

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE



ÉCOLE NATIONALE POLYTECHNIQUE
LABORATOIRE DE COMMANDE DE PROCESSUS
DEPARTEMENT AUTOMATIQUE



Mémoire de projet de fin d'études pour l'obtention du diplôme d'Ingénieur d'Etat
en Automatique.

**Modélisation et Commande d'Ordre Fractionnaire d'un Robot Parallèle de Type Delta
"ISIR88"**

Réalisé par :

BOUDEHANE Fadal allah

LABCHRI Amine

Sous la Direction de :

Pr. BOUKHETALA Djamel

Dr. BOUDJEDIR Chems Eddine

Composition du jury :

Président :	M. El Madjid BERKOUK	Professeur	ENP
Rapporteurs :	M. Djamel BOUKHETALA	Professeur	ENP
	M. Chems Eddine BOUDJEDIR	Docteur	ENP
Examineur :	M. Rachid ILOUL	MCA	ENP

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE



ÉCOLE NATIONALE POLYTECHNIQUE
LABORATOIRE DE COMMANDE DE PROCESSUS
DEPARTEMENT AUTOMATIQUE



Mémoire de projet de fin d'études pour l'obtention du diplôme d'Ingénieur D'état
en Automatique.

**Modélisation et Commande d'Ordre Fractionnaire d'un Robot Parallèle de Type Delta
"ISIR88"**

Réalisé par :

BOUDEHANE Fadal allah

LABCHRI Amine

Sous la Direction de :

Pr. BOUKHETALA Djamel

Dr. BOUDJEDIR Chems Eddine

Composition du jury :

Président :	M. El Madjid BERKOUK	Professeur	ENP
Rapporteurs :	M. Djamel BOUKHETALA	Professeur	ENP
	M. Chems Eddine BOUDJEDIR	Docteur	ENP
Examineur :	M. Rachid ILOUL	MCA	ENP

ENP 2020

نبذة مختصرة

الغرض من هذه المذكرة هو التصميم والتحكم في الروبوت الآلي ذي الأضلع المتوازية Delta ISIR88 ، والذي يشتهر بسرعته ودقته العاليتين، كما يستخدم في العديد من مجالات الصناعة. سيتم إنشاء نمذجة هذا الروبوت باستخدام فرضيات مبسطة، قد تؤدي الى إمكانية فقدان المعلومات التي سيتم تعويضها بواسطة قوانين التحكم (PD، PDOF، التحكم عن طريق الانزلاق) باستخدام خوارزمية التحسين PSO. سيتم القيام باختبارات المتانة عن طريق تغييرات الكتلة وتجسيدها بالمحاكاة وكذلك مقارنة النتائج التي تم الحصول عليها من قوانين التحكم المختلفة.

الكلمات المفتاحية: روبوت ISIR88، نمذجة والتحكم في روبوتات الصناعية، التحكم بمتحكمات ذات درجة كسرية PDOF ، خوارزمية التحسين PSO، التحكم بوضع الانزلاق.

Abstract

The aim of this project is the modeling and control of the parallel delta robot ISIR88, used in several fields in industry and known for its high speed and precision. The modeling of this robot is established using simplifying hypotheses, hence the possibility of losing information which will be compensated by control laws (PD, PDOF, Sliding Mode Control) where PSO algorithm will be introduced. Robustness tests for parametric variations are established by simulation as well as the comparison of the results obtained from the different commands.

Keywords : ISIR88 Robot, PSO Algorithm, Proportional Derivative (PD), Fractional Order Proportional Derivative (FOPD), Sliding Mode control SM.

Résumé

Ce Projet de fin d'étude a pour but la modélisation et la commande du robot parallèle Delta ISIR88 réputé pour sa rapidité et sa précision et utilisé dans plusieurs domaines d'industrie. La modélisation de ce robot est établit à l'aide d'hypothèses simplificatrices d'où la possibilité de perte d'informations qui sera compensée par des lois de commandes (PD, PDOF, Commande par Mode Glissant) en mettant en contribution l'algorithme d'optimisation PSO. Des tests de robustesses aux variations paramétriques sont établis par simulation ainsi que la comparaison des résultats obtenus des différentes commandes.

Mots clé : Robot ISIR 88, Algorithme PSO, Proportionnel dérivé PD, Proportionnel dérivé d'ordre fractionnaire PDOF, commande par mode glissant.

Dédicace

Nous dédions ce travail à nos chers parents pour tous leurs sacrifices, leur amour, et leurs prières tout au long de nos études.

À nos frères et sœurs que nous chérissons.

À nos familles.

À nos amis.

Remerciement

Nous remercions Dieu le tout-puissant de nous avoir donné le courage, la santé et de nous avoir accordé son soutien durant les périodes les plus difficiles.

*Nous remercions vivement nos encadreurs, Monsieur **BOUKHETALA Djamel** et **BOUDJEDIR Chems Eddine** de nous avoir aidés dans notre travail en nous conseillant et orientant pour l'accomplissement de ce dernier et surtout d'avoir cru en nos compétences.*

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre travail et d'avoir bien voulu l'examiner et l'enrichir de leurs propositions.

Nous remercions chaleureusement nos familles et tous nos amis pour leur soutien et leurs encouragements.

Nous souhaitons aussi remercier tous les enseignants de l'Ecole Nationale Polytechnique, et en particulier nos professeurs d'automatique qui nous ont encadrés auparavant et tous nos enseignants pour les connaissances qu'ils nous ont transmises, leur disponibilité et leurs efforts.

Que tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste travail trouvent ici l'expression de notre sincère gratitude.

Merci à toutes et à tous.

Table Des Matières

Liste Des Figures	
Liste Des Tableaux	
Liste Des Symboles	
Liste Des Abréviations	
Introduction Générale	15
CHAPITRE 1	17
Généralités Sur Les Robots Parallèles	17
1.1 Introduction	18
1.2 Robots sériels et robot parallèles	18
1.2.1 Robots sériels	18
1.2.1.1 Avantages	19
1.2.1.2 Inconvénients	19
1.2.2 Robots Parallèles.....	19
1.2.2.1 Avantages	20
1.2.2.2 Inconvénients	21
1.2.3 Conclusion	21
1.3 Exemples D'architectures Parallèles.....	21
1.3.1 Plateforme De Stewart	21
1.3.2 Robot Parallèle Plan.....	22
1.3.3 Le MinAngle.....	23
1.3.4 Robot Delta.....	23
CHAPITRE 2	25
ROBOT DELTA	25
2.1 Introduction	26
2.2 Caractéristique Du Robot Delta.....	27
2.3 Constituants Et Description Du Robot Delta.....	28

2.4 Robot Delta ISIR 88.....	30
2.4.1 Introduction	30
2.4.2 Paramètres Géométriques du Delta ISIR 88.....	31
2.4.3 Volume De Travail Du Robot Isir88.....	32
2.4.4 Singularités Du Robot Delta.....	34
2.5 Génération De Trajectoire	36
2.6 Conclusion.....	38
CHAPITRE 3	39
Modélisation Du Robot ISIR 88.....	39
3.1 Introduction	40
3.2 Modélisation Géométrique	40
3.2.1 Validation Du MGD Et MGI.....	42
3.3 Modèle Cinématique.....	45
3.3.1 Modèle Cinématique Direct.....	45
3.3.2 Modèle Cinématique Inverse.....	46
3.3.3 Matrice Jacobienne	46
3.4 Accélération.....	48
3.5 Modèle Dynamique.....	48
3.5.1 Introduction	48
3.5.2 Hypothèses Simplificatrices	49
3.5.3 Paramètres Dynamiques	50
3.5.4 Modèle Dynamique Base Sur Le Principe Du Travail Virtuel.....	51
3.5.5 Conclusion.....	53
CHAPITRE 4	54
Commande PD classique et d'ordre fractionnaire du Robot ISIR 88.....	54
4.1 Introduction	55
4.2 Régulateur Proportionnel Dérivé.....	55

4.2.1 Simulation Avec PD.....	56
4.3 Régulateur D'ordre Fractionnaire.....	57
4.3.1 Introduction.....	57
4.3.2 Généralité Sur Le calcul Fractionnaire :.....	58
4.3.3 Approche De Synthèse D'un Régulateur Proportionnel Dérivé D'ordre Fractionnaire D'un Système De Second Ordre.....	60
4.3.4 Calcul Des Paramètres Du Régulateur PDOF.....	63
4.4 Critère Des Variations De Performances.....	68
4.5 Comparaison Entre Le Pd Classique Et Le Pd D'ordre Fractionnaire.....	69
4.6 Commande Pd Et PDOF Par Algorithme D'optimisation PSO.....	72
4.6.1 Introduction.....	72
4.6.2 Optimisation Par Essaim De Particules.....	72
4.6.3 Application Du PSO Pour Le PD Et Le PDOF.....	73
4.6.4 Comparaison Entre Le PD Classique Et Le PD D'ordre Fractionnaire Calculés Par PSO.....	76
4.7 Commande Par Mode De Glissement.....	78
4.7.1 Introduction.....	78
4.7.2 Commande Par Mode Glissement D'ordre Entier.....	79
4.7.2.1 Elaboration De La Commande uj	79
4.7.2.2 Analyse De La Stabilité.....	80
4.7.3 Commande Par Mode Glissement D'ordre Non Entier.....	81
4.7.3.1 Elaboration De La Commande uj D'ordre Fractionnaire.....	81
4.7.3.2 Analyse De La Stabilité.....	81
4.7.4 Comparaison Entre Le Pd Et Le Pd D'ordre Fractionnaire Calculés Par Mode Glissement.....	82
4.7.4.1 Test De Robustesse.....	86
4.7.5 Conclusion.....	87
Conclusion Générale.....	89

Table Des Matières

Référence Bibliographique91

Liste Des Tableaux

Tableau 2.1 Longueurs paramétriques	33
Tableau 2.2 Angles caractéristiques	33
Tableau 2.3 Paramètres du volume de travail de l'ISIR 88	35

Liste Des Figures

Figure 1.1 – Photo et graphe d’agencement du robot sériel SCARA	18
Figure 1.2 – Robot dans l’industrie automobile	19
Figure 1.3 – Robot dessinateur	19
Figure 1.4 – Photo et graphe d’agencement du robot parallèle Mitsubishi	20
Figure 1.5 – Exemple d’une structure parallèle [4]	20
Figure 1.6 – Plateforme de Stewart	22
Figure 1.7 – Simulateur de vol : Application de la plateforme de Stewart	22
Figure I.8 – Robot parallèle plan à articulations prismatiques [6]	23
Figure 1.9 – Robot MinAngle	23
Figure 1.10 – Robot Delta d’ABB	24
Figure 2.1 – Raymond Claver, inventeur du robot Delta	26
Figure 2.2 – Le Flexpicker de ABB	27
Figure 2.3 – Delta M-1 de FANUC	27
Figure 2.4 – Chaîne d’assemblage de base de micro delta	27
Figure 2.5 – Schéma descriptif des composants du robot Delta [6]	29
Figure 2.6 – Robot Delta ISIR 88 et son unité de commande [6]	30
Figure 2.7 – Aperçu détaillée de la courroie	31
Figure 2.8 – Définition des différents paramètres d’une chaîne cinématique	31
Figure 2.9 – Volume de travail du Robot Delta [21]	33
Figure 2.10 – Les singularités du robot Delta	35
Figure 2.11 – Trajectoire parcourue $Z(t)$ en fonction de $X(t)$	37
Figure 2.12 – Trajectoire de référence prise-dépose	38
Figure 3.1 – Modèle géométriques	40

Figure 3.2 – Géométrie du Delta [1]	41
Figure 3.3 – Vue de dessus de la base du Delta [6]	41
Figure 3.4 – Paramètres géométriques d'un Delta actionnée	42
Figure 3.5 – Schéma de principe pour la validation du MGD et MGI	43
Figure 3.6 – Trajectoire des coordonnées cartésiennes désirées	43
Figure 3.7 – Courbes des angles q_1 , q_2 et q_3	44
Figure 3.8 – Courbes des coordonnées x, y et z obtenues	44
Figure 3.9 – MDD et MGD	53
Figure 4.1 – Structure de commande en boucle fermée	56
Figure 4.2 – Trajectoire effectuée, erreurs de poursuite articulaires et commandes en couple dans le cas d'un régulateur PD	57
Figure 4.3 – Structure du régulateur PD d'ordre fractionnaire [24]	60
Figure 4.4 – Courbes des deux équations de kd en fonction de μ [17]	63
Figure 4.5 – Réponses du modèle non linéaire et modèle approximé	64
Figure 4.6 – Courbes de Kd en fonction de μ	65
Figure 4.7 – Tracé de Bode du système avec régulateur en BO	66
Figure 4.8 – Trajectoire effectuée, erreurs de poursuite articulaires et commandes en couple dans le cas d'un régulateur PDOF(FOMCON)	67
Figure 4.9 – Erreurs de poursuite articulaires et commandes en couple dans le cas d'un régulateur PDOF avec l'approximation d'Oustaloup	68
Figure 4.10 – RMS des erreurs, RMS des couples et MAXAE des erreurs (FOMCON)	70
Figure 4.11 – RMS des erreurs, RMS des couples et MAXAE des erreurs (Oustaloup) ...	71
Figure 4.12 – Schéma représentant l'optimisation par essaim de particules	73
Figure 4.13 – Erreur et couple du PD avec PSO	74

Figure 4.14 – Erreur et couple du PDOF avec PSO	75
Figure 4.15 – RMS des erreurs, RMS des couples et MAXAE des erreurs	77
Figure 4.16 – Structure de la commande équivalente [20]	79
Figure 4.17 – La commande et l’erreur pour la surface PD avec la fonction signe	82
Figure 4.18 – La commande et l’erreur pour la surface PDOF avec la fonction signe	83
Figure 4.19 – La commande et l’erreur pour la surface PD avec la fonction proposée dans [23]	84
Figure 4.20 – La commande et l’erreur pour la surface PDOF avec la fonction proposée dans [23]	85
Figure 4.21 – RMS des erreurs, RMS des couples et MAXAE des erreurs	87

Liste Des Symboles

q_i	Coordonnées articulaires.
τ	Constante de temps
ω_{cg}	Fréquence de coupure
φ_m	Marge de phase
μ	Ordre de l'opérateur fractionnaire
t	Temps en secondes
ξ	Nombre petit
δ	Constantes
${}_aD_t^\mu$	Opérateur fractionnaire

Liste Des Abréviations

BF	Boucle Fermée
BO	Boucle Ouverte
CRONE	Commande Robuste d'Ordre Non Entier
ENP	Ecole Nationale Polytechnique
EPFL	Ecole Polytechnique Fédérale de Lausanne
LCP	Laboratoire de commande de processus
MAXAE	Maximum Absolute Error
MCD	Modèle Cinématique Directe
MCI	Modèle Cinématique Inverse
MDD	Modèle Dynamique Direct
MDI	Modèle Dynamique Inverse
MGD	Modèle Géométrique Direct
MGI	Modèle Géométrique Inverse
PD	Proportionnel Dérivée
PDF	Proportionnel Dérivée d'ordre Fractionnaire
PFE	Projet de Fin d'Etude
PID	Proportionnel Intégrale Dérivée
PSO	Particle Swarm Optimization
RMS	Root Mean Square
MG	Mode glissant

Introduction Générale

Durant ces dernières années, l'intérêt porté aux robots en général et aux bras manipulateurs en particulier ne cesse d'augmenter, le domaine de la robotique a connu des avancées fulgurantes aussi bien dans le domaine de la recherche scientifique qu'en applications industrielles.

Le terme Robot dérive du mot tchèque << Robota >> qui veut dire travail forcé ou asservissement. La robotique est l'ensemble des techniques permettant la conception et la réalisation de robots, elle touche différents domaines d'application tel que le domaine industriel, domestique, médical, militaire ...etc

Dans le cadre de notre projet de fin d'étude et pour les raisons qu'on vient d'évoquer, nous avons choisi de travailler sur le robot ISIR88 à 3 degrés de liberté conçu par l'équipe de Clavel de l'EPFL et disponible au niveau du Laboratoire de Commande des Processus de l'Ecole Nationale Polytechnique. Nous allons, présenté dans le premier et le deuxième chapitre les robots parallèles en générale et le robot ISIR88 en particulier. Ce robot est très utilisé en industrie pour les multiples qualités dont il jouit et qui sont incontestablement très convoitées par les industriels grâce à sa rapidité et sa haute précision. On note aussi que ce robot est très utilisé dans divers applications comme, par exemple, le conditionnement des produits, la télé-chirurgie, l'agriculture ... etc.

Dans le troisième chapitre, nous allons présenter une modélisation du robot basée sur le modèle de Coudrey [1], qui présente le modèle géométrique, donnant la relation entre les angles articulaires et les coordonnées cartésiennes, le modèle cinématique, qui donne la relation entre les vitesses et le modèle dynamique, qui représente l'équation entre la commande (couple) et les coordonnées articulaires, ainsi que leurs inverses. Nous avons constaté que ce modèle est assez complexe, ce qui le rend difficile à manipuler analytiquement.

Le dernier chapitre est consacré à la synthèse de lois de commande et leur implémentation sur le modèle du robot. Ainsi, plusieurs approches sont proposées allant du régulateur PD et modes glissant classiques jusqu'à leurs versions d'ordre fractionnaire en introduisant préalablement la notion du calcul fractionnaire. Les paramètres de différentes commandes ont été optimisés en utilisant les algorithmes PSO. Des comparaisons en termes de performances

dynamiques entre les approches classiques d'ordre entier et celles d'ordre fractionnaires sont présentées.

CHAPITRE 1

Généralités Sur Les Robots Parallèles

1.1 Introduction

Durant ces dernières années, le secteur de la robotique ne cesse de gagner en développement surtout grâce à l'apparition des robots parallèles. Ces derniers sont une catégorie de robots présentant des chaînes cinématiques fermées avec un champ d'architectures cinématiques très vaste. Cette structure présente un grand nombre d'avantages comparée à celle des robots séries, ce qui lui permet d'avoir diverse domaines d'application qui s'étendent de la micromanipulation à la manipulation de charges élevés.

Les performances des robots parallèles sont complémentaires et généralement à l'opposé de celles des robots sériels. Ils ouvrent donc de nouvelles pistes pour la robotisation de certaines tâches. Ils représentent aujourd'hui une faible part du marché pour les robots industriels commercialisés, mais ils s'imposent naturellement pour certaines applications. Pour d'autres, ils sont en concurrence avec leurs homologues sériels.

Dans ce chapitre, nous allons présenter les robots parallèles et les robots sériels en faisant une comparaison entre ces deux types de robots en évoquant leurs avantages et leurs inconvénients, puis nous donnerons des exemples sur les robots parallèles présents dans l'industrie.

1.2 Robots Sériels et Robot Parallèles

1.2.1 Robots Sériels

C'est une catégorie de robot qui est structurée sous forme d'une chaîne cinématique ouverte qui comporte plusieurs corps reliés entre eux à l'aide d'une articulation à un seul degré de liberté, chaque articulation est munie d'un actionneur (linéaire ou rotatif).

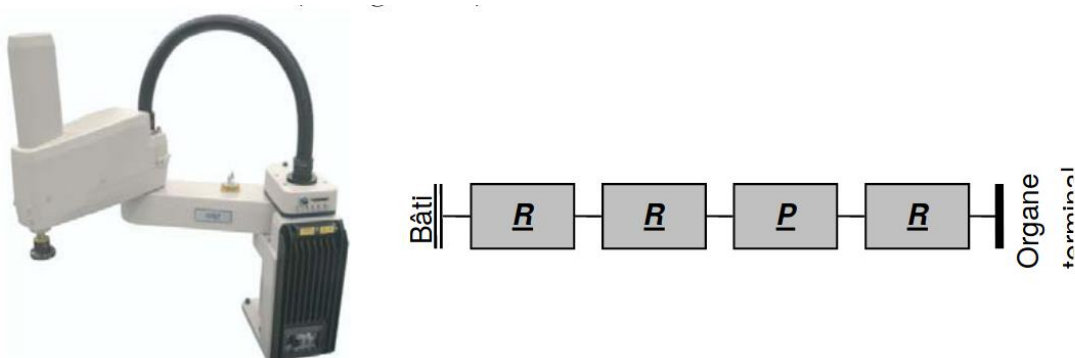


Figure 1.1 – Photo et graphe d'agencement du robot sériel SCARA.

Ces robots touchent plusieurs domaines d'application, ils sont utilisés pour le soudage et l'assemblage dans l'industrie surtout dans le secteur de l'automobile. Ils peuvent aussi

accomplir des tâches de loisir comme produire un texte ou dessiner en imitant les gestes d'un humain [4].

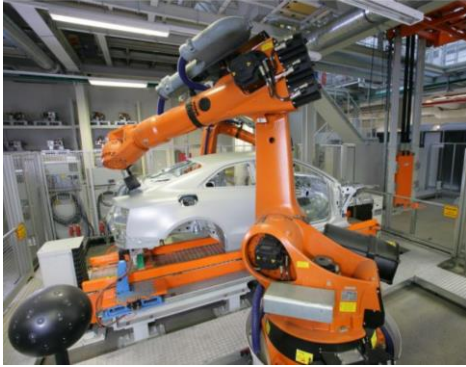


Figure 1.2 – robot dans l'industrie automobile

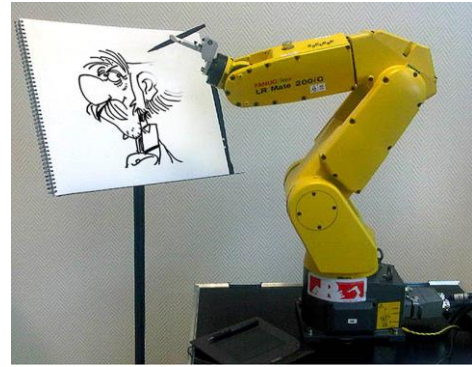


Figure 1.3 – robot dessinateur

1.2.1.1 Avantages

- La structure de chaîne ouverte lui permet d'avoir un grand volume de travail.
- La modélisation et la commande du robot est simple.

1.2.1.2 Inconvénients

La disposition en série des éléments implique que chaque corps doit supporter sa masse ainsi que celle du corps suivant cela entraîne un manque de rigidité.

Des erreurs de position peuvent être engendrées à cause de l'inertie élevée qui est due aux masses réparties sur toute la chaîne cinématique (actionneurs et organes de transmission). Chaque articulation a une erreur de positionnement et en démarrant de la base jusqu'à l'organe terminal on aura un cumul d'erreurs en fonction du nombre d'articulations.

Le robot sériel comporte des avantages intéressantes mais ses inconvénients sont à ne pas négliger, c'est pour cela que les chercheurs en robotique ont réfléchi à un moyen de contourner ces inconvénients d'où l'apparition d'une nouvelle structure qui est la structure parallèles.

1.2.2 Robots Parallèles

Comme mentionné précédemment, on retrouve dans ce type de robots une structure de chaînes cinématiques fermées qui lient entre une plateforme fixe (base du robot) à une plaque mobile (nacelle).

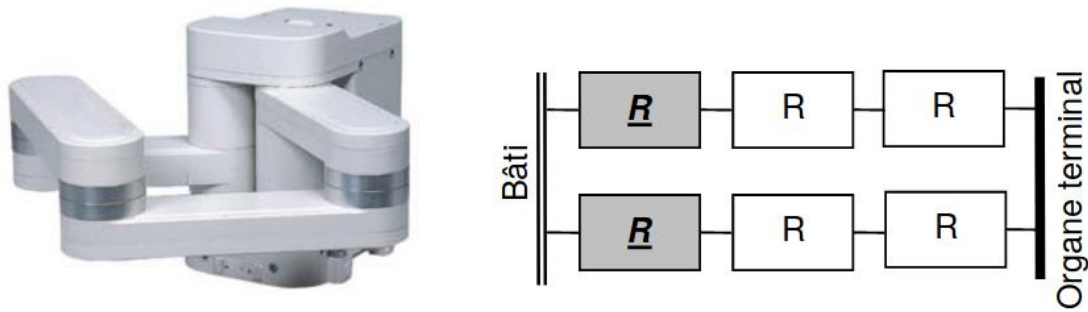


Figure 1.4 – Photo et graphe d’agencement du robot parallèle Mitsubishi.

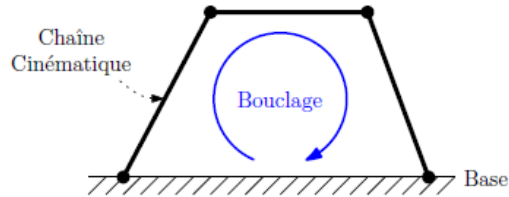


Figure 1.5 – Exemple d’une structure parallèle [4]

Ils sont utilisés dans le domaine industriel depuis les années 1990. Ils représentent aujourd’hui une faible part de marché pour les robots industriels commercialisés, mais ils s’imposent naturellement pour certaines applications. Ils sont utilisés pour des applications industrielles classiques et surtout pour des opérations de prise-dépose a haute cadence.

1.2.2.1 Avantages

Un avantage principal du robot est de fixer les actionneurs et les capteurs sur la base fixe du robot ce qui permet la diminution de la masse mobile du robot contrairement au robot sériel [1].

Pour un poids équivalent à celui du robot sériel, le robot parallèle peut soulever des charges plus importantes.

La rigidité mécanique est très bonne ce qui confère à la structure parallèle une bonne précision [1].

Ces robots sont capables d’atteindre des dynamiques élevées et des accélérations importantes avec de très faibles couples engendrés

Le processus de fabrication est facile car le robot parallèle présente une configuration symétrique des chaînes cinématiques.

1.2.2.2 Inconvénients

L'espace de travail est relativement réduit à cause de sa structure fermée du robot.

La modélisation et la commande est moins facile que celle du robot sériel car le système est généralement complexe.

Le passage par un point singulier entraîne l'instabilité du robot (système hautement non linéaire)

1.2.3 Conclusion

Les robots séries et les robots parallèles présentent des configurations différentes (voir opposées), chacun d'entre eux bénéficie de plusieurs avantages et souffre de plusieurs inconvénients donc on ne pourra pas trancher entre ces deux derniers. Le choix d'utilisation reste en fonction du cahier de charge et des performances souhaitées.

1.3 Exemples d'Architectures Parallèles

Il existe plusieurs structures possibles pour les robots parallèles.

1.3.1 Plateforme de Stewart

La plateforme de Stewart, aussi connue sous le nom de positionneur hexapode, est un type de robot parallèle à six degrés de liberté : les trois coordonnées de translation mais aussi les angles de tangage, roulis et lacet. Les six jambes sont actionnées pour changer de longueur et fluctuer l'orientation de la plateforme. A une position donnée du plateau supérieur correspond six longueurs uniques des jambes.

Son application la plus connue est le simulateur de vol. La raison de cette utilisation est la masse élevée de la cabine à laquelle on doit faire subir des mouvements selon les six degrés de liberté (3 translations et 3 rotations).

Les figures suivantes nous donnent une idée de cette structure.



Figure 1.6 – Plateforme de Stewart.



Figure 1.7 – Simulateur de vol : Application de la plateforme de Stewart

1.3.2 Robot Parallèle Plan

On retrouve plusieurs configurations pour cette structure, c'est un robot à 3 degrés de liberté vu qu'il nous donne un positionnement plan plus une rotation. Le mouvement des bras est assuré par l'intermédiaire d'actionneurs linéaires ou rotatifs.

Exemple d'application : positionnement précis de pièces pour effectuer des tâches de perçage.

La structure du robot est présentée dans la figure 1.8.

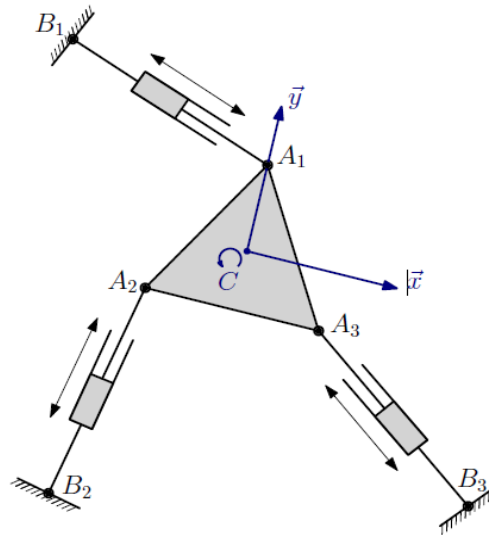


Figure I.8 – Robot parallèle plan à articulations prismatiques [6].

1.3.3 Le MinAngle

La structure du MinAngle est constituée de trois chaînes cinématiques identiques flexibles. Grâce à elles, l'effecteur peut faire des déplacements suivant les axes X, Y et Z, c'est donc un robot parallèle a trois degrés de liberté.

La figure 1.9 nous donne un aperçu de ce robot.

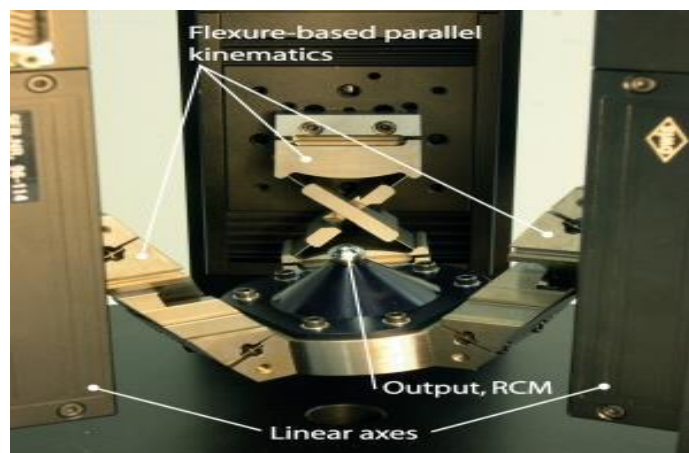


Figure 1.9 – Robot MinAngle.

1.3.4 Robot Delta

Le robot Delta est un robot à trois degrés de libertés possédant trois chaînes cinématiques indépendantes identiques. La caractéristique essentielle qui est identique pour tous les robots

Delta est l'utilisation de barres parallèles. Ces barres sont fixées entre elles à une petite barre de maintien par des liaisons rotules (voir figure 1.10). C'est cette particularité qui permet à la nacelle de rester parallèle à la base et d'avoir ainsi seulement des mouvements de translation.



Figure 1.10 – Robot Delta d'ABB

Ce modèle figure parmi les robots Deltas les plus utilisés dans l'industrie.

CHAPITRE 2

ROBOT DELTA

2.1 Introduction

Dans ce chapitre, nous allons nous intéresser à une des structures parallèles citées dans le chapitre précédent qui est celle du Robot Delta.

En 1985, le robot delta fut inventé et breveté par une équipe de recherche dirigée par Raymond Clavel (Figure 2.1) professeur à l'Ecole polytechnique fédérale de Lausanne (EPFL), Laboratoire de systèmes robotiques (LRSO), il marqua le début de l'ère des applications de la robotique parallèle légère.

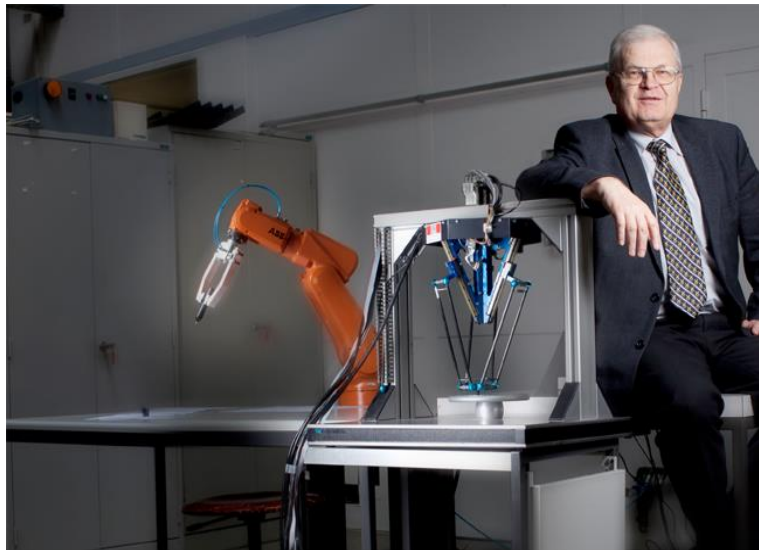


Figure 2.1 – Raymond Clavel, inventeur du robot Delta

Une licence pour la construction de tels robots a été accordée à Demarex, mais suite à des restructurations commerciales, d'autres compagnies ont obtenu le droit de construction des robots deltas, plus de 10.000 exemplaires ont été vendus depuis.

Aujourd'hui, le brevet principal du robot delta est expiré et depuis plusieurs années, le robot est commercialisé par plusieurs compagnies sous différents noms. Bosch Packaging Technology le produit sous les noms Presto, Paloma, Delfi et Astor, ABB sous le nom de Flexpicker (figure 2.2) et FANUC sous le nom M-1 (figure 2.3).

De même, le Centre Suisse d'Electronique et de Microtechnique propose deux modèles de micro-robot Delta qui sont utilisés généralement pour l'industrie horlogère, l'assemblage de microsystèmes (capteurs, moteurs) et de composants montés en surface (CMS) (figure 2.4).

Grâce à sa rapidité et sa précision, le domaine d'application du robot Delta reste très vaste, il est utilisé dans des opérations d'assemblage de haute précision des composants électroniques. On le retrouve également dans le domaine médical notamment pour la chirurgie (figure 2.5) et la rééducation.



Figure 2.2 – Le Flexpicker de ABB



Figure 2.3 – Delta M-1 de FANUC



Figure 2.4 – Chaîne d'assemblage de base de micro Delta

2.2 Caractéristique du Robot Delta

Comme on l'a mentionné dans le chapitre 1, le robot delta fait partie de la famille des robots parallèles, on constate quatre caractéristiques principales qui le différencient des robots sériels :

- La plaque de base du robot est reliée à l'organe terminal par le biais de trois chaînes cinématiques identiques (pour le Delta conventionnel), grâce à cette configuration la rigidité

et la précision du robot sont fortement augmentées mais elle entraîne une diminution de l'espace de travail.

-Le robot delta est muni de trois degrés de liberté de translation (XYZ) plus un quatrième degré de liberté de rotation (thêta), ce dernier est négligé car il est assuré par un mécanisme indépendant. On s'intéresse qu'aux mouvements de translation grâce à la faculté de la nacelle (plaque mobile) à garder constamment une configuration horizontale parallèle à la base du robot et ce, de manière passive, sans aucun asservissement ou control (propriété obtenue par construction mécanique du robot).

- La nacelle effectue uniquement des mouvements de translation dans le repère XYZ grâce aux barres parallèles qui constituent un parallélogramme.

- Les moteurs chargés d'actionner le robot sont fixés aux niveaux des extrémités de la plaque de base du robot, ce qui a pour effet de diminuer considérablement les masses mobiles du robot et donc leurs inerties, permettant ainsi de grandes vitesses et accélérations (l'accélération peut atteindre 50 G) dans l'utilisation de ce robot.

Les articulations rotoïdes des bras tournants sont actionnées de deux façons différentes, la première manière se fait grâce à l'utilisation de moteurs rotatifs (servomoteur CC ou CA) fixés sur les extrémités de la base du robot, quand à la deuxième, elle est réalisée grâce à l'utilisation d'actionneurs linéaires (vérins pneumatiques ou hydrauliques) fixés directement aux barres parallèles.

De ce fait, on distingue deux types de robots Delta : le Delta linéaire et le Delta rotatif.

2.3 Constituants et Description du Robot Delta

Dans ce projet, notre travail s'est focalisé sur le robot Delta à actionneur rotatif Etant donné sa forte utilisation par rapport aux autres Delta dans le milieu industriel.

La figure 2.5 nous donne un aperçu du schéma descriptif des composants du robot Delta à actionneur rotatif.

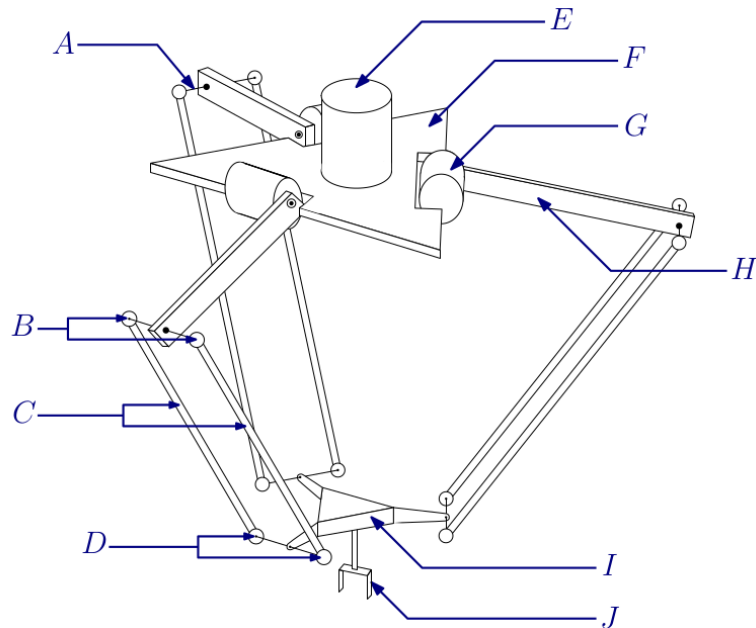


Figure 2.5 – schéma descriptif des composants du robot Delta [6].

Avec :

A : Barre de liaison entre le bras (H) et une paire de barres parallèles (C).

B : Rotules supérieures.

C : Barres parallèles. Ces deux barres forment, avec deux barres de Liaison (A), un parallélogramme mobile, grâce aux rotules disposées aux quatre angles de ce parallélogramme. Les barres parallèles du Delta permettent de mouvoir la nacelle en translation seulement (selon les trois axes spatiaux), les différentes rotations sont éliminées par l'effet de la configuration mécanique du robot (de manière passive).

D : Rotules inférieures.

E : Connexion de la base au battis fixe (à titre indicatif).

F : Base du robot Delta sur laquelle sont fixées les trois chaînes cinématiques (bouclées) du robot.

G : Actionneur d'une chaîne cinématique (articulation active). Cet actionneur constitue le seul élément du robot accessible directement par une commande ou pour des mesures (par le biais de son instrumentation). Celui-ci est fixé sur la base et actionne un seul bras (H).

H : Bras d'une chaîne cinématique du Delta. Transmet le mouvement du moteur (G) vers la nacelle (I) à travers les barres parallèles (A).

I : Nacelle du robot Delta. La nacelle est portée par trois chaînes cinématiques identiques. La nacelle constitue un support pour l'organe terminal (J).

J : Organe terminal du robot Delta (Pince, . . .).

2.4 Robot Delta ISIR 88

2.4.1 Introduction

Dans la suite de ce projet, nous allons consacrer notre travail sur le Robot Delta ISIR 88, il est disponible au niveau du Laboratoire de Commande des Processus (LCP) de l'Ecole Nationale Polytechnique (ENP), il est présenté dans la (figure 2.6), on aperçoit que ce dernier est contrôlé par l'intermédiaire d'un système de commande numérique.



Figure 2.6 – Robot Delta ISIR 88 et son unité de commande [6]

Ce robot fait partie de la classe des Deltas à actionneurs rotatifs, ses entraînements sont des moteurs à courant continu Brushless identiques qui font office d'actionneurs pour ses bras, ces moteurs ont la particularité d'être munis d'un réducteur à courroie dentée (figure 2.7) ce qui implique l'introduction d'un coefficient de réduction (égale à 12) car la transmission entre le moteur et le bras du robot n'est pas directe.

L'équation du couple au niveau du bras devient :

Couple au niveau du bras = $12 \times$ couple au niveau du moteur.

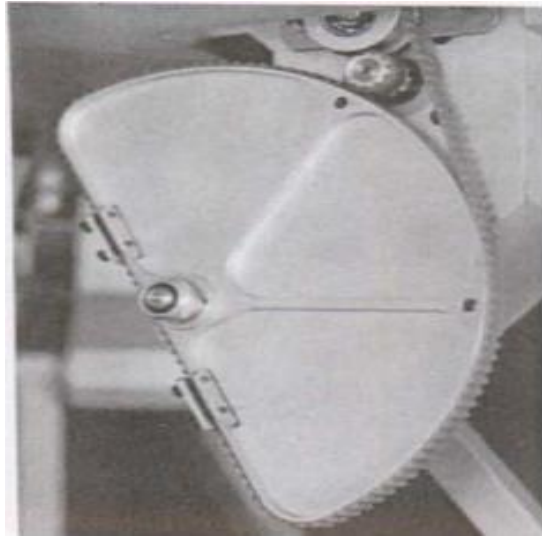


Figure 2.7 – Aperçu détaillée de la courroie

2.4.2 Paramètres Géométriques du Delta ISIR 88

La figure 2.8 représente l'une des manières de paramétrage d'une chaîne cinématique parmi les trois chaînes du Robot Delta 88. Cette approche se base sur les angles et les longueurs reliant la nacelle et l'origine fixe avec les points intermédiaires C_i (les centres des articulations passives).

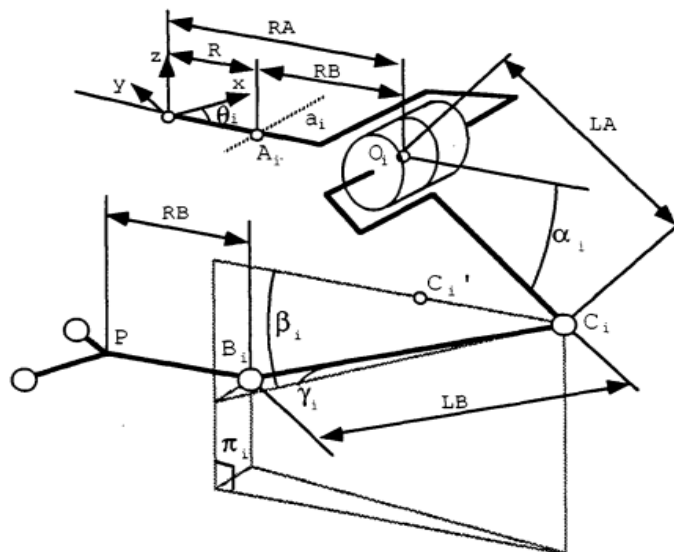


Figure 2.8 – Définition des différents paramètres d'une chaîne cinématique.

Le tableau 2.1 nous donne une description et les valeurs des longueurs paramétriques du ISIR88

Paramètre	Description	Valeur
L_A	Longueur du bras	200 mm
L_B	Longueur d'une barre parallèle	380 mm
R_A	La distance entre le centre de la base fixe et l'axe de rotation du bras	150 mm
R_B	Distance entre le centre de la nacelle et le coté du parallélogramme.	25 mm
R	$R_A - R_B$	125 mm

Tableau 2.1 Longueurs paramétriques

Le tableau 2.2 nous donne une description et les valeurs des angles caractéristiques du ISIR88.

Paramètre	Description
α_i	Angle entre le $i^{\text{ème}}$ bras et le plan de la base fixe, il sera considéré positif lorsque le bras est situé du côté de la nacelle
β_i	Angle entre le plan du parallélogramme i et le plan horizontal, mesuré dans un plan vertical π_i contenant le bras i
γ_i	Angle entre le plan vertical π_i et une des barres du parallélogramme i .
θ_i	Angle entre le plan π_i et le plan Oxz

Tableau 2.2 Angles caractéristiques.

2.4.3 Volume De Travail Du Robot Isir88

Le volume de travail du robot Delta est la portion de l'espace pouvant être atteinte par son organe terminal situé sur sa nacelle. Il est déterminé par l'intersection des volumes atteignables par les trois chaînes cinématiques constituant le Delta prises indépendamment.

La connaissance de l'espace de travail du robot est primordiale. En effet, la trajectoire qu'on désire être suivie par la nacelle doit impérativement appartenir à cet espace.

On prendra en considération que les dimensions des constituants sont finies et qu'ils ont des mouvements relatifs restreints, cela implique que le mouvement de la nacelle est limité à une portion de l'espace. On devra donc imposer certaines limites géométriques à certains mouvements afin d'éviter des ambiguïtés lors du calcul des transformations des coordonnées et les singularités du Delta.

Limite 1 : $-\gamma_{max} \leq \gamma \leq \gamma_{max}$ avec $\gamma_{max} = 40^\circ$

La cause de cette limitation est due au fait que les contraintes de construction des articulations aux deux extrémités des barres parallèles limitent cet angle [21].

Limite 2 : $30^\circ \leq \alpha_i + \beta_i \leq 180^\circ$

Cette limitation est appliquée pour les deux raisons, lorsque la somme des deux angles donne une valeur inférieure à 30° , ça entrainera la présence d'interférences entre le bras et les barres parallèles ainsi qu'entre les barres parallèles et les moteurs des bras. Lorsque la somme des deux angles devient supérieure à 180° ceci donne deux ensembles de consignes articulaires pour un seul point de l'espace opérationnel.

Le volume de travail du IRIS 88 est donné (en c) dans la figure suivante :

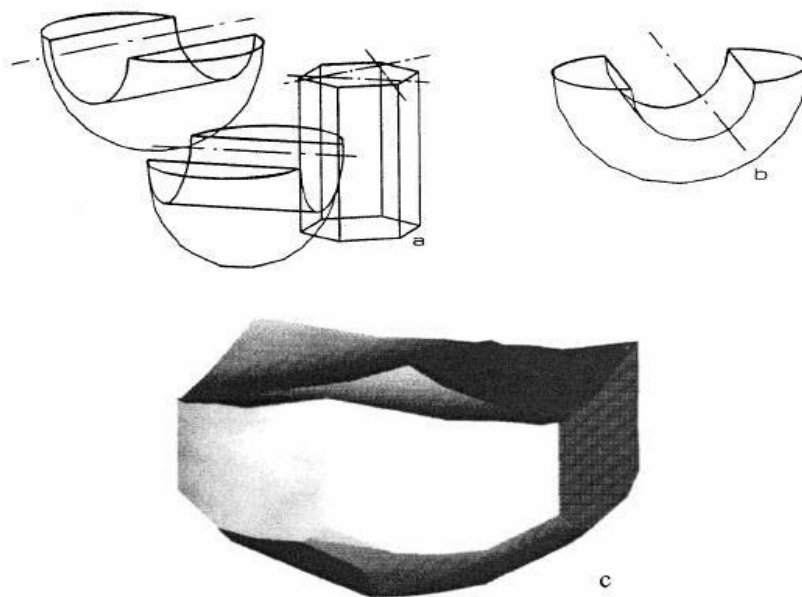


Figure 2.9 – Volume de travail du Robot Delta [21].

Ce dernier est obtenu approximativement par l'intersection du prisme hexagonal droit (en a) avec les trois corps de révolution (b) dont les axes sont indiqués (en a).

Les développements et les justifications pour l'obtention de ce volume de travail sont détaillés dans [21].

Les valeurs paramètres ainsi que leurs définitions du volume de travail du Robot Delta ISIR 88 sont données dans le tableau suivant :

Paramètre	Définition	Valeur (ISIR 88)
D	Diamètre du cylindre inscrit dans le volume.	489 mm
S	La surface de la projection du volume sur un plan horizontal.	20,7 104 mm ²
H	Hauteur du volume.	183 mm
V	Le volume.	37,9 106 mm ³

Tableau 2.3 Paramètres du volume de travail de l'ISIR 88

On constate que le volume de travail est petit comparé à la longueur de la chaîne cinématique (L_A+L_B) qui est de 580 mm environ, c'est dû au fait que la fermeture de la chaîne qui conduit à la restriction des mouvements de la nacelle et par conséquent le volume de travail se restreint.

2.4.4 Singularités du Robot Delta

L'étude des singularités du robot est une étape primordiale, en ce qui concerne les robots séries, les points singuliers résultent d'une perte en degrés de libertés. Contrairement aux robots parallèles où ils résultent du gain en degrés de libertés.

En effet, le passage par un point singulier peut avoir pour conséquence la rotation non désirée de la nacelle ou un mouvement de translation non souhaité, à ce moment, la position de la nacelle n'est plus contrôlable et cela impliquerait que l'efficacité du dispositif de commande sera remise en cause.

On constate quatre configurations singulières du Delta qui sont données dans la figure 2.10 [21].

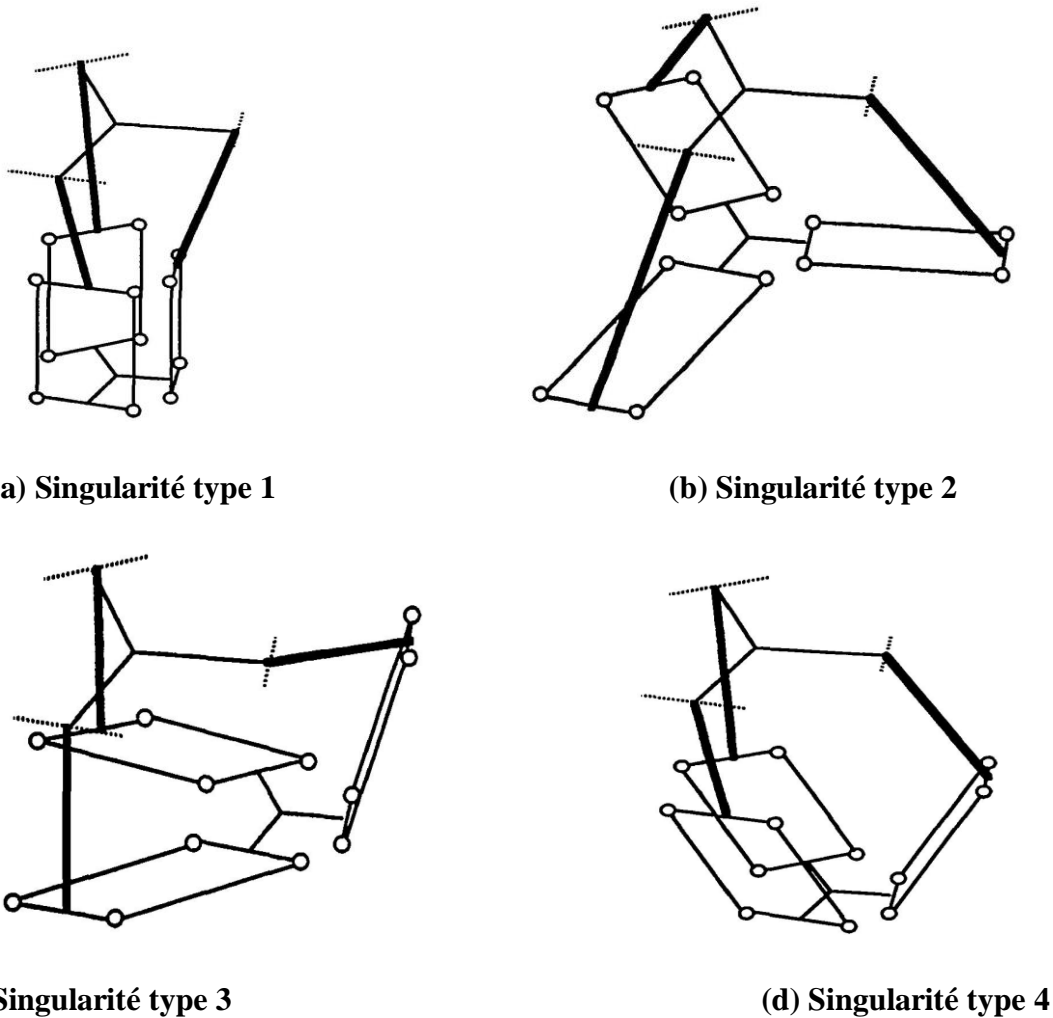


Figure 2.10 – Les singularités du robot Delta

La définition totale de la position de la nacelle et sa stabilité sont les indices de non singularité et cela se traduit par deux conditions [21] :

- Les barres parallèles (les lignes d'action des forces) sont situées dans 3 plans -différents non parallèles
- 2 lignes d'action au maximum sont parallèles.

A partir des deux conditions précédentes, on obtient les quatre configurations singulières montrées dans la figure précédente :

- a) Les trois parallélogrammes sont parallèles (figure 2.10a) ;
- b) Les trois parallélogrammes sont sur le même plan horizontal (figure 2.10b) ;
- c) Deux des trois parallélogrammes ont leurs barres sur le même plan (figure 2.10c) ;

d) Deux des trois parallélogrammes ont leurs barres parallèles (figure 2.10d).

2.5 Génération de Trajectoire

A présent, nous allons nous intéresser au générateur de trajectoire de la nacelle.

Ce dispositif nous permet d'imposer à cette dernière un chemin de référence qui devra impérativement appartenir à l'espace de travail de notre robot défini précédemment, la nacelle devra donc se déplacer selon la trajectoire délivrée par le générateur.

La trajectoire peut être définie sous plusieurs aspects, en effet elle peut soit être définie par une courbe géométrique ou une séquence de points définie soit en coordonnées cartésiennes (x , y et z) ou bien en coordonnées articulaire (α_1 , α_2 et α_3).

Toutefois, cet aspect géométrique est fortement liés au profil temporel, ce dernier concerne l'évolution de l'accélération, de la vitesse et de la position dans le temps, car une erreur géométrique engendrera une erreur temporelle et vice versa.

On précisera aussi que la trajectoire souhaitée peut être sous différents types énoncés ci-dessous :

- Chemin libre entre deux points (sans asservissement de trajectoire).
- Trajectoire entre deux points avec suivi de chemin souhaitée.
- Trajectoire entre deux points en passant par des points intermédiaires reliés entre eux par des chemins libres.
- Trajectoire entre deux points en passant par des points intermédiaires reliés entre eux par des chemins imposés.

Ces trajectoires s'expriment le plus souvent par des fonctions mathématiques à base d'interpolation polynomiale.

Dans notre travail, nous avons choisi de prendre une trajectoire entre deux points reliés entre eux par un chemin sous forme de demi-ellipse pour que le robot Delta effectue la tâche de prise-dépose (Figure 2.11).

En prenant le cas le cas d'un ellipsoïde selon l'axe x , nous avons les équations suivantes pour $0 \leq t \leq t_f$:

$$r(t) = 10 \left(\frac{t}{t_f} \right)^3 - 15 \left(\frac{t}{t_f} \right)^4 + 6 \left(\frac{t}{t_f} \right)^5$$

$$\left\{ \begin{array}{l} X(t) = x_i \pm A(1 - \cos(\frac{\pi r(t)}{D})) \\ Y(t) = y_i \\ Z(t) = z_i \pm B \sin(\frac{\pi r(t)}{D}) \end{array} \right. \quad (2.1)$$

Ou : $D = \pi \sqrt{\frac{1}{2}(A^2 + B^2)}$

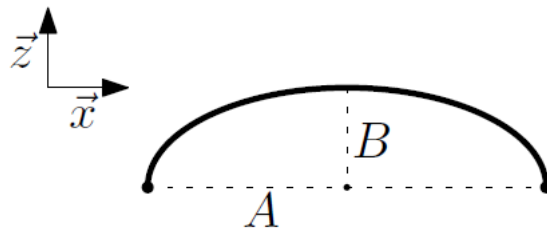


Figure 2.11 – Trajectoire parcourue Z(t) en fonction de X(t)

Ces équations nous permettent d’avoir un libre choix de fixer les paramètres de la loi de mouvement selon :

- La distance du parcours;
- Le temps de parcours de la distance;
- La durée de la phase d’accélération et de vitesse constantes;
- L’accélération et la vitesse maximale.

La génération de trajectoire nous permet donc de prédire le mouvement de la nacelle et de s’assurer que la consigne reste dans l’espace de travail.

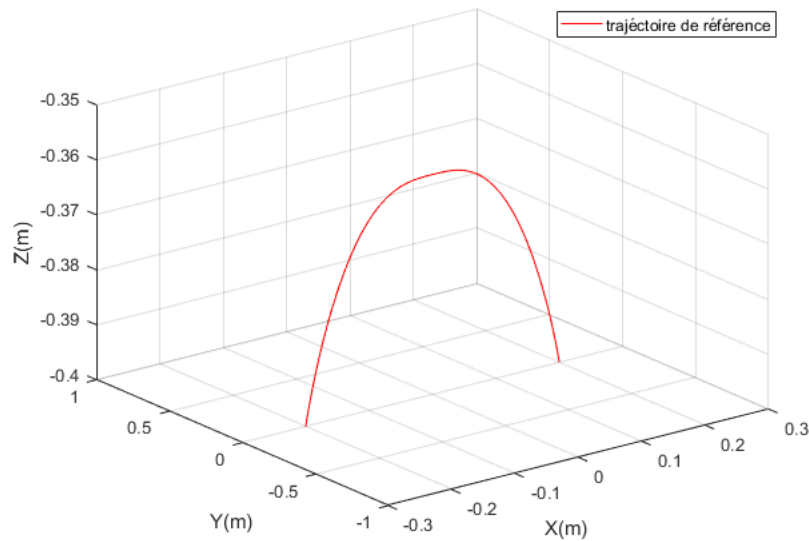


Figure 2.12 – Trajectoire de référence prise-dépose

2.6 Conclusion

Dans ce chapitre nous avons présenté des généralités sur les robots manipulateurs, leurs types et leurs fonctions. Nous avons constaté le besoin des applications industrielles en termes de rapidité et précision.

En second lieu, nous avons introduit le robot Delta en définissant ses caractéristiques et ses composants. Ensuite, nous nous sommes focalisés sur le Delta ISIR 88 qui sera le robot utilisé pour la suite de ce présent projet, nous avons traité sa géométrie, son volume de travail et ses singularités qui sont essentiels pour la modélisation du robot. Pour finir, nous avons défini la génération de trajectoire qui aura une forme ellipsoïdale pour la suite de notre travail et qui sera indispensable pour la commande de notre robot. A présent, nous pouvons passer à la modélisation du Delta ISIR 88.

CHAPITRE 3

Modélisation Du Robot ISIR 88

3.1 Introduction

Après avoir défini les paramètres géométriques du robot Delta ISIR 88, il devient nécessaire d'élaborer un modèle mathématique permettant de décrire le fonctionnement du système de la façon la plus précise possible. Ce modèle sera utilisé aussi bien pour la synthèse des lois de commande que pour la simulation numérique.

Dans ce chapitre, nous allons essayer de modéliser le robot delta afin d'obtenir le modèle géométrique, cinématique et dynamique qui sont d'une importance capitale pour la commande du robot.

3.2 Modélisation Géométrique

Pour le Robot Delta ISIR 88 à trois degrés de liberté et à actionnement rotatif, la modélisation géométrique consiste à trouver des équations liant les coordonnées cartésiennes x , y et z de la nacelle aux trois articulations rotoïdes q_i du robot. Voir La figure 3.1

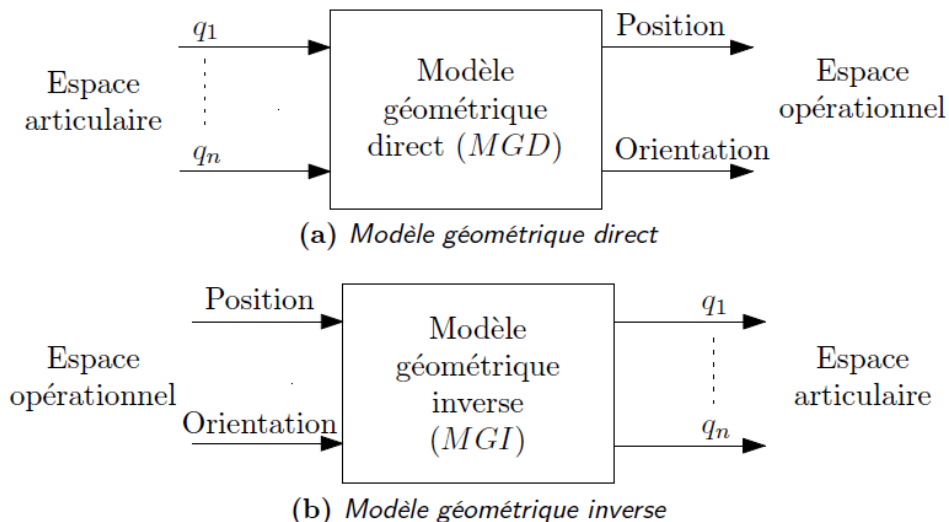


Figure 3.1 – Modèles géométriques.

Le modèle géométrique direct (MGD) nous donne (x, y, z) en fonction de (q_1, q_2, q_3) .

Le modèle géométrique inverse (MGI) nous donne (q_1, q_2, q_3) en fonction de (x, y, z) .

La triple symétrie du robot nous permet de travailler sur une seule chaîne cinématique (les trois chaînes cinématiques sont identiques).

Pour des raisons de simplification, la nacelle est considérée comme un point matériel P en considérant qu'elle adopte uniquement un mouvement de translation grâce à la particularité du robot.

Le référentiel absolu $\{R\}$ est choisi comme montré sur la figure 3.2, c'est-à-dire au centre du triangle dessiné par les axes des trois moteurs, l'axe z vers le haut, et l'axe x étant perpendiculaire à l'axe du moteur 1. L'indice i ($i = 1, 2, 3$) est utilisé pour identifier le numéro du bras (chaîne). Chaque bras est séparé par un angle de 120° . un repère $\{R_i\}$ est associé à un bras i correspondant qui est situé au même endroit que $\{R\}$ mais tourné autour de (oz) d'un angle $\varphi_i = 0^\circ, 120^\circ, 240^\circ$, pour les bras 1, 2 et 3 respectivement comme montré dans la figure 3.3 [1]

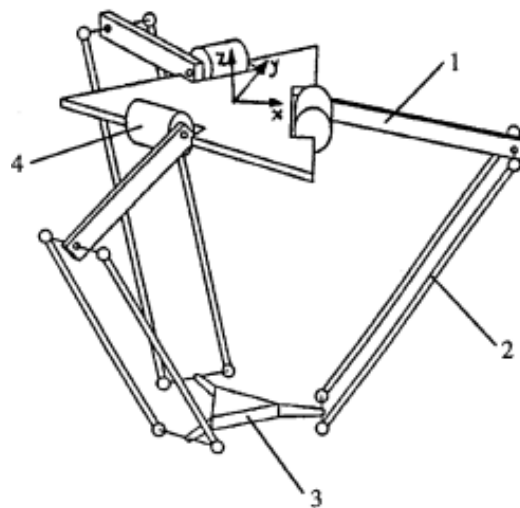


Figure 3.2 – Géométrie du Delta [1].

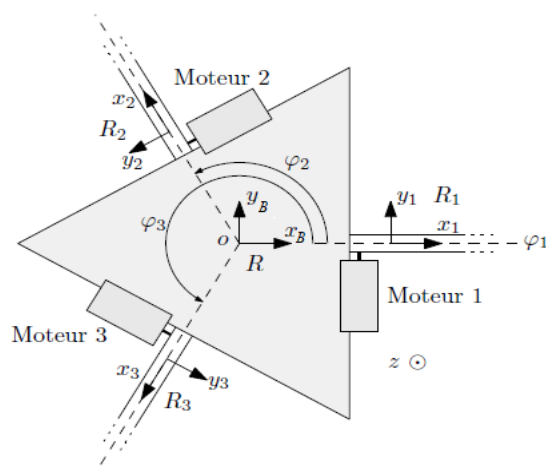


Figure 3.3 – Vue de dessus de la base du Delta [6].

La transformation entre le repère $[R_i]$ et le repère $[R_0]$ est donnée par la matrice de rotation suivante :

$$R_i^R = \begin{bmatrix} \cos(\varphi_i) & -\sin(\varphi_i) & 0 \\ \sin(\varphi_i) & \cos(\varphi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

Comme mentionné précédemment, la nacelle ne peut effectuer que des mouvements de translation donc elle garde à chaque déplacement la même orientation que le repère $[R]$. Ceci nous permet de considérer que la distance entre le repère absolu $[R]$ et le moteur est

$$R = RA - RB.$$

et $P = B_1 = B_2 = B_3$. Voir la figure 3.4.

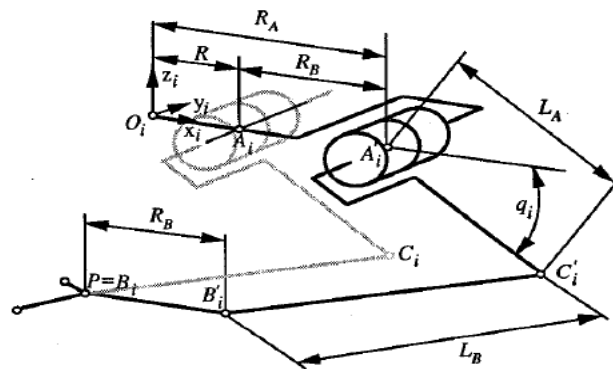


Figure 3.4 – Paramètres géométriques d'un Delta actionnée.

Ces approximations nous donnent un modèle géométrique approximatif de celui du système réel sans trop affecter les résultats.

Grace à ces définitions, on pourra établir le modèle géométrique direct et inverse comme proposé dans [21] et [3]. Cependant, dans les deux modèles proposés, il faudra prendre en considération les singularités mathématiques de type « 0/0 » qui apparaissent dans le modèle géométrique inverse, ce qui peut entraîner à une perte de contrôle du robot pour des faibles valeurs de vitesses. Afin d'éliminer ces singularités un nouveau modèle est proposé dans [4].

3.2.1 Validation du MGD et MGI

Le robot Delta est généralement utilisé en industrie pour des opérations de prise-dépose c'est pour cela qu'on a utilisé un générateur de trajectoire pour la validation du MGD et MGI.

On donne à l'entrée du MGI une trajectoire (figure 3.6) pour chaque coordonnée cartésienne (x , y et z). On obtient à la sortie les valeurs des angles q_1 , q_2 et q_3 en radian (figure 3.7).

Pour la validation du MGD on donne à l'entrée les mêmes valeurs des angles obtenus et on compare les courbes des coordonnées cartésiennes obtenues à la sortie du MGD (Figure 3.8) avec la trajectoire donnée initialement. La validation du MGD et MGI a été établie sous Simulink (Figure 3.5).

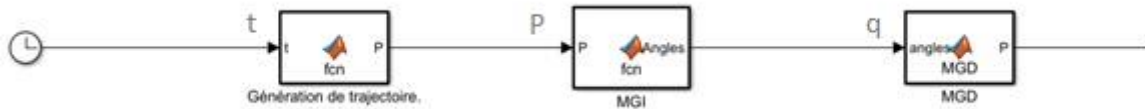


Figure 3.5 – Schéma de principe pour la validation du MGD et MGI

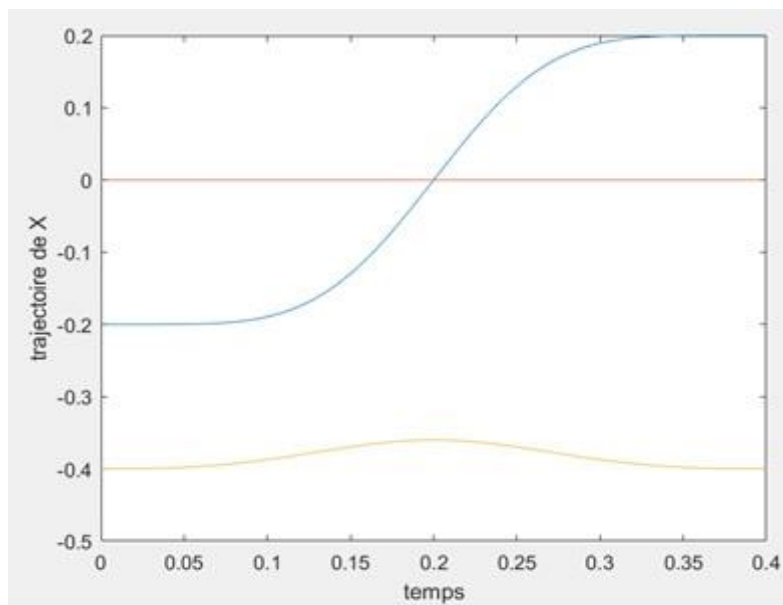


Figure 3.6 – trajectoire des coordonnées cartésiennes désirées

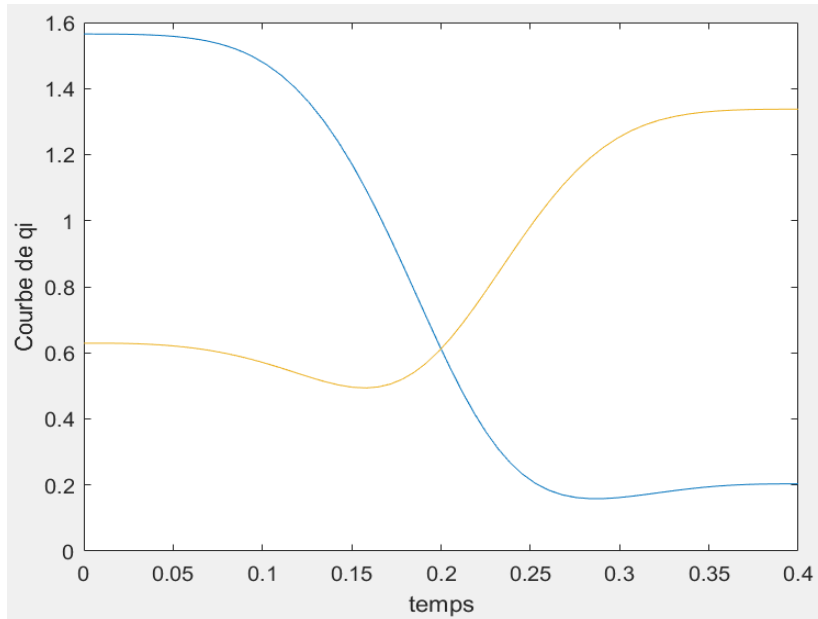


Figure 3.7 – Courbes des angles q_1 , q_2 et q_3

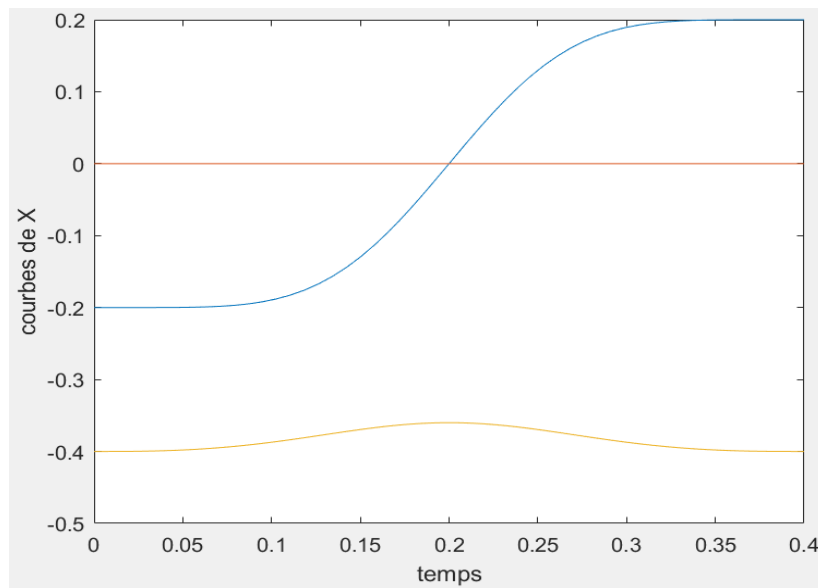


Figure 3.8 – Courbes des coordonnées x , y et z obtenues

On remarque que la trajectoire donnée initialement est identiques aux courbes des x , y et z obtenue à la sortie du MGD donc les modèles géométriques direct et inverse que nous avons établis sont validés.

3.3 Modèle Cinématique

La détermination du modèle cinématique consiste à trouver la matrice jacobienne J qui relie les vitesses opérationnelles $(\dot{x}, \dot{y}, \dot{z})$ avec les vitesses articulaires $(\dot{q}_1, \dot{q}_2, \dot{q}_3)$.tel que :

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

Pour les robots séries, une approche systématique peut être utilisée pour trouver la matrice jacobienne. Malheureusement, pour les robots parallèles, la détermination de cette matrice est plus difficile.

Pour le cas du robot Delta, Coudrey a pu déterminer cette matrix grâce au calcul numérique des dérivées partielles du modèle géométrique direct [1].

3.3.1 Modèle Cinématique Direct

Le modèle cinématique direct offre le passage des vitesses de la nacelle \dot{x}, \dot{y} et \dot{z} vers les vitesses articulaires \dot{q}_1, \dot{q}_2 et \dot{q}_3 .

$$\dot{x} = \begin{bmatrix} \frac{\partial f_x}{\partial q_1} & \frac{\partial f_x}{\partial q_2} & \frac{\partial f_x}{\partial q_3} \\ \frac{\partial f_y}{\partial q_1} & \frac{\partial f_y}{\partial q_2} & \frac{\partial f_y}{\partial q_3} \\ \frac{\partial f_z}{\partial q_1} & \frac{\partial f_z}{\partial q_2} & \frac{\partial f_z}{\partial q_3} \end{bmatrix} \dot{q} \quad (3.3).$$

A noter que J est une matrice 3x3 présentée sous la forme générale suivante :

$$J = \begin{bmatrix} \frac{\partial f_x}{\partial q_1} & \frac{\partial f_x}{\partial q_2} & \frac{\partial f_x}{\partial q_3} \\ \frac{\partial f_y}{\partial q_1} & \frac{\partial f_y}{\partial q_2} & \frac{\partial f_y}{\partial q_3} \\ \frac{\partial f_z}{\partial q_1} & \frac{\partial f_z}{\partial q_2} & \frac{\partial f_z}{\partial q_3} \end{bmatrix} \quad (3.4)$$

Tel que $\dot{x} = f(q)$ représente le modèle géométrique direct. f_x, f_y et f_z sont des fonctions liées aux composants x, y et z respectivement.

La dérivée partielle pour le premier terme de la matrice est calculée comme montré dans l'équation suivante :

$$\frac{\partial f_x}{\partial q_1} = \frac{f_x(q_1 + \Delta, q_2, q_3) - f_x(q_1, q_2, q_3)}{\Delta} \quad (3.5)$$

Et ainsi de suite pour les autres termes. Δ doit être choisi assez petit pour réduire autant que possible les erreurs dans le calcul de J , mais suffisamment grand pour éviter le bruit numérique. Cette formulation nécessite quatre évaluations du modèle géométrique qui sont $f_x(q_1 + \Delta, q_2, q_3), f_x(q_1, q_2 + \Delta, q_3), f_x(q_1, q_2, q_3 + \Delta), f_x(q_1, q_2, q_3)$ pour le calcul de la matrice jacobienne.

3.3.2 Modèle Cinématique Inverse

Le modèle cinématique inverse offre le passage des vitesses articulaires \dot{q}_1, \dot{q}_2 et \dot{q}_3 vers les vitesses de la nacelle \dot{x}, \dot{y} et \dot{z} .

Nous reprenons l'équation 3.2 en multipliant les deux membres de l'équation par J^{-1} , nous obtenons alors :

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = J^{-1} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \quad (3.6)$$

A noter que la matrice jacobienne J doit être inversible, c'est-à-dire ($\det(J) \neq 0$).

Ceci nous mène à l'étude des points singuliers qui sont définis comme les points de l'espace pour lesquels la matrice J n'est pas inversible ($\det(J) = 0$).

3.3.3 Matrice Jacobienne

Comme mentionnée précédemment, la matrice jacobienne des robots parallèles est difficile à calculer, pour cela une deuxième méthode a été développée. Elle se base sur la considération d'un ensemble de contraintes reliant les variables de position de la nacelle aux coordonnées relatives. Cette méthode a été appliquée pour première fois au robot delta par Guglielmetti [7]. Nous décrivons ci-dessous une version simplifiée de cette dernière formulation.

Les trois équations de contrainte dans le cas du robot DELTA peuvent être choisies comme :

$$|\overrightarrow{C_i B_i}|^2 - L_b^2 = 0 \quad i=1, 2, 3 \quad (3.7)$$

Cela signifie que la longueur de l'avant-bras doit être constante.

Soit S_i le vecteur $\overrightarrow{C_i B_i}$. L'équation précédente peut alors s'écrire :

$$s_i^T s_i - L_b^2 = 0 \quad i=1, 2, 3 \quad (3.8)$$

Avec :

$$s_i = \overline{C_i B_i} = \overline{O_i B_i} - (\overline{O_i A_i} + \overline{O_i C_i}) \quad (3.9)$$

Donc :

$$s_i = \begin{pmatrix} x \\ y \\ z \end{pmatrix} - R_i^R \left[\begin{pmatrix} R \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} L_A \cos q_i \\ 0 \\ -L_A \sin q_i \end{pmatrix} \right] \quad i=1, 2, 3 \quad (3.10)$$

La dérivée de l'équation 3.8 nous donne :

$$s_i^T \dot{s}_i + \dot{s}_i^T s_i = 0 \quad i=1, 2, 3 \quad (3.11)$$

Le produit scalaire étant commutatif, la dernière équation dévient :

$$s_i^T \dot{s}_i = 0 \quad i=1, 2, 3 \quad (3.11)$$

En dérivant l'équation (3.10) on obtient :

$$\dot{s}_i = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} + R_i^R \left[\begin{pmatrix} L_A \cos q_i \\ 0 \\ -L_A \sin q_i \end{pmatrix} \right] \dot{q}_i = \dot{X} + b_i \dot{q}_i \quad (3.12)$$

Avec :

$$b_i = R_i^R \begin{pmatrix} L_A \cos q_i \\ 0 \\ -L_A \sin q_i \end{pmatrix} \quad i=1, 2, 3 \quad (3.13)$$

On injecte les équations 3.10 et 3.13 dans l'équation 3.13, on obtient :

$$\begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix} \dot{X} + \begin{pmatrix} s_1^T b_1 & 0 & 0 \\ 0 & s_2^T b_2 & 0 \\ 0 & 0 & s_3^T b_3 \end{pmatrix} \dot{q} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.14)$$

Après réarrangement de cette dernière équation on obtient :

$$\dot{X} = J \dot{q}$$

Avec :

$$J = \left(\begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix} \right)^{-1} \begin{pmatrix} s_1^T b_1 & 0 & 0 \\ 0 & s_2^T b_2 & 0 \\ 0 & 0 & s_3^T b_3 \end{pmatrix} \quad (3.15)$$

Il est à noter que l'expression de la matrice jacobienne dépend en plus des variables articulaires de la position de la nacelle, d'où la nécessité d'utiliser le modèle géométrique directe.

3.4 Accélération

En dérivant l'équation 3.14 et faisant quelques transformation on obtient :

$$\ddot{X} = - \left(\begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix} \right)^{-1} \left(\begin{bmatrix} \dot{s}_1^T \\ \dot{s}_2^T \\ \dot{s}_3^T \end{bmatrix} J + K \right) \dot{q} + J \ddot{q} \quad (3.16)$$

Avec :

$$K = \begin{bmatrix} \dot{s}_1^T b_1 + s_1^T \dot{b}_1 & 0 & 0 \\ 0 & \dot{s}_2^T b_2 + s_2^T \dot{b}_2 & 0 \\ 0 & 0 & \dot{s}_3^T b_3 + s_3^T \dot{b}_3 \end{bmatrix}$$

La formule de \dot{s}_i a été calculée auparavant et \dot{b}_i est donnée par :

$$\dot{b}_i = -R_i^R \begin{pmatrix} LA \sin qi \\ 0 \\ LA \cos qi \end{pmatrix} \dot{q}_i \quad i=1, 2, 3. \quad (3.17)$$

A noter que J peut être identifié dans l'équation 3.16 comme le terme multipliant \ddot{q} .

3.5 Modèle Dynamique

3.5.1 Introduction

Le développement de modèles dynamiques de robots a été un sujet de recherche intense au cours de la dernière décennie. La difficulté principale est de trouver une solution suffisamment représentative du système réel et qui peut facilement être calculée en temps réel

pour la mise en œuvre de la commande. Pour les structures parallèles, le problème est encore plus complexe que pour les structures série, principalement en raison de la difficulté analytique présentée par les interdépendances des variables communes. La recherche de méthodes généralement applicables à la modélisation de tels robots a été réalisée, mais leurs résultats ne se prêtent pas facilement à un vrai traitement du temps. Une voie qui a souvent été explorée est de couper le mécanisme de la chaîne fermée aux articulations passives et de considérer d'abord la dynamique du robot arborescence ainsi créé. La condition de fermeture est ensuite assouplie soit par l'utilisation de multiplicateurs de Lagrange [8] ou l'application du principe de travail virtuel d'Alembert au moyen d'une matrice jacobienne [9], pour les mécanismes parallèles, l'application directe de la méthode Newton-Euler a également été étudiée [10] [11]. l'application de cette méthode au robot DELTA peut être trouvée dans [4] ou [12]. La méthode basée sur le principe du travail virtuel [13], sur la formalisation de Lagrange [14] [15] ou sur le principe d'Hamilton [16] a également été étudiée

La plus part des chercheurs conviennent qu'un modèle complet, ou on tient compte de toutes les masses et les inerties conduit à des solutions très compliquées, pour y remédier, certaines hypothèses simplificatrice sont utilisées.

Le modèle dynamique proposé dans notre travail est développé en se basant sur ces hypothèses simplificatrices ci-dessous.

3.5.2 Hypothèses Simplificatrices

La complexité du modèle dynamique du robot Delta provient principalement du mouvement des avant-bras (barres parallèles). Pour simplifier ce problème on négligera leurs inerties de rotation. Cette hypothèse n'est pas très restrictive en raison de l'utilisation de fibres de carbone dans leur construction. Ainsi, la force entre la nacelle et le bras est dans une direction donnée par l'orientation de l'avant-bras [1]. Le modèle développé pour le robot DELTA est basé sur la méthode de Newton-Euler avec les hypothèses simplificatrices suivantes :

Les inerties de rotation des avant-bras sont négligées.

A des fins analytiques, les masses des avant-bras sont séparées en deux parties et sont placées à leurs deux extrémités. $2/3$ à l'extrémité supérieure (coude) et $1/3$ à l'extrémité inférieure (la nacelle). Pour justifier cette démarche, on s'appuiera sur fait que l'inertie I d'une barre de

longueur L et de masse m par rapport à l'une des extrémités est égale à $\frac{1}{3}ml^2$. ce qui est équivalent à placer $1/3$ de la masse à l'extrémité inférieure en suivant le raisonnement précédent,

Les effets de frottement et l'élasticité sont négligés.

3.5.3 Paramètres Dynamiques

Pour la suite, afin d'établir le modèle dynamique, les paramètres dynamiques énoncée ci-dessous ont été utilisées.

Pour la nacelle la masse totale utilisée est :

$$m_{nt} = m_n + m_{charge\ utile} + 3(1 - r)m_{ab} \quad (3.18)$$

Avec :

m_n , la masse de la charge $m_{charge\ utile}$ et les 3 masses rapportées de chacun des 3 avant-bras où $r=2/3$ est le rapport de masse des avant-bras située à l'extrémité supérieure.

Au niveau des bras, la position du centre de masse du bras est donnée par :

$$r_{GB} = L_A \frac{\frac{1}{2}m_{br} + m_c + m_{ab}}{m_b} \quad (3.19)$$

Avec :

$$m_b = m_{br} + m_c + r m_{ab} \quad (3.20)$$

Ou :

m_{br} : la masse du bras.

m_c : la masse du coude.

m_{ab} : la masse de l'avant-bras.

L'inertie à considérer au niveau du bras I_{bi} est la somme de l'inertie du moteur après réduction et rapporté au bras I_m plus l'inertie du bras lui-même I_{br} plus l'inertie du système de réduction rapporté au bras I_{red} . Elle est donnée par :

$$I_{bi} = I_m + I_{br} + I_{red} \quad (3.21)$$

Avec :

$$I_{br} = L_A^2 \left(\frac{1}{3} m_{br} + m_c + r m_{ab} \right) \quad (3.22)$$

A noter que le facteur de réduction étant égale à 12, l'inertie réduction sera multiplié par 12.

Grace au théorème de Huygens, les inerties sont ramenées au niveau du bras.

3.5.4 Modèle Dynamique Basé sur le Principe du Travail Virtuel

A l'aide des hypothèses simplificatrices mentionnées précédemment, on pourra réduire le robot delta en quatre corps qui sont : les trois bras supérieurs et la nacelle.

On constate alors deux types de forces qui agissent sur la nacelle : la force de gravité G_n et la force d'inertie F_n . leurs équations sont données respectivement :

$$G_n = m_{nt} [0 \quad 0 \quad -g]^T \quad (3.23)$$

$$F_n = m_{nt} \ddot{X} \quad (3.24)$$

\ddot{X} et \ddot{q} sont liées par :

$$\ddot{X} = J\ddot{q} + \dot{J}\dot{q} \quad (3.25)$$

On remplace dans (3.24) :

$$F_n = m_{nt} (J\ddot{q} + \dot{J}\dot{q}) \quad (3.26)$$

Pour trouver la contribution de ces forces a chacun des moteurs, on multiplie leurs deux équations par la transposée de la matrice jacobienne :

$$\tau_n = J^T F_n = J^T m_{nt} (J\ddot{q} + \dot{J}\dot{q}) \quad (3.27)$$

$$\tau_{Gn} = J^T G_n = J^T m_{nt} [0 \quad 0 \quad -g]^T \quad (3.28)$$

Selon le principe du travail virtuel, la contribution de toutes les forces non inertielles doit être égale à la contribution de toutes les forces inertielles. Ce principe appliqué au niveau articulaire donne :

$$\tau + \tau_{Gn} + \tau_{Gb} = I_b \ddot{q} + \tau_n \quad (3.29)$$

τ est le vecteur des couples qui doivent être appliqués aux moteurs avec :

$$\tau = 12 \tau_m \quad (3.30)$$

τ_{Gb} est le couple produit par la force gravitationnelle des bras :

$$\tau_{Gb} = m_b r_{Gb} [\cos(q_1) \cos(q_2) \cos(q_3)]^T \quad (3.31)$$

I_b est la matrice d'inertie des bras dans l'espace articulaire :

$$I_b = \begin{bmatrix} I_{b1} & 0 & 0 \\ 0 & I_{b2} & 0 \\ 0 & 0 & I_{b3} \end{bmatrix} \quad (3.32)$$

En injectant les équations obtenues précédemment dans (3.29) on obtient :

$$\tau = (I_b + J^T m_{nt} J) \ddot{q} + J^T m_{nt} \dot{J} \dot{q} - \tau_{Gn} - \tau_{Gb} \quad (3.33)$$

A présent, on pourra identifier les termes suivant :

La matrice de masse du robot :

$$A = (I_b + J^T m_{nt} J) \quad (3.33a)$$

Les contributions de Coriolis et centrifuges :

$$\tau_{c\&c} = J^T m_{nt} \dot{J} \quad (3.33b)$$

La partie gravité :

$$\tau_{Gravité} = -\tau_{Gn} - \tau_{Gb} = -J^T m_{nt} [0 \ 0 \ g]^T - \tau_{Gb} \quad (3.33c)$$

De ce développement, nous pourrons ainsi définir le modèle dynamique direct qui nous permettra d'obtenir les angles rotoïdes à partir des couples développés par les moteurs, et de définir le modèle dynamique inverse qui nous permettra d'obtenir les couples à partir des angles rotoïdes (figure 3.8).

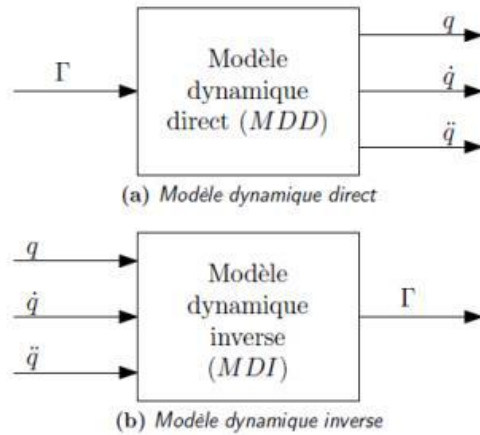


Figure 3.9 – MDD et MGD

3.5.5 Conclusion

Nous avons constaté que le robot Delta présente une structure parallèle assez complexe et difficile à modéliser contrairement au robot série qui possède une méthode systématique pour sa modélisation. Néanmoins, la particularité de triple symétrie que possède le Delta nous a permis de travailler sur une seule chaîne cinématique et à grâce à des hypothèses simplificatrices, nous avons pu obtenir le modèle géométrique, cinématique et dynamique qui sont proche des modèles réels. A présent, nous pouvons passer à la dernière étape de notre projet qui est la commande du robot Delta.

CHAPITRE 4

Commande PD Classique et d'Ordre Fractionnaire du Robot ISIR 88

4.1 Introduction

Dans ce chapitre, nous allons aborder la commande du Robot Delta ISIR 88. Ce robot est réputé pour ses accélérations particulièrement élevées d'où la nécessité d'une synthèse de loi de commande robuste pour garantir un positionnement précis et la poursuite d'une trajectoire désirée. Pour ce faire, nous allons traiter la partie théorique qui consiste à synthétiser les régulateurs que nous allons utiliser pour commander notre robot à savoir la commande Proportionnel Dérivé (PD), Proportionnel Dérivé d'ordre fractionnaire (PDOF) et commande par mode glissant classique (ordre entier) et d'ordre non entier en mettant en contribution les modèles étudiés dans le chapitre précédent et d'autres notions qui seront nécessaires pour l'élaboration de la commande à savoir le calcul d'ordre fractionnaire, les critères RMS(Root Mean Square) et MAXAE (Maximum Absolute Error), la méthode des Moments Généralisés et l'Optimisation par Essaim de Particules (PSO). Ensuite, nous allons simuler les lois de commande développées sous MATLAB-SIMULINK puis comparer les résultats obtenus (les erreurs, les couples et les tests de robustesse) entre les différentes commandes.

4.2 Régulateur Proportionnel Dérivé

Un régulateur classique parmi les plus anciens, connu pour sa simplicité et la facilité d'implémentation.

La commande PD est de la forme :

$$U_i(t) = K_{p_i}e_i(t) + K_{d_i}\dot{e}_i(t) \quad i=1, 2, 3. \quad (4.3)$$

Avec :

$U(t) \in \mathbb{R}^3$: représente la commande (le couple dans notre cas).

$e(t) \in \mathbb{R}^3$: représente l'erreur de poursuite ($e(t) = q_d(t) - q(t)$), et $\dot{e}(t)$ sa dérivée.

K_p, K_d : représentent les paramètres du régulateur.

i : numéro de l'articulation.

Ces paramètres ont été élaborés par ajustement en simulation pour garantir un couple lisse et une erreur acceptable. Les valeurs obtenues sont :

$$K_p = \text{diag} (25 ; 25 ; 25) \text{ et } K_d = \text{diag} (1.5 ; 1.5 ; 1.5).$$

Ce régulateur va être implémenté comme présenté dans la figure (4.1).

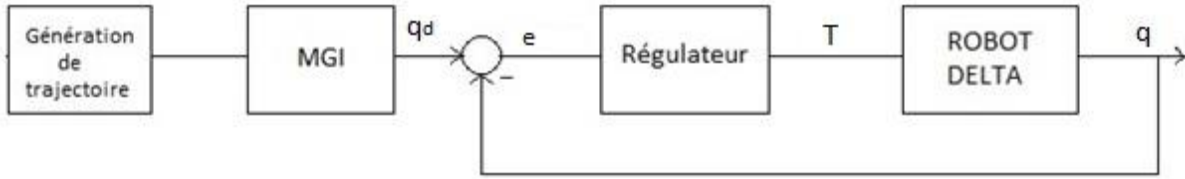


Figure 4.1 – Structure de commande en boucle fermée

Cette structure présente une commande décentralisée de sorte que chaque articulation est commandée par un régulateur PD et sera maintenue pour le régulateur PDOF.

4.2.1 Simulation avec PD

Le choix des paramètres a été effectué par simulation sous MATLAB/SIMULINK sans masse manipulée ($m = 0g$). Une limitation sur la commande (couple) est imposée avec une valeur égale à $|14.4| Nm$.

Tandis que, la trajectoire désirée adoptée dans tous notre travail est de type elliptique.

De la figure 4.2, on remarque que la trajectoire désirée et la trajectoire de la nacelle sont quasi superposées ce qui se traduit par une bonne poursuite de trajectoire. Pour l'erreur en position, on constate qu'elle est de l'ordre de 10^{-3} avec une valeur maximale de $-0.0041rad$. Pour le couple, on remarque que la valeur maximale atteinte est égale a $-1.9Nm$. Les courbe des erreurs et des couples sont parfaitement lisses (absence d'oscillation).

Ces résultats montrent que le régulateur PD est un régulateur qui a pu commander un système non linéaire assez complexe. Néanmoins, Pour ce type de régulateur linéaire, la robustesse n'est pas toujours garantie. Il existe dans la littérature d'autres types de commandes qui offrent plus de manœuvre par rapport aux variations paramétriques. Entre outre, la commande d'ordre fractionnaire et la commande par mode glissant qui ont ont été avantageux dans différents domaines d'applications. Nous allons essayer ces approches sur notre système.

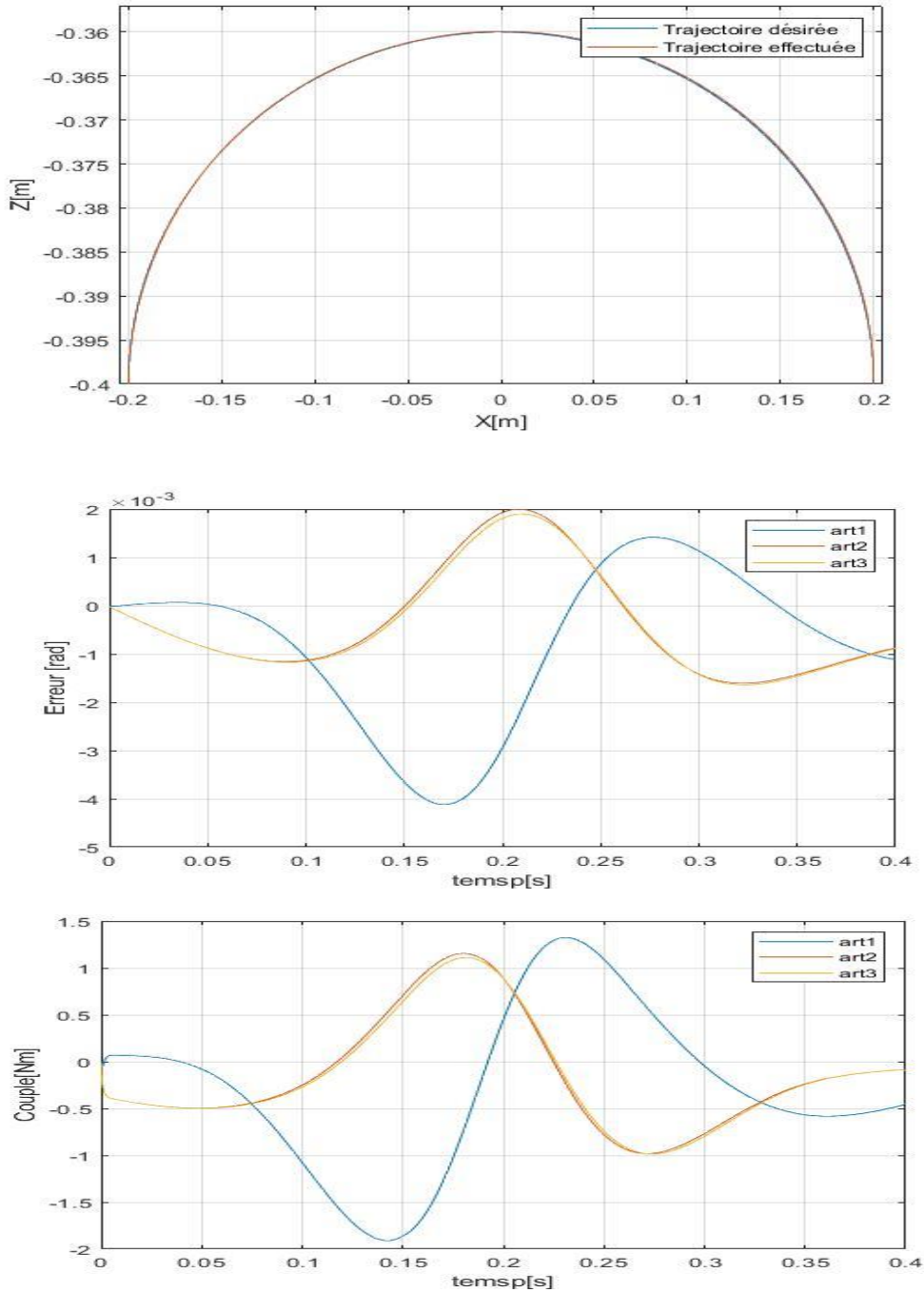


Figure 4.2 – Trajectoire effectuée, erreurs de poursuite articulaires et commandes en couple dans le cas d'un régulateur PD.

4.3 Régulateur d'Ordre Fractionnaire

4.3.1 Introduction

Comme mentionné précédemment, le régulateur proportionnel dérivé est réputé pour sa simplicité de synthèse. Cependant, un autre régulateur similaire à ce dernier a été le centre

d'intérêt des chercheurs durant ces dernières décennies, il s'agit du régulateur proportionnel dérivé d'ordre fractionnaire. En effet, plusieurs systèmes ont manifesté un comportement dynamique d'ordre fractionnaire tel que les systèmes viscoélastiques, la polarisation d'interfaces, le comportement cardiaque ...

C'est seulement durant ces dernières années que l'on retrouve un progrès fulgurant dans l'étude de ce régulateur qui peut servir de fondation dans la théorie des systèmes et la théorie de la commande. Entre autre, un travail intensif est dédié à l'application des concepts d'ordre fractionnaire dans le domaine de l'ingénierie. En effet, l'idée d'utiliser le calcul d'ordre fractionnaire dans la boucle de contrôle des systèmes date du début des années 60. Oustaloup a développé la Commande Robuste d'Ordre non Entier (CRONE) et a appliqué cette technique dans divers domaines des systèmes de commande [8]. En profitant des propriétés avantageuses des systèmes d'ordre fractionnaire, ce régulateur permettait d'assurer la robustesse de la commande dans une bande de fréquences donnée. Depuis cette initiative, les chercheurs se sont penchés sur le développement de ce type de régulateur.

4.3.2 Généralité sur le Calcul Fractionnaire :

Le calcul d'ordre fractionnaire est un sujet mathématique datant de plus de 300 ans. En effet, la généralisation du calcul différentiel et intégral sur l'ordre non entier (fractionnaire) a été proposée pour la première fois par Leibniz et l'Hopital. Depuis de nombreux chercheurs a l'instar de Liouville 1832, Holmgren 1864 et Riemann 1953 ont effectué des travaux de recherche sur ce domaine [24].

4.3.2.1 Définition de la Dérivée d'Ordre Fractionnaire

L'opérateur différo-intégral noté ${}_aD_t^\mu$ où a et t sont les limites de l'opération et $\mu \in R$ est l'ordre fractionnaire est donné par :

$${}_aD_t^\mu = \begin{cases} \frac{d^\mu}{dt^\mu} & , \mu > 0 \\ 1 & , \mu = 0 \\ \int_a^t (d\tau)^{-\mu} & , \mu < 0 \end{cases} \quad (4.4)$$

On trouve dans la littérature d'autres définitions de cet opérateur [25], telles que celles de Cauchy, Grunwald et Caputo

Selon la définition de Riemann Liouville (RL) :

$${}_a D_t^\mu f(t) = \frac{1}{\Gamma(m-\mu)} \frac{d^m}{dt^m} \int_\mu^t \frac{f(\tau)}{(t-\tau)^{1-(m-\mu)}} d\tau \quad (4.5)$$

Selon la définition de Caputo :

$${}_a D_t^\mu f(t) = \frac{1}{\Gamma(m-\mu)} \frac{d^m}{dt^m} \int_a^t \frac{f^m(\tau)}{(t-\tau)^{1-(m-\mu)}} d\tau \quad (4.6)$$

Selon la définition de Gründwald-Leitnikov [20] :

$${}_a D_t^\mu f(t) = \lim_{h \rightarrow 0} \frac{1}{h} \sum_{k=0}^{\infty} (-1)^k \binom{m}{k} f(t - k h) \quad (4.7)$$

La simulation numérique pour des systèmes modélisés par des équations différentielles d'ordre fractionnaire nécessite la transformation de la Laplace de la dérivée d'ordre non entier afin de synthétiser aisément des lois de commandes. En effet, cet outil mathématique a été élargi pour rester applicable pour le cas fractionnaire et ne présente pas beaucoup de différences par rapport au cas classique. Pour des conditions initiales nulles, la dérivée d'ordre non entier est donnée par :

$$L \left\{ \frac{d^\mu f(t)}{dt^\mu} \right\} = s^\mu L \{ f(t) \} \quad (4.8)$$

Il existe plusieurs toolbox Matlab permettant l'étude des systèmes d'ordre non entier. Parmi les plus avérées la toolbox FOMCON. Elle propose plusieurs fonctionnalités telles que l'identification, la modélisation, la simulation et même l'implémentation des régulateurs fractionnaires sous différentes formes. Elle jouit remarquablement d'une simplicité d'utilisation grâce aux interfaces graphiques dont elle est dotée. On en trouve "impid", "iopid_tune" et "fomcon".

D'autre part, la méthode D'Oustaloup [26] permet l'approximation de l'opérateur fractionnaire par une fonction rationnelle (appelée aussi filtre récursif d'Oustaloup) en utilisant une distribution récursive de zéros et de pôles d'ordre entier, repartis dans une bande de fréquence limitée. L'approximation de l'opérateur s^μ dans une bande de fréquence $[w_b, w_h]$ est donnée par [20] :

$$\hat{G}(s) = K \prod_{k=-N}^N \frac{s + w'_k}{s + w_k} \quad (4.9)$$

Avec w_b et w_h sont respectivement les basses et les hautes fréquences et $(2N + 1)$ est l'ordre de filtre.

K , w_k et w'_k sont donnés par :

$$K = \omega_h^\alpha$$

$$\omega'_k = \omega_b \left(\frac{\omega_h}{\omega_b} \right)^{\frac{k+N+0.5(1-\alpha)}{2N+1}} \quad (4.10)$$

$$\omega_k = \omega_b \left(\frac{\omega_h}{\omega_b} \right)^{\frac{k+N+0.5(1+\alpha)}{2N+1}}$$

Dans notre travail, nous avons choisi de prendre $N = 2$ et la marge d'approximation

$(\omega_b = 10^{-2}; \omega_h = 10^2)rad/s$.

4.3.3 Approche de Synthèse d'un Régulateur Proportionnel Dérivé d'Ordre Fractionnaire d'un Système de Second Ordre

Nous avons adopté l'approche présentée dans [20].

Contrairement au PD classique, le PDOF comporte un troisième paramètre « μ ».

La fonction de transfert du régulateur est donnée par l'équation suivante :

$$C(s) = Kp(1 + Kd s^\mu) \quad 0 < \mu < 1 \quad (4.11)$$

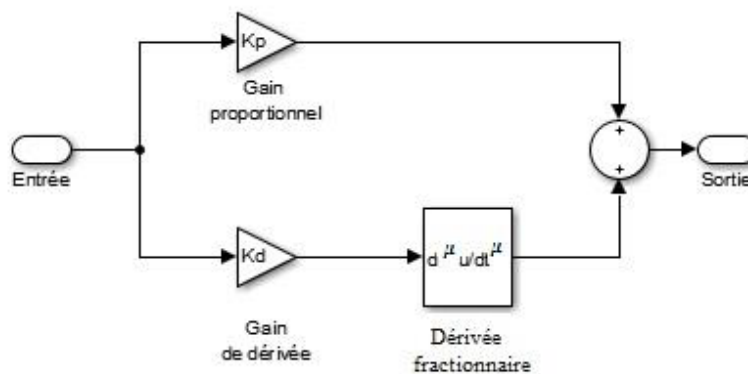


Figure 4.3 – structure du régulateur PD d'ordre fractionnaire [27]

Dans la suite, nous avons utilisé une méthode de synthèse du régulateur PDOF pour la classe des systèmes de second ordre. Cette méthode est utilisée pour des applications de commande de mouvement [17].

La fonction de transfert d'un système de second ordre est donnée dans l'équation suivante :

$$G(s) = \frac{1}{s(\tau s + 1)} \quad (4.12)$$

Le principe de cette méthode de réglage, est de garantir que la fréquence de coupure (crossover frequency) et la marge de phase sont respectées. En outre, la dérivée de l'argument du système en boucle ouverte par rapport à la fréquence est forcée d'être nulle pour une fréquence de coupure donnée, de sorte que le système en boucle fermée soit robuste.

La phase et le gain d'un système du second ordre sont données respectivement dans (4.13) et (4.14) :

$$\arg[G(j\omega)] = -\tan^{-1}(\omega\tau) - \frac{\pi}{2} \quad (4.13)$$

$$|G(j\omega)| = \frac{1}{\omega\sqrt{(1 + (\omega\tau)^2)}} \quad (4.14)$$

La fonction de transfert du système avec régulateur en boucle ouverte est :

$$L(s) = C(s) G(s) \quad (4.15)$$

On impose une marge de phase φ_m et fréquence de coupure ω_{cg} pour que :

$$\arg[L(j\omega_{cg})] = -\pi + \varphi_m \quad (4.16)$$

$$|L(j\omega_{cg})| = 0_{dB} \quad (4.17)$$

Avec :

$$\left. \frac{d \arg[G(j\omega)]}{d\omega} \right|_{\omega_{cg}} = 0 \quad (4.18)$$

A partir du développement de ces trois équations dans [17] nous obtenons :

$$K_d = \frac{1}{(\omega_{cg})^\mu} \tan \left[\varphi_m + \tan^{-1}(\omega_{cg}\tau) - \frac{\mu\pi}{2} + \pi \right] \cos \frac{(1-\mu)\pi}{2} - \frac{1}{(\omega_{cg})^\mu} \sin \frac{(1-\mu)\pi}{2} \quad (4.19)$$

$$K_d = \frac{-B \pm \sqrt{B^2 - 4A^2(\omega_{cg})^{2\mu}}}{2A(\omega_{cg})^{2\mu}} \quad (4.20)$$

Avec :

$$A = \frac{\tau}{1+(\tau\omega_{cg})^2}, \quad B = 2A(\omega_{cg})^\mu \sin \frac{(1-\mu)\pi}{2} - \mu(\omega_{cg})^{\mu-1} \cos \frac{(1-\mu)\pi}{2}.$$

$$1 = \frac{K_p \sqrt{\left(1 + K_d (\omega_{cg})^\mu \cos \left(\frac{\mu\pi}{2}\right)\right)^2 + K_d (\omega_{cg})^\mu \sin \left(\frac{\mu\pi}{2}\right)^2}}{\sqrt{\omega_{cg}(1 + (\omega_{cg} \tau)^2)}} \quad (4.21)$$

A présent, on peut résoudre (4.19), (4.20) et (4.21) pour trouver les paramètres K_d , K_p et μ . Une méthode graphique peut être utilisée comme outil pratique et simple pour obtenir ces paramètres. La procédure pour la synthèse du régulateur PDOF est la suivante :

- 1) Donner ω_{cg} , la fréquence de coupure.
- 2) Donner φ_m , la marge de phase désirée.
- 3) Dessiner la courbe K_d en fonction de μ selon (4.19).
- 4) Dessiner la courbe K_d en fonction de μ selon (4.20).
- 5) Obtenir μ et K_d à partir du point d'intersection des deux courbes dessinées (Figure 4.4).
- 6) Calculer K_p de (4.21).

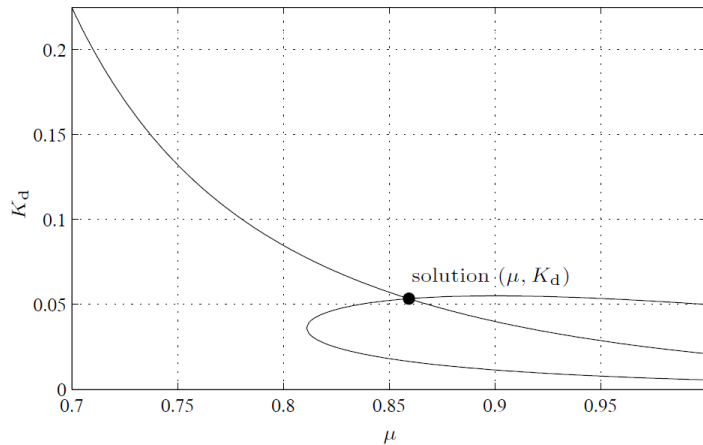


Figure 4.4 – Courbes des deux équations de kd en fonction de μ [17].

4.3.4 Calcul des Paramètres du Régulateur PDOF

Afin de trouver la fonction de transfert de second ordre qui est nécessaire pour le calcul du régulateur, nous avons identifié le système non linéaire par la méthode des Moments Généralisés présentée dans [22].

Cette méthode se base sur la relation entre les moments de l'entrée et les moments de la sortie et qui est valable pour tout type d'entrée contrairement à la méthode classique et donc nous allons utiliser un échelon comme entrée pour trouver l'ordre de grandeur des constantes de notre système ainsi qu'une approximation au tour d'un point de fonctionnement qui a été choisi auparavant.

Entre autres, nous avons identifié le MDD en prenant comme entrée le couple et en sortie la vitesse articulaire, puis nous avons intégré la FT en multipliant par $\frac{1}{s}$ pour obtenir l'angle articulaire (la position). Nous avons obtenu la fonction de transfert suivante :

$$G(s) = \frac{1.4476}{s(0.0776s + 1)} \quad (4.22)$$

Pour valider l'indentification, nous avons comparé la réponse du modèle identifié avec la réponse du MDD (modèle non linéaire) pour un couple en entrée de valeur -0.2766 Nm, voir (Figure 4.5).

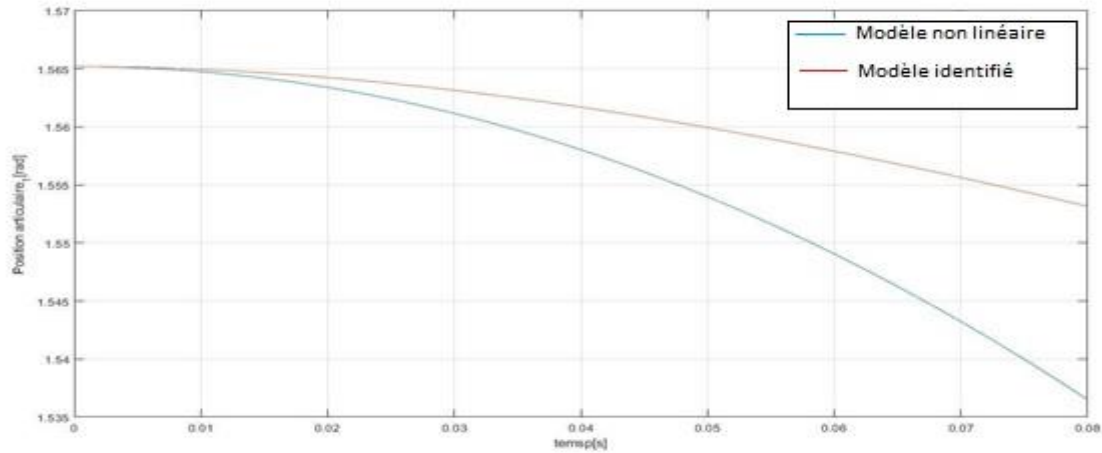


Figure 4.5 – Réponses du modèle non linéaire et modèle approximé

D'après cette figure, on constate que le comportement dynamique du modèle identifié suit correctement celui du modèle non linéaire. L'erreur statique entre les deux modèles peut être réduite en modifiant légèrement la valeur du gain k .

Nous pouvons conclure que notre système peut être approximé par une fonction de transfert d'ordre 2 avec un intégrateur, un gain $k = 1.4476$ et une constante de temps égale à 0.0776 s.

Une fois que le système a été identifié, nous pouvons appliquer la procédure de calcul du PDOF.

On prend la fonction de transfert suivante :

$$G(s) = \frac{1}{s(0.0776 s + 1)} \quad (4.23)$$

On choisit φ_m égale à 60° pour une fréquence de coupure de 25 rad s^{-1} .

Lors du choix de la marge de phase désirée il ne faut pas forcer le système en choisissant une marge de phase proche de 45° car ça entrainera des oscillations dans la commande.

On dessine les deux courbes de Kd en fonction de μ (Figure 4.6) :

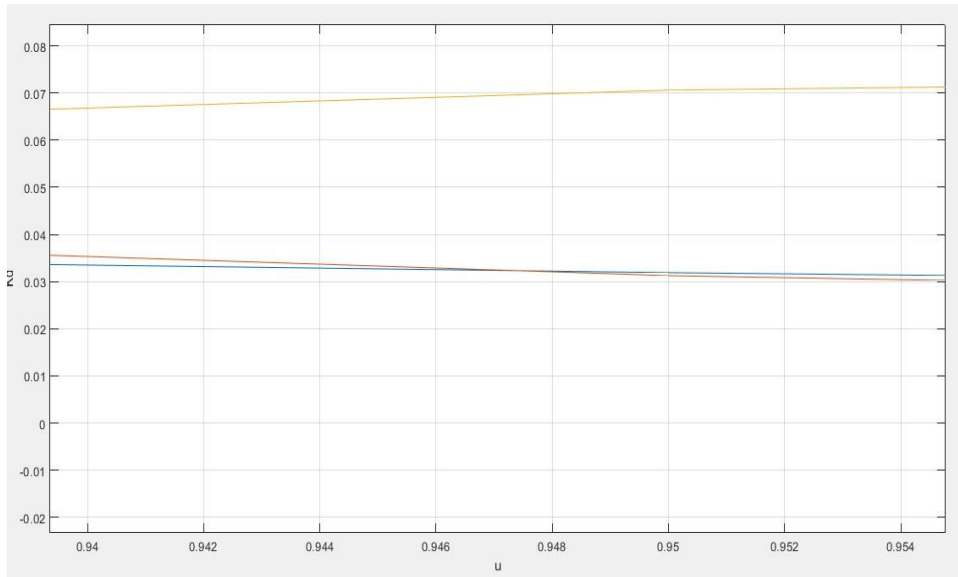


Figure 4.6 – Courbes de K_d en fonction de μ .

Le point d'intersection correspond à $K_d = 0.0323$ et $\mu = 0.9474$.

Pour ces deux valeurs on obtient $K_p = 43.3463$.

On divise le K_p par le gain du système k .

Pour la structure suivante :

$$C(s) = K_p' + K_d' s^\mu \quad 0 < \mu < 1 \quad (4.24)$$

$$K_p' = \frac{K_p}{k} = 30.0057.$$

$$K_d' = K_p' \cdot K_d = 0.9692.$$

$$\mu = 0.9474.$$

Pour valider les résultats obtenus, on trace le diagramme de Bode du système avec régulateur en BO (Figure 4.7) et on compare la marge de phase et la fréquence de coupure avec φ_m et w_{cg} imposées.

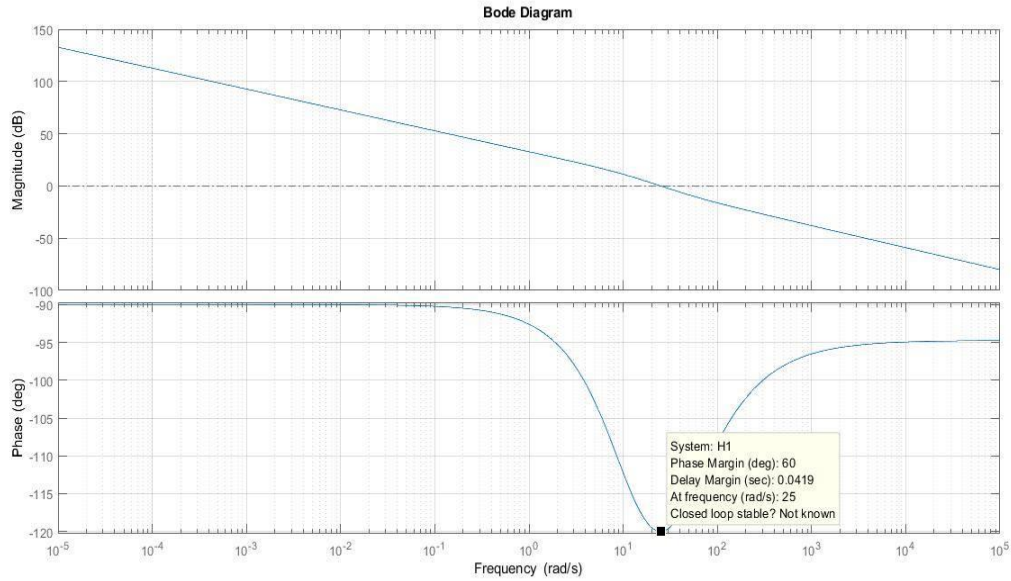


Figure 4.7 – Tracé de Bode du système avec régulateur en BO.

On remarque que les valeurs de φ_m et ω_{CG} , sont identiques, on pourra conclure que la méthode utilisée est validée.

Les résultats de la simulation du robot avec régulateur en BF sont donnés dans la figure 4.8 :

On remarque que la trajectoire désirée et la trajectoire de la nacelle sont quasi superposées ce qui se traduit par une bonne poursuite de trajectoire. Pour l'erreur en position, on constate qu'elle est de l'ordre de 10^{-3} avec une valeur maximale de -0.0027rad . Pour le couple, on remarque que la valeur maximale atteinte est égale a -1.2Nm , par contre, on constate la présence de légères fluctuations pendant un laps de temps petit du a la forte non linéarité du robot.

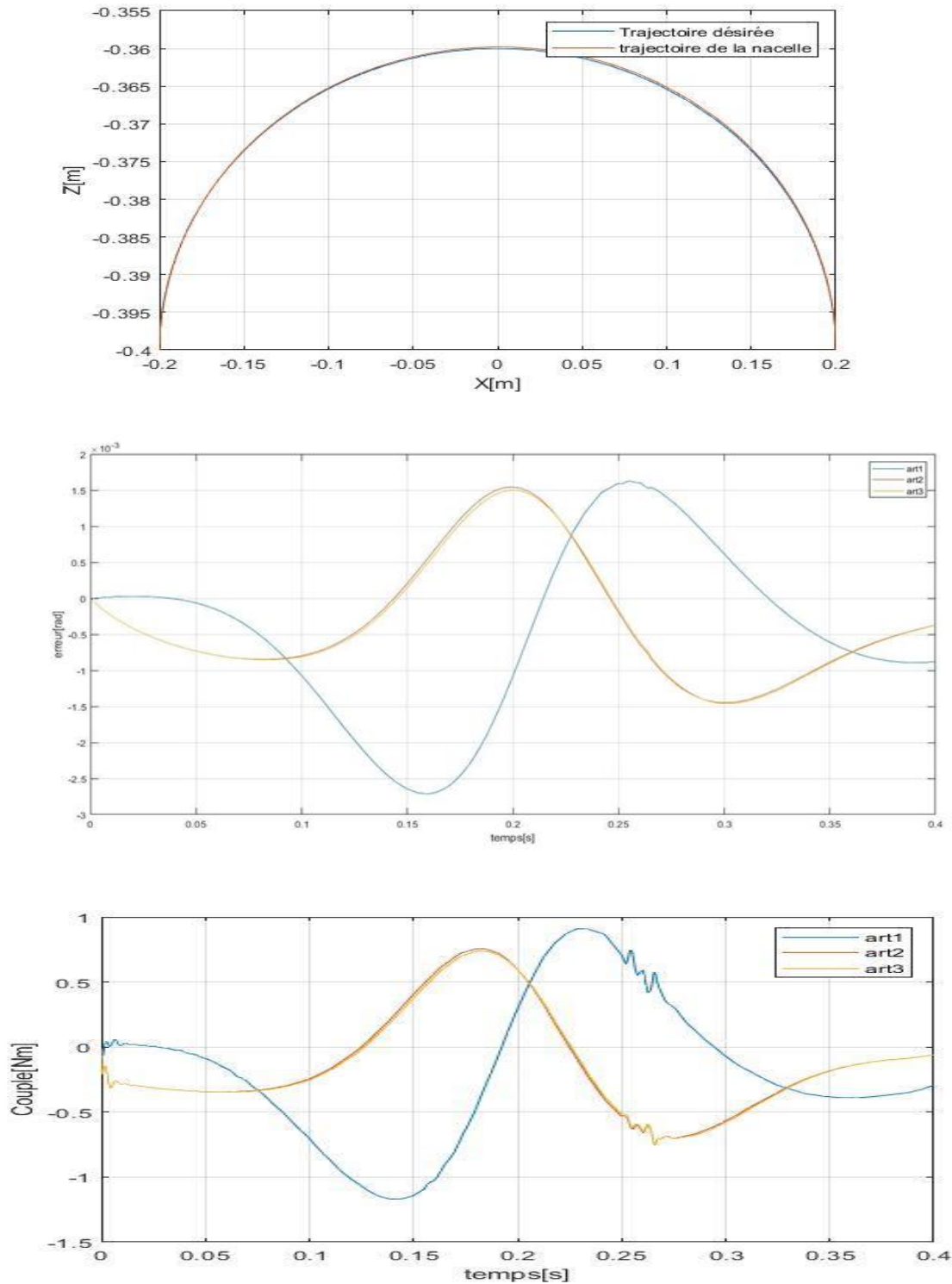


Figure 4.8 – Trajectoire effectuée, erreurs de poursuite articulaires et commandes en couple dans le cas d'un régulateur PDOF (FOMCON)

On remplace l'opérateur fractionnaire de FOMCON par le filtre d'Oustaloup mentionné précédemment.

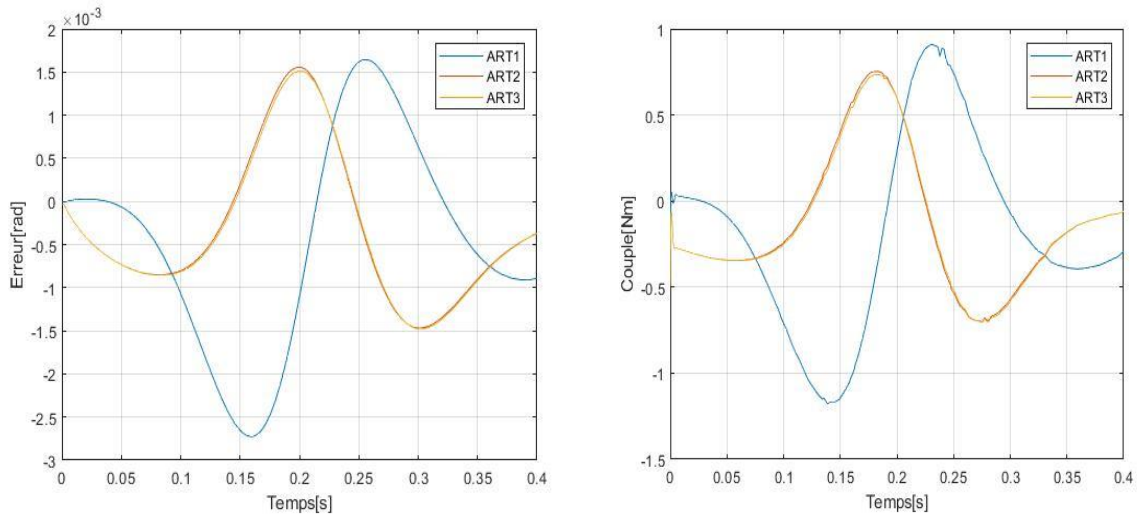


Figure 4.9 – Erreurs de poursuite articulaires et commandes en couple dans le cas d'un régulateur PDOF avec l'approximation d'Oustaloup

Comme le montre la figure (4.9), nous remarquons que les résultats obtenus précédemment n'ont pas changé à l'exception du couple où on constate une forte diminution des oscillations (quasiment absentes). Cela nous permet de déduire que les oscillations sont introduites par l'opérateur fractionnaire de FOMCON et que ce dernier peut être mieux approximé par un filtre d'Oustaloup.

Nous pouvons conclure que le régulateur proportionnel dérivé d'ordre fractionnaire nous donne des résultats satisfaisants.

4.4 Critère des Variations de Performances

Root Mean Square Error représente la valeur quadratique de l'erreur de poursuite qui est une mesure de précision pour comparer les erreurs de prévision de différents modèles pour un ensemble de données particulier. Sa formule est donnée par :

$$RMSE = \sqrt{RMSE_{\alpha_1}^2 + RMSE_{\alpha_2}^2 + RMSE_{\alpha_3}^2} \quad (4.25)$$

Avec : $RMSE_{\alpha_i} = \sqrt{\frac{1}{N} \sum_{j=1}^N e_i(j)^2}$

Où :

N : nombre d'échantillons.

$e_i(j)$: L'erreur de poursuite pour l'articulation α_i dans l'échantillon.

$RMSE_{\alpha_i}$: est RMS de l'erreur de poursuite pour l'articulation α_i ;

La formule reste valable pour la RMS de la commande.

Maximum Absolute Error calcule la moyenne de la différence absolue entre deux valeurs (prédite et réelle). Sa formule est donnée par :

$$\text{MaxAE} = \max(\text{MaxAE}_{\alpha_1}, \text{MaxAE}_{\alpha_2}, \text{MaxAE}_{\alpha_3}) \quad (4.26)$$

Avec : $\text{MaxAE}_{\alpha_i} = \max(|x_i - x_{di}|)$

Tel que : $e_i = x_i - x_{di}$.

e_i : l'erreur de poursuite pour l'articulation α_i .

4.5 Comparaison entre le PD Classique et le PD d'Ordre Fractionnaire

A présent, Nous allons faire une comparaison entre les résultats du PD et du PDOF obtenus précédemment. Pour cela, nous testerons la robustesse de ces deux régulateurs pour des valeurs de masses suivantes : 0, 20, 40, 60, 80 et 100g. La comparaison a été faite en utilisant les critères RMS et MAXAE cités dans le paragraphe précédent.

La figure 4.10 nous donne les résultats de la comparaison entre les RMS des erreurs et des couples des deux régulateurs et les MAXAE des erreurs.

Pour des valeurs croissantes de la masse utile, nous remarquons que l'erreur et le couple augmentent pour les deux régulateurs. Les RMSE du PDOF et du PD sont comprises entre [0.0014 ,0.0021] rad et [0.0018 ,0.0028] rad respectivement. La RMS des couples du PDOF et du PD sont comprise entre [0.71 ,1.02] Nm et [0.9 ,1.35] Nm respectivement. Cependant pour chaque masse, les valeurs obtenues du PDOF sont inférieures aux valeurs obtenues du PD. On remarquera aussi que pour une masse égale à 60g, la RMSE du PDOF a brusquement augmenté et a atteint sa valeur maximale ce qui a entraîné une valeur maximale de la RMS du couple. Ce phénomène peut se justifier par une mauvaise approximation de l'opérateur fractionnaire FOMCON.

On constate aussi que la valeur du MAXAE du PDOF est à chaque fois inférieure à la valeur du MAXAE du PD pour chaque masse.

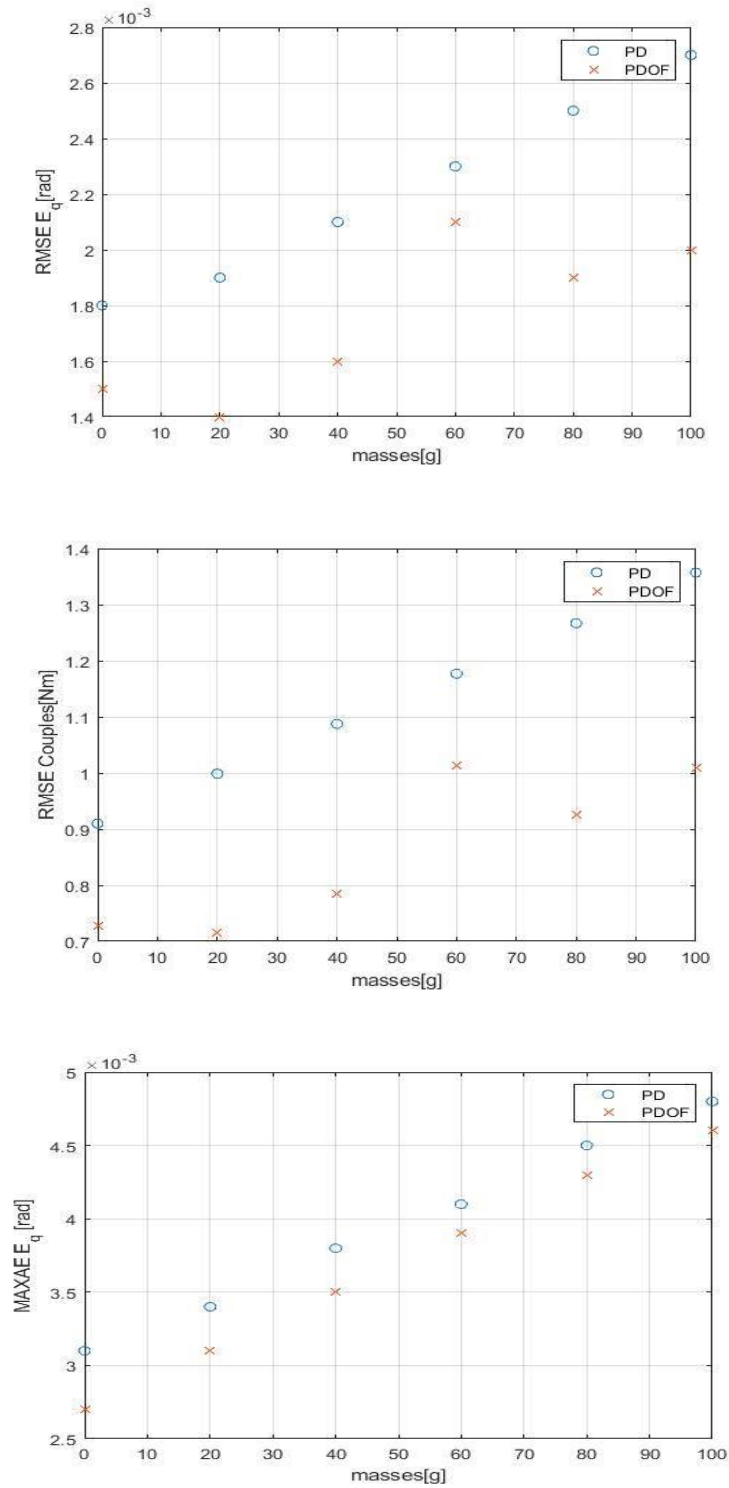


Figure 4.10 – RMS des erreurs, RMS des couples et MAXAE des erreurs (FOMCON).

En changeant l'opérateur FOMCON par celui d'Oustaloup, on remarque de la figure 4.11 que les résultats obtenus précédemment n'ont pas vraiment changé sauf la RMS erreur et couple qui n'ont pas augmenté brusquement à $m = 60g$ ce qui confirme l'explication précédente.

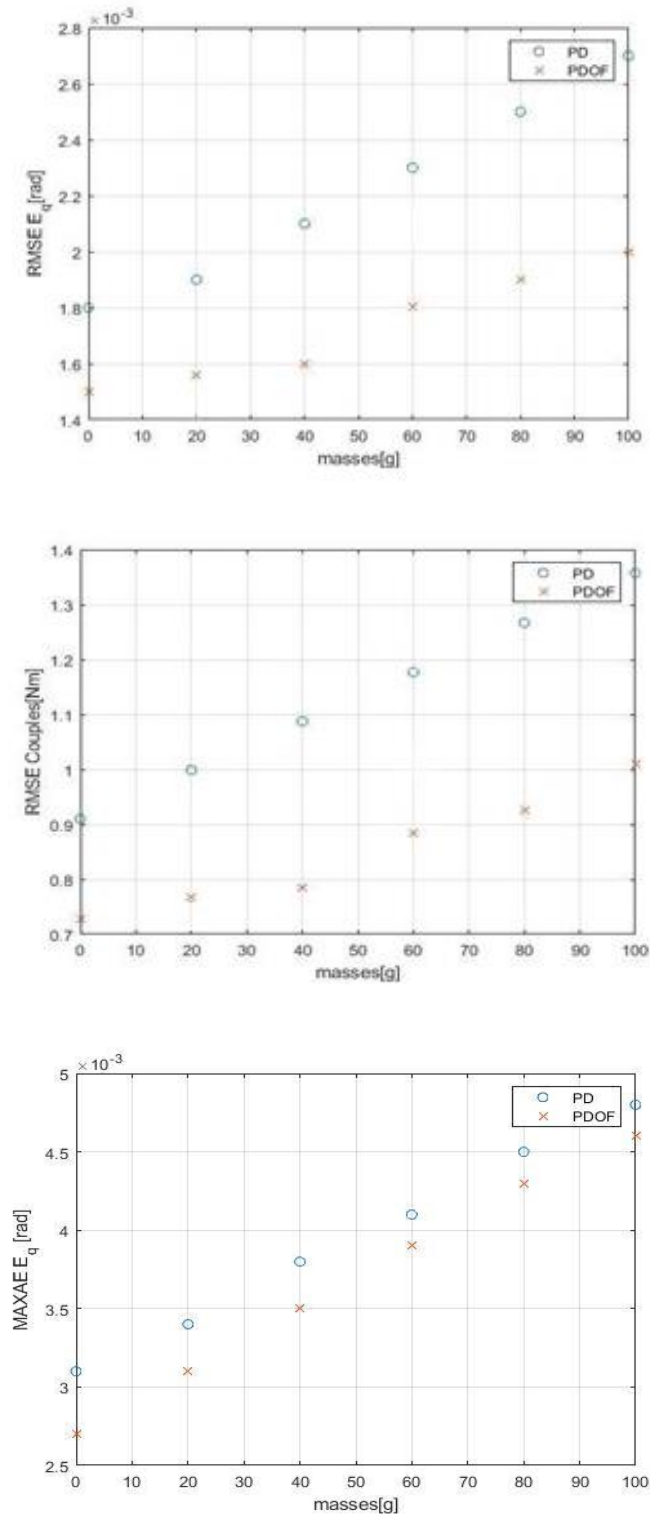


Figure 4.11 – RMS des erreurs, RMS des couples et MAXAE des erreurs(Oustaloup).

Les résultats qu'on vient de citer montrent que la performance et la robustesse du PDOF sont nettement meilleure que celles du PD classique, en effet pour une commande (couple) plus économique énergétiquement on obtient une erreur plus inférieure. On utilisera le filtre d'Oustaloup pour les prochaines simulations.

4.6 Commande PD et PDOF par Algorithme d'Optimisation PSO

4.6.1 Introduction

L'optimisation est une branche des mathématiques cherchant à modéliser, à analyser et à résoudre analytiquement ou numériquement les problèmes qui consistent à minimiser ou maximiser une fonction sur un ensemble donné. Elle joue un rôle très important dans la théorie du contrôle et de la commande.

En effet, de nombreux systèmes susceptibles d'être décrits par un modèle mathématique sont optimisés. La qualité des résultats et des prédictions dépend de la pertinence du modèle, de l'efficacité de l'algorithme et des moyens pour le traitement numérique.

Il existe beaucoup de méthodes d'optimisation dont les méthodes heuristiques et méta-heuristiques qui sont les plus répandues et en fil d'actualité. Or, on a choisi d'employer l'optimisation par essais particulaires (OEP ou PSO) pour trouver les valeurs des paramètres du régulateur qui donne une commande optimale.

4.6.2 Optimisation par Essaim de Particules

L'optimisation par essais particulaires (*OEP* ou *PSO* particule swarm optimisation en anglais) est une méta-heuristique d'optimisation, inventée par Russel Eberhart (ingénieur en électricité) et James Kennedy (socio-psychologue) en 1995.

Cet algorithme s'inspire à l'origine du monde des vivants. Il s'appuie notamment sur un modèle développé par Craig Reynolds à la fin des années 1980, permettant de simuler le déplacement d'un groupe d'oiseaux. Une autre source d'inspiration, revendiquée par les auteurs, James Kennedy et Russel Eberhart, est la socio-psychologie.

Cette méthode d'optimisation se base sur la collaboration des individus entre eux. D'ailleurs, elle a des similarités avec les algorithmes de colonies de fourmis, qui s'appuient eux aussi sur le concept d'auto-organisation. Cette idée veut qu'un groupe d'individus peu intelligents puisse posséder une organisation globale complexe.

Ainsi, grâce à des règles de déplacement très simples (dans l'espace des solutions), les particules peuvent converger progressivement vers un minimum local. Cette méta-heuristique semble cependant mieux fonctionner pour des espaces en variables continues.

Cela donne l'équation de mouvement suivante :

$$V_i(k + 1) = \text{Inertie} + \text{cognitive} + \text{sociale}$$

$$V_i(k + 1) = \omega V_i(k) + c_1 r_1 (Pbest_i - x_i(k)) + c_2 r_2 (Gbest - x_i(k)) \quad (4.27)$$

$Pbest_i$: est la meilleure position de la particule i .

$Gbest$: est la meilleure position calculée.

x_i : la position actuelle de la particule.

ω, c_1, c_2 : Constantes.

r_1, r_2 : variables aléatoires généralement entre [1,2].

L'équation de la position serait donc :

$$x_i(k + 1) = x_i(k) + V_i(k + 1) \quad (4.28)$$

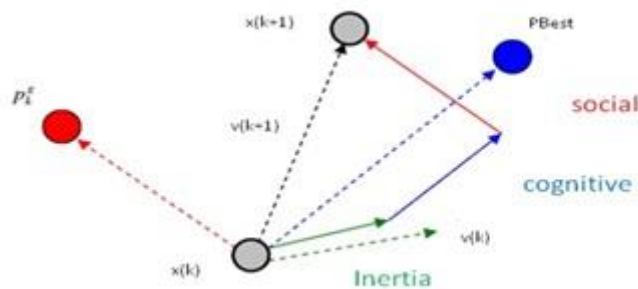


Figure 4.12 – Schéma représentant l'optimisation par essaim de particules

4.6.3 Application du PSO pour le PD et le PDOF

Tout d'abord, on définit la fonction objective (4.29) que nous allons minimiser grâce à l'algorithme PSO

$$J = \sum_{i=1}^N \sum_{j=1}^3 |e_j(i)| \quad (4.29)$$

A noter que ce critère est multiplié par 10.

Les paramètres du régulateur PD sont : $K_p = 112$ et $K_d = 0.0959$.

Avec : $J = 318.8088 \text{ rad}$.

Les résultats de la simulation sont donnés dans la figure 4.13

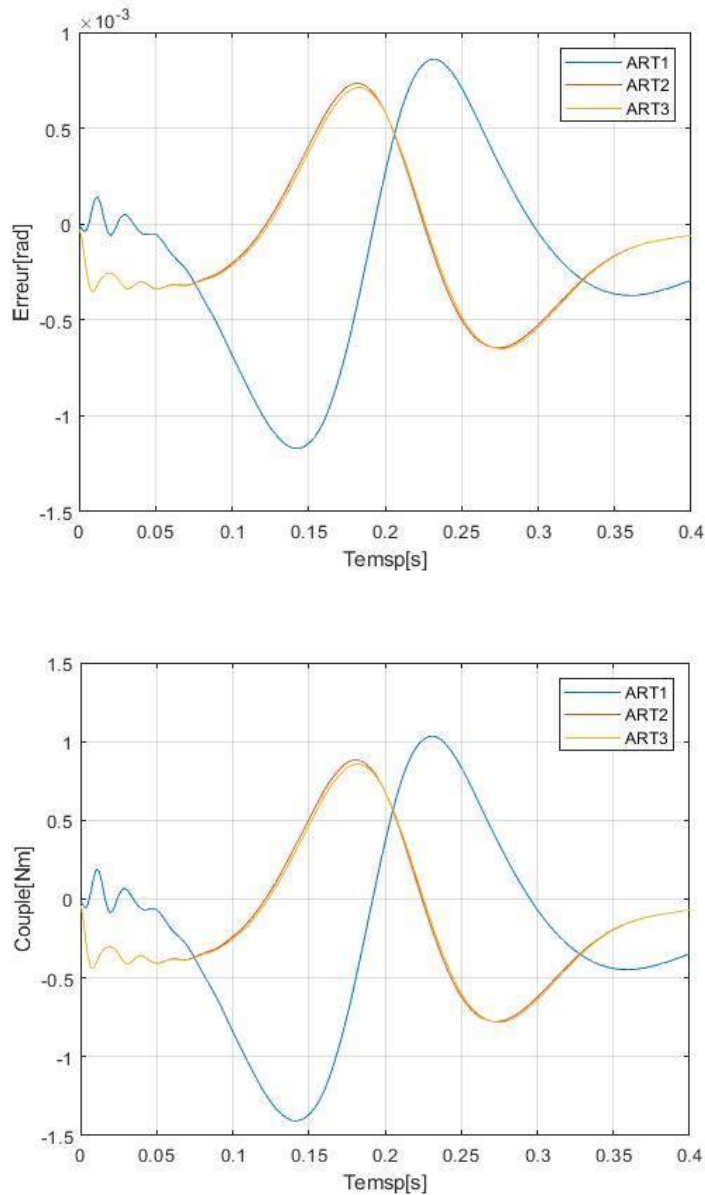


Figure 4.13 – Erreur et couple du PD avec PSO

Par rapport au PD précédent, on remarque que l'erreur de position est toujours de l'ordre de 10^{-3} mais a légèrement diminuée et a atteint une valeur maximale de 1.2 rad. On constate que le couple n'a pas vraiment changé ce qui nous permet de dire que pour la même commande on obtient une erreur inférieure. Cependant, on remarque l'apparition d'oscillation dans le couple au démarrage pendant un laps de temps égale à 0.05 s, cela est dû aux paramètres qu'on a trouvés via PSO qui forcent l'obtention d'une erreur minimale. Ces oscillations peuvent endommager les actionneurs, c'est le seul inconvénient vu que le PD (non optimisé) présente un couple lisse.

En utilisant l'algorithme d'optimisation PSO pour le calcul des paramètres du régulateur PDOF avec filtre d'Oustaloup on trouve $K_p = 99.941$, $K_d = 0.6067$ et $\mu = 0.924$.

Avec : $J = 274.9162 \text{ rad}$.

Les résultats de la simulation sont donnés dans la figure 4.14

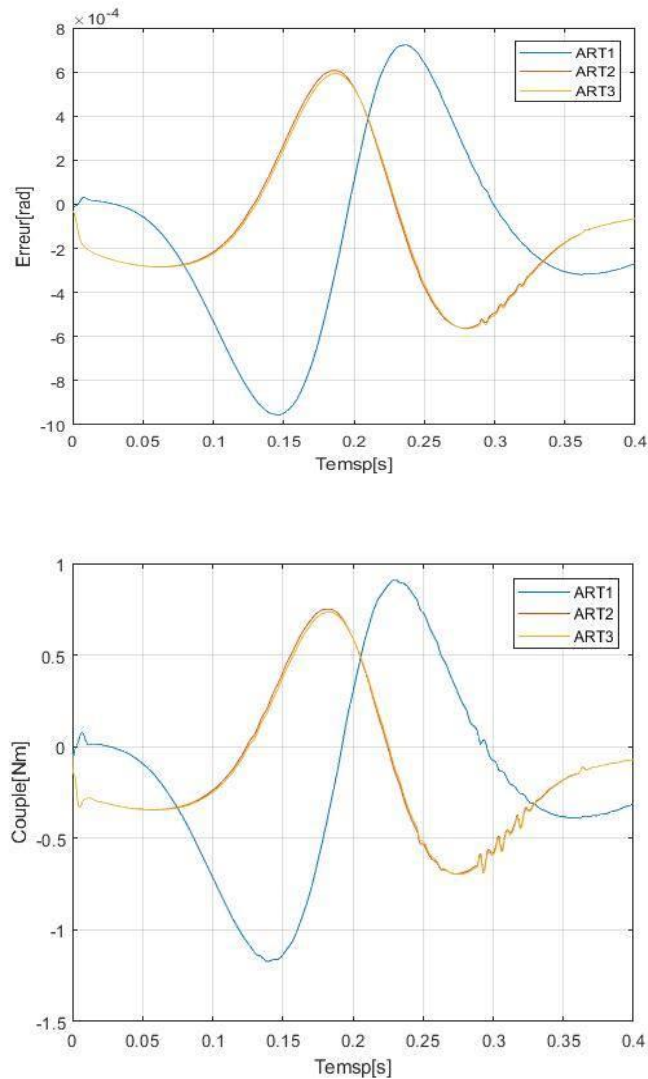


Figure 4.14 – Erreur et couple du PDOF avec PSO

Par rapport au PDOF précédent, On remarque que l'erreur de position a nettement diminué (elle est de l'ordre de 10^{-4} quand à la précédente elle est de l'ordre de 10^{-3}). On constate que le couple n'a pas vraiment changé ce qui nous permet de dire que pour la même commande on obtient une erreur 10 fois inferieure a c'elle du PDOF (non optimisé). Cependant, on remarque l'apparition d'oscillations dans le couple dans l'intervalle de temps [0.25s 0.35s] dues à l'optimisation qui impose une erreur minimale. On peut conclure que le PSO nous

donne de très bons résultats en ce qui concerne l'erreur de position mais introduit des oscillations dans la commande qui peuvent entraîner l'endommagement des actionneurs.

4.6.4 Comparaison entre le PD Classique et le PD d'Ordre Fractionnaire Calculés par PSO

A présent, Nous allons faire une comparaison entre le PD et le PDOF obtenues précédemment. Pour cela, nous testerons la robustesse de ces deux régulateurs pour des valeurs de masses suivantes : 0, 20, 40, 60, 80 et 100g. La comparaison a été faite en utilisant les critères RMSE et MAXAE cité dans le paragraphe 4.4.

La figure 4.15 nous donne les résultats de la comparaison entre les RMSE des erreurs et des couples des deux régulateurs et les MAXAE des erreurs.

Pour des valeurs croissantes de la masse utile, nous remarquons que l'erreur et le couple augmentent pour les deux régulateurs. Les RMSE du PDOF et du PD sont comprises entre $[6.2, 9.5] 10^{-4}$ rad et $[7.7, 11] 10^{-4}$ rad respectivement. La RMS des couples du PDOF et du PD sont comprise entre $[0.75, 1.2]$ Nm et $[0.92, 1.4]$ Nm respectivement. Cependant pour chaque masse, les valeurs obtenues du PDOF sont inférieures aux valeurs obtenues du PD. On remarque aussi que la RMSE et RMSC du PDOF augmentent de manière linéaire (absence de changement brusque de leurs valeurs pour des masses croissantes grâce à une bonne approximation de l'opérateur fractionnaire Oustaloup). On constate aussi que pour $m = 100g$ la RMSE de PDOF est inférieure à la RMSE du PD pour $m = 60g$, ce qui nous permet de déduire que le PDOF est nettement plus robuste que le PD.

On constate aussi que la valeur du MAXAE du PDOF est à chaque fois inférieure à la valeur du MAXAE du PD pour chaque masse.

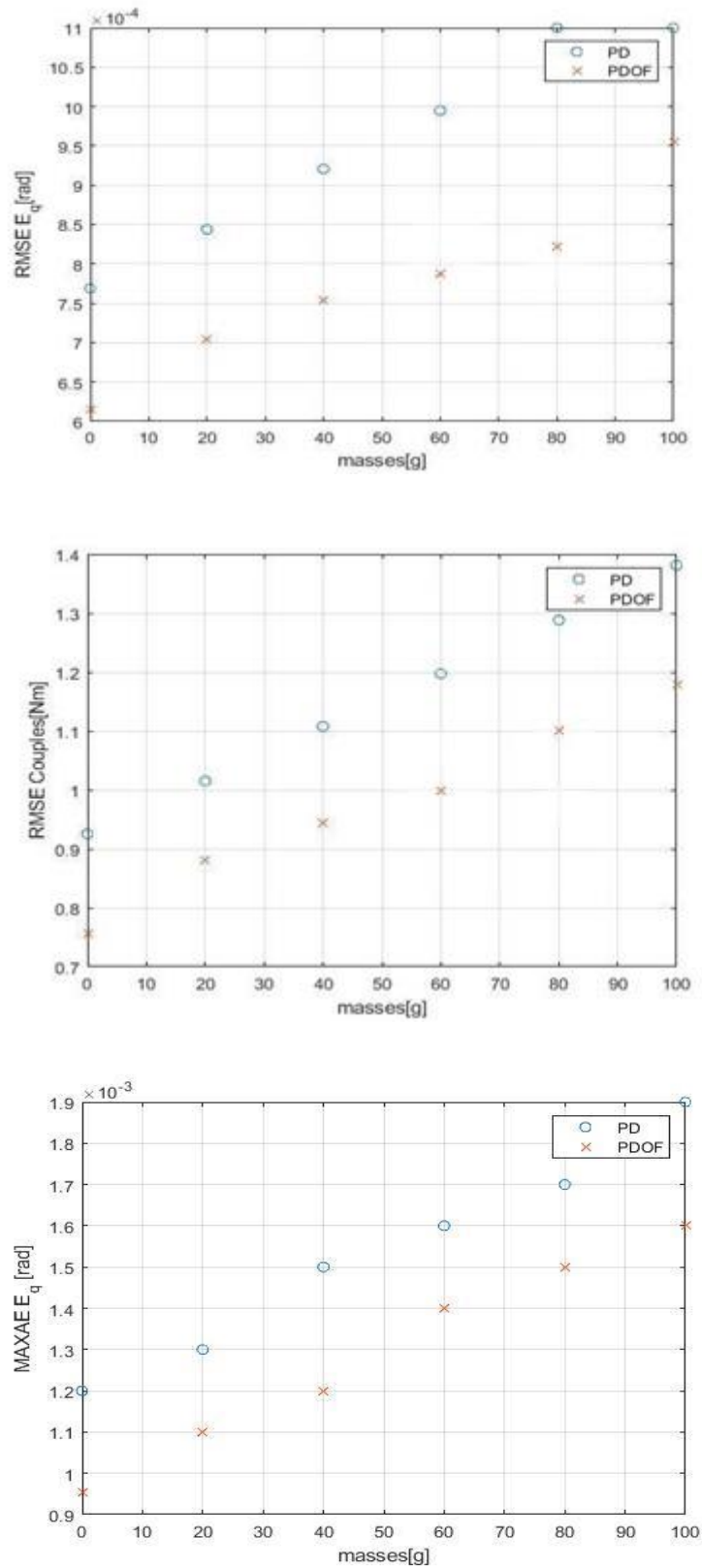


Figure 4.15 – RMS des erreurs, RMS des couples et MAXAE des erreurs.

Les résultats qu'on vient de citer montrent que la performance et la robustesse du PDOF sont nettement meilleures que celles du PD classique et cela même en utilisant un algorithme

d'optimisation pour calculer des paramètres des régulateurs, en effet pour une commande (couple) plus économique énergétiquement on obtient une erreur plus inférieure. Mais ces aspects positifs ne doivent pas néanmoins masquer un inconvénient majeur qui est l'apparition d'oscillations dans la commande.

4.7 Commande par Mode de Glissement

4.7.1 Introduction

La commande par mode glissant est une méthode qui est très utilisée dans l'asservissement des systèmes et qui a prouvé son efficacité à travers les travaux de recherche théoriques effectués et pratiques. Cette approche nous permet de garantir une convergence rapide et en temps fini des erreurs ainsi qu'une grande robustesse par rapport aux erreurs de modélisation.

Cette commande est principalement élaborée en trois étapes [20] :

- Premièrement, on choisit une surface de glissement $s(x, t)$ qui représente une relation adéquate entre les variables d'états du système, elle définit une équation différentielle et détermine donc totalement la dynamique du système à condition qu'il reste dans cette surface.
- Deuxièmement, on détermine une loi de commande attractante qui devra forcer les états du système à converger vers la surface de glissement qui mène vers la position souhaitée.
- Troisièmement, on vérifie la condition de stabilité qui peut être confirmée par la méthode directe de Lyapunov qui permet de vérifier la stabilité d'un état d'équilibre sans passer par la résolution de l'équation d'état du système.

La commande par mode glissant présente des avantages très intéressants. Néanmoins, cette approche procède de manière discontinue et agit directement sur les actionneurs dans la plus parts des cas ce qui peut engendrer une excitation des modes cachés pas forcément pris en compte dans la modélisation.

Dans ce qui suit, nous allons utiliser une surface de glissement appropriée pour le calcul de la commande d'ordre entier (surface de glissement de type proportionnel dérivé) et une surface de glissement appropriée pour le calcul de la commande d'ordre non entier (surface de glissement de type proportionnel dérivé d'ordre fractionnaire).

4.7.2 Commande par Mode Glissement d'Ordre Entier

Nous allons utiliser la surface de glissement de type Slotine donnée dans (4.30) [20].

$$s(x) = \left(\frac{d}{dx} + K_p \right)^{r-1} e(x) \quad (4.30)$$

x : Variable à commander.

$e(x)$: L'erreur de poursuite $e(x) = x - x_d$.

K_p : Une constante positive qui interprétera la dynamique de la surface

r : Le degré relatif du système, peut être interpréter comme le nombre de fois qu'il faut dériver la sortie par rapport au temps pour faire apparaître l'entrée u de manière explicite.

Pour notre système, le degré relatif $r = 2$ pour chaque variable articulaire. En effet, on obtient la surface de glissement sous forme de régulateur proportionnel dérivé (PD) suivante :

$$s_j = K_{pj} e_j + \dot{e}_j, \quad e_j = x_j - x_{jd} \quad j = 1,2,3. \quad (4.31)$$

4.7.2.1 Elaboration de la Commande u_j

Pour le calcul de la commande, nous avons opté pour la structure de commande équivalente présentée dans la figure 4.16 :

L'élaboration de la commande a été détaillée dans [20] en choisissant la structure type commande équivalente (figure 4.16). La commande globale u_j qui stabilise le système est la somme de u_{eq} la commande équivalente obtenue de $\dot{s}(x, t) = 0$ et u_n qui est la commande discontinue ou attractante.

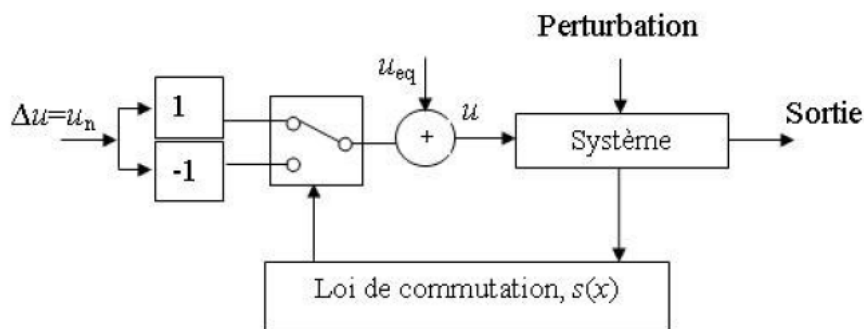


Figure 4.16 – Structure de la commande équivalente [20]

Tel que :

$$u_n = -K_s \text{sign}(s) \quad K_s > 0 \quad (4.32)$$

$$u_{eq} = \frac{-1}{b_j(X)} [K_{Pj} \dot{e}_j + f_j(X) - \ddot{x}_{jd}] \quad (4.33)$$

La commande globale qui satisfait la condition de stabilité est :

$$u_j = \frac{-1}{b_j(X)} [K_{Pj} \dot{e}_j + f_j(X) - \ddot{x}_{jd} + K_{sj} \text{sign}(s_j)] \quad (4.34)$$

Avec :

$$\frac{1}{b_j(X)} = A(q), \quad f_j(X) = A^{-1}(q) (\tau_{c\&c} \dot{q} + \tau_{Gr}) \quad \text{et} \quad \ddot{x}_{jd} = \ddot{q}_{jd}$$

4.7.2.2 Analyse de la Stabilité

En utilisant la première méthode de Lyapunov, on va vérifier la stabilité de la commande. La fonction de Lyapunov définie positive choisie comme suit :

$$V(x) = \frac{1}{2} s^2(x) \quad (4.35)$$

La dérivée de cette fonction est :

$$\dot{V}(x) = s(x) \dot{s}(x) \quad (4.36)$$

Pour que $V(x)$ puisse décroître, il suffit que $\dot{V}(x)$ soit négative. De [20] on obtient :

$$\dot{s}_j = d_j - K_{sj} \text{sign}(s_j) \quad (4.37)$$

d_j est la perturbation qui agit sur le système (voir la figure 4.16). Pour notre cas, elle est considérée comme étant nulle. D'où :

$$s_j \dot{s}_j = -K_{sj} s_j \text{sign}(s_j) \quad (4.38)$$

Ce qui implique :

$$s_j \dot{s}_j \leq -K_{sj} |s_j| \quad (4.39)$$

Pour que le système soit stable, il faut que \dot{V} soit négative donc il suffit que $K_{sj} > 0$.

4.7.3 Commande par Mode Glissement d'Ordre Non Entier

On utilise la surface de glissement de type Slotine pour $r = 2$ définie dans (4.27) [20] :

$$s_j = K_{pj} e_j + D_t^{\alpha_j} e_j \quad 0 < \alpha_j < 1 \quad (4.40)$$

${}_a D_j$: Dérivateur fractionnaire.

α_j : Ordre de dérivation.

Selon la propriété de Caputo présentée dans [20], la surface de glissement peut être réécrite comme suit :

$$s_j = K_{pj} e_j + D_t^{\alpha_j - 1} \dot{e}_j \quad (4.41)$$

4.7.3.1 Elaboration de la Commande u_j d'Ordre Fractionnaire

La synthèse de la commande u_j se fait de la même manière que pour la commande d'ordre entier. La commande globale u_j qui stabilise le système est la somme de u_{eq} la commande équivalente obtenue de $\dot{s}(x, t) = 0$ et u_n la commande discontinue qui peut prendre une forme modifiée et être d'ordre fractionnaire pour des raisons de simplification de calcul et satisfaction de la condition de stabilité. Donc u_j est d'ordre fractionnaire tel que :

$$u_{eq} = \frac{-1}{b_j(X)} [K_{pj} D_t^{(1-\alpha)} (\dot{e}_j) + f_j(X) - \ddot{x}_{jd}] \quad (4.42)$$

La commande globale est donc :

$$u_j = \frac{-1}{b_j(X)} [K_{pj} D_t^{(1-\alpha)} (\dot{e}_j) + f_j(X) - \ddot{x}_{jd} + D_t^{(1-\alpha)} (K_{sj} \text{sign}(s_j))] \quad (4.43)$$

4.7.3.2 Analyse de la Stabilité

Comme illustré dans [20]

$$\dot{s}_j = D_t^{(1-\alpha)} (d_j) + K_{pj} \dot{e}_j(0) - K_{sj} (\text{sign}(s_j)) + K_{sj} \text{sign}(s_j(0)) \quad (4.44)$$

Si on suppose que :

$$K_{pj} \dot{e}_j(0) + K_{sj} \text{sign}(s_j(0)) = 0 \quad (4.45)$$

Et avec une perturbation nulle. On obtient :

$$\begin{aligned} s_j \dot{s}_j &= -K_{sj} s_j \text{sign}(s_j) \\ &\leq -K_{sj} |s_j| \end{aligned} \quad (4.46)$$

Donc la condition de stabilité est vérifiée pour $K_{s_j} > 0$.

$$\text{Autrement, si } \left| K_{p_j} \dot{e}_j(0) + K_{s_j} \text{sign}(s_j(0)) \right| < \xi_j < \infty$$

$$s_j \dot{s}_j \leq (\xi_j - K_{s_j}) |s_j| \quad (4.47)$$

Alors la condition de stabilité est vérifiée pour $K_{s_j} > \xi_j$.

4.7.4 Comparaison entre le Pd et le PD d'Ordre Fractionnaire Calculés par Mode Glissement

Les paramètres K_{p_j} , K_{s_j} et α ont été calculés à l'aide de l'algorithme d'optimisation PSO en utilisant le critère à optimiser précédent (4.29).

Pour la surface de glissement de la forme proportionnelle dérivée, on obtient :

$$K_{p_j} = 483.9904, K_{s_j} = 479.7431.$$

Avec une fonction cout égale à : $J = 1.06$ rad.

Les résultats de la simulation sont donnés dans la figure 4.17

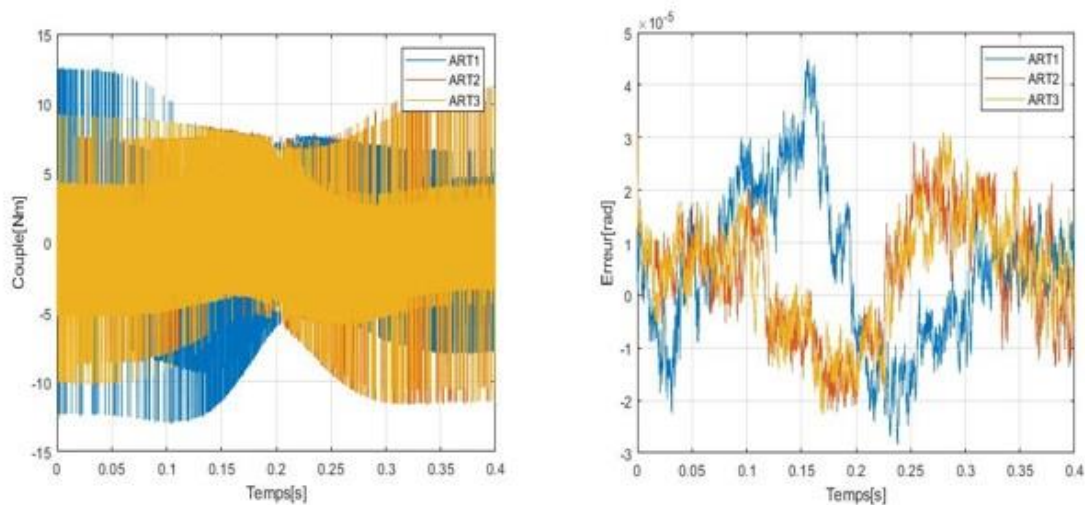


Figure 4.17 – la commande et l’erreur pour la surface PD avec la fonction signe.

Par rapport aux régulateurs PD précédents, On remarque que l’erreur de position a nettement diminué (elle est de l’ordre de 10^{-5} quant aux précédentes elles sont de l’ordre de 10^{-3}). Cependant, le couple a fortement augmenté et a atteint une valeur maximale de 12.5 Nm (10 fois plus important). En plus de cela, la commande n’est plus lisse (le couple présente trop de oscillations) et cela à cause de la fonction Signe qui est certes attractante pour l’erreur mais entraine le phénomène de Chattering qui est nocif pour les actionneurs du robot.

On conclue que cette commande nous donne un très bon résultat concernant l'erreur mais pour un cout énergétique important. On conclue que l'usage de la fonction signe ne donne un résultat non acceptable.

Pour la surface de glissement de la forme proportionnelle dérivée d'ordre fractionnaire, les valeurs sont : $K_{pj} = 440.2, K_{sj} = 407.15$ et $\alpha = 0.91$

Avec une fonction cout égale à : $J = 0.9501$ rad.

Les résultats de la simulation sont donnés dans la figure 4.18

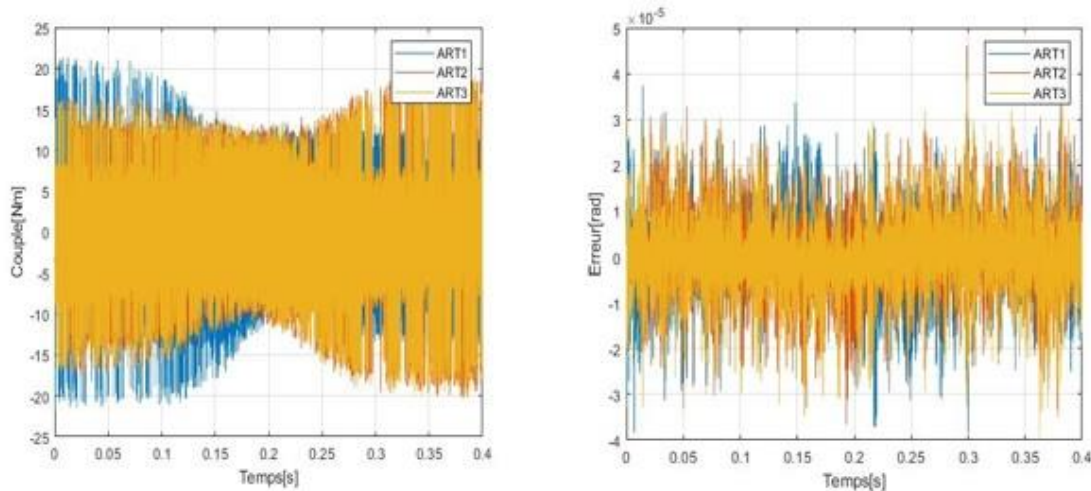


Figure 4.18 – La commande et l'erreur pour la surface PDOF avec la fonction signe

Par rapport aux régulateurs PDOF précédents, On remarque que l'erreur de position a nettement diminué (elle est de l'ordre de 10^{-5} quant aux précédentes elles sont de l'ordre de 10^{-4} pour le PDOF optimisé et de 10^{-3} pour le PDOF non optimisé). Cependant, le couple a fortement augmenté et a atteint une valeur maximale de 21 Nm (20 fois plus important). En plus de cela, la commande n'est plus lisse (le couple présente trop d'oscillations) et cela à cause de la fonction Signe qui est certes attractante pour l'erreur mais entraine le phénomène de Chattering qui est nocif pour les actionneurs du robot.

On conclue que cette commande nous donne un très bon résultat concernant l'erreur mais en contrepartie, un cout énergétique important. On conclue que l'usage de la fonction signe ne donne pas un résultat non acceptable.

On remplace la fonction signe par la fonction suivante élaborée dans [23] pour diminuer les oscillations.

$$M_{\delta}(s_j) = \frac{s_j}{|s_j| + \delta_{j0} + \delta_{j1}|e_j|} \quad (4.48)$$

On garde les mêmes valeurs de K_{pj} , K_{sj} et α précédentes : et on ajuste les valeurs des paramètres δ_{j0} à 0.5 et δ_{j1} à 10. On obtient :

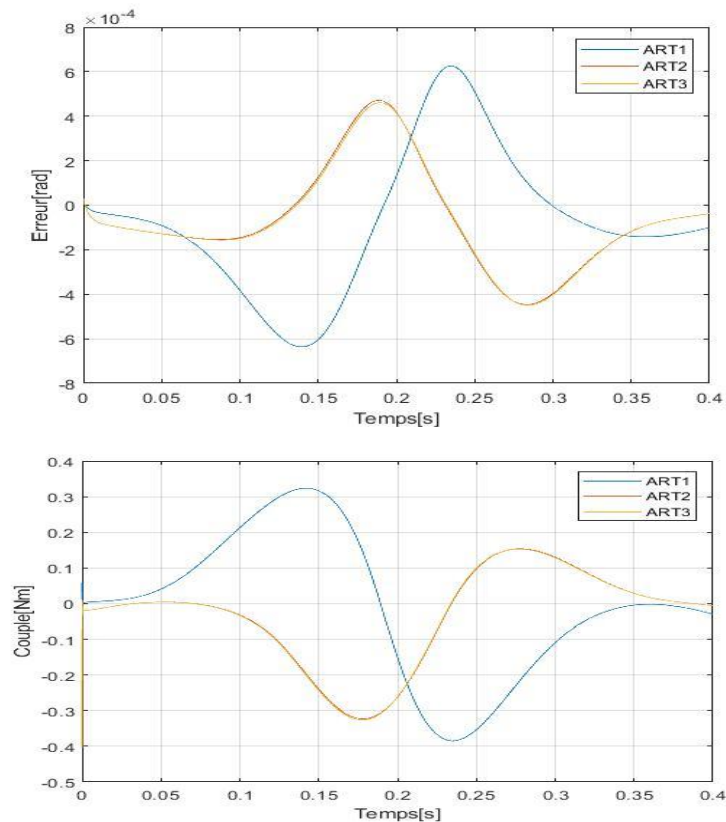


Figure 4.19 – La commande et l’erreur pour la surface PD avec la fonction attractante proposée dans [23]

En changeant la fonction d’attraction Signe, on remarque que l’erreur de position a augmenté (elle est de l’ordre de 10^{-4}). Quant au couple, on constate qu’il a fortement diminué et a atteint une valeur maximale de -0.38 Nm (Hors mis à $t=0$ ou nous remarquons que le couple de démarrage est trop élevé et égale à -0.4 Nm ce qui est dû aux paramètres initiaux imposés (C'est à dire que le système dépense de l’énergie pour se mettre à la position de démarrage)(le terme $f_j(X)$ dans la commande u_j)En plus de cela, les courbe obtenues sont parfaitement lisses (absence d’oscillations), ce qui est un résultat intéressant vu qu’on obtient une erreur acceptable pour une commande énergétiquement minimale comparée aux régulateurs PD calculés auparavant

Pour la commande par mode glissant PDOF, on obtient :

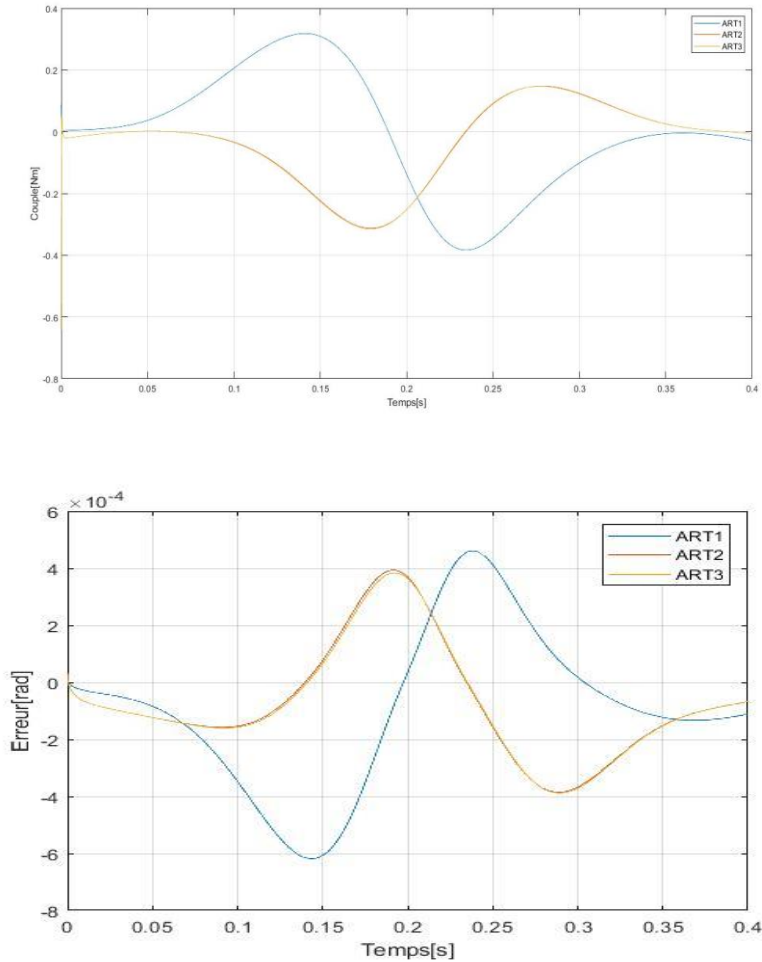


Figure 4.20 – La commande et l’erreur pour la surface PDOF avec la fonction proposée dans [23]

En changeant la fonction d’attraction Signe, Nous remarquons que l’erreur de position a augmenté (elle est de l’ordre de 10^{-4}). Quant au couple, on constate qu’il a fortement diminué et a atteint une valeur maximale de -0.38 Nm (Hors mis à $t=0$ ou nous remarquons que le couple de démarrage est élevé et égale a -0.6Nm qui est dû aux paramètres initiaux imposés (pour les mêmes raisons que pour le cas précédent) En plus de cela, les courbe obtenues sont parfaitement lisses (absence d’oscillations), ce qui est un résultat intéressant vu qu’on obtient une erreur acceptable pour une commande plus économique énergétiquement comparée aux régulateurs PDOF calculés auparavant.

4.7.4.1 Test de Robustesse

A présent, Nous allons faire une comparaison entre la commande par mode glissant PD et PDOF obtenues précédemment. Pour cela, nous testerons la robustesse de ces deux régulateurs pour des valeurs de masses suivantes : 0, 20, 40, 60, 80 et 100g. La comparaison a été faite en utilisant les critères RMSE et MAXAE.

La figure 4.21 nous donne les résultats de la comparaison entre les RMS des erreurs et des couples des deux régulateurs et les MAXAE des erreurs.

Pour des valeurs croissantes de la masse utile, nous remarquons que l'erreur et le couple augmentent pour les deux régulateurs. Les RMSE du PDOF et du PD sont comprises entre $[4.15, 4.25.36] 10^{-4}$ rad et $[4.65, 4.75] 10^{-4}$ rad respectivement. La RMS des couples du PDOF et du PD sont comprise entre $[3.2, 4.9]$ Nm et $[3.3, 5.1]$ Nm respectivement. Cependant pour chaque masse, les valeurs obtenues du PDOF sont inférieures aux valeurs obtenues du PD. On constate aussi que pour les différentes masses utilisées, les valeurs du RMSE et RMS couple augmente linéairement ainsi que pour une masse utile égale à 100g, la RMSE du PDOF est toujours inférieure à celle du PD quelle que soit la valeur de la masse utile ce qui nous permet de déduire que le PDOF est nettement plus robuste que le PD.

Les résultats qu'on vient de citer montrent que la performance et la robustesse du PDOF mode glissement d'ordre non entier sont largement meilleures que celles du PD mode glissement d'ordre entier), en effet pour une commande (couple) plus économique énergétiquement ont obtient une erreur inferieure.

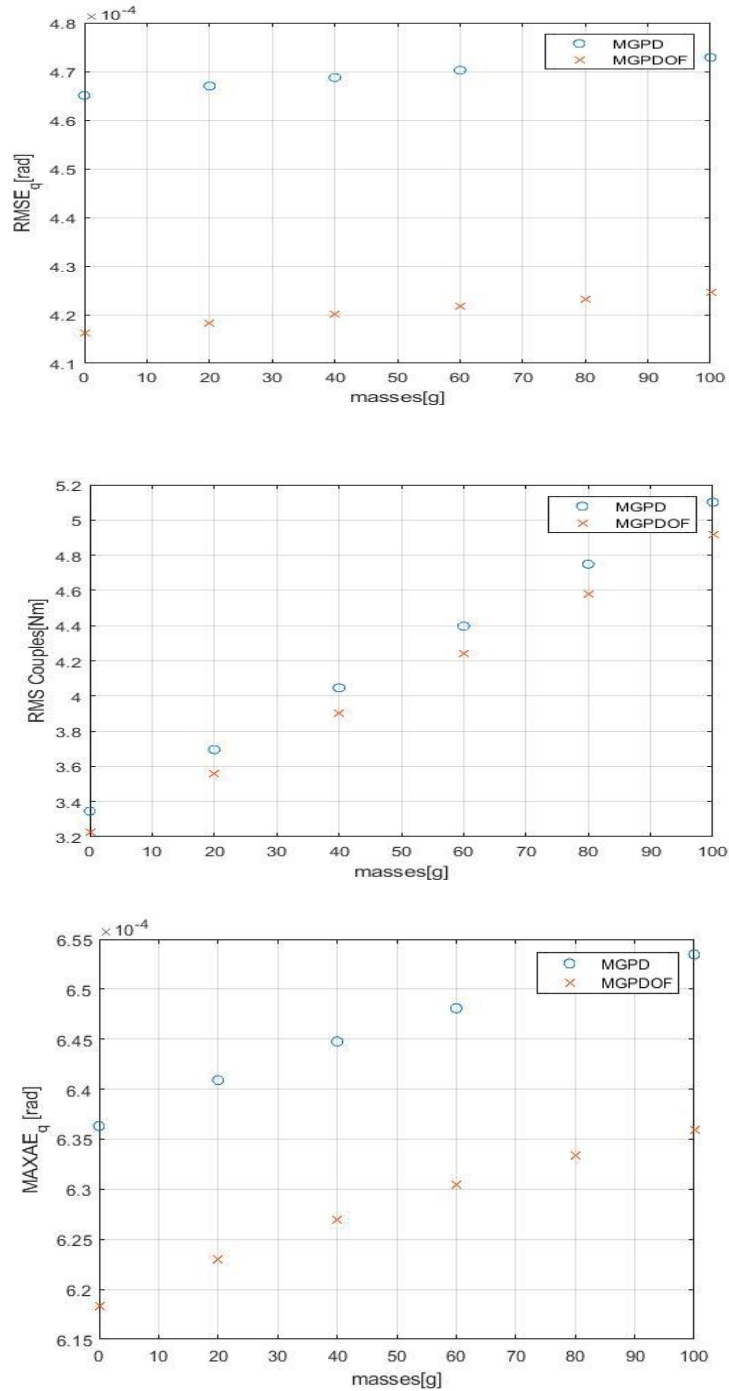


Figure 4.21 – RMS des erreurs, RMS des couples et MAXAE des erreurs

4.7.5 Conclusion

Dans ce chapitre nous nous sommes intéressés au problème de commande du robot Delta ISIR88. Pour ce faire, nous avons fait recours à plusieurs types de commandes à savoir PD, PD d'ordre fractionnaire et Commande par mode de Glissement classique et d'ordre fractionnaire. L'étude a comporté le développement théorique, la simulation et la comparaison des résultats obtenus en mettant en contribution des notions qui ont été nécessaires pour l'élaboration de la loi de

commande à savoir le calcul d'ordre fractionnaire, les critères RMS et MAXAE, la méthode des Moments Généralisés et l'algorithme (PSO).

La commande par mode glissement d'ordre fractionnaire a présenté de meilleures performances et la meilleure robustesse par rapport aux autres commandes. L'explication des résultats obtenus est développée dans le chapitre présent.

Conclusion Générale

Notre travail s'est focalisé sur deux aspects du robot Delta ISIR 88 disponible au Laboratoire de Commande des Processus : La modélisation et la commande.

En ce qui concerne la modélisation, le robot Delta ISIR88 présente une structure parallèle complexe ce qui entraîne l'absence de méthodes systématiques qui permettraient de modéliser ce dernier contrairement au robot série qui présente une structure simple et facile à modéliser (méthode systématique permettant la modélisation). Par ailleurs, nous nous sommes inspirés des travaux de modélisation du robot Delta déjà faits auparavant. En effet, la modélisation géométrique, cinématique et dynamique que nous avons développée dans notre projet se base principalement sur le travail de Coudrey. Bien que les développements de ces modèles s'est basé sur des hypothèses simplificatrices vu la complexité de notre système ce qui a engendré une perte d'information sur la dynamique du robot mais la modélisation obtenue reste assez fiable et cette perte est compensée par des régulateurs robustes. La validité du modèle a ensuite été confirmée.

Concernant la commande du robot, nous nous sommes intéressés au régulateur proportionnel dérivé classique, proportionnel dérivé d'ordre fractionnaire et la commande par mode glissant d'ordre entier (surface de glissement sous forme PD) et d'ordre fractionnaire (surface de glissement sous forme PDOF).

En premier lieu, les paramètres du PD classique ont été obtenus par ajustement en simulation par contre ceux du PDOF en appliquant une méthode de synthèse appropriée pour un système représenté par une fonction de transfert de second ordre qui consiste à imposer une marge de phase et une fréquence de coupure désirées, pour ensuite calculer les paramètres du régulateur d'une manière systématique en utilisant des équations reliant ces paramètres entre eux. Ensuite, nous avons à nouveau déterminé les paramètres de ces deux régulateurs grâce à l'algorithme d'optimisation par Essaim de Particules (PSO).

En dernier lieu, nous avons appliqué une commande par mode glissant d'ordre entier de surface sous forme PD et d'ordre non entier de surface sous forme PDOF. Cette méthode consiste à choisir une surface de glissement puis élaborer une commande stabilisante du système ensuite les paramètres de cette dernière sont déterminés par PSO. A noter que la

fonction signe a introduit le phénomène de réticence (Chattering) qui est néfaste aux actionneurs ce qui nous a mené à la remplacer par une autre fonction attractante pour éliminer ce problème ainsi que l'approximation de l'opérateur fractionnaire par filtre d'Oustaloup donne de meilleurs résultats comparée a l'approximation de FOMCON

A partir des résultats obtenus par simulation sous MATLAB/SIMULINK, une comparaison a été faite entre ces régulateurs. Les meilleure performances en terme de poursuite de trajectoire ont été enregistrées dans le cas de la commande par mode glissant d'ordre non entier de surface sous forme PDOF qui aussi manifeste une meilleure robustesse comparée aux autres régulateurs.

En guise de conclusion, le travail que nous avons fait confirme l'efficacité de la commande d'ordre fractionnaire pour l'asservissement du robot Delta ISIR88 ce qui ouvre le champ d'application de cette dernière sur d'autres systèmes.

Malheureusement, nous n'avons pas pu effectuer la partie expérimentale qui consiste à implémenter les régulateurs synthétisés dans le chapitre 4 à cause des conditions difficiles engendrés par la crise sanitaire (COVID-19). En guise de perspective, nous suggérons l'application de ces commandes sur le robot delta ISIR88.

Référence Bibliographique

- [1] Alain CODOUREY, *Dynamic Modelling and Mass Matrix Evaluation of the DELTA Parallel Robot for Axes Decoupling Control* [en ligne], IEEE, 1996. Disponible sur : <https://ieeexplore.ieee.org/document/568973>
- [2] Stefan Staicu, *Modèle matriciel en dynamique du robot parallèle Stewart-Gough*, Revue roumaine des sciences techniques [en ligne], septembre 2009, No 3, P. 187–200, disponible sur : https://www.researchgate.net/publication/235338625_Model_e_matriciel_en_dynamique_du_robot_parallele_Stewart-Gough
- [3] Sternheim, F. *Computation of the direct and inverse geometric models of the DELTA4 parallel robot*, Robotersysteme, 1987, pp 401-417.[consulté le 10/02/2020], disponible sur : https://link.springer.com/chapter/10.1007%2F978-1-4471-0885-6_29
- [4] Codourey, A. *Contribution B la commande des robots rapides et precis. Application au robot DELTA B entrainement direct*, Ph.D. Thesis, Swiss Federal Institute of Technology Lausanne (EPFL), No. 922 (in french). 1991.
- [5] Laboratoire de Systèmes Robotiques, Life science research office.[consulté le 13/02/2020] disponible sur : <http://lsro.epfl.ch/>.
- [6] KHELIFI Oualid et AZZOUg Larbi, *Modelisation et Commande d'un Robot Delta*, projet de fin d'études : automatique, Alger : ENP, 2010, p105
- [7] Philippe Guglielmetti, *Model-based control of Fast Parallel Robots: A global Approach in Operational Space*, Ph.D. Thesis, Swiss Federal Institute of Technology Lausanne, Janvier 1944.
- [8] Kleinfinger, J.-F. *Modelisation dynamique des robots 2 chaine cinematique simple, arborescente ou fennee, en vue de leur commande*, Ph. D. Thesis, ENSM, Nantes, France, 1986.
- [9] Kokkinis, T., Stoughton R. *Dynamics and control of closed-chain robot arms with application to a new direct-drive robot arm*, International Journal of Robotics and Automation, Vol. 6, No. 1, pp. 25-34, 1991.

- [10] Pierrot, F., Dauchez, P., Fournier, A., *HEXA: a fast six-DOF fully-parallel robot*, *Proceedings of '91 ICAR*, International Conference on Advanced Robotics, Pisa, Italy, juin 1991.
- [11] Reboulet, C., Berthomieu, T. *Dynamic models of a six degree of freedom parallel manipulator*, *Proceedings of '91 ICAR*, International Conference on Advanced Robotics, Pisa, Italy, juin 1991
- [12] Codourey, A., Clavel, R., Burckhardt, C.W, *Control Algorithm and Controller for the Direct Drive Delta Robot*, *Proc. of the IFAC Symposium on Robot Control*, pp. 169-177, Septembre 1991.
- [13] Zhang, C-D., Song, S-M., *An Efficient Method for Inverse Dynamics of Manipulators Based on the Virtual Work. Principle*, *Journal of Robotic Systems* 10(5), DD. 605-627, 1993.
- [14] Lebret, G., Liu, K., Lewis, F. L. *Dynamic Analysis and Control of a Stewa Platform Manipulator*, *Journal of Robotic Systems* 10(5), pp. 629-655, 1993.
- [15] G.lebret, K.liu, F.L.lewis, *Dynamics of a Parallel Manipulator*, *Journal of Robotic Systems* [en ligne], 1991, 629-655. Disponible sur:
<<https://www.onlinelibrary.wiley.com/doi/abs/10.1002/rob.4620100506>>
- [16] Miller, K., *The proposal of a New Model of Direct Drive Robot DELTA-4 Dynamics*, *Proceedings of '93 ICAR*, International Conference on Advanced Robotics, Tokyo, Japan Novembre 1993.
- [17] Concepción A. Monje, YangQuan Chen, Blas M. Vinagre, Dingyü Xue, Vicente Feliu. *Fractional-order Systems and Controls Fundamentals and Applications* [en ligne], London: Springer-Verlag London Limited 2010 [2020]. PDF. Disponible sur:
<<https://www.springer.com/gp/book/9781849963343>>
- [18] Chems Eddine Boudjedir, Djamel Boukhetala, and Mohamed Bouri. *Iterative Learning Control of a Parallel Delta Robot*, Springer Nature Switzerland AG [en ligne], M. Chadli et al. (Eds.): ICEECA, 2017, LNEE 522, pp. 72–83 disponible sur :
<https://link.springer.com/chapter/10.1007%2F978-3-319-97816-1_6>
- [19] J. Kennedy et R. Eberhart, *Particle swarm optimization*, IEEE International Conference on Neural Networks. Proceedings, vol. 4, novembre 1995.

- [20] Noureddine BOUARROUDJ, *Commandes par Mode de Glissement d'Ordre Fractionnaire Optimisées par Métaheuristiques. Application aux Systèmes Non Linéaires et Interconnectés*, Thèse de Doctorat en Sciences en Génie Électrique, Février 2017.
- [21] CLAVEL,Reymond. *Conception d'un robot parallèle rapide à 4 degrés de liberté*.146 p. Thèse de Doctorat : microtechnique. Lausanne, EPFL. 1991.
[Consultée le 22/02/2020]
- [22] Anne François, José Ragod, Alain Richard. *Identification assistée par ordinateur : La méthode des moments*. Le Nouvel, Automatismes.1984,
- [23] DJAMEL Boukhetala, FARES Boudjema, TAREK Madani, Mohamed Seghir Boucherit. *A new decentralized variable Structure control for robot Manipulators, International Journal of Robotics and Automation* [en ligne]. 2003, vol.18, No.1, [consulté le 15/06/2020], disponible sur :
<https://www.researchgate.net/publication/287008250_A_new_decentralized_variable_Structure_control_for_robot_Manipulators>
- [24] V. Shekher , P.Rai et O.Prakash, *Tuning and Analysis of Fractional Order PID Controller*, In : International Journal of Electronic and Electrical Engineering. 2012, Vol 5, No 1, 0974-2174.
- [25] Abdelelah Kidher Mahmood, *Design Optimal Fractional Order PID Controller Utilizing Particle Swarm Optimization Algorithm and Discretization Method (IJESE)*, ISSN, 2013. 2319–6378, Vol 1, Issue-10.
- [26] A. Oustaloup, F. Levron, B. Mathieu, and F. M. Nanot, "Frequency-Band Complex Noninteger Differentiator: Characterization and Synthesis", *IEEE transactions on circuits and systems-I: fundamental theory and applications*, vol. 47, nO. 1, pp. 25-39, Janvier 2000.
- [27] Abdeldjalil BENSALID, Cherif MOKRAOUI, *Modélisation et Commande d'un Robot de Type Delta de la Simulation à l'Expérimentation*, Projet de fin d'étude : Automatique. Alger : Ecole nationale polytechnique, 2019, 103p.