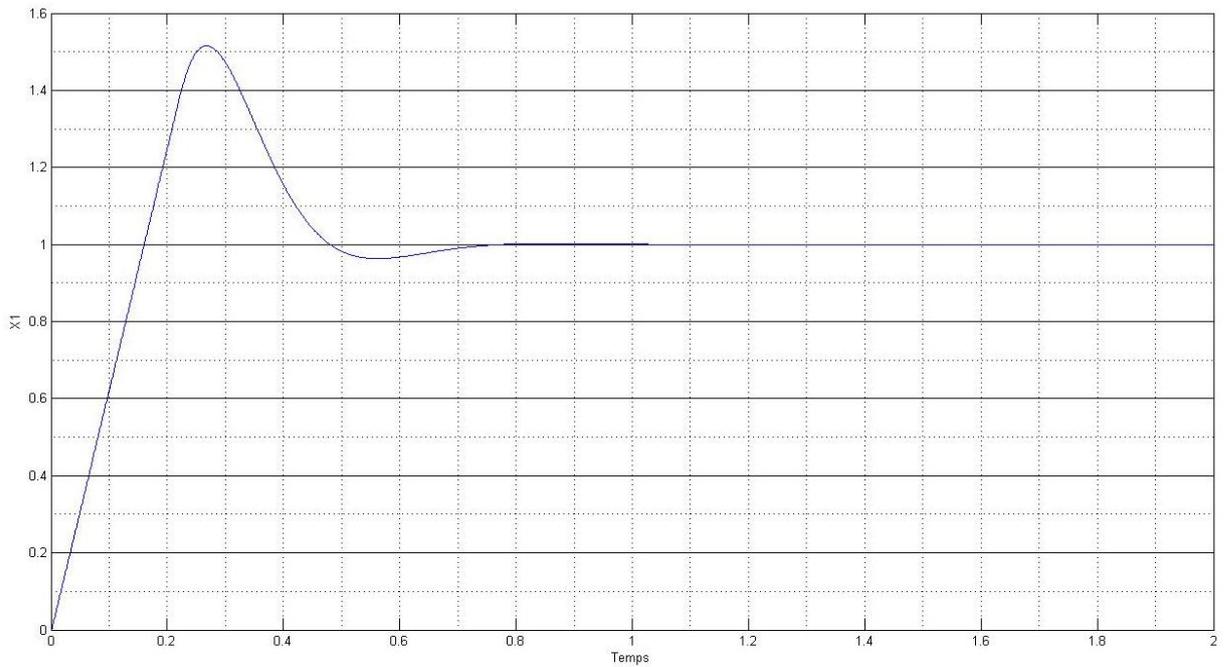


Figure 2. 13 – L'angle q_1 avec commande PDC

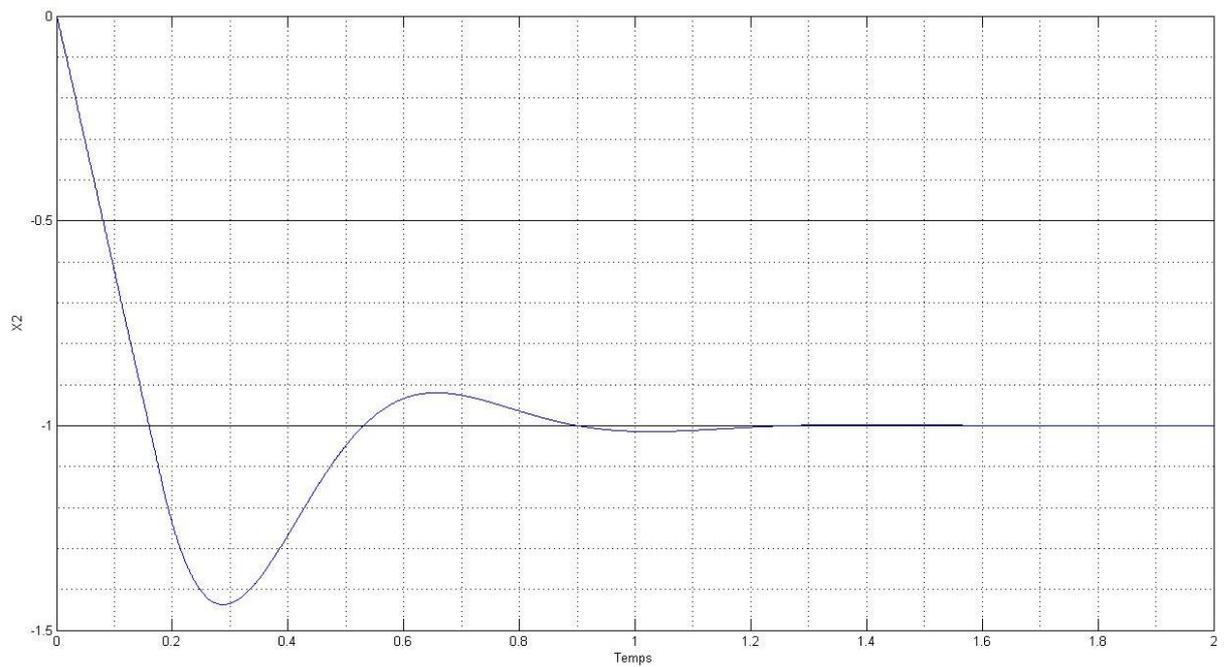


Figure 2. 14 – L'angle q_2 avec commande PDC

Interprétation des résultats de simulation : Les résultats de simulation montre la bonne performance de poursuite de la référence avec une erreur inférieure à 0.1%. Les figures ci-dessus montrent que la commande PDC permet d'assurer la convergence en temps fini, temps de réponse court. D'autre part, elle permet de maintenir de bonnes performances de poursuite.

Conclusion

Ce chapitre a permis de présenter les modèles flous de type T-S et les concepts de base relatifs à la stabilité et la stabilisation des modèles flous de type T-S. Ainsi, des conditions suffisantes de stabilité ont été présentées. Celles-ci se basent sur la seconde méthode de Lyapunov et sont, si possible, écrites sous la forme de LMI afin d'être résolues par les outils classiques de l'optimisation convexe.

Notons que les résultats présentés dans ce chapitre ne tiennent pas compte des incertitudes de modélisation et des perturbations externes lors de la synthèse de la loi de commande. Ce point sera abordé au chapitre suivant par l'élaboration d'une commande adaptative robuste pour la classe des modèles flous T-S incertains et perturbés.

Chapitre 3

Commande adaptative multimodèle floue directe avec modèle de référence

Dans ce chapitre, nous représentons une commande floue adaptative directe avec modèle de référence (MRAFC) pour les systèmes non linéaires MIMO dont la structure est représentée par le modèle flou Takagi-Sugeno. La loi d'adaptation du MRAFC estime les paramètres de commande, de sorte qu'il assure le suivi asymptotique du signal de référence pour les systèmes avec des variations paramétriques. La loi de contrôle développé et la loi d'adaptation garantissent la bornitude de tous les signaux dans le système en boucle fermée. En outre, l'état de système suit l'état du modèle de référence asymptotiquement avec le temps pour tout signal d'entrée de référence bornée. Pour vérifier la validité et l'efficacité de la commande MRAFC, l'analyse représentée et les techniques de conception sont appliquées à la commande de poursuite de robot manipulateur présenté dans le chapitre 1.

3.1. Loi d'adaptation

Considérons de nouveau la loi de commande de chapitre 2, supposons que K_j^* et L_j^* dans (2.23) existe, c'est-à-dire, il y a une flexibilité structurelle suffisante pour atteindre l'objectif de commande alors nous appliquons la loi de commande suivante :

$$u(t) = \frac{\sum_{j=1}^r \mu_j(x(t)) \{-K_j x(t) + L_j r(t)\}}{\sum_{j=1}^r \mu_j(x(t))} \quad (3.1)$$

Où K_j et L_j sont les estimations de K_j^* et L_j^* respectivement, à générer par une loi d'adaptation appropriée.

En ajoutant et en soustrayant le terme d'entrée désirée :

$$u(t) = \frac{\sum_{j=1}^r \mu_j(x(t)) \{-(K_j^* x(t) + L_j^* r(t))\}}{\sum_{j=1}^r \mu_j(x(t))} \quad (3.2)$$

Au model (2.8) et en utilisant (2.25), on obtient :

$$\begin{aligned} \dot{x}(t) &= \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (A_m)_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} x(t) \\ &+ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (B_m)_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} r(t) \\ &+ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) B_i (K_j^* x(t) - L_j^* r(t) + u(t))}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} \end{aligned} \quad (3.3)$$

En outre, en ajoutant et en soustrayant le terme d'entrée estimé multiplié par :

$$\frac{\sum_{i=1}^r \omega_i(x(t)) B_i}{\sum_{i=1}^r \omega_i(x(t))}$$

A savoir

$$\frac{\sum_{i=1}^r \omega_i(x(t)) B_i}{\sum_{i=1}^r \omega_i(x(t))} \left[\frac{\sum_{j=1}^r \mu_j(x(t)) \{-K_j x(t) + L_j r(t)\}}{\sum_{j=1}^r \mu_j(x(t))} \right]$$

Au modèle de référence (2.22), on obtient :

$$\begin{aligned} \dot{x}_m(t) &= \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (A_m)_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} x_m(t) \\ &+ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (B_m)_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} r(t) \\ &+ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) B_i (K_j x(t) - L_j r(t) + u(t))}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} \end{aligned} \quad (3.4)$$

En utilisant le modèle de référence (3.4), nous pouvons exprimer (3.3) en termes de l'erreur de poursuite défini comme $e = x - x_m$, c'est-à-dire :

$$\begin{aligned} \dot{e}(t) &= \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (A_m)_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} e(t) \\ &+ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) B_i (-\tilde{K}_j x(t) + \tilde{L}_j r(t))}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} \end{aligned} \quad (3.5)$$

Avec :

$$\tilde{K}_j = K_j - K_j^* \quad \tilde{L}_j = L_j - L_j^*$$

Dans l'équation dynamique (3.5), les B_i sont inconnus. Nous supposons que L_j^* sont soit définie positive ou négative et on définit $\Gamma_j^{-1} = L_j^* \text{sign}(l_j)$, où $l_j = 1$ si L_j^* est définie positive et $l_j = -1$ si L_j^* est définie négative. Puis $B_i = B_m L_j^{*-1}$, (3.5) devient:

$$\begin{aligned} \dot{e}(t) &= \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (A_m)_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} e(t) \\ &+ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) (B_m)_{ij} L_j^{*-1} (-\tilde{K}_j(t) x(t) + \tilde{L}_j(t) r(t))}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} \end{aligned} \quad (3.6)$$

En utilisant la dynamique de l'erreur de poursuite (3.6), nous pourrions obtenir la loi d'adaptation pour les paramètres de commande K_j et L_j . Nous supposons que la loi d'adaptation a la structure générale [Y. Cho et al., 2007] suivante:

$$\begin{aligned}\dot{K}_j &= F_j(x(t), x_m(t), r(t), (t)) \\ \dot{L}_j &= G_j(x(t), x_m(t), r(t), (t))\end{aligned}\quad (3.7)$$

Où G_j et F_j sont des fonctions de signaux connus qui doivent être choisis de telle sorte que l'équilibre des équations dynamiques (3.6) et (3.7) a des propriétés de stabilité désirées.

$$K_{je} = K_j^* L_{je} = L_j^* e_e = 0 \quad (3.8)$$

Nous proposons la fonction de Lyapunov [Y. Cho et al., 2007] suivante :

$$V(e, \tilde{K}_j, \tilde{L}_j) = e^T P e + \sum_{j=1}^r \text{tr}(\tilde{K}_j^T \Gamma_j \tilde{K}_j + \tilde{L}_j^T \Gamma_j \tilde{L}_j) \quad (3.9)$$

Où $P = P^T > 0$ est une matrice commune définie positive solution de l'équation de Lyapunov $(A_m)^T P + P(A_m)_{ij} < -Q_{ij}$ pour tous $Q_{ij} = Q_{ij}^T > 0$, dont l'existence est garantie par l'hypothèse de stabilité de A_m , et le Théorème 2.3 Après quelques manipulations mathématiques [Y. Cho et al., 2007], nous obtenons :

$$\begin{aligned}\dot{V}(t) &= -e^T \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) Q_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} e(t) \\ &+ 2\text{tr} \left\{ - \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) \tilde{K}_j^T \Gamma_j (B_m)_{ij}^T \text{sgn}(l_j)}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} P e x^T + \sum_{j=1}^r \tilde{K}_j^T \Gamma_j \dot{\tilde{K}}_j \right\} \\ &+ 2\text{tr} \left\{ \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) \tilde{L}_j^T \Gamma_j (B_m)_{ij}^T \text{sgn}(l_j)}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} P e r^T + \sum_{j=1}^r \tilde{L}_j^T \Gamma_j \dot{\tilde{L}}_j \right\}\end{aligned}\quad (3.10)$$

Dans les deux derniers termes de (3.10), si on met :

$$\sum_{j=1}^r \tilde{K}_j^T \Gamma_j \dot{\tilde{K}}_j = \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) \tilde{K}_j^T \Gamma_j (B_m)_{ij}^T \text{sgn}(l_j)}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} P e x^T \quad (3.11)$$

$$\sum_{j=1}^r \tilde{L}_j^T \Gamma_j \dot{\tilde{L}}_j = - \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) \tilde{L}_j^T \Gamma_j (B_m)_{ij}^T \text{sgn}(l_j)}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} P e r^T \quad (3.12)$$

Pour que \dot{V} soit définie négative, c'est-à-dire :

$$\dot{V}(t) = -e^T \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) Q_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} e(t) \quad (3.13)$$

Alors le choix le plus simple pour la loi d'adaptation pour que \dot{V} soit définie négative est :

$$\dot{\tilde{K}}_j = \dot{K}_j(t) = \left(\frac{\sum_{i=1}^r \omega_i(x) (B_m)_{ij}^T}{\sum_{i=1}^r \omega_i(x)} \right) \left(\frac{\mu_j}{\sum_{j=1}^r \mu_j(x)} \right) \text{sgn}(l_j) P e x^T \quad (3.14a)$$

$$\dot{\tilde{L}}_j = \dot{L}_j(t) = - \left(\frac{\sum_{i=1}^r \omega_i(x) (B_m)_{ij}^T}{\sum_{i=1}^r \omega_i(x)} \right) \left(\frac{\mu_j}{\sum_{j=1}^r \mu_j(x)} \right) \text{sgn}(l_j) P e r^T \quad (3.14b)$$

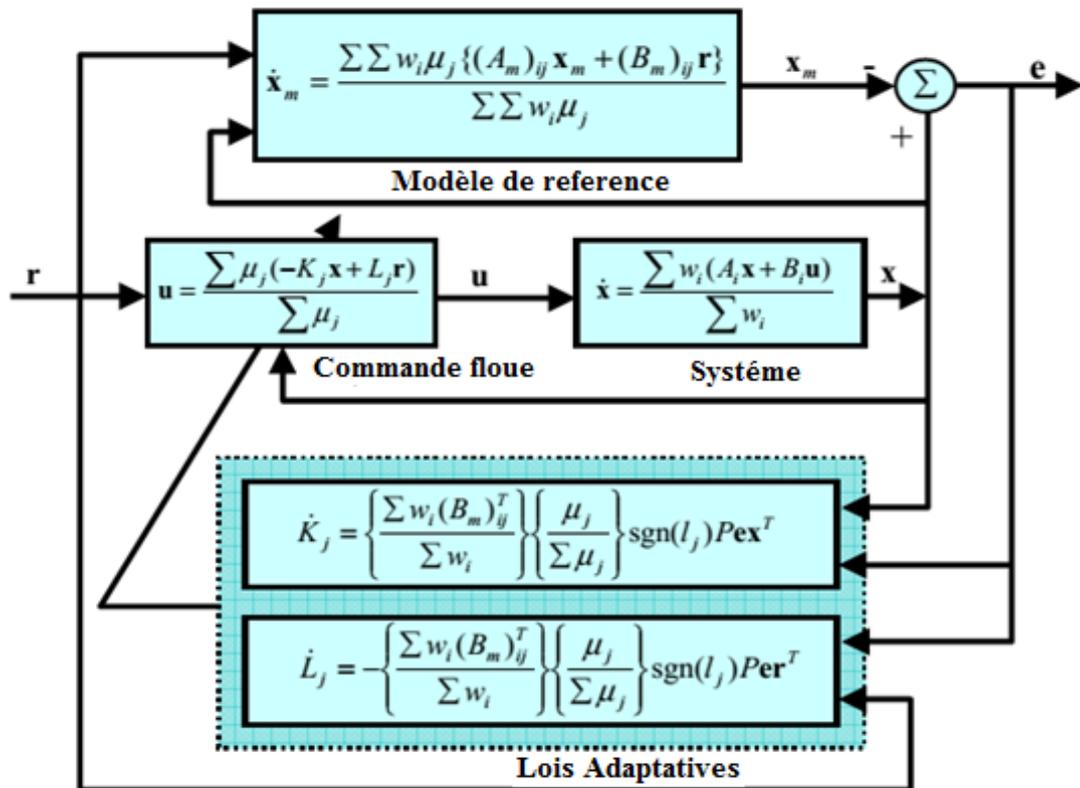


Figure 3. 1 – Schéma bloc de la commande adaptative directe MRAFC

3.2. Stabilité de la commande MRAFC :

Toutes les quantités dans le côté droit de (3.14a) et (3.14b) sont connus ou mesurable. Par conséquent la loi d'adaptation (3.14a) peut être mise en œuvre pour la commande MRAFC du système flou TS. La figure 3.1 illustre la configuration du système de commande MRAFC. Le modèle de référence est utilisé pour spécifier la réponse idéale que le système de contrôle flou devrait suivre. Le système est supposé contenir des paramètres inconnus, mais sa structure est connue. Le contrôleur flou est conçu à partir des systèmes flous dont les paramètres sont réglables. La loi d'adaptation ajuste les paramètres de contrôle K et L tels que l'état X de système non linéaire suit l'état de modèle de référence X_m .

En utilisant des arguments évoqués précédemment. Nous établissons le théorème qui montre les propriétés de la commande MRAFC. La loi de commande (3.1) ainsi que la loi d'adaptation (3.14a) garantit la bornitude pour tous les signaux dans le système en boucle fermée. En outre, l'état X de système non linéaire suit l'état du modèle de référence X_m asymptotiquement avec le temps pour toute entrée borné de signal de référence r.

Théorème 3.1 (Stabilité de la commande MRAFC) :

Considérons le modèle de système (2.8) et le modèle de référence (2.22) avec la loi de commande (3.1) et la loi d'adaptation (3.14a). Supposons que l'entrée r et l'état du modèle de référence X_m sont uniformément bornées. Alors, la loi de commande (3.1) et la loi d'adaptation (3.14a) garantie que :

- (i) $K(t), L(t), e(t)$ sont bornés
- (ii) $e(t) \rightarrow 0$ quand $t \rightarrow \infty$

Preuve :

A partir de (3.9) et (3.13), il découle directement que V est une fonction de Lyapunov pour le système (3.6) et (3.7), ce qui implique que l'équilibre donné par (3.8) est uniformément stable, ce qui, à son tour, implique que les trajectoires $\tilde{K}(t), \tilde{L}(t), e(t)$ sont bornées pour tout $t > 0$. Comme $e = x - x_m$ et $x_m \in L_\infty$, alors $x \in L_\infty$. À partir de (3.1) et $r \in L_\infty$, nous avons aussi $u \in L_\infty$; Par conséquent, tous les signaux dans le système en boucle fermée sont bornés. Maintenant, pour montrer que $e \in L_2$. A partir de (3.9) et (3.13), nous concluons que V est bornée et non croissante avec le temps, elle a une limite, à savoir :

$$\lim_{t \rightarrow \infty} V(e(t), \tilde{K}_j(t), \tilde{L}_j(t)) = V_\infty < \infty \quad (3.15)$$

A partir de (3.13) et (3.15), il en résulte que :

$$\int_0^{\infty} e^T \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) Q_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} e(t) d\tau = - \int_0^{\infty} \dot{V} d\tau = V_0 - V_{\infty} \quad (3.16)$$

Où $V_0 = V(e(0), \tilde{K}_j(0), \tilde{L}_j(0))$

D'autre part, de $0 \leq \omega_i(x(t)) \leq 1, 0 \leq \mu_j(x(t)) \leq 1$ et $\lambda_{\min}(Q_{ij}) \|e\|^2 \leq e^T Q_{ij} e \leq \lambda_{\max}(Q_{ij}) \|e\|^2$ nous avons :

$$\begin{aligned} (\lambda_{\min}(Q_{ij}))_{\min} \|e\|^2 &\leq e^T \frac{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t)) Q_{ij}}{\sum_{i=1}^r \sum_{j=1}^r \omega_i(x(t)) \mu_j(x(t))} e \\ &\leq (\lambda_{\max}(Q_{ij}))_{\max} \|e\|^2 \end{aligned} \quad (3.17)$$

Où :

$$\begin{aligned} (\lambda_{\min}(Q_{ij}))_{\min} &= \min\{\lambda_{\min}(Q_{11}), \dots, \lambda_{\min}(Q_{rr})\} \\ (\lambda_{\max}(Q_{ij}))_{\max} &= \min\{\lambda_{\max}(Q_{11}), \dots, \lambda_{\max}(Q_{rr})\} \end{aligned}$$

En utilisant (3.17) et (3.16), et une manipulation mathématique simple, nous avons :

$$\frac{V_0 - V_{\infty}}{(\lambda_{\min}(Q_{ij}))_{\min}} \leq \int_0^{\infty} \|e\|^2 d\tau \leq \frac{V_0 - V_{\infty}}{(\lambda_{\max}(Q_{ij}))_{\max}}$$

Ce qui implique que $e \in L_2$ car $\dot{e}, K_j, L_j, r \in L_{\infty}$, et il résulte de (3.5) que $e \in L_{\infty}$, qui conjointement avec $e \in L_2$, implique que $e(t) \rightarrow 0$ quand $t \rightarrow \infty$ [Y. Cho et al., 2007].

3.3. Application et analyse :

Dans cette section, la validité et l'efficacité de la commande MRAFC représenté sont examinés par la simulation de poursuite de trajectoire pour le manipulateur de robot présenté dans le chapitre 1.

L'objectif de la commande adaptatif de poursuite d'un robot manipulateur est de donner une loi de commande adaptative pour le couple de l'actionneur u afin de rendre la trajectoire du robot manipulateur comportant des incertitudes paramétrique plus proche des trajectoires désirées $q_d(t)$ avec la précision et la stabilité souhaitée.

Le gain de commande par retour d'états K_i et les L_i de chaque contrôleur sont mis à jour par la loi d'adaptation (3.14a) pour que le système en boucle fermée suive le modèle de référence.

Les valeurs initiales de K_i et L_i sont calculées à partir des paramètres nominaux de systèmes (tableau 1.1). Dans cette simulation, nous concevons les paramètres initiaux du K_i et L_i de sorte que le système en boucle fermée et y compris la commande par retour d'états a les mêmes valeurs propres que le modèle de référence, et les sorties et les entrées de système en boucle fermée seront découplés.

Maintenant, en utilisant (3.14a), nous avons la loi d'adaptation pour mettre à jour les éléments de K_i et L_i pour que le système en boucle fermée suit le modèle de référence.

$$\dot{K}_j(t) = \frac{\mu_j}{\sum_{j=1}^r \mu_j(x)} \text{sgn}(l_j) B_m^T P e x^T$$

$$\dot{L}_j(t) = -\frac{\mu_j}{\sum_{j=1}^r \mu_j(x)} \text{sgn}(l_j) B_m^T P e r^T$$

Ou

$$B_m^T = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P = \begin{bmatrix} 2.1957 & 0 & 0.0059 & 0 \\ 0 & 1.6520 & 0 & 0.0059 \\ 0.0059 & 0 & 0.0014 & 0 \\ 0 & 0.0059 & 0 & 0.0020 \end{bmatrix}$$

Simulation :

Dans cette section, on propose d'illustrer l'efficacité des lois d'adaptation dans ce chapitre. Pour évaluer la robustesse de notre commande, les simulations ont été effectuées pour le système présenté dans le chapitre 1, ayant des variations paramétriques. Les paramètres du système sont variés à 2 secondes après le démarrage de la simulation.

La précision et la robustesse de notre commande a été étudiée avec deux types de trajectoire; poursuite de trajectoire linéaire et poursuite de trajectoire circulaire. Le manipulateur est simulé pour suivre la trajectoire linéaire générée à la position initiale (1.9, 0.5) et d'atteindre à la position finale (0.1, -1.3) - au bout de 4 secondes, la vitesse du mouvement linéaire souhaitée est (0,64 m/s). La vitesse angulaire du mouvement circulaire souhaitée est $\frac{\pi}{2} \text{rad/s}$. Toutes ces simulations ont été réalisées en utilisant MATLAB avec un intervalle d'échantillonnage de 1ms.

Nous avons effectué des simulations pour étudier les effets de petites variations paramétriques sur les réponses du système, les paramètres du système sont modifiés à 2 secondes après le

démarrage de la simulation, les paramètres initiaux sont les paramètres nominaux de système (tableau 1.1) de chapitre 1, et à partir de 2 secondes:

$$M1=3Kg$$

$$M2=6Kg$$

Les figures ci-dessous montrent l'évolution temporelle du vecteur de sortie, de l'erreur de sortie et de la poursuite de trajectoire linéaire par le contrôleur conçu dans la première partie de ce chapitre.

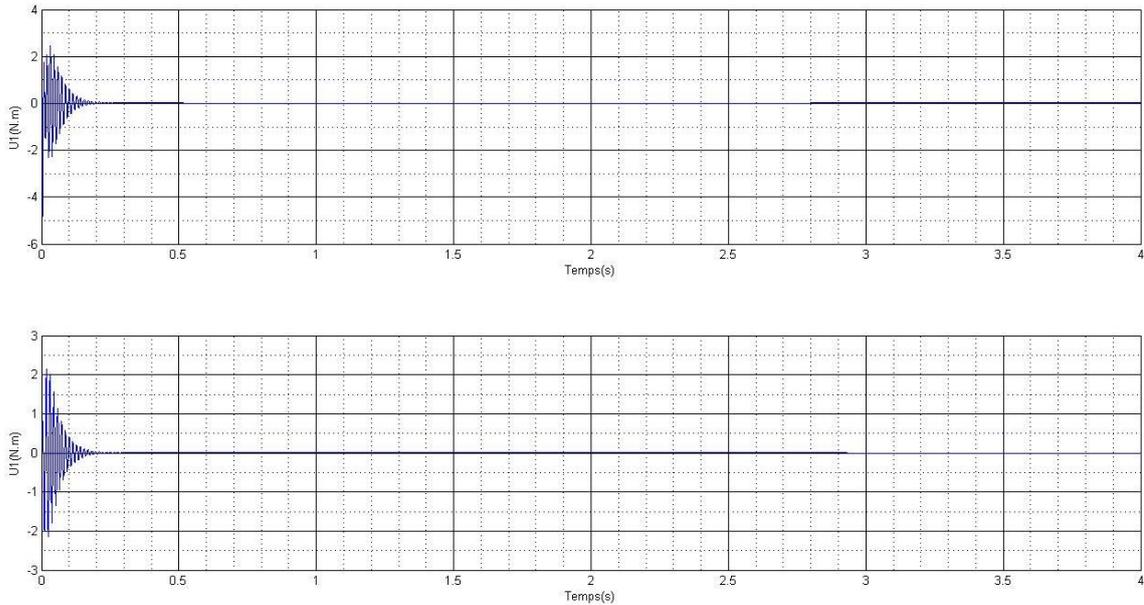


Figure 3. 2 – Commande du système pour une trajectoire linéaire dans l'espace cartésien

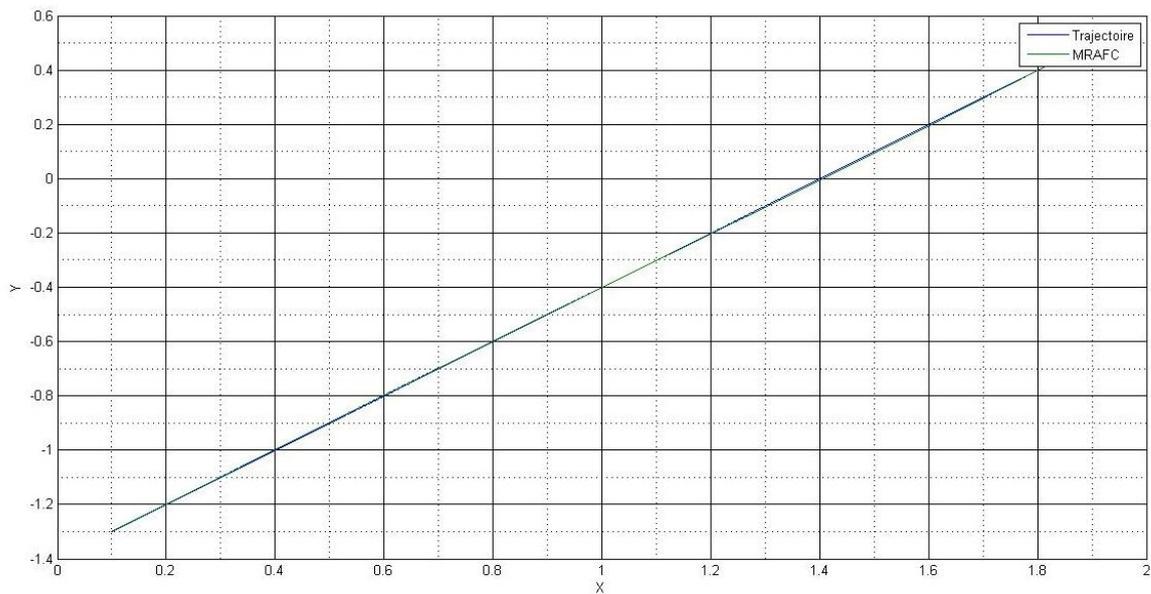


Figure 3. 3 – Réponses de système sur une trajectoire linéaire dans l'espace cartésien

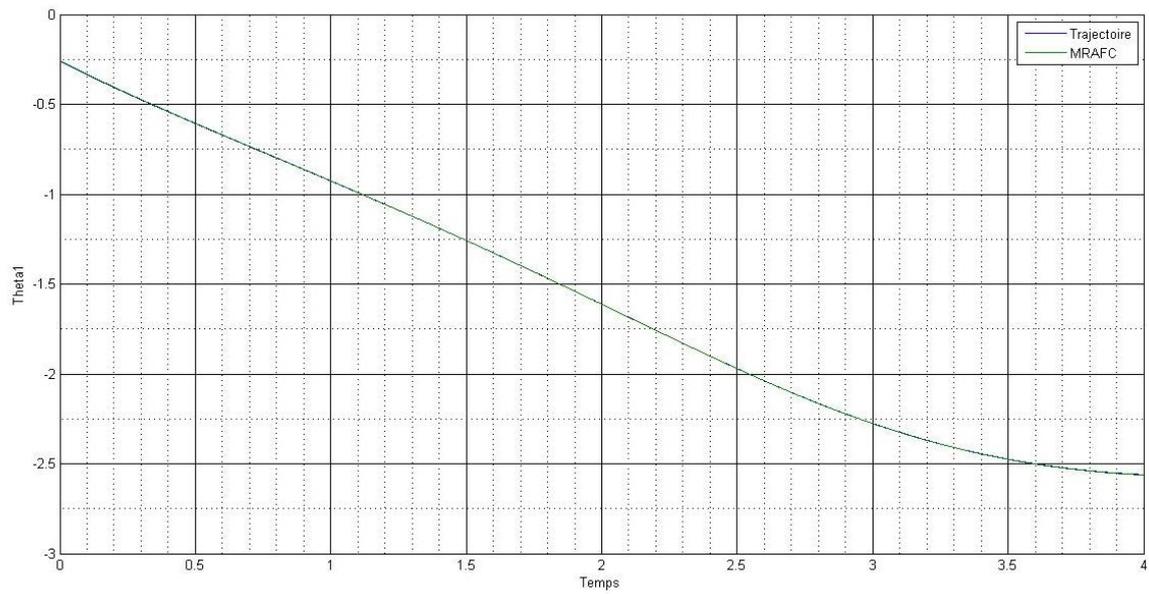


Figure 3. 4 – L'angle q_1 d'une trajectoire linéaire

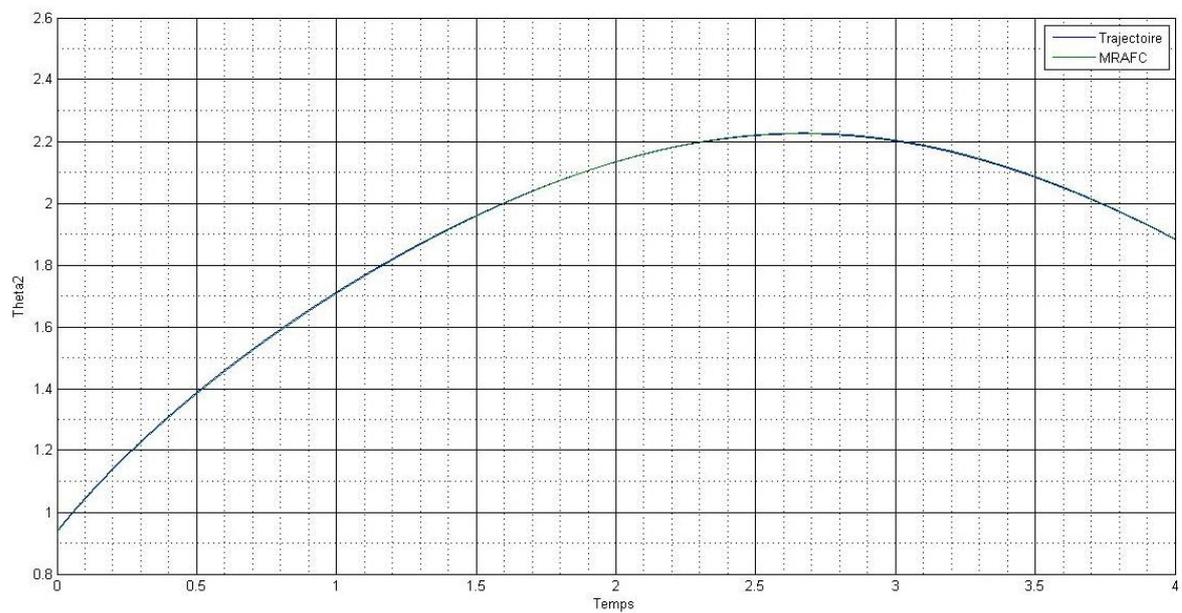


Figure 3. 5 – L'angle q_2 d'une trajectoire linéaire

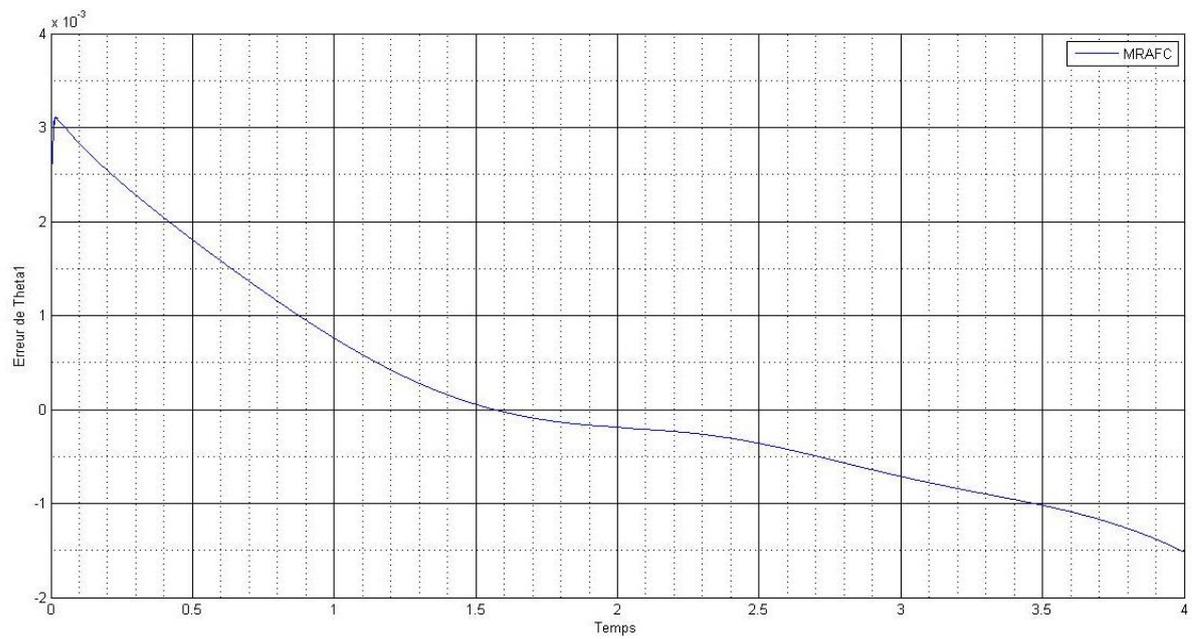


Figure 3. 6 – L'erreur de l'angle q_1 sur une trajectoire

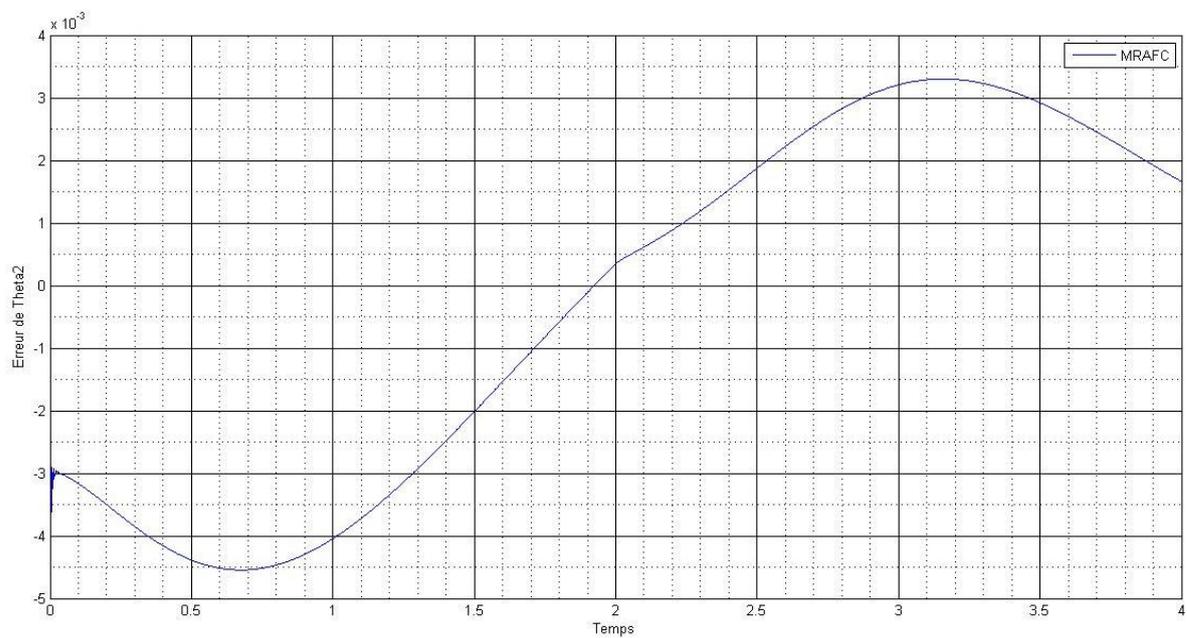


Figure 3. 7 – L'erreur de l'angle q_2 sur une trajectoire

Les figures ci-dessous montrent l'évolution temporelle du vecteur de sortie, de l'erreur de sortie et de la poursuite de trajectoire circulaire par le contrôleur conçu dans la première partie de ce chapitre.

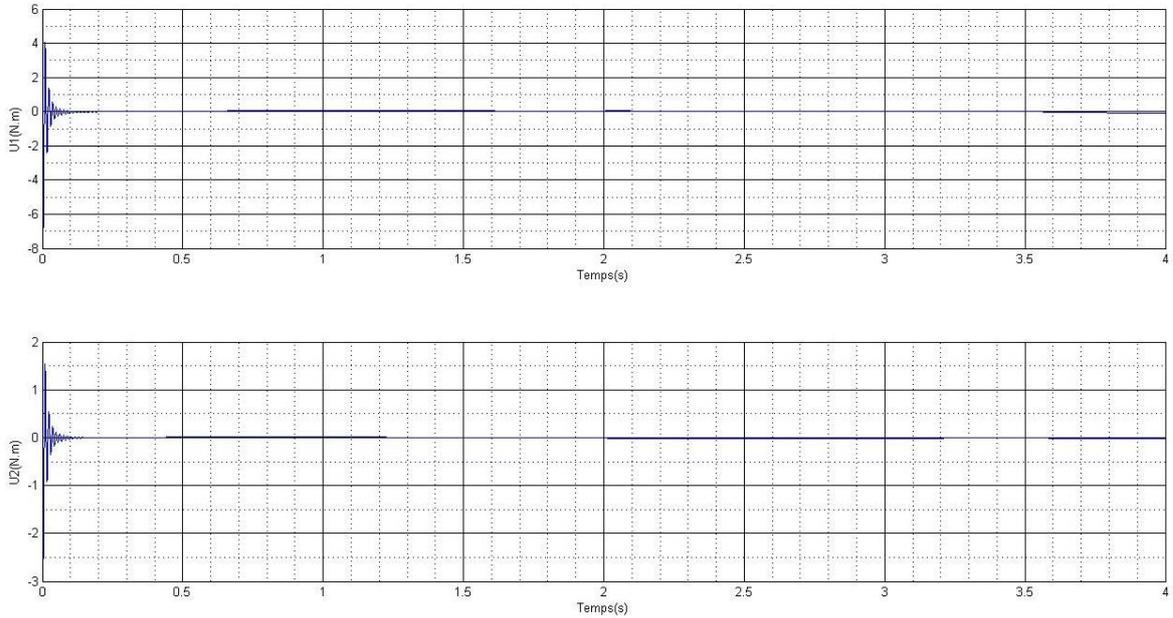


Figure 3. 8 – Commande du système pour une trajectoire circulaire

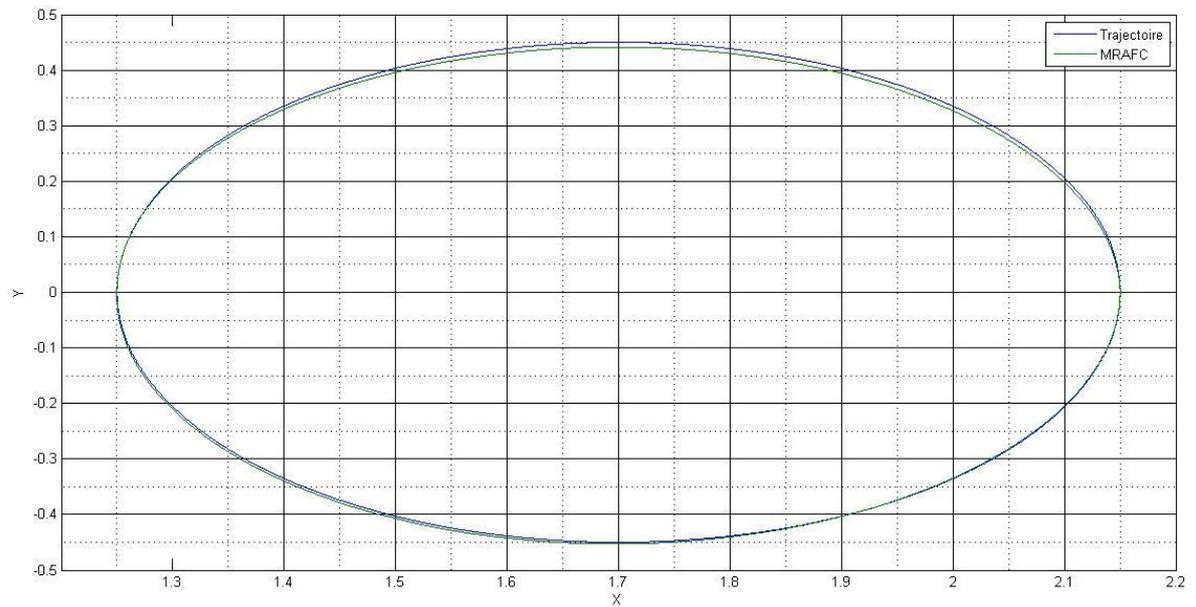


Figure 3. 9 – Réponses de système sur une trajectoire circulaire

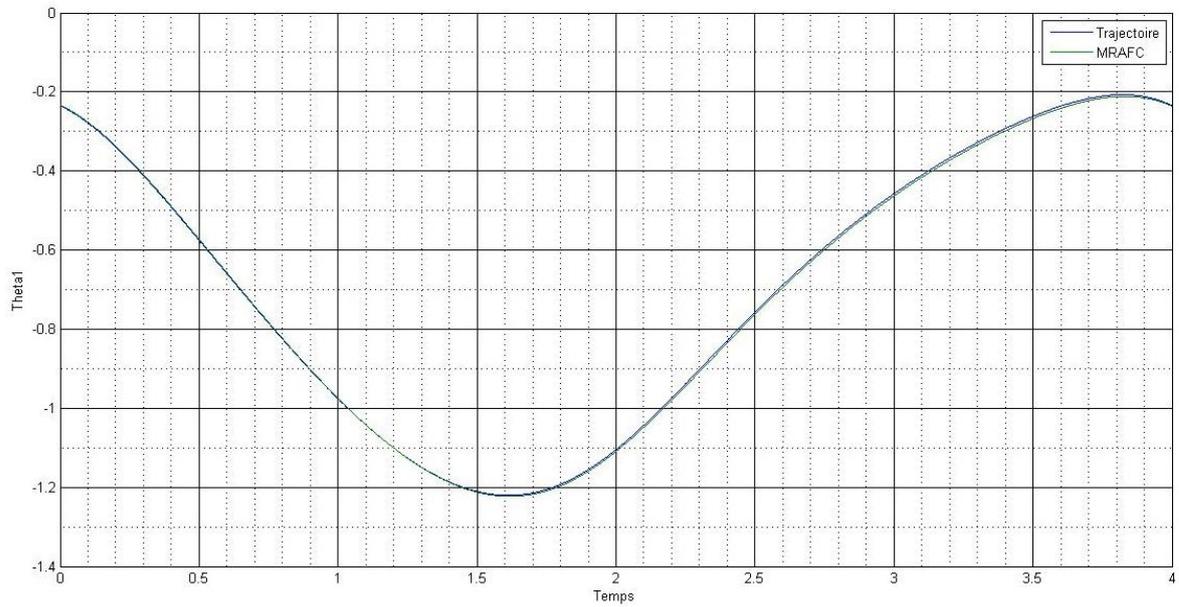


Figure 3. 10 – L'angle q_1 d'une trajectoire

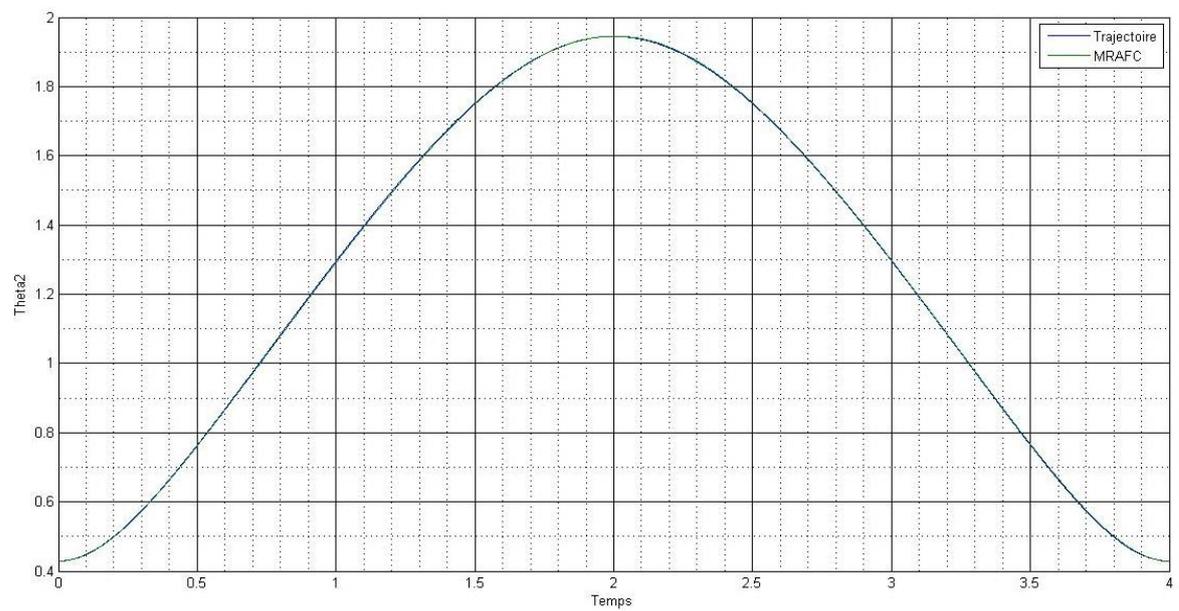


Figure 3. 11 – L'angle q_2 d'une trajectoire

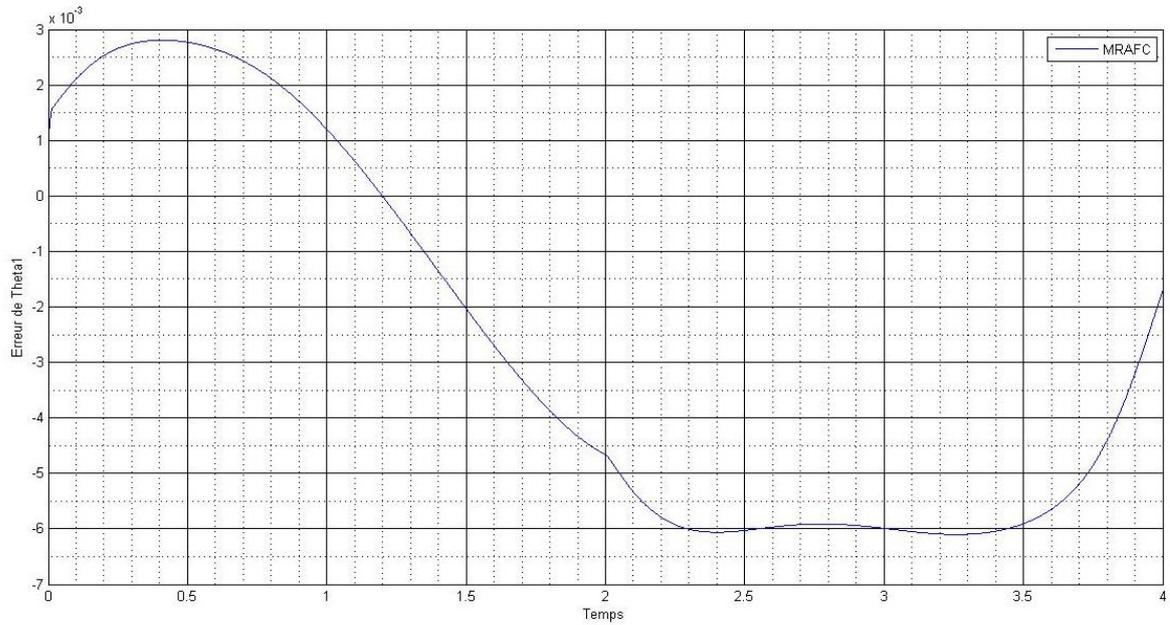


Figure 3. 12 – L’erreur de l’angle q_1 sur une trajectoire circulaire

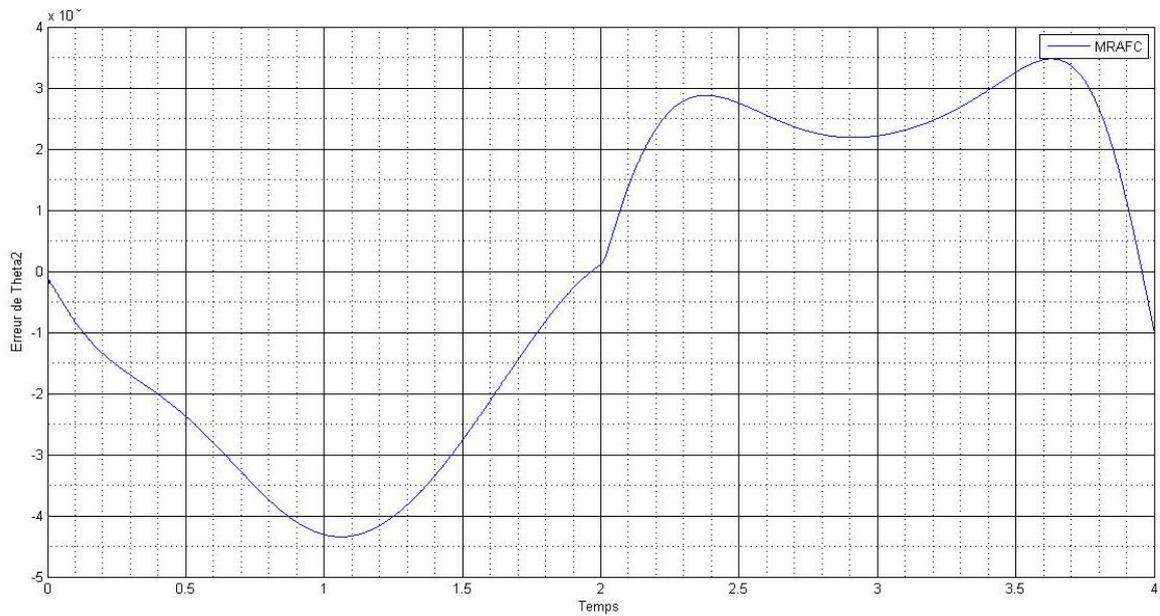


Figure 3. 13 – L’erreur de l’angle q_2 sur une trajectoire circulaire

Interprétation des résultats de simulation : Les résultats de simulation montre la dynamique de rejet de perturbation et de poursuite de la référence avec une erreur inférieure à 0.1%. Les figures ci-dessus montrent que la commande MRAFC permet d’assurer la convergence en temps fini. D’autre part, elle permet de maintenir de bonnes performances de poursuite.

Conclusion :

Dans ce chapitre, nous avons développé une commande adaptatif direct avec modèle de référence (MRAFC) pour les systèmes MIMO de Takagi-Sugeno dont la structure est supposée connue, mais les paramètres inconnus. La loi d'adaptation du MRAFC ajuste les paramètres du contrôleur en ligne de telle sorte que la sortie de système suit la sortie du modèle de référence. La loi d'adaptation développée garantit le bornitude de tous les signaux dans le système en boucle fermée et assure que l'état de système suit l'état du modèle de référence asymptotiquement avec le temps pour tout signal d'entrée de référence bornée. Dans l'ensemble du processus de conception, pas de contraintes strictes et une connaissance préalable de système contrôlée sont nécessaires, et la stabilité asymptotique du système de contrôle peut être garantie. Pour vérifier la validité et l'efficacité du régime MRAFC, les techniques d'analyse et de conception représentées ont été appliquées à la commande de suivi de robots manipulateur. Dans la conception de contrôle, la commande MRAFC a été combinée avec le contrôleur PD et utilisé comme stabilisateur de l'état de modèle de référence. A partir des résultats de simulation, nous concluons que le système de contrôle représenté peut effectivement atteindre le suivi de trajectoire, même pour le manipulateur de robot avec quantité relativement importante d'incertitudes paramétriques.

Chapitre 4

Etude comparative des performances de la commande MRAFC avec d'autres méthodes de commande

L'objectif de ce chapitre est d'évaluer les performances de la commande MRAFC, le contrôleur adaptatif associé à un contrôleur PID est simulé pour la poursuite d'une trajectoire du bras manipulateur, et les résultats sont discutés dans ce chapitre, du point de vue comparatif, trois contrôleurs ont été simulés pour la poursuite d'une trajectoire du bras manipulateur à deux degrés de liberté présentée dans le premier chapitre. Le premier est un contrôleur PD classique; le second est un contrôleur adaptatif glissant [Y Cho et al., 2007]; et le troisième est notre contrôleur adaptatif direct avec modèle de référence conçu dans le chapitre précédent. Ainsi, nous rappelons dans un premier temps très brièvement les schémas ainsi que les valeurs des paramètres des lois PD et Sliding mode.

Ensuite, nous présentons les performances de poursuite des trois contrôleurs, ainsi la démarche adoptée pour la simulation des différents cas proposées, Finalement, ces trois contrôleurs ont été simulés pour évaluer leurs robustesse avec des différents cas de variations paramétriques et trajectoires.

4.1. Présentation des différents contrôleurs :

Le contrôleur proportionnel dérivé (PD) proposé dans le chapitre 1, est utilisé comme premier schéma comparatif avec la forme générale suivante:

$$\tau = -K_p.e - K_d.\dot{e} \quad (4.1)$$

Où les gains K_p et K_d ont les valeurs suivantes : $K_d = 100I$ et $K_p=20*K_d$.

Le deuxième contrôleur est la commande adaptative glissante avec une estimation des paramètres inconnus ou variables dans le temps. La commande glissante [Y Cho et al., 2007] prend la loi suivante :

$$\tau = Y.\hat{a} - K_d.s \quad (4.2)$$

Qui comprend le terme $Y.\hat{a}$ et un simple retour PD $K_d.s$, où :

$$S = \dot{e} + \Lambda e \quad (4.3)$$

Le paramètre estimé \hat{a} est actualisé par la loi d'adaptation suivante :

$$\dot{\hat{a}} = -\Gamma Y^T s \quad (4.4)$$

Les composants de la matrice Y sont explicitement écrits comme suite :

$$Y_{11} = \ddot{q}_{r1} \quad (4.5a)$$

$$Y_{12} = \ddot{q}_{r1} + \ddot{q}_{r2} \quad (4.5b)$$

$$Y_{13} = (2\ddot{q}_{r1} + \ddot{q}_{r2})\cos(q_2) - (\dot{q}_2\dot{q}_{r1} + (\dot{q}_1 + \dot{q}_2)\dot{q}_d)\sin(q_2) \quad (4.5c)$$

$$Y_{21} = 0 \quad (4.5d)$$

$$Y_{22} = \ddot{q}_{r1} + \ddot{q}_{r2} \quad (4.5e)$$

$$Y_{23} = \ddot{q}_{r1}\cos(q_2) + \dot{q}_1\dot{q}_{r1}\sin(q_2) \quad (4.5f)$$

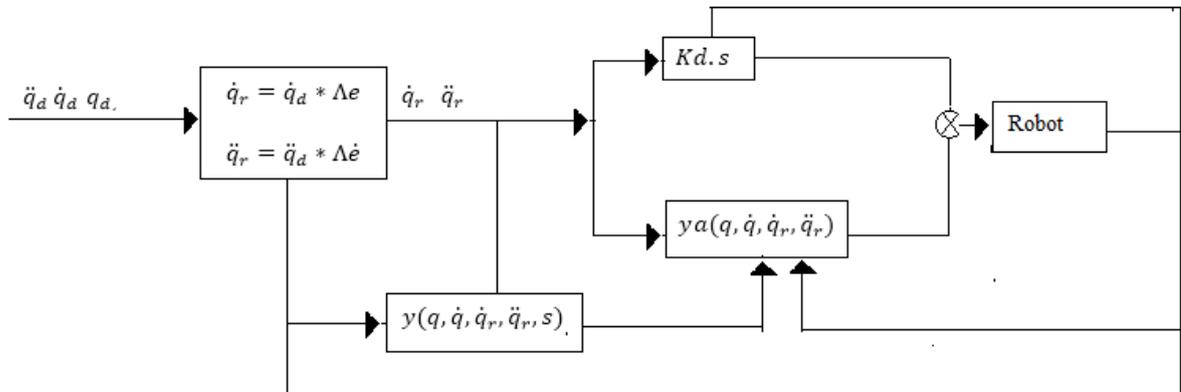


Figure 4. 1– Structure de la commande adaptative glissante [Su et al, 1993]

[Slotine et Li 1991] [Slotine et Sastru 1983] ont utilisé les matrices de gain et d'adaptation suivante :

$$A = 20I \quad Kd = 100I$$

4.2. Etude comparative sur une trajectoire sinusoïdale :

Dans un premier temps, l'objectif est d'illustrer les bonnes performances de poursuite de la commande MRAFC par rapport aux autres contrôleurs.

La simulation montre le comportement du système sur une poursuite d'une trajectoire sinusoïdale de la forme suivante :

$$Xd(1) = \left(\frac{\pi}{2}\right) \sin(t\pi) \quad (4.6)$$

$$Xd(2) = \left(\frac{\pi}{2}\right) \sin(t\pi) \quad (4.7)$$

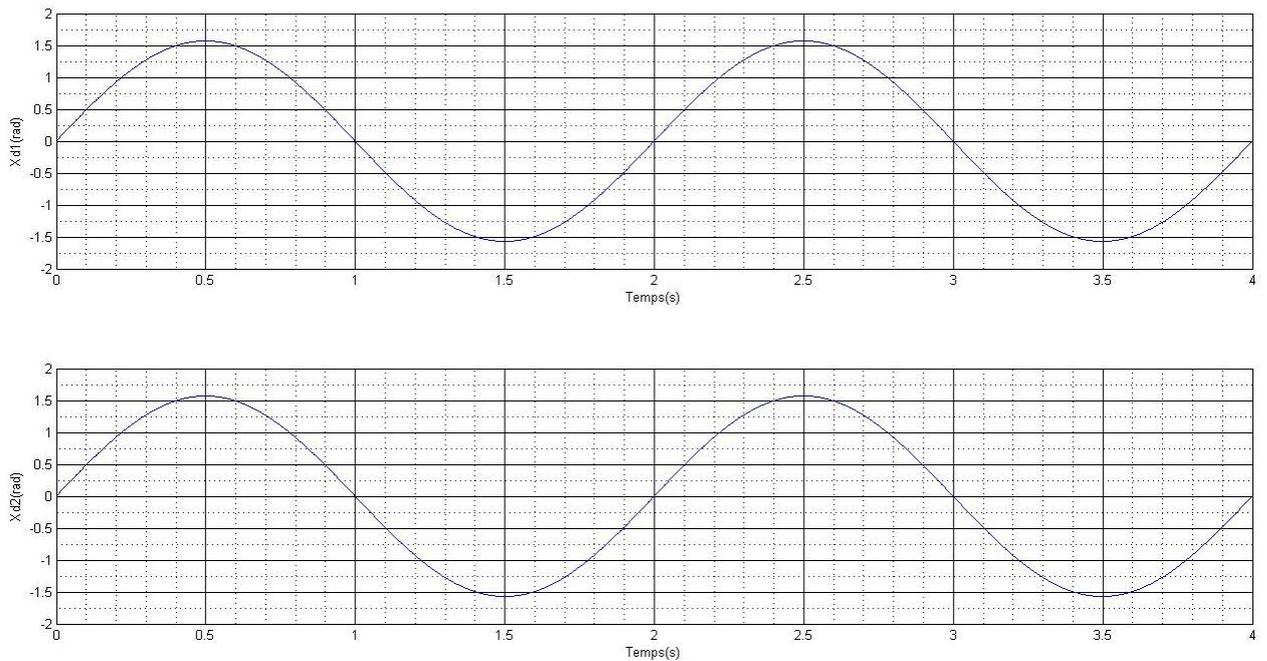


Figure 4. 2 – Trajectoire sinusoïdale

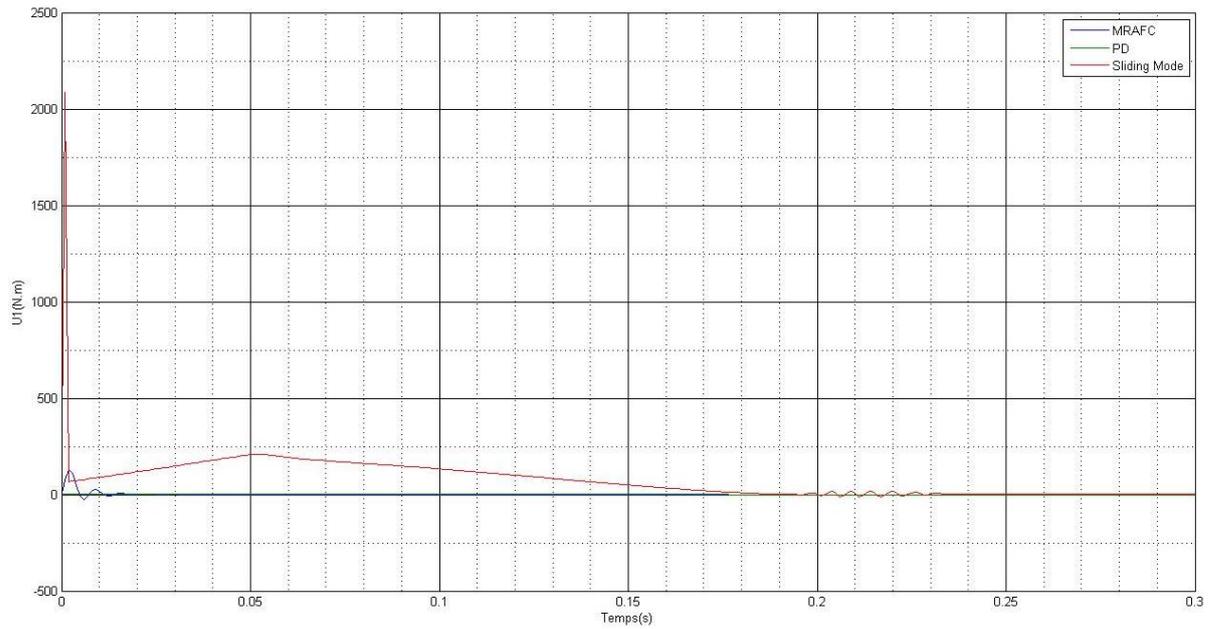


Figure 4. 3 – Signale de commande pour trajectoire sinusoïdale

Les figures ci-dessous montrent l'évolution temporelle du vecteur de sortie, de l'erreur de sortie et de la poursuite de trajectoire sinusoïdale par les trois contrôleurs cités dans la première partie de ce chapitre.

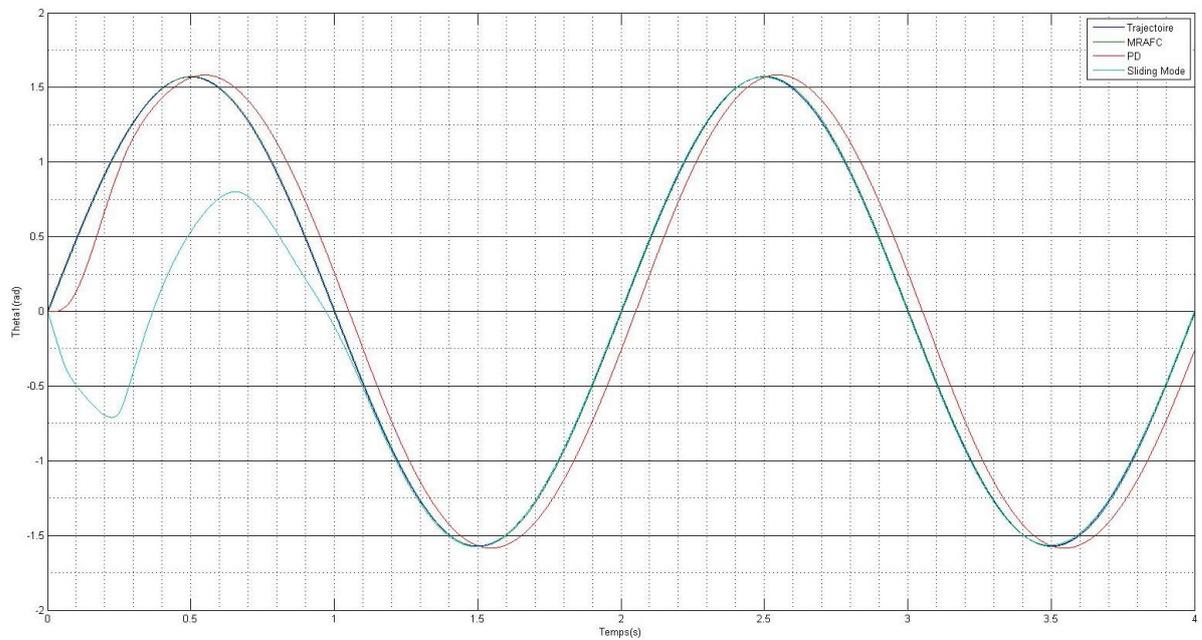


Figure 4. 4 – L'angle q_1 d'une trajectoire sinusoïdale

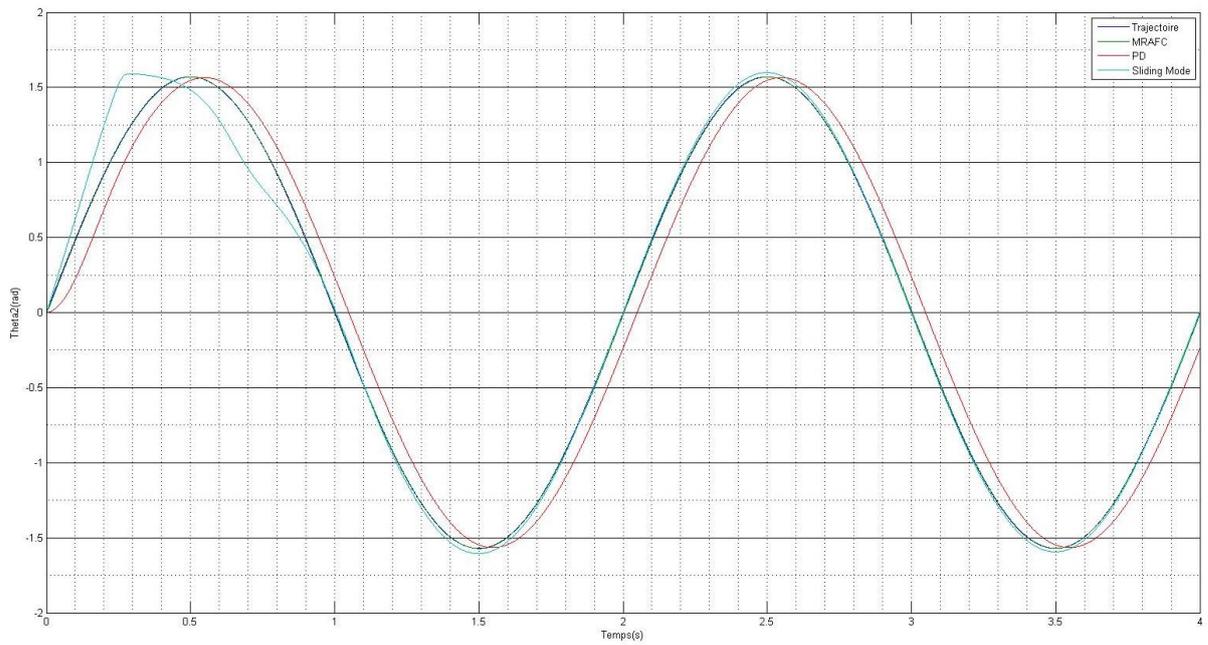


Figure 4. 5 – L'angle q_2 d'une trajectoire sinusoïdale

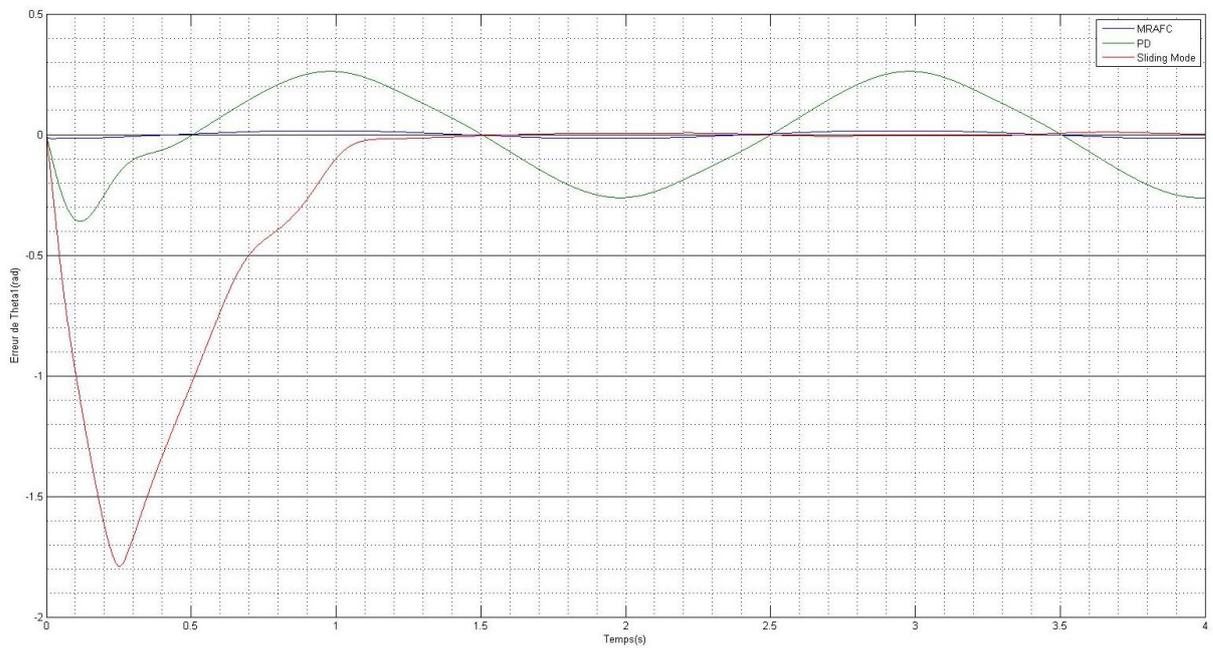


Figure 4. 6 – L'erreur de l'angle q_1 d'une trajectoire sinusoïdal avec des petites variations paramétriques

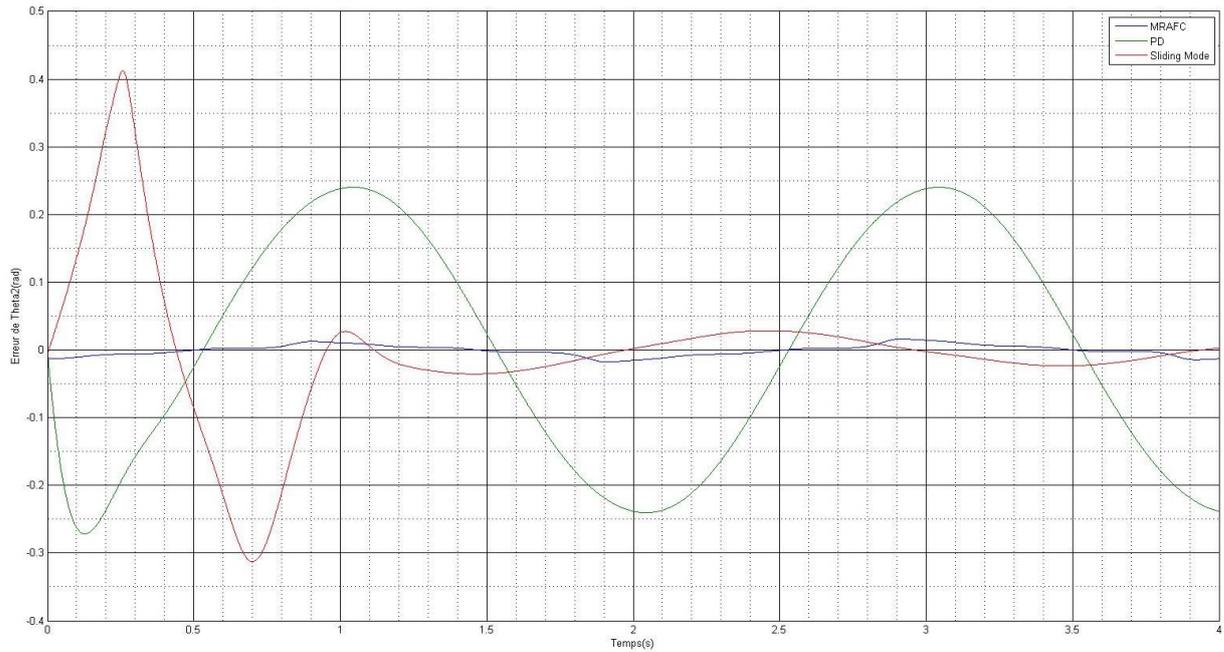


Figure 4. 7 – L’erreur de l’angle q_2 d’une trajectoire sinusoidal avec des petites variations paramétriques

Interprétation des résultats de simulation : Les résultats de simulation montre la dynamique de rejet de perturbation et de poursuite de la référence avec une erreur inférieure à 0.1%. Les figures ci-dessus montrent que la commande MRAFC permet d’assurer la convergence en temps fini, temps de réponse court. D’autre part, elle permet de maintenir de bonnes performances de poursuite.

Il est clair que la commande adaptative floue est bien meilleure que les autres techniques. Elle permet d’obtenir une convergence rapide asymptotique et une meilleure précision. De plus, il n’exige pas un modèle mathématique précis du système sous cette commande.

4.3. Etude comparative avec plusieurs cas de variations paramétriques :

Pour évaluer la robustesse de notre contrôleur, les simulations ont été effectuées pour le même système ayant des variations paramétriques. Le premier cas est le cas où les paramètres du système sont légèrement modifiés à 2 secondes après le démarrage de la simulation. Les paramètres de deuxième cas simulé avec de grandes variations en même temps que le premier cas. Les paramètres de système utilisés dans les simulations sont présentés dans le tableau 1.1 du chapitre 1.

La précision et la robustesse de chaque contrôleur ont été étudiées avec deux types de trajectoire; poursuite de trajectoire linéaire et poursuite de trajectoire circulaire. Le manipulateur est simulé pour suivre la trajectoire linéaire générée à la position initiale (1.9, 0.5) et d'atteindre à la position finale (0.1, -1.3) - au bout de 4 secondes, la vitesse du mouvement linéaire souhaitée est (0,64 m/s). La vitesse angulaire du mouvement circulaire souhaitée est $\frac{\pi}{2} rad/s$. Toutes ces simulations ont été réalisées en utilisant MATLAB avec un intervalle d'échantillonnage de 1ms.

Cas1 : Petites variations paramétriques

Nous avons effectué des simulations pour étudier les effets de petites variations paramétriques sur les réponses du système, dans ce cas-là les paramètres du système sont modifiés à 2 secondes après le démarrage de la simulation, les paramètres initiaux sont les paramètres nominaux de système (tableau 1.1), et à partir de 2 secondes:

$$M_1=3Kg$$

$$M_2=6Kg$$

Les figures ci-dessous montrent l'évolution temporelle du vecteur de sortie, de l'erreur de sortie et de la poursuite de trajectoire linéaire par les trois contrôleurs cités dans la première partie de ce chapitre.

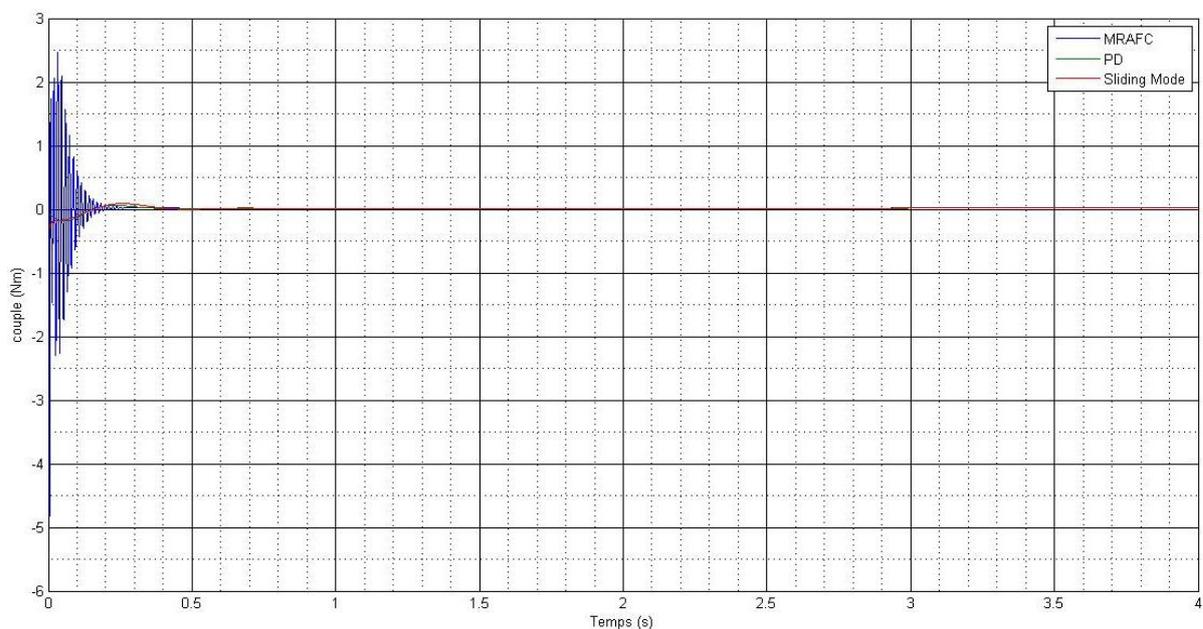


Figure 4. 8 – Signale de commande pour trajectoire droite avec des petites variations paramétriques

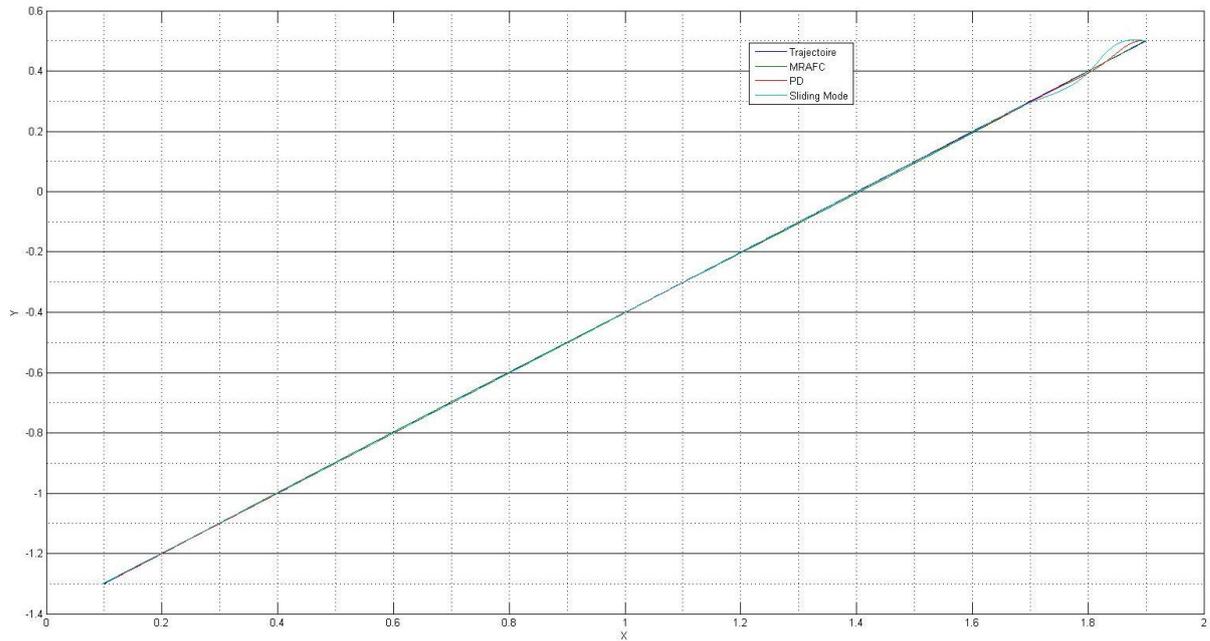


Figure 4. 9 – Réponses de système sur une trajectoire linéaire dans l'espace cartésien avec des petites variations paramétriques

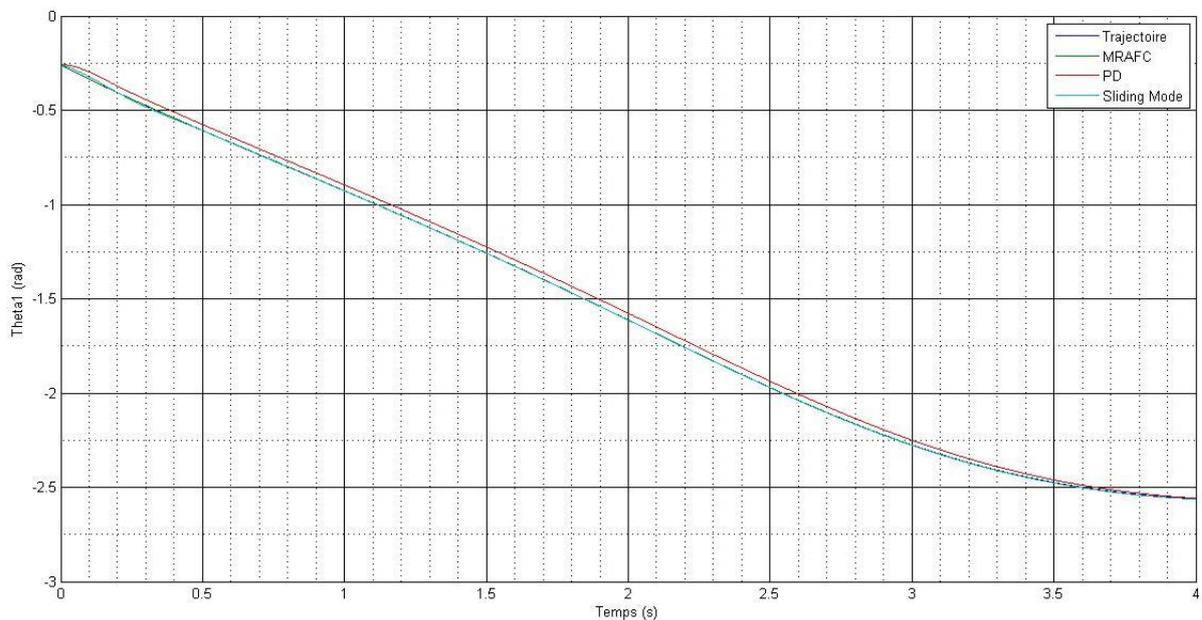


Figure 4. 10 – L'angle q_1 d'une trajectoire linéaire avec des petites variations paramétriques

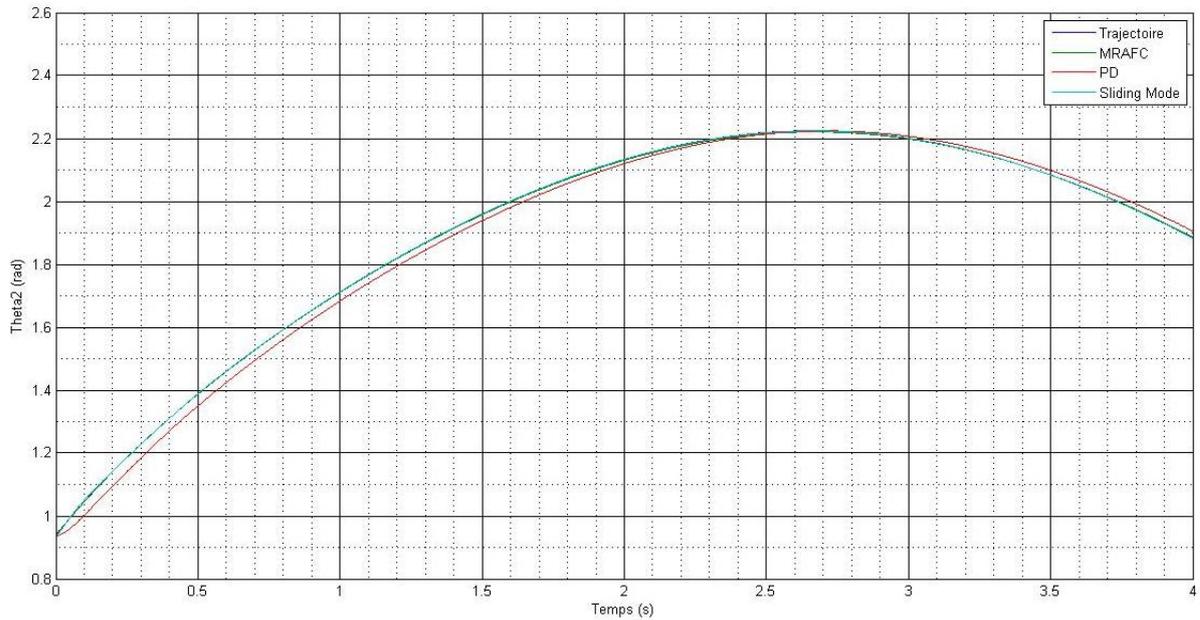


Figure 4. 11 – L'angle q_2 d'une trajectoire linéaire avec des petites variations paramétriques

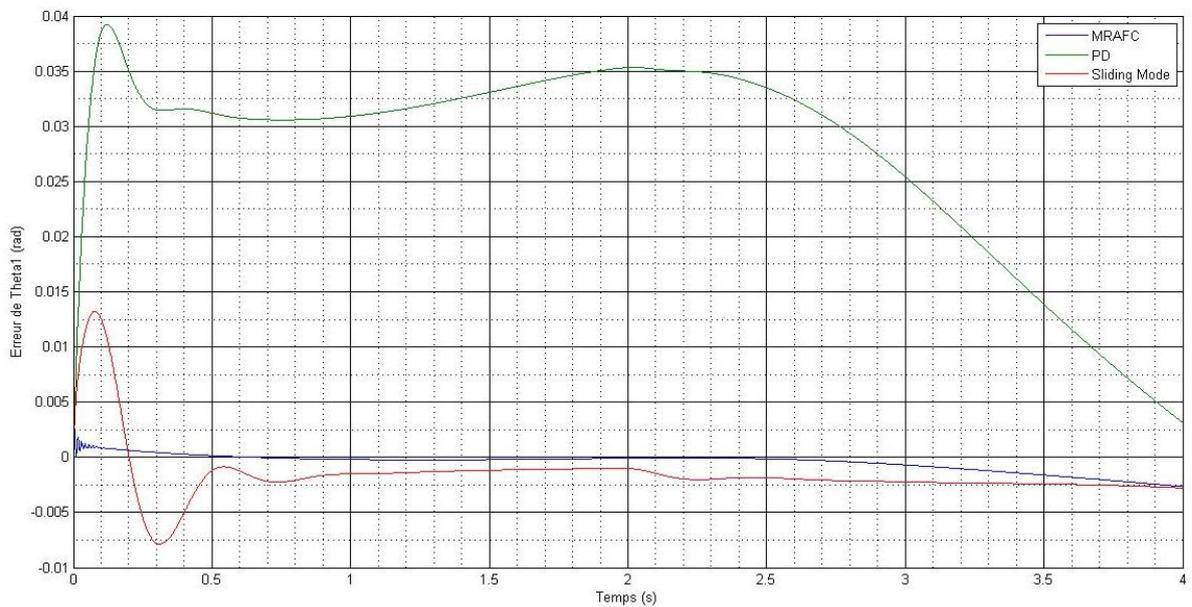


Figure 4. 12 – L'erreur de l'angle q_1 sur une trajectoire linéaire avec des petites variations paramétriques

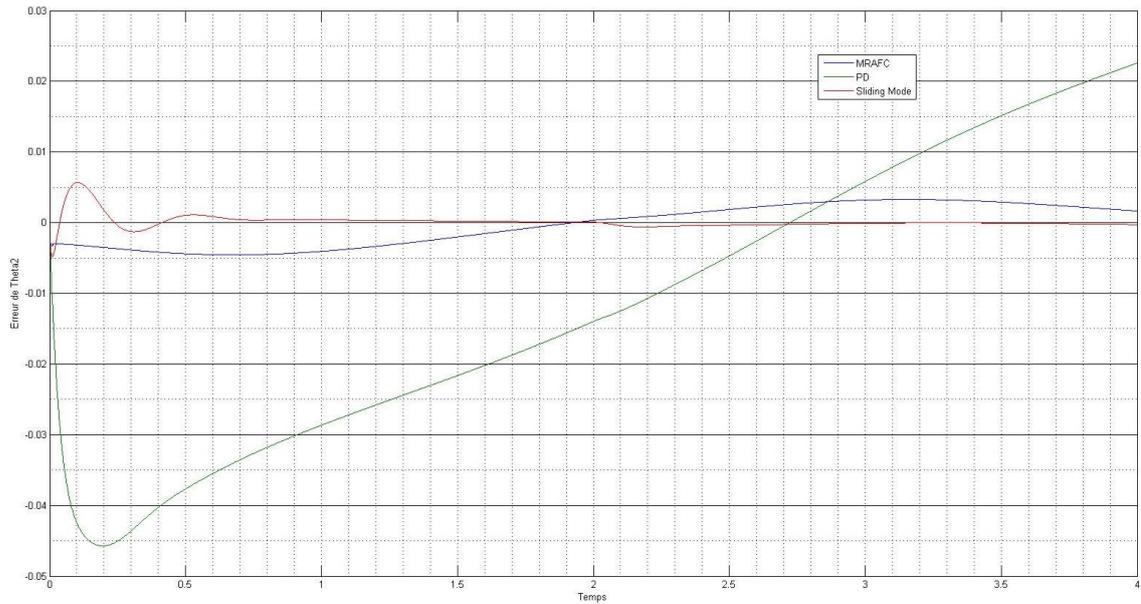


Figure 4. 13 – L’erreur de l’angle q_2 sur une trajectoire linéaire avec des petites variations paramétriques

Les figures ci-dessous montrent l’évolution temporelle du vecteur de sortie, de l’erreur de sortie et de la poursuite de trajectoire circulaire par les trois contrôleurs cités dans la première partie de ce chapitre.

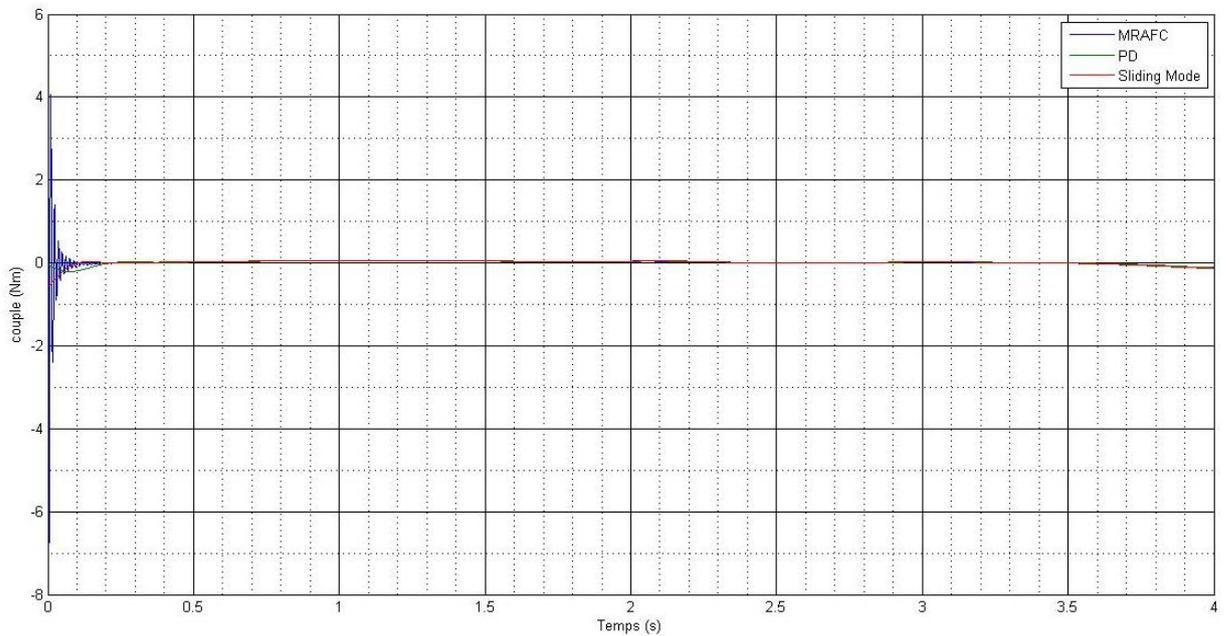


Figure 4. 14 – Signale de commande pour une trajectoire circulaire avec des petites variations paramétriques

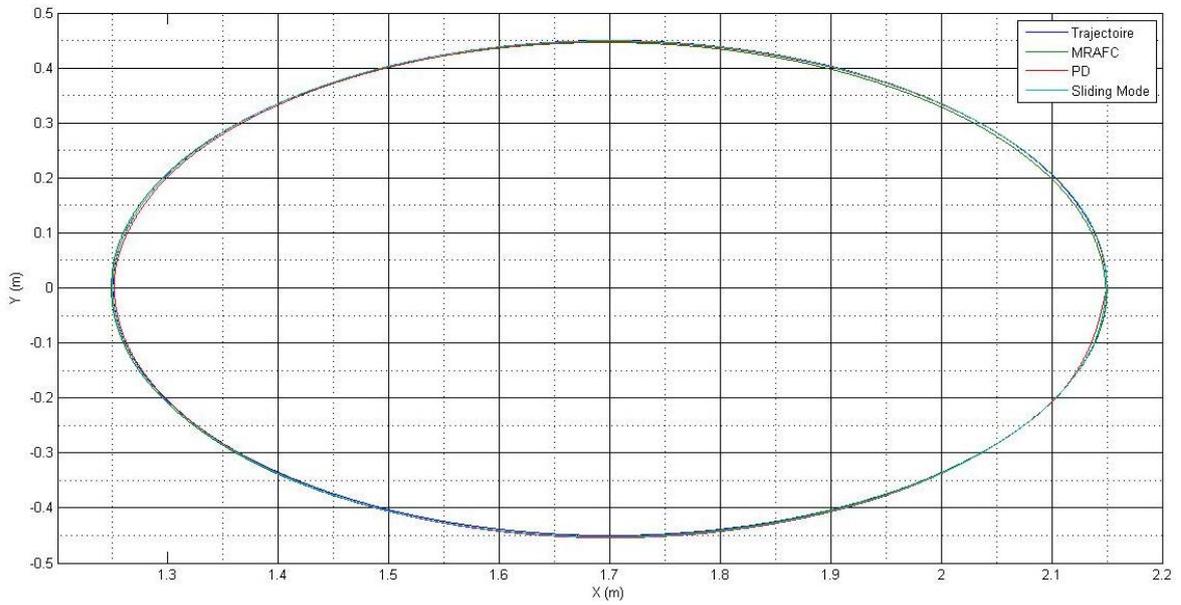


Figure 4. 15 – Réponses de système sur une trajectoire circulaire dans l'espace cartésien avec des petites variations paramétriques

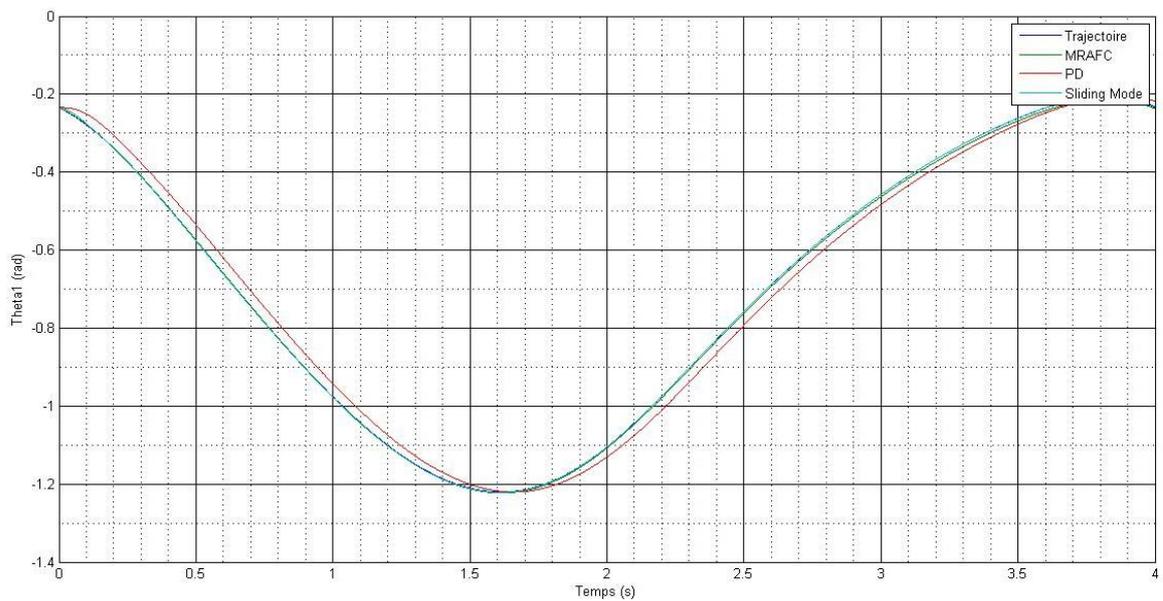


Figure 4. 16 – L'angle q_1 d'une trajectoire circulaire avec des petites variations paramétriques

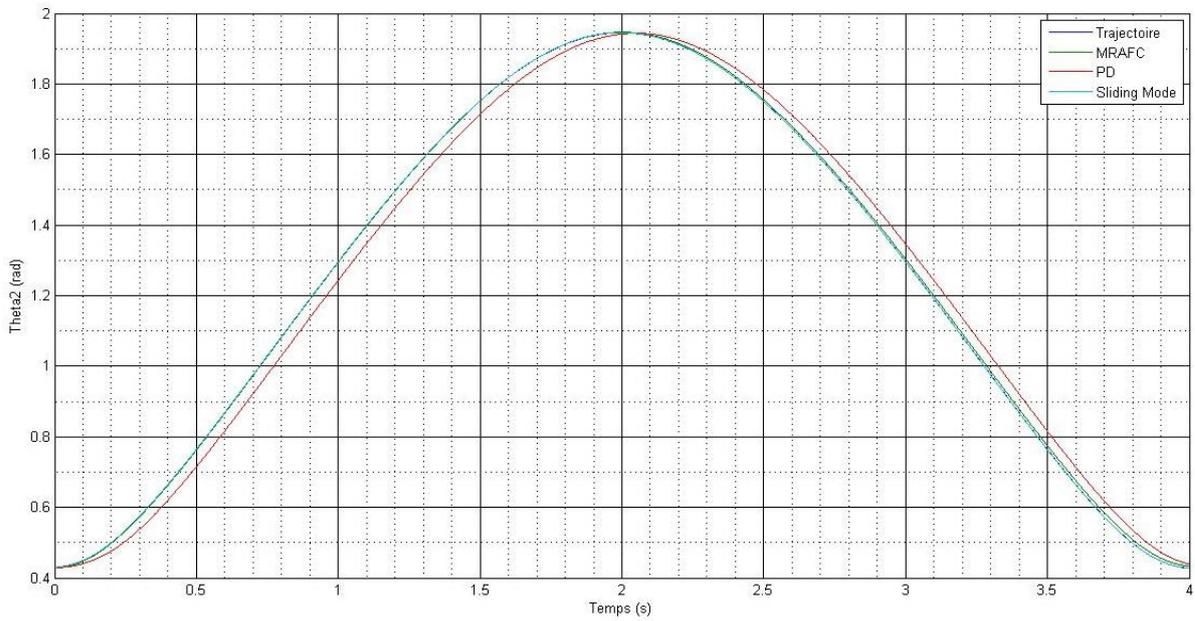


Figure 4. 17 – L'angle q_2 d'une trajectoire circulaire avec des petites variations paramétriques

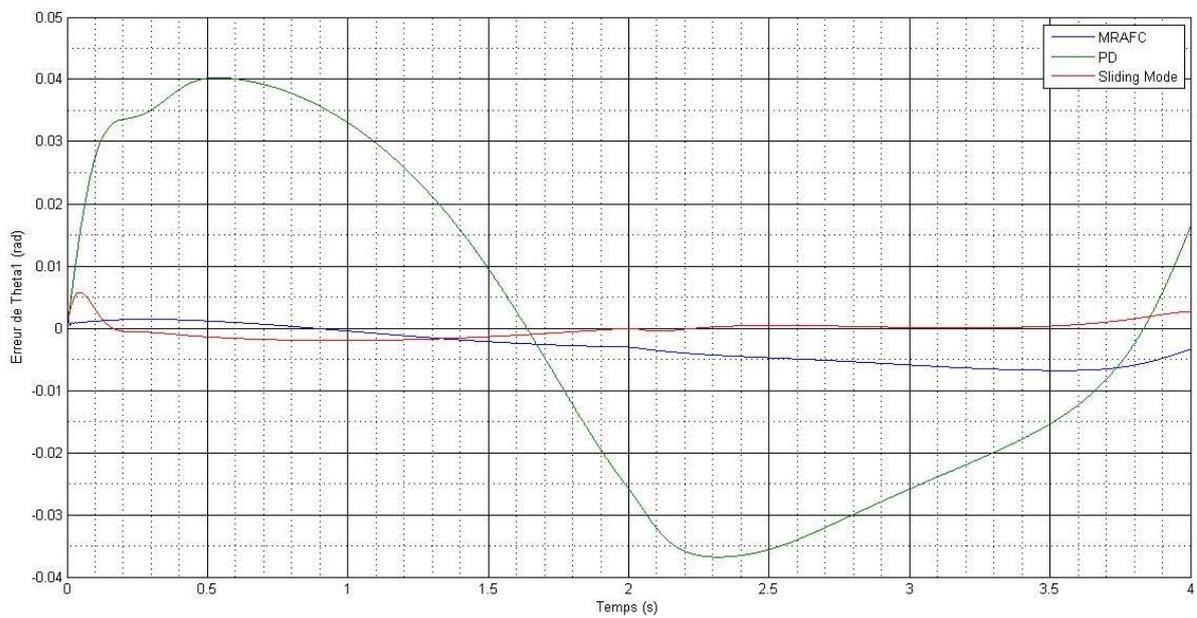


Figure 4. 18 – L'erreur de l'angle q_1 sur une trajectoire circulaire avec des petites variations paramétriques

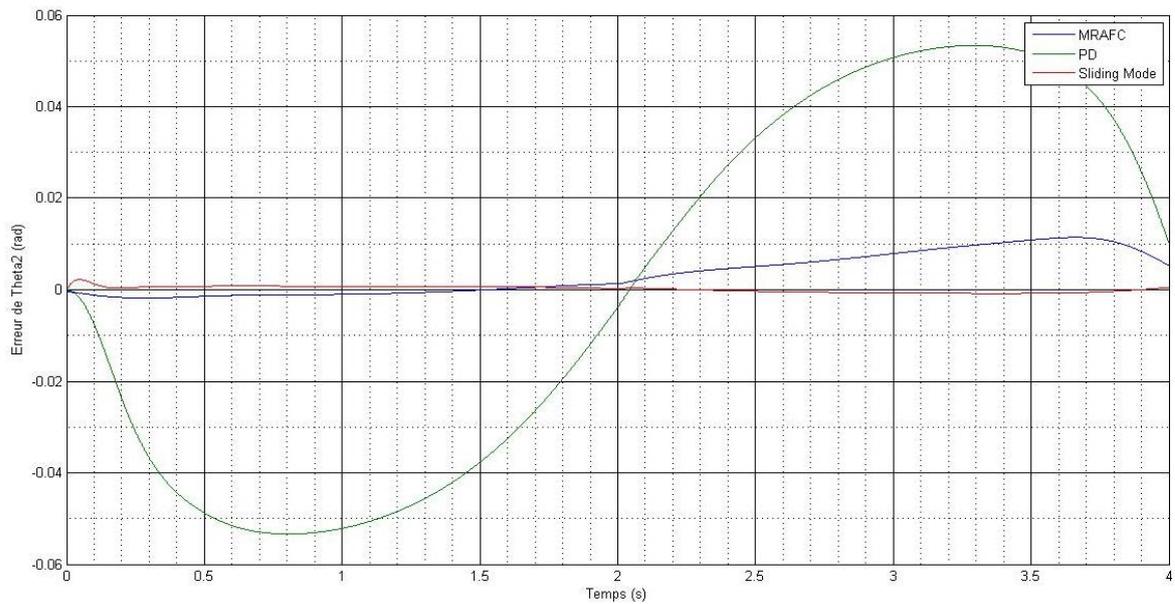


Figure 4. 19 – L’erreur de l’angle q2 sur une trajectoire circulaire avec des petites variations paramétriques

Interprétation des résultats de simulation :

Les résultats de simulation montrent la dynamique de rejet de perturbation et de poursuite de la référence (figure 5.8 et 5.13) avec une erreur inférieure à 2%. Seul le contrôleur PD a montré une réponse sensible aux variations paramétriques. Aussi à partir des résultats de simulation, on peut déduire que la commande adaptative représenté peut faire face aux variations paramétriques sans erreur de décalage, alors que la commande glissante a une certaine erreur de décalage.

Cas 2 : Grande variations paramétrique

Nous avons aussi effectué des simulations pour étudier les effets de grandes variations paramétriques sur les réponses du système, dans ce cas-là les paramètres du système sont modifiés à 2 secondes après le démarrage de la simulation avec une variation paramétrique de la forme suivante (Figure 4.18):

$$M1 = 2\sin(2\pi t)^2 + 6 \quad M2 = 4\sin(2\pi t)^2 + 12$$

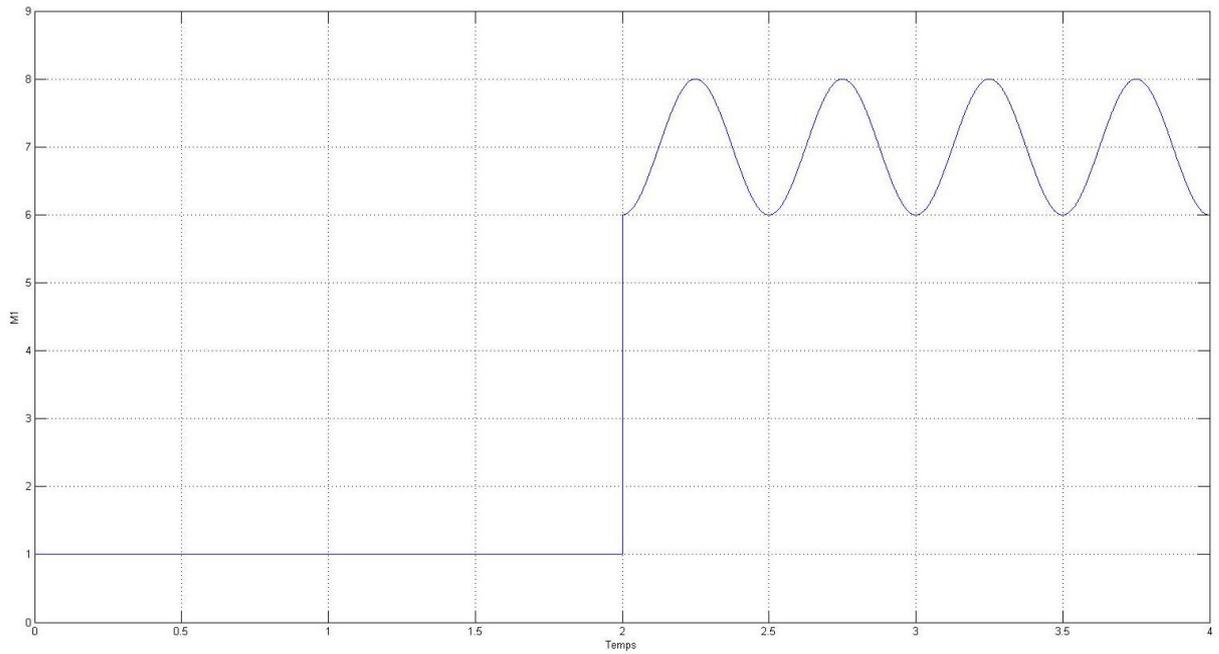


Figure 4. 20 – Grande variations paramétriques de M1

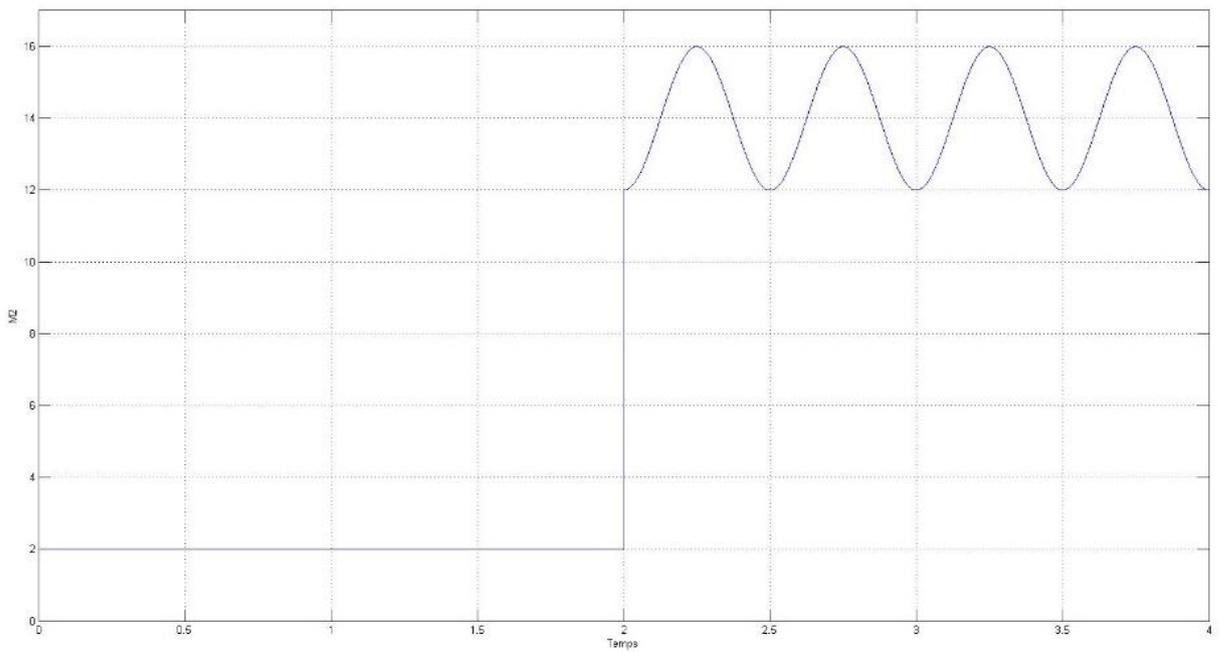


Figure 4. 21 – Grande variations paramétriques de M2

Les figures ci-dessous montrent l'évolution temporelle du vecteur de sortie, de l'erreur de sortie et de la poursuite de trajectoire linéaire par les trois contrôleurs cités dans la première partie de ce chapitre.

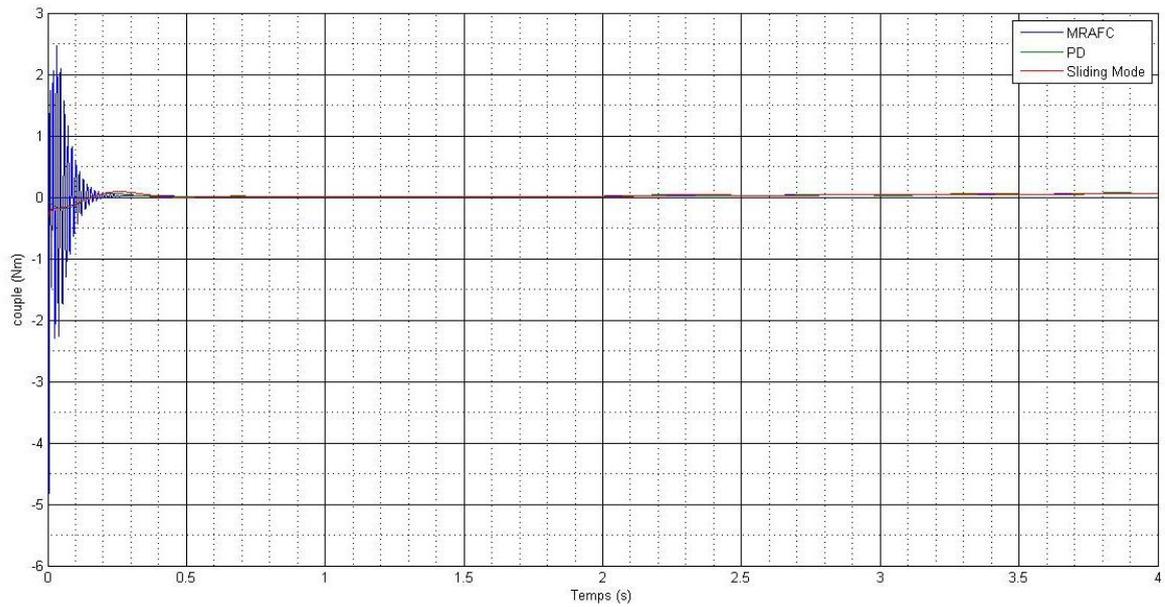


Figure 4. 22 – signale de commande pour une trajectoire linéaire avec des grandes variations paramétriques

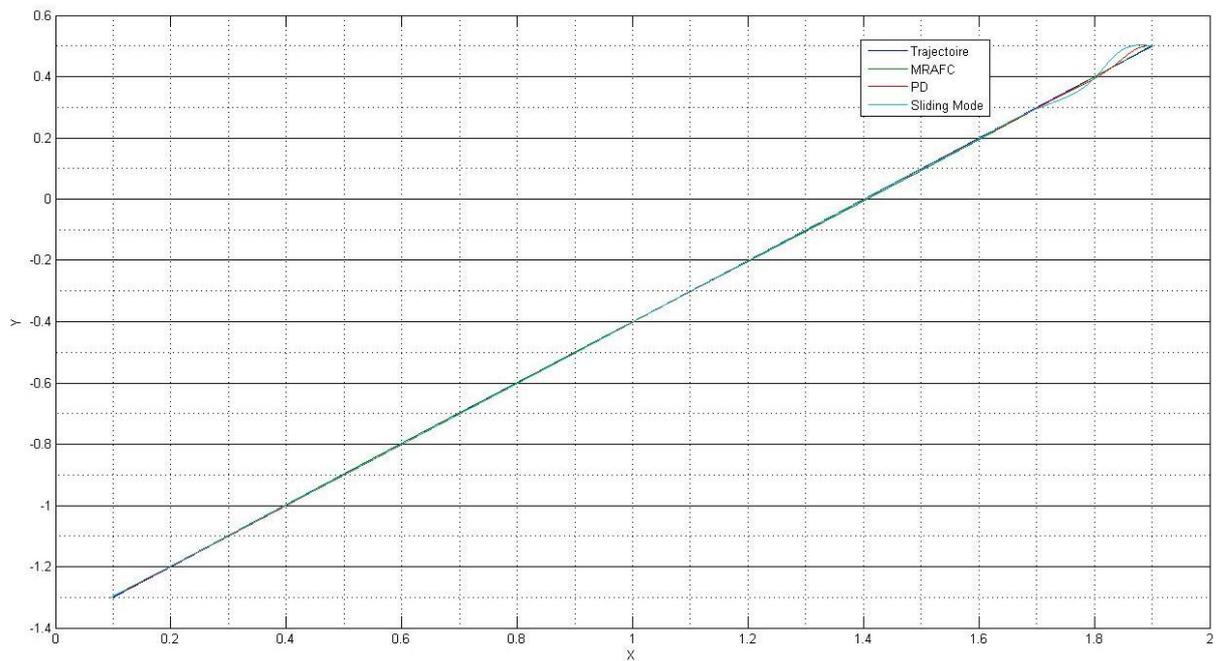


Figure 4. 23 – Réponses de système sur une trajectoire linéaire dans l'espace cartésien avec des grandes variations paramétriques

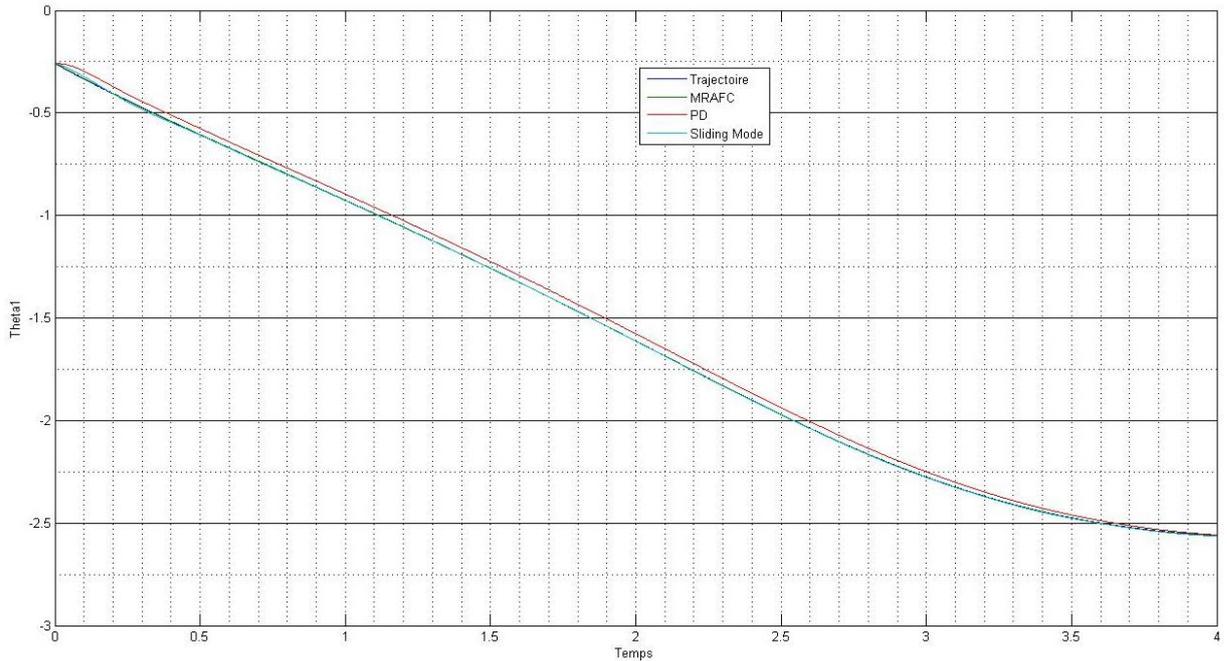


Figure 4. 24 – L'angle q_1 d'une trajectoire linéaire avec des grandes variations paramétriques

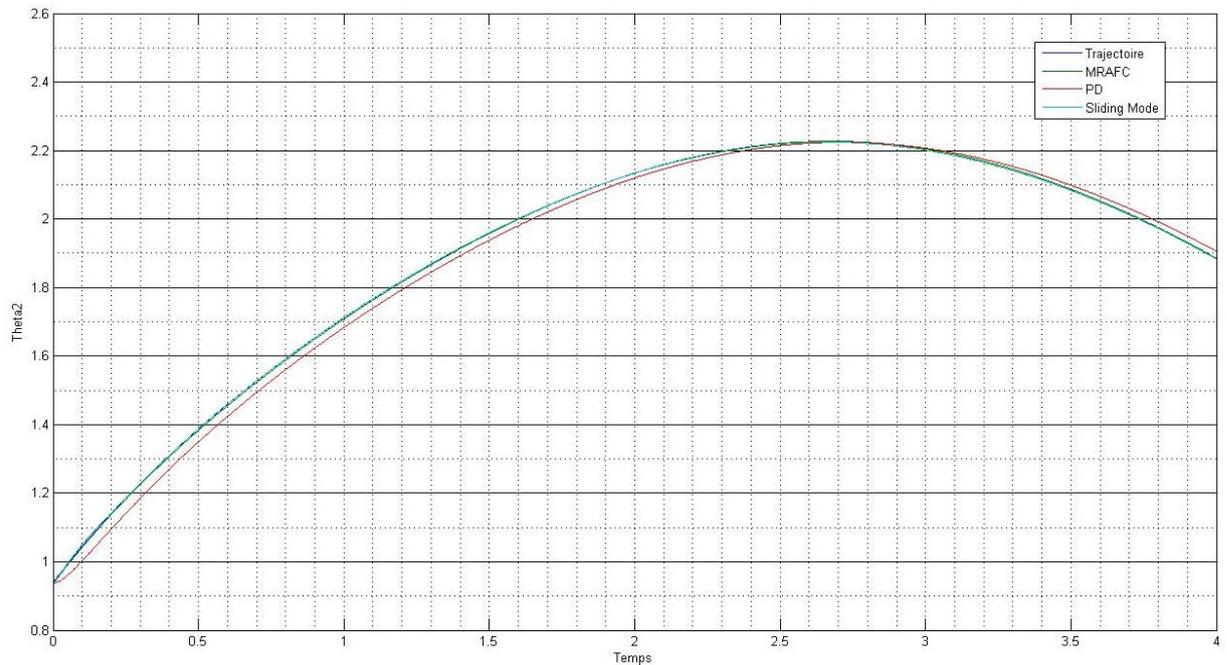


Figure 4. 25 – L'angle q_2 d'une trajectoire linéaire avec des grandes variations paramétriques

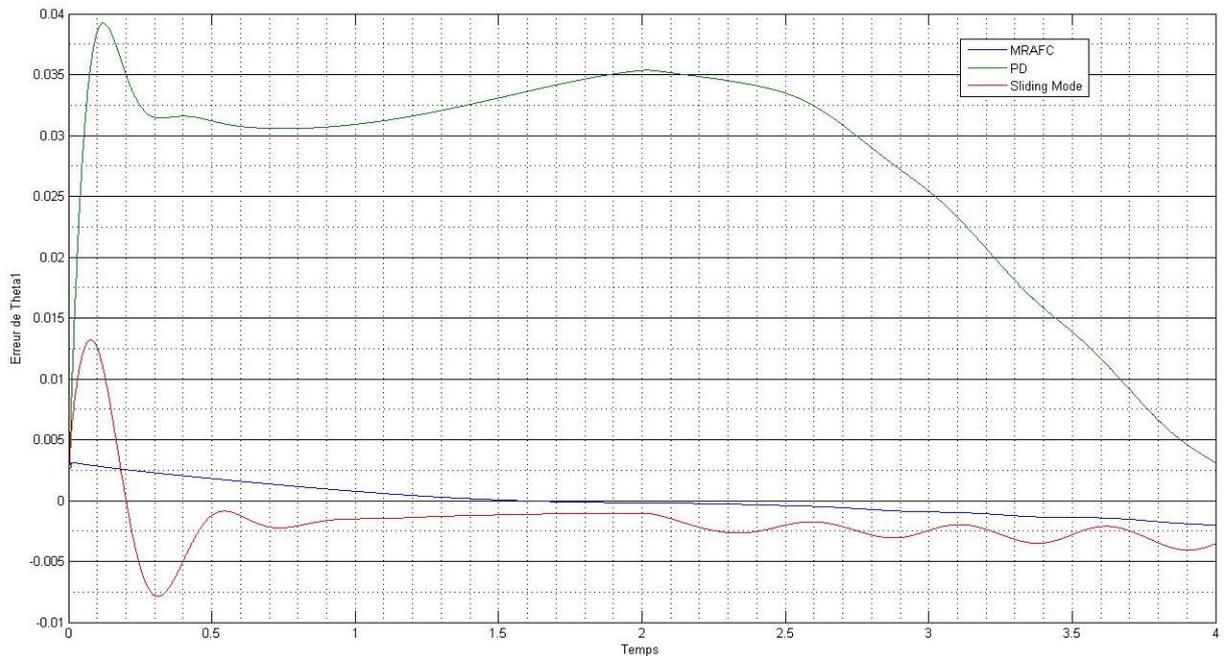


Figure 4. 26 – L'erreur de l'angle q_1 sur une trajectoire linéaire avec des grandes variations paramétriques

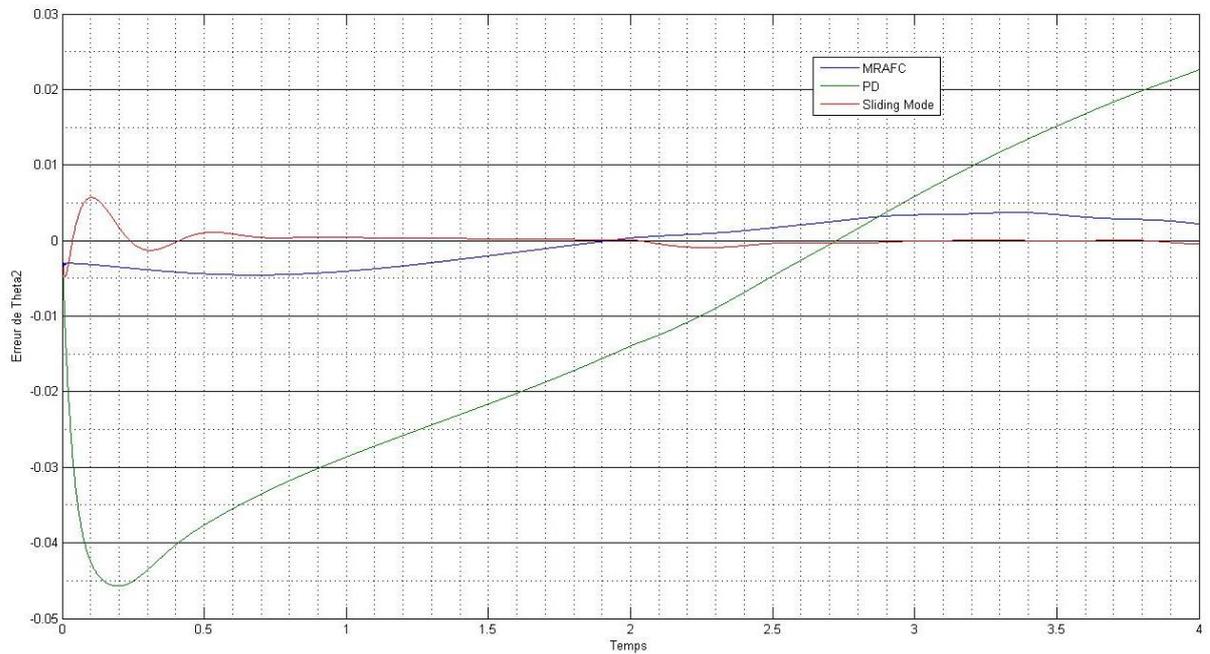


Figure 4. 27 – L'erreur de l'angle q_2 sur une trajectoire linéaire avec des grandes variations paramétriques

Les figures ci-dessous montrent l'évolution temporelle du vecteur de sortie, de l'erreur de sortie et de la poursuite de trajectoire circulaire par les trois contrôleurs cités dans la première partie de ce chapitre.

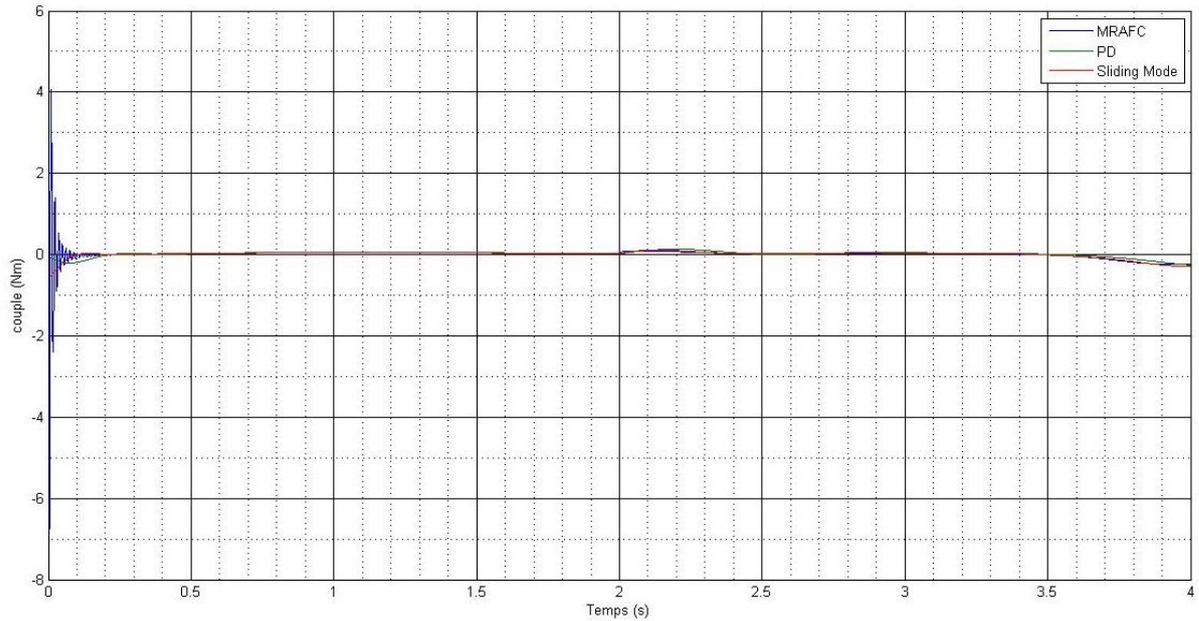


Figure 4. 28 – signale de commande pour une trajectoire circulaire avec des grandes variations paramétriques

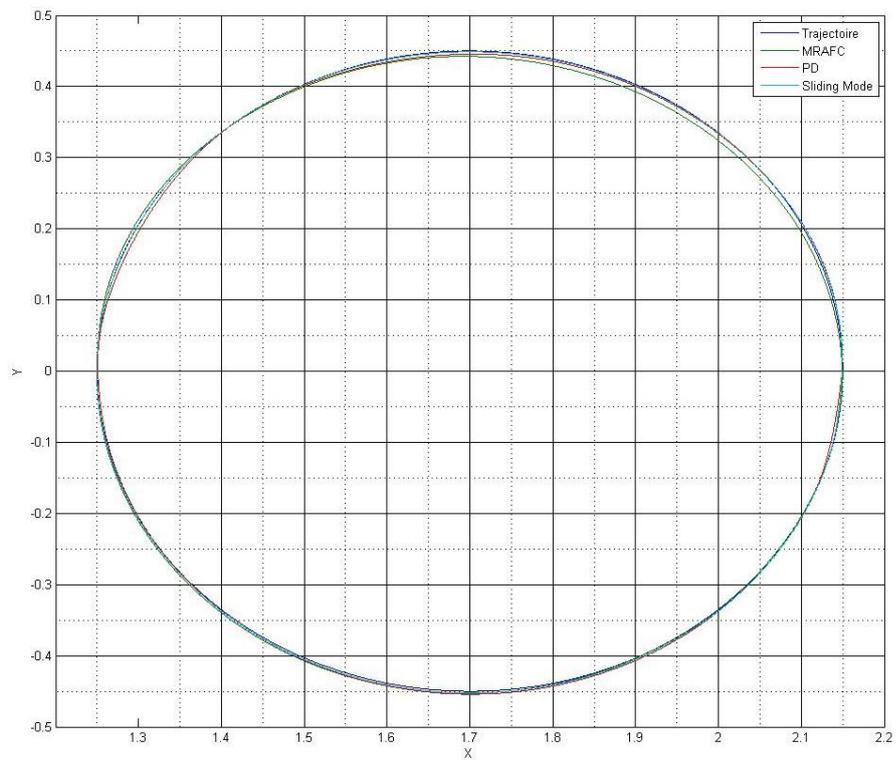


Figure 4. 29 – Réponses de système sur une trajectoire circulaire dans l'espace cartésien avec des grandes variations paramétriques

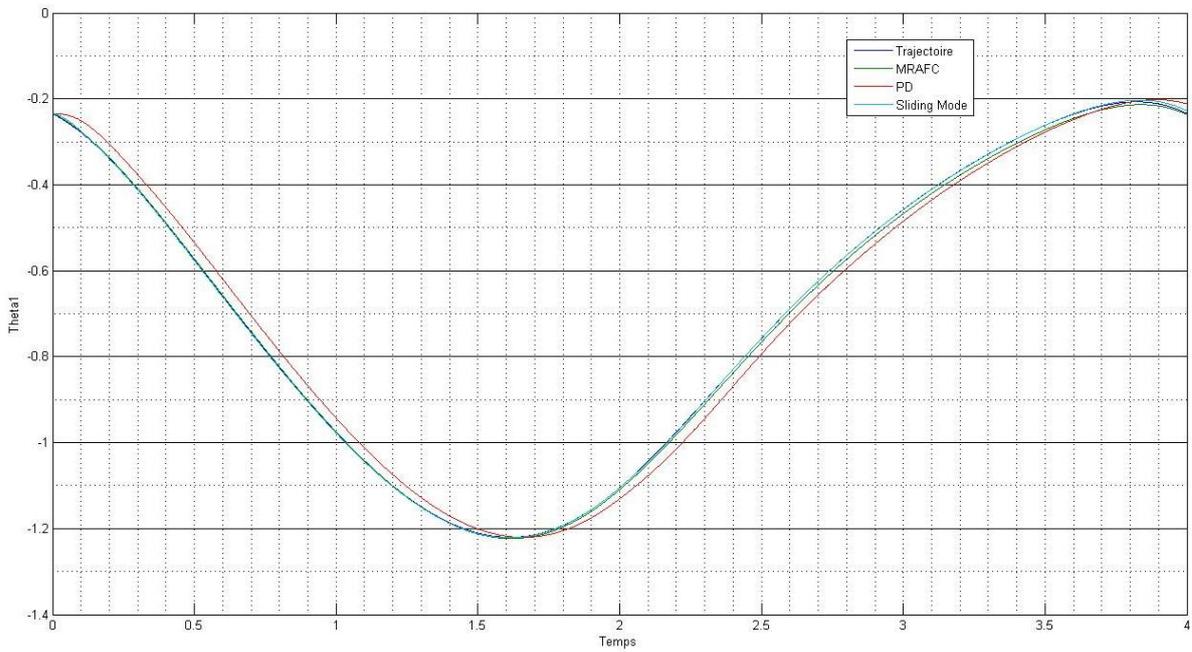


Figure 4. 30 – L'angle q_1 d'une trajectoire circulaire avec des grandes variations paramétriques

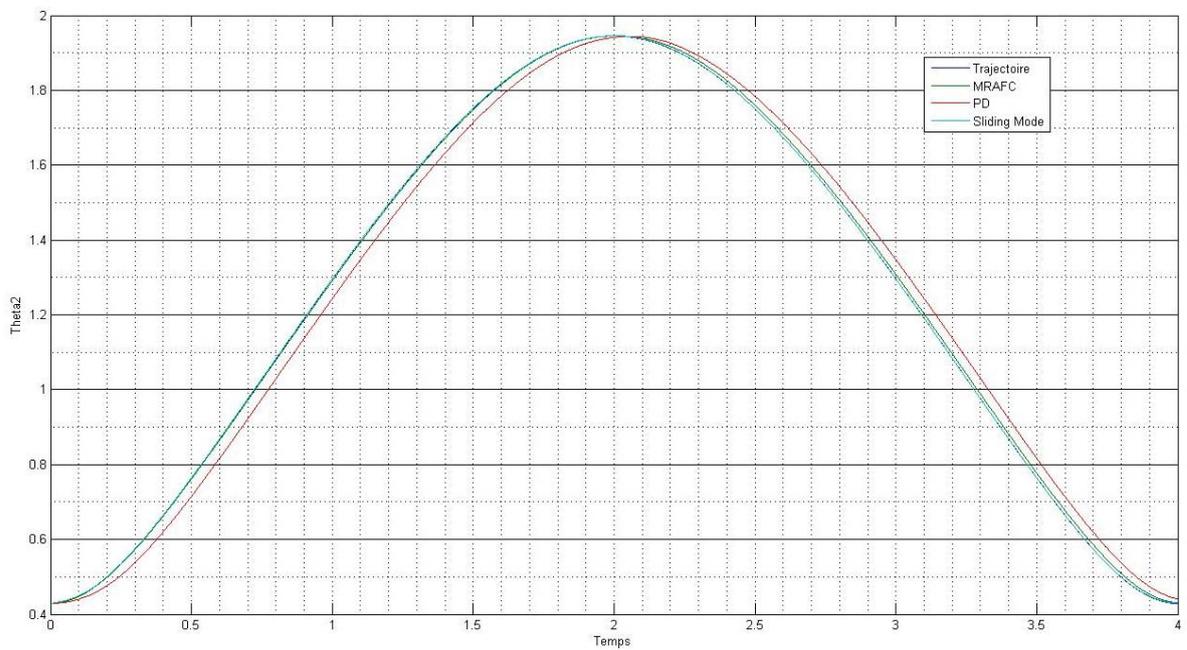


Figure 4. 31 – L'angle q_2 d'une trajectoire circulaire avec des grandes variations paramétriques

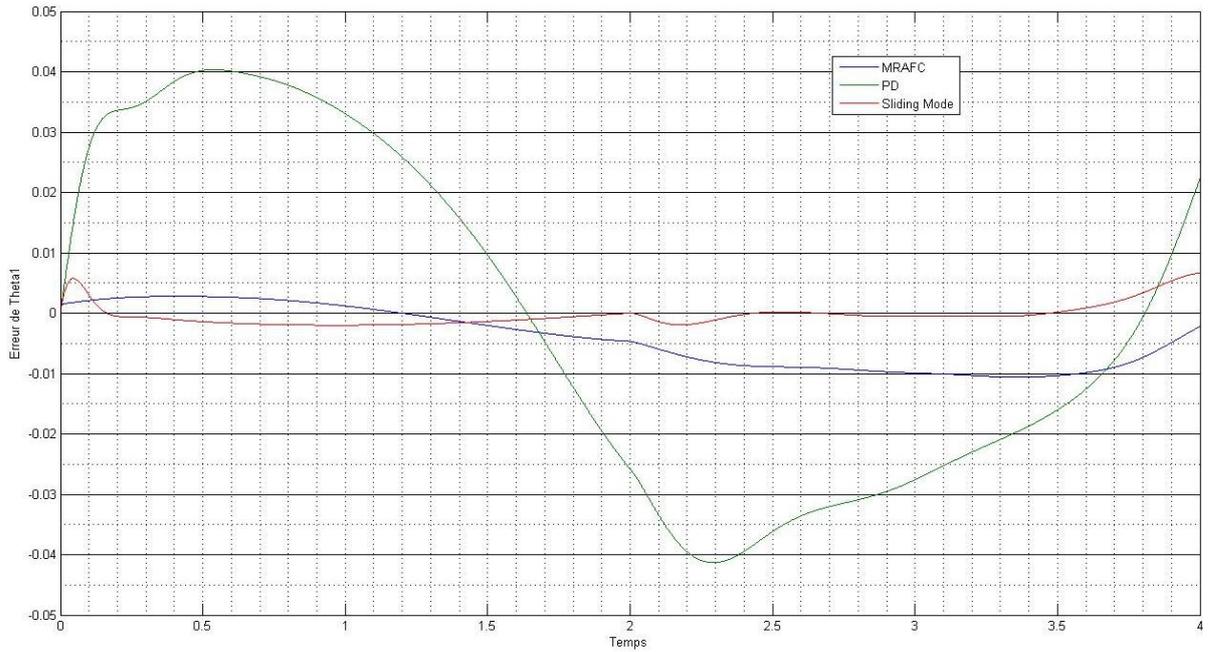


Figure 4. 32 – L’erreur de l’angle q_1 sur une trajectoire circulaire avec des grandes variations paramétriques

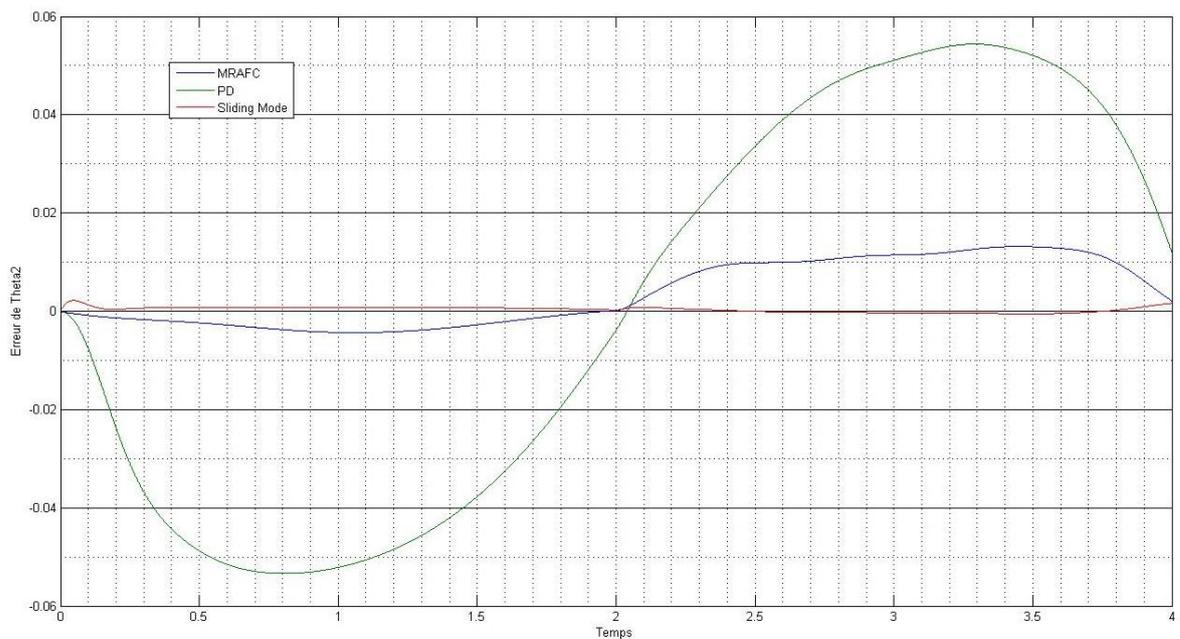


Figure 4. 33 – L’erreur de l’angle q_2 sur une trajectoire circulaire avec des grandes variations paramétriques

Interprétation des résultats de simulation : Les résultats de simulation montre la dynamique de rejet de perturbation et de poursuite de la référence qui suit parfaitement sa référence (figure 4.20 et 4.25) avec une erreur inférieure à 2% (figure 4.23, 4.24, 2.28 et 2.29). D’après les résultats de simulation, le contrôleur PD ne montre pas une forte capacité d’adaptation avec de grandes variations paramétriques, alors que la commande développée dans notre travail peut stabiliser le système très rapidement.

Conclusion :

Dans ce chapitre nous avons étudié le comportement et les performances de poursuite et de rejet de perturbation de la commande MRAFC comparée à d'autres contrôleurs sur des différents cas de trajectoires et de variation paramétriques. Les points suivants peuvent être soulignés à partir des résultats de simulation:

- (1) Le commande appliqué e présente des propriétés excellentes pour la réponse transitoire telle que temps de réponse très petit, un léger dépassement, par rapport à ceux du contrôleur PD et le contrôleur adaptatif glissant.
- (2) La commande floue adaptative conçue peut atteindre efficacement la trajectoire de poursuite pour les systèmes avec de grandes variations paramétriques.
- (3) le système de contrôle appliqué montre un grand retard de phase et plus petite erreur transitoire pour toutes les situations simulées.

Conclusion générale :

Les travaux présentés dans ce mémoire appliquent des résultats obtenus sur les modèles flous de Takagi-Sugeno afin de synthétiser une commande adaptative pour un système non-linéaire, qui est celui d'un bras manipulateur.

Dans le contexte de la modélisation, l'approche multimodèle a été choisie pour les nombreux avantages qu'elle apporte au niveau de l'analyse de stabilité et de la synthèse de commande pour les systèmes non-linéaires. Cette structure a fait l'objet de nombreuses études dans différents domaines comme l'identification, la commande ou l'estimation.

Après avoir établi le modèle non linéaire de bras manipulateur, l'approche par linéarisation a été utilisée pour l'obtention de modèle flous de Takagi-Sugeno. Nous avons vu que cette dernière offrait des modèles flous équivalents aux modèles non linéaires sans perte d'information, bien qu'elle nécessite la connaissance de la dynamique du système.

Sur la base du modèle flou de Takagi-Sugeno obtenu, nous avons réalisé une commande adaptative directe avec modèle de référence (MRAFC). La loi d'adaptation développée garantit la bornitude de tous les signaux dans le système en boucle fermée et assure que l'état de système suit l'état du modèle de référence asymptotiquement avec le temps pour tout signal d'entrée de référence bornée.

Les résultats proposés dans ce mémoire ouvrent un certain nombre de perspectives :

- Etendre l'application de la commande MRAFC à des bras de robot plus complexes.
- Implémentation réelle de la commande sur les bras de robot.
- Application de la commande sur d'autres types de systèmes.

Bibliographie

- [Abonyi et al., 1999] Abonyi, J., Babuška, R., Setnes, M., Verbruggen, H., et Szeifert, E. (1999). *Constrained parameter estimation in fuzzy modeling*. In *Fuzzy Systems Conference Proceedings, 1999. FUZZ-IEEE'99*. 1999 IEEE International, volume 2, pages 951–956. IEEE.31
- [Bezzaoucha, 2013] Bezzaoucha, S. (2013). *Commande tolérante aux défauts de systèmes non linéaires représentés par des modèles de TakagiSugeno*. PhD thesis, Université de Lorraine. 31, 34
- [Boyd et al., 1994] Boyd, S. P., El Ghaoui, L., Feron, E., et Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*, volume 15. SIAM. 74
- [Chekireb et al., 2003] H. Chekireb, M. Tadjine, et D. Bouchaffra, “*Direct adaptive fuzzy control of nonlinear system class with applications*,” *Control and Intelligent Systems*, vol. 31 no. 2, pp. 113-121,2003.
- [Delarminat, 1993] P. deLarminat *Automatique : commande des systèmes linéaires* 1993
- [Dombre et Khalil 2007] E. Dombre et W. Khalil *Modeling, Performance Analysis and Control of Robot Manipulators* 2007
- [Dubuisson, 1990] B. Dubuisson, “*Diagnostic et reconnaissance des formes*,” Hermès, Paris.
- [Elkhalil 2011] M. Elkhalil. *Modélisation, observation et commande : une approche multimodèle*. Automatic Control Engineering. Université de Caen; Université de Gabès (Tunisie), 2011.
- [Gasso et al., 1999] K. Gasso, G. Mourot, A. Boukhriss et J. Ragot, “*Optimisation de la structure d’un modèle de Takagi-Sugeno*,” Actes des rencontres francophones (LFA), Valenciennes, France, 1999.
- [Gasso et al., 2000]. K. Gasso, “*Identification des systèmes dynamiques non-linéaires : approche multi-modèles*,” Thèse de doctorat de l’INPL, Décembre, 2000
- [Guerra et al., 1999] T. M. Guerra, L. Vermeiren, F. Delmotte et P. Borne, “*Lois de commande pour systèmes flous continus Lois de commande pour systèmes flous continus*,” *Journal Européen des Systèmes Automatisés*, 33(4) : 489-527, 1999
- [Henrion et Garulli , 2004] D. Henrion et A. Garulli, “*Stabilization of LPV Systems*,” *Lecture Notes in Control and Information Sciences*, Springer, 2004
- [Kawamoto et al., 1992] S. Kawamoto, K. Tada, A. Ishigame et T. Taniguchi, “*An approach to stability analysis of second order fuzzy systems*,” *IEEE International Conference on Fuzzy Systems*: 1427–1434. 1992.

- [Khalil 1987] KHALIL W., KLEINFINGER J.-F., “*Minimum operations and minimum parameters of the dynamic models of tree structure robots*”, IEEE J. Robotics and Autom. p. 517-526, 1987.
- [Khalil 1988] W.Khalil, “*Modélisation et commande des robots*”, Édition Hermès, Paris, 1988
- [Khalil, 1996] H. Khalil, “*Nonlinear systems*,” Prentice hall, Inc. second edition, USA, 1996.
- [Khosla 86] KHOSLA P.K., “*Real-time control and identification of directdrive manipulators*”, PhD Thesis, Carnegie Mellon University, Pittsburgh, 1986.
- [Kleinfinger 86a] KHALIL W., KLEINFINGER J.-F., “*A new geometric notation for open and closed-loop robots*”, Proc. IEEE Int. Conf. on Robotics and Automation, San Francisco, Avril 1986, p. 1174-1180.
- [Krzysztof. 98] Krzysztof K.- *Modelling and identification in robotics:Advances in Industrial Control*. Springer.1998.
- [Kuo, 1995] Kuo "Automatic control systems" Prentice Hall Editions 1995
- [Lallemand 1994] Lallemand J.P. et Zeghloul S. *Robotique Aspects fondamentaux, Modélisation mécanique CAO robotique-Commande*.Masson, 1994.
- [Liapounoff, 1907] Liapounoff, A. (1907). “*Problème général de la stabilité du mouvement*,” Annales de la faculté des sciences de Toulouse, (9):203–474. (traduction en français du mémoire Russe).
- [Liberzon, 2003] D. Liberzon, “*Switching in systems and control*,” Edition Boston Birkauer, 2003.
- [Lin et Er, 2001] D. Lin et M.J. Er, “*A new approach for stabilising nonlinear systems with time delay*,” European Control Conference,Porto, Portugal, 2001.
- [Luh 80b] LUH J.Y.S., WALKER M.W., PAUL R., “*On-line computational scheme for mechanical manipulators*”, ASME, J. of Dyn. Syst., Measurements and Control, Vol. 102, No. 2, p. 69-76, 1980.
- [Ma et al., 1998] X. J. Ma, Z. Q. Sun et Y. Y. He, “*Analysis and design of fuzzy controller and fuzzy observer*,”Transactions on Fuzzy Systems, 6(1): 41-50, 1998.
- [Morère, 2001] Y. Morère, “*Mise en oeuvre de lois de commande pour les modèles flous de type Takagi-Sugeno*,” Thèse de doctorat, LAMIH-SF, de l’Université de Valenciennes et du Hainaut-Cambrésis, 2001
- [Oustaloup et Mathieu, 1999] A. Oustaloup et B. Mathieu “*La commande crone*” Hermès Science Publications,Paris, 1999.
- [Sastry, 1999] S. Sastry, “*Nonlinear systems: analysis, stability, and control*,” Springer-Verlag, New York, 1999.
- [Slotine et Li 1991] J. J. Slotine et W. Li, *Applied Nonlinear Control*, Prentice-Hall, 1991.

- [Slotine et Sastry 1983] J. J. Slotine et S. S. Sastry, “*Tracking control of nonlinear systems using sliding surfaces, with application to robot manipulators,*” International Journal of Control, vol. 38, no. 2, pp. 465-492, 1983.
- [Su et al, 1993] Chun-Yi Su. Tin-Pui Leung et Yury Stepanenko, “*Real-Time Implementation of Regressor-Based Sliding Mode Control Algorithm for Robotic Manipulators,*” IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 40, NO. 1, FEBRUARY 1993
- [Takagi et Sugeno, 1985] T. Takagi et M. Sugeno, “*Fuzzy identification of systems and its application to modelling and control,*” Transactions on Systems Man and Cybernetics, 1115: 116-132,1985.
- [Tanaka et al., 1998] K. Tanaka, T. Ikeda et H. O. Wang, “*Fuzzy regulators and fuzzy observers: Relaxed stability conditions and LMI-based designs,*” Transactions on Fuzzy Systems, 6 (2), Mai 1998.
- [Takagi et Sugeno, 1985] T. Takagi et M. Sugeno, “*Fuzzy identification of systems and its application to modelling and control,*” Transactions on Systems Man and Cybernetics, 1115: 116-132,1985.
- [Tanaka et Sugeno, 1992] K. Tanaka et M. Sugeno, “*Stability analysis and design of fuzzy control systems,*” Fuzzy Sets and Systems, 45(2): 135-156,1992.
- [Tanaka et Wang, 2001] K. Tanaka et H.O. Wang, “*Fuzzy control systems design and analysis : a linear matrix inequality approach,*” John Wiley & Son Eds, New York, Wiley-Inter science ISBN 0-471-32324-190000, 2001.
- [Wang et al., 1996] H. O. Wang, K. Tanaka et M. Griffin, “*An approach to fuzzy control of non linear systems: Stability and design issues,*” Transactions on Fuzzy Systems, (4): 14-23, 1996.
- [Y. Cho et al., 2007] Y. Cho, K. Seo, H. Lee, “*A Direct Adaptive Fuzzy Control of Nonlinear Systems with Application to Robot Manipulator Tracking Control*”, International Journal of Control, Automation, and Systems, vol. 5, no. 6, pp. 630-642, Décembre 2007.
- [Zerar et al., 2009] M. Zerar, F. Cazaurang et A. Zolghadri, “*Coupled LPV and Flatnessbased Approach for Space Reentry Vehicles Guidance,*” Control Theory and Applications 3(8) 1081-1092, 2009
- [Zhou et Doyle, 1998] K. Zhou and J.C. Doyle, “*Robust optimal control*”, Printice Hall Int. Ed., New Jersey, 1996.

Annexe A

A.1 Matrices Ai et Bi pour les 12 modèles locaux :

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 30.1593 & 30.1593 \\ 0 & 0 & -51.1032 & -30.1593 \end{bmatrix} \quad A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 10.0531 & -10.0531 \\ 0 & 0 & -30.9970 & 10.0531 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -10.0531 & 10.0531 \\ 0 & 0 & 30.9970 & -10.0531 \end{bmatrix} \quad A_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -30.1593 & -30.1593 \\ 0 & 0 & 51.1032 & 30.1593 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 726.4029 & 0 & 0 \\ 0 & -1397.5360 & 0 & 0 \end{bmatrix} \quad A_6 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 347.4101 & 0 & 0 \\ 0 & -702.7158 & 0 & 0 \end{bmatrix}$$

$$A_7 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 347.4101 & 0 & 0 \\ 0 & -702.7158 & 0 & 0 \end{bmatrix} \quad A_8 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 726.4029 & 0 & 0 \\ 0 & -1397.5360 & 0 & 0 \end{bmatrix}$$

$$A_9 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -30.1593 & -30.1593 \\ 0 & 0 & 51.1032 & 30.1593 \end{bmatrix} \quad A_{10} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -10.0531 & 10.0531 \\ 0 & 0 & 30.9970 & -10.0531 \end{bmatrix}$$

$$A_{11} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 10.0531 & -10.0531 \\ 0 & 0 & -30.9970 & 10.0531 \end{bmatrix} \quad A_{12} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 30.1593 & 30.1593 \\ 0 & 0 & -51.1032 & -30.1593 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix} \quad B_3 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix}$$

$$B_4 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix} \quad B_5 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -1.8333 \\ -1.8333 & 3.7083 \end{bmatrix} \quad B_6 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -1.8333 \\ -1.8333 & 3.7083 \end{bmatrix}$$

$$B_7 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -1.8333 \\ -1.8333 & 3.7083 \end{bmatrix} \quad B_8 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -1.8333 \\ -1.8333 & 3.7083 \end{bmatrix} \quad B_9 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix}$$

$$B_{10} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix} \quad B_{11} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix} \quad B_{12} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.3333 & -0.3333 \\ -0.3333 & 0.6805 \end{bmatrix}$$

A.2 Matrices A_i et B_i pour les 27 modèles locaux :

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 12.7254 & 0 \\ 0 & 0 & -28.2715 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 12.7254 \\ 0 & 0 & -15.5460 & -12.7254 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A_7 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -12.7254 \\ 0 & 0 & 15.5460 & 12.7254 \end{bmatrix}$$

$$A_9 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -12.7254 & 0 \\ 0 & 0 & 28.2715 & 0 \end{bmatrix}$$

$$A_{11} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 119.6901 & 0 & 0 \\ 0 & -316.5995 & 0 & 0 \end{bmatrix}$$

$$A_{13} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -53.5020 & 0 & 0 \\ 0 & 119.6901 & 0 & 0 \end{bmatrix}$$

$$A_{15} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -53.5020 & 0 & 0 \\ 0 & 119.6901 & 0 & 0 \end{bmatrix}$$

$$A_{17} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 119.6901 & 0 & 0 \\ 0 & -316.5995 & 0 & 0 \end{bmatrix}$$

$$A_{19} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -12.7254 & 0 \\ 0 & 0 & 28.2715 & 0 \end{bmatrix}$$

$$A_{21} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -12.7254 \\ 0 & 0 & 15.5460 & 12.7254 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 6.3627 & 6.3627 \\ 0 & 0 & -21.9087 & -6.3627 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 6.3627 & -6.3627 \\ 0 & 0 & -6.3627 & 6.3627 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -6.3627 & 6.3627 \\ 0 & 0 & 6.3627 & -6.3627 \end{bmatrix}$$

$$A_8 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -6.3627 & -6.3627 \\ 0 & 0 & 21.9087 & 6.3627 \end{bmatrix}$$

$$A_{10} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 173.1921 & 0 & 0 \\ 0 & -436.2896 & 0 & 0 \end{bmatrix}$$

$$A_{12} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -40.8159 & 0 & 0 \\ 0 & 42.4707 & 0 & 0 \end{bmatrix}$$

$$A_{14} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A_{16} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -40.8159 & 0 & 0 \\ 0 & 42.4707 & 0 & 0 \end{bmatrix}$$

$$A_{18} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 173.1921 & 0 & 0 \\ 0 & -436.2896 & 0 & 0 \end{bmatrix}$$

$$A_{20} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -6.3627 & -6.3627 \\ 0 & 0 & 21.9087 & 6.3627 \end{bmatrix}$$

$$A_{22} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -6.3627 & 6.3627 \\ 0 & 0 & 6.3627 & -6.3627 \end{bmatrix}$$

