



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

École Nationale Polytechnique
Département d'Automatique



Mémoire de projet de fin d'études

pour l'obtention du diplôme d'Ingénieur d'État en Automatique

Modélisation et commande à retard et d'ordre fractionnaire d'un robot parallèle de type Delta

Réalisé par :

M. CHALAL Ahmed

Mlle. MEDDOUR Imane

Supervisé par :

Pr. BOUKHETALA Djamel

Dr. BOUDJEDIR Chems

Eddine

Présenté et soutenu publiquement le 4 Juillet 2023.

Membres du jury :

Président	M. BOUCHERIT MOHAMED SEGHIR	Pr	ENP
Promoteur	M. BOUKHETALA DJAMEL	Pr	ENP
	M. BOUDJEDIR CHEMS EDDINE	Dr	USTHB
Examineur	M. CHAKIR MESSAOUD	Pr	ENP

ENP 2023



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

École Nationale Polytechnique
Département d'Automatique



Mémoire de projet de fin d'études

pour l'obtention du diplôme d'Ingénieur d'État en Automatique

Modélisation et commande à retard et d'ordre fractionnaire d'un robot parallèle de type Delta

Réalisé par :

M. CHALAL Ahmed

Mlle. MEDDOUR Imane

Supervisé par :

Pr. BOUKHETALA Djamel

Dr. BOUDJEDIR Chems

Eddine

Présenté et soutenu publiquement le 4 Juillet 2023.

Membres du jury :

Président	M. BOUCHERIT MOHAMED SEGHIR	Pr	ENP
Promoteur	M. BOUKHETALA DJAMEL	Pr	ENP
	M. BOUDJEDIR CHEMS EDDINE	Dr	USTHB
Examineur	M. CHAKIR MESSAOUD	Pr	ENP

ENP 2023

Dédicace

“

*En signe de gratitude et de respect, je dédie ce modeste
travail :*

*Aux êtres les plus chers au monde qui n'ont pas cessé de
me fournir leur soutien et leurs encouragements : mes
parents qui ont éclairé mon chemin et qui m'ont encouragé
et soutenu dans les moments difficiles de ma carrière, ceux
qui ont fait de moi ce que je suis.*

À mes frères Mohamed et Abdennour.

À toute ma famille.

À tous mes amis qui m'ont soutenu.

”

Imane

Dédicace

“

*En signe de gratitude et de respect, je dédie ce modeste
travail :*

*A celle qui m'a donnée vie, celle qui me nourrit d'amour,
celle qui m'a appris tout ce que je sais et ma soutenu toute
ma vie, à ma chère mère.*

*A mon idole, mon exemple à suivre, le héros de toute mes
histoire, a mon père.*

*A mon défunt grand père Messaoud, et à mon défunt frère
d'âme Fares.*

A toute ma famille et a tout ceux qui mon aidé et soutenu.

”

Ahmed

Remerciement

Tout d'abord, nous remercions Dieu pour nous avoir donné le courage, la volonté et la patience de réaliser ce travail.

Nous remercions nos parents, qui nous ont soutenus tout au long de nos études.

Nous tenons à exprimer notre gratitude à nos encadrants, Pr. Djamel BOUKHETALA et Dr. Chems Eddine BOUDJEDIR, pour leur soutien constant, leurs conseils et leurs éclaircissements tout au long de la réalisation de ce projet de fin d'étude.

Nous voudrions également remercier par avance les membres du jury d'avoir pris le temps de lire notre travail.

ملخص

يتعلق العمل المعروض في هذه الوثيقة بنمذجة روبوت من نوع دلتا والتحكم فيه. نبدأ بمراجعة الفئات المختلفة للروبوتات ، وبشكل أخص الروبوتات المتوازية ، يليها عرض تقديمي لروبوت دلتا ومجالات استعماله. يتبع ذلك نمذجة شاملة لروبوت ISIR Delta 88. تم تصميم مختلف المتحكمات و المقارنة بين نتائجها (PD ، PD بأس كسري ، ASMC مع الوضع الانزلاقي) تم إختيار جميع عوامل المتحكمات المقترحة بواسطة خوارزمية جينية وراثية. ينتهي هذا العمل بإستنتاج عام يسرد النتائج المختلفة التي تم تحديدها خلال المشروع.

كلمات مفتاحية: روبوت دلتا, نمذجة و تحكم, تحكم بأس كسري, التحكم بالإعتماد على المعلومات السابقة, خوارزمية وراثية .

Abstract

The work presented in this document concerns the modeling and the control of a Delta robot. We begin with a review of the different categories of robots, more specifically parallel robots, followed by a presentation of the Delta robot and its fields of application. This is followed by a global modeling of the Delta ISIR 88 robot. Several controllers were synthesized and compared (PD, fractional order PD, TDC with sliding mode and ASMC TDC). All of the controller parameters are optimized by a genetic algorithm.

Keywords : Delta Robot ISIR88, modeling and control, fractional order controllers, TDC, genetic algorithm.

Résumé

Le travail présenté dans ce mémoire concerne la modélisation et la commande d'un robot Delta . Nous commençons par une présentation des catégories de robots, plus précisément les robots parallèles, suivi d'une présentation du robot Delta avec leur domaines d'application. Par la suite, une modélisation globale du robot Delta ISIR 88 a été présentée. Plusieurs commandes ont été synthétisées et comparées (PD, PD d'ordre fractionnaire, TDC avec mode glissant et ASMC TDC). Les paramètres de l'ensemble des commande proposées sont optimisés a l'aide d'un algorithme génétique,méta heuristique.

Mots clés : Robot Delta ISIR88, modélisation et commande, commande d'ordre fractionnaire, TDC, algorithme génétique.

Table des matières

Liste des Figures

Liste des Tableaux

Liste des Abbreviations

Introduction Générale	13
1 Robot Delta	15
1.1 Introduction	16
1.2 Classification des robots	16
1.3 Les manipulateurs industriels	19
1.3.1 Robot sériel	20
1.3.2 Robot parallèle	21
1.3.3 Comparaison entre les robots sériels et parallèles	21
1.3.4 Historique des robots parallèles	22
1.4 Le robot Delta	23
1.4.1 Caractéristique du robot Delta	24
1.4.2 Application et domaine d'utilisation	25
1.4.2.1 Prise-dépose	25
1.4.2.2 Impression 3d	25
1.5 Description du robot Delta ISIR88 disponible au laboratoire de commande des processus (LCP)	26
1.6 Conclusion	28
2 Modélisation du robot Delta	29
2.1 Introduction	30
2.2 Modélisation géométrique	30
2.2.1 Modèle géométrique direct	30
2.2.2 Modèle géométrique inverse	35

2.3	Modélisation cinématique	36
2.3.1	Modèle cinématique direct	37
2.3.2	Modèle cinématique inverse	38
2.4	Modèle des accélérations	38
2.5	La dynamique de l'actionneur	39
2.6	Paramètres dynamiques	39
2.7	Calcul du modèle dynamique en utilisant le principe du travail virtuel . . .	40
2.8	Singularités du robot Delta	42
2.9	Volume de travail	43
2.10	Génération de mouvement et de trajectoire	44
2.10.1	Lois de mouvement	45
2.10.1.1	Loi Bang-Bang	45
2.10.1.2	Loi Trapèze	46
2.10.2	Trajectoire	47
2.11	Conclusion	49
3	Commande PD d'ordre fractionnaire	50
3.1	Introduction	51
3.2	Opérateurs d'ordre fractionnaire	51
3.2.1	Définitions	51
3.2.1.1	Définition de Grünwald-Leitnikov	51
3.2.1.2	Définition de Riemann-Liouville	52
3.2.1.3	Définition de Caputo	52
3.2.2	Propriétés des opérateurs d'ordre fractionnaire	52
3.2.3	Transformée de Laplace des opérateurs d'ordre fractionnaire	53
3.2.3.1	Transformée de Laplace de la dérivée d'ordre fractionnaire	53
3.2.3.2	Transformée de Laplace de l'intégrale d'ordre fractionnaire	54
3.2.4	Approximation analogique des opérateurs d'ordre fractionnaire . . .	54
3.3	Régulateur PID	55
3.4	Commande par PD	55
3.5	Commande par PD d'ordre fractionnaire	56
3.6	L'algorithme génétique	57
3.7	Simulation	59
3.7.1	Test de poursuite	59
3.7.2	Test de robustesse	61
3.7.3	Comparaison et discussion	62

3.8	Conclusion	63
4	La commande "Time Delay Control"	64
4.1	Introduction	65
4.2	Time Delay Control	65
4.3	La TDC par mode glissement d'ordre fractionnaire	67
4.3.1	Le mode glissant et le phénomène de Chattering	67
4.3.2	Critère de choix de la surface	68
4.3.3	Conception de la commande et analyse de la converge	70
4.4	Commande adaptative avec mode glissement basée sur la TDC	71
4.4.1	Conception de la commande	71
4.4.2	Analyse de la convergence	72
4.5	Simulation	74
4.5.1	Test de poursuite	75
4.5.2	Test de robustesse	78
4.5.3	Comparaison et discussion	80
4.6	Conclusion	81
	Conclusion Générale	82
	Bibliographie	84

Table des figures

1.1	Bras manipulateur Kuka	16
1.2	Robot-café	17
1.3	Robot mobile Rover	17
1.4	Humanoïde de Boston Dynamics	18
1.5	Robot éducatif	18
1.6	Robot de recherche	19
1.7	Représentation d'un point	20
1.8	Robot SCARA Fanuc SR-3iA	21
1.9	Robot parallèle Fanuc	21
1.10	Un des premiers robots parallèles, inventé en 1928 par Gwinnett	22
1.11	Plateforme de Gough 1947	23
1.12	Robot Delta	24
1.13	Robot Delta prise-dépose	25
1.14	Imprimantes 3D Delta	26
1.15	Robot Delta ISIR88	26
2.1	Modèle géométrique direct du robot Delta	30
2.2	Bras du Delta	31
2.3	Vue du dessus de la base du Delta	31
2.4	Schéma d'un bras du Delta	32
2.5	Schéma simplifié d'un bras du Delta	32
2.6	Sphère de centre P et de rayon l_2 passant par C_1	33
2.7	Modèle géométrique inverse du robot Delta	35
2.8	Actionneur avec réducteur	39
2.9	Force gravitationnelle agissant sur la partie supérieure du bras d'un robot Delta	41
2.10	Représentation des 4 types de singularités du robot Delta	42
2.11	Volume de travail considéré du Delta	44
2.12	Génération de mouvement dans l'espace articulaire	45

2.13	Génération de mouvement dans l'espace opérationnel	45
2.14	Loi de mouvement Bang-bang	46
2.15	Loi de mouvement en Trapèze	46
2.16	Projection de la coordonnée curviligne $s(t)$ donnant $x(t)$; $y(t)$; $z(t)$	47
2.17	Trajectoire ellipsoïde	48
3.1	Schéma de commande du robot Delta avec régulateur PD	56
3.2	Structure d'un régulateur PD d'ordre fractionnaire	56
3.3	Hierarchie dans l'AG	58
3.4	Poursuite de trajectoire dans l'espace opérationnel	59
3.5	Erreurs de position opérationnelle obtenus par les deux régulateurs	60
3.6	Couples obtenus par la simulation des deux régulateurs	60
3.7	Poursuite de trajectoire dans l'espace opérationnel avec une masse de 250g	61
3.8	Erreurs de position opérationnelle obtenus par les deux régulateurs avec une masse de 250g	61
3.9	Couples obtenus par la simulation des deux régulateurs avec une masse de 250g	62
4.1	Schéma du régulateur TDC	67
4.2	Fonction "saturation"	68
4.3	Effet de b_1 sur la convergence de l'erreur	69
4.4	Effet de α sur la convergence de l'erreur	69
4.5	Structure de commande TDC S_1	70
4.6	Structure de commande ASMC	72
4.7	Erreurs de position opérationnelle obtenus en utilisant les deux méthodes TDC avec la fonction sign	75
4.8	Couples de commande obtenus par les deux méthodes TDC avec la fonction sign	76
4.9	Poursuite de trajectoire dans l'espace opérationnel	77
4.10	Erreurs de position opérationnelle obtenus en utilisant les deux méthodes TDC	77
4.11	Couples de commande obtenus par les deux méthodes TDC	78
4.12	Poursuite de trajectoire dans l'espace opérationnel avec une masse de 250g et une perturbation	78
4.13	Erreurs de position opérationnelle obtenus par les deux méthodes TDC avec une masse de 250g	79
4.14	Couples obtenus par la simulation des deux méthodes TDC avec une masse de 250g et une perturbation	79

Liste des tableaux

1.1	Tableau comparatif entre robot sériel et robot parallèle	21
1.2	Paramètres physiques du robot	27
3.1	Tableau d'effet de chaque action	55
3.2	Performances à travers le critère RMSE	62
3.3	Performances de poursuite pour une charge de 250 g à travers le critère RMSE	63
4.1	Performances selon le critère RMSE	80
4.2	Performances de poursuite pour une charge de 250 g et une perturbation à travers le critère RMSE	80

Liste des Abbreviations

ASMC	<i>Adaptative Sliding Mode Control</i>
ENP	<i>Ecole Nationale Polytechnique</i>
EPFL	<i>École Polytechnique Fédérale de Lausanne</i>
FOPD	<i>Proportionnel Dérivée d'Ordre Fractionnaire</i>
ISO	<i>International Standard Organization</i>
LCP	<i>Laboratoire de Commande des Processus</i>
LSRO	<i>Laboratoire de Système Robotiques</i>
MCD	<i>Modèle Cinématique Direct</i>
MCI	<i>Modèle Cinématique Inverse</i>
MDD	<i>Modèle Dynamique Direct</i>
MGD	<i>Modèle Géométrique Direct</i>
MGI	<i>Modèle Géométrique Inverse</i>
PD	<i>Proportionnel Dérivée</i>
RIA	<i>Robot Institute of America</i>
RMSE	<i>Root Mean Square Error</i>
RUR	<i>Rossum's Universal Robots</i>
TDC	<i>Time Delay Control</i>
TDCSM	<i>Time Delay Control with Sliding Mode</i>
TDE	<i>Time Delay Estimation</i>

Introduction Générale

La robotique est la discipline s'intéressant au fonctionnement et à l'utilisation des robots pour effectuer des tâches de manière automatique. À ses débuts, les connaissances et les moyens techniques ne permettaient d'envisager que des tâches extrêmement simples, dans des environnements maîtrisés et connus auxquels le robot est dédié. Cependant de nos jours, l'objectif de la robotique est beaucoup plus ambitieux. Il consiste à construire des robots capables d'évoluer dans des environnements variés, connus ou non. Les robots doivent également réaliser des tâches de plus en plus complexes, de la manière la plus autonome possible. Ces tâches sont généralement exécutées avec un degré de précision élevé. En effet, ces robots doivent être dotés de systèmes de commande assez évolués afin de satisfaire les performances exigées.

Dans le cadre de ce projet de fin d'études, nous avons travaillé sur une famille de robots manipulateurs assez particulière : les robots parallèles. Nous nous sommes focalisé sur le cas du robot Delta ISIR88 à 4 degré de liberté conçu par l'équipe de Clavel de l'EPFL, disponible au niveau du Laboratoire de Commande des Processus de l'École Nationale Polytechnique.

Le présent mémoire se compose de quatre chapitres, en plus de l'introduction générale et de la conclusion générale.

Dans le premier chapitre, nous présenterons les différentes catégories des robots plus précisément les robots parallèles, on commence par un bref historique, puis une présentation du robot Delta, ses domaines d'application et quelques notions nécessaires pour entamer notre sujet principal qui est la modélisation et la commande du robot Delta.

Dans le deuxième chapitre, nous donnons le modèle géométrique direct et inverse, cinématique direct et inverse ainsi que le modèle dynamique du robot Delta ISIR88 disponible au Laboratoire de Commande des Processus (LCP) avec quelques caractéristiques et spécifications géométriques nécessaires pour notre étude comme l'espace de travail, les configurations singulières, et la génération de la trajectoire.

Dans le troisième chapitre, on commence par une introduction mathématique aux opérateurs fractionnaires puis on présentera le régulateur PD et PD d'ordre fractionnaire, ainsi que les résultats des simulations de ces deux régulateurs sur le robot Delta ISIR88 et la comparaison entre leurs résultats. Les paramètres des deux régulateurs sont calculés à l'aide d'un algorithme d'optimisation génétique, un algorithme métaheuristique qui estime les gains et les ordres fractionnaires.

Dans le quatrième chapitre, nous allons synthétiser théoriquement les commandes que nous allons appliquer sur le robot Delta ISIR88 à savoir la commande Time delay control

(TDC) simple, la TDC avec mode glissant d'ordre fractionnaire, afin de tenter d'obtenir de meilleurs résultats on s'est proposé d'améliorer ce dernier en faisant deux changements, le premier étant d'améliorer l'estimateur TDE, le deuxième est l'introduction d'un gain adaptatif. Le chapitre se termine par une étude comparative entre les deux commandes et leurs résultats.

Enfin, on clôture le projet de fin d'étude par une conclusion générale englobant les perspectives et les investigations futures à poursuivre.

Chapitre 1

Robot Delta

1.1 Introduction

Le mot robot est un vaste terme qui peut évoquer différentes images dans nos têtes, mais en réalité, le mot robot découle du mot tchèque « robota » qui signifie « travail, corvée », et fut utilisée la première fois dans la pièce de théâtre (science-fiction) R. U. R. (Rossum's Universal Robots) écrite en 1920 par l'auteur Karel Čapek¹ [1]. De nos jours on peut définir un robot comme une machine conçue pour effectuer diverses tâches avec une grande précision, vitesse et exactitude de manière automatique ou télécommandée. Ils peuvent être utilisés dans un large éventail de situations, allant des chaînes de fabrication et d'assemblage à la médecine jusqu'à l'exploration scientifique et aux missions spatiales. Ils se présentent sous différentes formes et tailles selon le besoin et l'environnement.

Dans ce chapitre, nous présenterons les catégories des robots plus précisément les robots parallèles avec quelques définitions de base, leurs domaines d'application et les notions nécessaires pour entamer notre sujet principal qui est la modélisation et la commande du robot Delta.

1.2 Classification des robots

Il existe de nombreuses façons de classer les robots, mais voici une classification courante des robots :

a. Les robots industriels

Il s'agit de robots utilisés dans le secteur de la fabrication pour effectuer des tâches telles que le soudage, la peinture et l'assemblage. Ils sont souvent de grande taille et très robustes, avec plusieurs axes de mouvement et une grande capacité de charge [2].



FIG. 1.1 : Bras manipulateur Kuka

b. Les robots de service

Il s'agit de robots conçus pour effectuer des tâches dans des environnements tels que les hôpitaux, les maisons et les bureaux. Ils peuvent être utilisés pour des tâches telles que le nettoyage, la sécurité et le service à la clientèle. Les robots aspirateurs, les robots de sécurité et les robots de téléprésence en sont des exemples.



FIG. 1.2 : Robot-café

c. Les robots mobiles

Il s'agit de robots conçus pour se déplacer dans leur environnement, grâce à des roues, des jambes ou d'autres moyens de locomotion. Les robots de livraison, les robots d'exploration et les robots militaires en sont des exemples.[2]

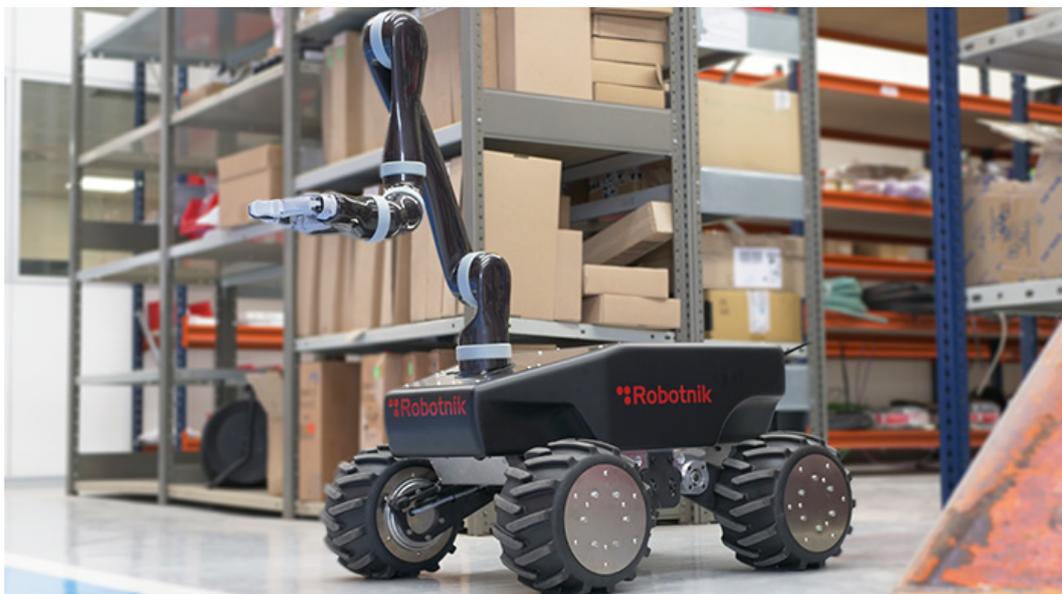


FIG. 1.3 : Robot mobile Rover

d. Les robots humanoïdes

Il s'agit de robots conçus pour ressembler à des humains et se déplacer comme eux. Ils ont souvent des bras, des jambes et une tête, et sont capables d'effectuer des tâches telles que marcher, saisir des objets et communiquer avec les humains.[2]



FIG. 1.4 : Humanoïde de Boston Dynamics

e. Les robots éducatifs

Il s'agit de robots conçus à des fins éducatives, par exemple pour enseigner des concepts de programmation ou de robotique à des étudiants. Il peut s'agir de simples kits ou de systèmes complexes.

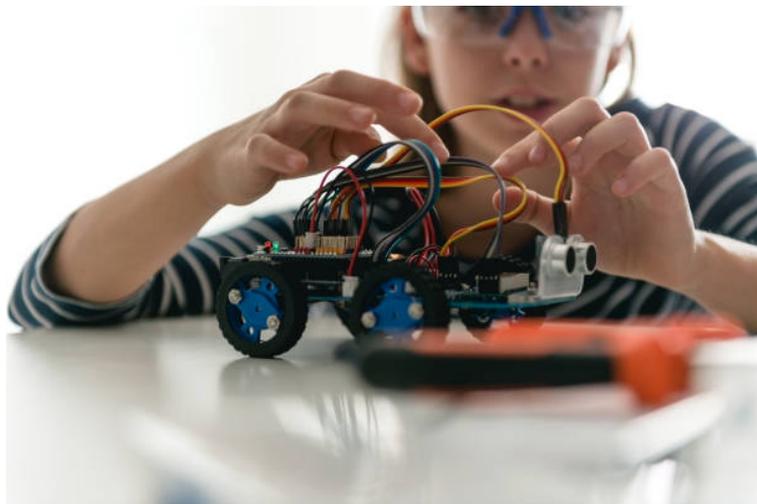


FIG. 1.5 : Robot éducatif

f. Les robots de recherche

Il s'agit de robots conçus à des fins de recherche, par exemple pour explorer de nouvelles technologies robotiques où tester de nouveaux algorithmes. Il peut s'agir de petits robots de laboratoire ou de systèmes à grande échelle destinés à l'exploration en plein air.



FIG. 1.6 : Robot de recherche

Il existe de nombreuses autres façons de classer les robots, et certains robots peuvent entrer dans plusieurs catégories en fonction de leurs capacités et de leurs applications. Dans ce qui suit, nous ne nous intéresserons qu'à un seul type de robot : **les robots industriels**.

1.3 Les manipulateurs industriels

Selon la RIA (Robot Institute of America) c'est un manipulateur qui doit être re-programmable multifonctionnel conçu pour déplacer des matériaux, des pièces, des outils ou tout autre dispositif spécialisé au moyen d'une série de mouvements programmés et d'accomplir une variété d'autres tâches.

L'ISO (International Standard Organization) l'a défini comme étant une machine mue par un mécanisme incluant plusieurs degrés de liberté, ayant souvent l'apparence d'un ou plusieurs bras se terminant par un poignet capable de tenir des outils, des pièces ou un dispositif d'inspection.

Avant de pouvoir s'approfondir plus dans le sujet des robots industriels, on doit définir les notions de base suivantes :

a. Les articulations

Une articulation lie deux corps successifs en limitant le nombre de degrés de liberté (n) de l'un par rapport à l'autre. En robotique, le cas le plus fréquent est $n = 1$ dans ce cas l'articulation est soit rotoïde soit prismatique.

Articulation rotoïde :

Il s'agit d'une articulation de type pivot réduisant le mouvement entre deux corps à une rotation autour d'un axe leur est commun. La situation relative entre deux corps

est donnée par l'angle autour du même axe.

Articulation prismatique :

Il s'agit d'une articulation de type glissière qui réduit le mouvement entre deux corps à une translation le long d'un axe commun. La position relative entre deux corps est mesurée par la distance le long du même axe.

b. L'espace cartésien ou opérationnel

L'espace opérationnel d'un robot est celui dans lequel est représentée la position de l'organe terminal (point P). On représente les coordonnées d'un point dans l'espace cartésien par le vecteur :

$$P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad (1.1)$$

Avec la représentation graphique (Figure 1.7) du point P :

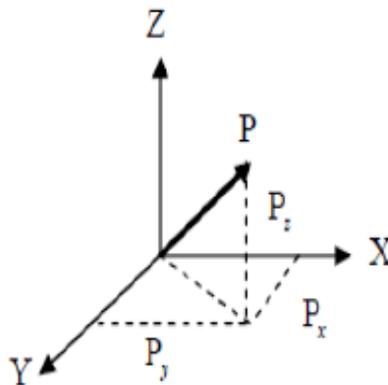


FIG. 1.7 : Représentation d'un point

c. L'espace articulaire

L'espace articulaire est celui dans lequel est représenté la position de tous les corps du robot, dans le cas de l'Orthoglide, c'est la position des 3 actionneurs (trois prismatiques). On représente les coordonnées articulaires par :

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (1.2)$$

Les robots manipulateurs viennent sous différentes formes taille et configuration selon le besoin, mais du point de vue du mécanisme, ils peuvent être divisés en 2 types principaux :

1.3.1 Robot sériel

Les robots sériels, caractérisés par le fait qu'une seule chaîne cinématique relie la base fixe à l'organe terminal mobile, cette chaîne est une succession de corps reliés entre eux par

une articulation d'un seul degré de liberté (rotatoire ou prismatique). Chaque articulation est munie d'un actionneur.



FIG. 1.8 : Robot SCARA Fanuc SR-3iA

1.3.2 Robot parallèle

Un robot parallèle est un mécanisme en chaîne cinématique fermée, constitué d'un organe terminal mobile et d'une base fixe, relié entre eux par des chaînes cinématiques indépendantes. Chacune de ces chaînes compte au plus deux segments articulés afin que la structure reste commandable. La motorisation s'effectue avec des actionneurs simples, un pour chaque chaîne, généralement fixée sur la base.



FIG. 1.9 : Robot parallèle Fanuc

1.3.3 Comparaison entre les robots sériels et parallèles

Une brève comparaison est présentée dans le tableau suivant [3] :

TAB. 1.1 : Tableau comparatif entre robot sériel et robot parallèle

Critères de comparaison	Robots sériel	Robots parallèles
Charge transportable	Faible	Grande
Masse mobile (inertie)	Grande	Faible
Rigidité	Faible	Grande
Précision	Médiocre	Grande
Vitesse et Accélération	Médiocre	Très grandes
Ratio volume de travail/encombrement	Grande	Faible
Volume de travail	Grande	Faible
Singularité dans l'espace de travail	Rare	Présence de singularité
Obtention du modèle géométrique direct	Simple	Difficile
Obtention du modèle géométrique inverse	Difficile	Simple
Synthèse des lois de commande	Difficile	Difficile
Conception	Simple	Difficile

Dans ce projet de fin d'étude, on s'intéresse au robot parallèle, plus précisément le robot Delta.

1.3.4 Historique des robots parallèles

Le premier exemple d'architecture parallèle fut développé par James E. Gwinnett en 1928 [4]. Il s'agit d'une plate-forme mobile destinée à l'industrie du divertissement (Figure 1.10).

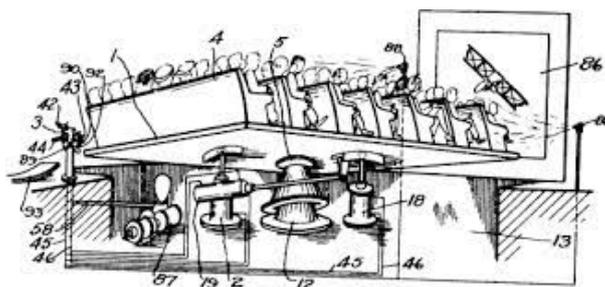


FIG. 1.10 : Un des premiers robots parallèles, inventé en 1928 par Gwinnett

En 1934, Willard L. G. Pollard a déposé un brevet pour une machine de pulvérisation de peinture. L'invention comporte un système de commande et un manipulateur. Le système de commande se compose de films perforés, et le manipulateur est essentiellement un robot à cinq barres.

En 1947 Gough a établi les principes de base d'un mécanisme cinématique avec une structure en boucle fermée (Figure 1.14) qui permet le positionnement et l'orientation d'une plate-forme mobile afin de tester l'usure des pneus[5].

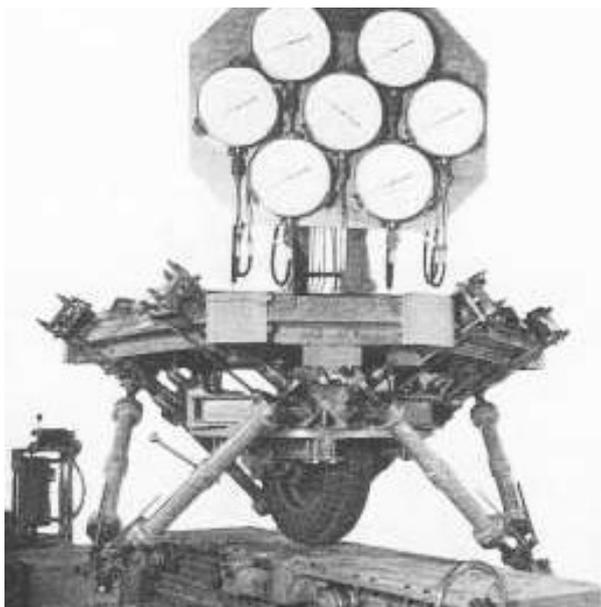


FIG. 1.11 : Plateforme de Gough 1947

Ce n'est que vingt ans plus tard qu'apparaît l'idée d'utiliser des hexapodes dans le domaine de l'aéronautique, lorsque le Dr. Stewart proposa un manipulateur à six degrés de liberté afin de réaliser des simulations de vol. Le développement important de la filière aéronautique dans les années 60 créa le besoin de machines permettant de déplacer de manière multi-directionnelle des masses importantes, tel que des cockpits d'avion. Les hexapodes ayant un rapport charge utile sur poids du robot très important, ils étaient donc particulièrement adaptés à ces tâches.

Le développement des plates-formes de Gough-Stewart a conduit à la création de manipulateurs spécifiques dans de nombreux domaines tels que l'assemblage ou le biomédical. Une autre architecture parallèle très répandue est l'architecture Delta, développée en 1986 par le Prof. Raymond Clavel. Ce robot, qui est à ce jour, le robot parallèle, le plus vendu dans le monde, marque l'arrivée des robots parallèles rapides destinés, entre autres, à des tâches de conditionnement. Le robot Par4, est un parfait exemple de l'engouement de l'industrie pour les robots parallèles rapides. En effet, une version commerciale du robot Par4, le robot Quattro développé par Adept, a ensuite vu le jour et est actuellement le robot le plus rapide du marché (240 cycles par minute). Il existe un grand nombre de cinématiques différentes de robots parallèles et leurs applications possibles sont nombreuses : simulateur de vol, machine-outil, robot de prise et dépose, positionneur de haute précision, robot médical, capteurs d'effort, etc. Aujourd'hui encore, de nouveaux domaines s'ouvrent pour ces robots notamment grâce à la recherche sur les robots redondants, les robots à câbles ou les robots ultra rapides [6].

1.4 Le robot Delta

Le robot Delta est un robot parallèle doté de trois bras reliés à une base fixe par des articulations rotoïde et de l'autre côté au barres parallèle à l'aide de 2 articulations universelles. Il utilise des parallélogrammes dans les bras, afin de préserver l'alignement de l'organe terminal, cette caractéristique est l'élément de conception fondamental du robot

qui lui permet de s'inscrire dans la famille des manipulateur parallèle à 3 degré de liberté (3ddl), car il peut effectuer des translation dans les 3 axes X,Y,Z.

Il a été mis au point en 1985, par le professeur Reymond Clavel, doctorant à l'époque à l'École polytechnique fédérale de Lausanne (EPFL) au Laboratoire de système robotiques (LSRO)[7].

En effet, après une visite avec son équipe du Laboratoire de Systèmes Robotiques à une usine de chocolat, l'idée d'un robot assurant des tâches de prise-dépose à grande vitesse et à faible charge lui est venue à l'esprit. Un travail de recherche fut ensuite couronné par l'invention du robot en question. Une invention qui lui a valu la réception en 1999 du Golden Robot Award, parrainée par ABB Flexible [8].

Le premier brevet fut accordé à R. Clavel en 1985, puis il a été vendu en 1988 à la compagnie swiss Demaurex SA (actuellement Bosch Packaging Technology Unit, Romanel). En 2007, le brevet s'ouvre au domaine public et dès lors plus de 100 entreprises ont adopté cette structure.



FIG. 1.12 : Robot Delta

1.4.1 Caractéristique du robot Delta

a. Point fort

Etant un robot parallèle, le robot Delta a plusieurs caractéristiques intéressantes :

- La nacelle étant reliée à la plaque fixe à l'aide de 3 bras ce qui permet de minimiser l'erreur de position et d'avoir une rigidité et précision phénoménale.
- Les actionneurs (moteur) sont situés dans la partie fixe du robot (la base), contrairement au bras robotique sériel où chaque moteur doit pouvoir mettre en mouvement toute la chaîne en aval, ce qui rend la partie mobile du robot Delta très légère d'où une inertie très faible.
- A cause de la faible inertie du robot, l'effecteur final peut atteindre de grandes vitesses et accélération tout en gardant une grande précision. Ils peuvent atteindre une vitesse de

200 prélèvements par minute et des accélérations de 50 G.

- Bras parallèle très léger et rigide ce qui permet de porté des charges portables importants comparé à la masse et taille du robot.
- Le robot Delta dispose d'un grand espace de travail par rapport à sa taille (pas d'encombrement spatiale), ce qui le rend idéal pour les applications où l'espace est limité.
- La possibilité d'ajouter un 4ème axe situé dans l'effecteur final qui peut ouvrir la porte à une vaste palette d'application industrielle.
- Facilité de production et de maintenance du robot à cause de sa symétrie.

b. Point faible

- Difficultée de modélisation (modèle géométrique direct et indirect).
- Difficultée de commande à cause du fort couplage entre les chaînes cinématiques du robot.
- Existence de zones et point de singularité qui limite le mouvement du robot.

1.4.2 Application et domaine d'utilisation

La 1ere application et la plus répondeue, c'est **prise-dépose**.

1.4.2.1 Prise-dépose

Grâce à l'inertie réduite du robot, sa rigidité, précision et la possibilité d'attacher tout type de ramasseur a la nacelle (pompe a vide, piston, pince), qu'il est utilisé excessive-ment dans le conditionnement de la marchandise (mise en boite) surtout dans le domaine alimentaire. Il est aussi utilisé dans les lignes d'assemblages électroniques ou une grande précision et vitesse sont requises.



FIG. 1.13 : Robot Delta prise-dépose

1.4.2.2 Impression 3d

Les imprimantes cartésiennes sont les plus courantes sur le marché et sont ainsi nom- mées en raison du système de coordonnées dimensionnelles utilisé, les axes X, Y et Z,

c'est-à-dire que la tête d'impression se déplace le long des trois axes cartésiens. Les robots Delta sont moins courants, mais se sont multipliés ces dernières années. Bien qu'ils utilisent également le système de coordonnées X, Y, Z, leur fonctionnement est différent. Les imprimantes Delta ont été créées dans l'idée d'augmenter la vitesse d'impression, pour atteindre 500-600mm/s sachant que les imprimantes cartésiennes sont limitées à 400 mm/s. En gros les par rapport aux imprimantes cartésiennes, les imprimantes Delta sont supérieures en termes de vitesse, de précision et qualité des pièces [9].



FIG. 1.14 : Imprimantes 3D Delta

1.5 Description du robot Delta ISIR88 disponible au laboratoire de commande des processus (LCP)

Ce robot parallèle à 4 degrés de liberté (x, y, z et j), spécialement dédié à la manipulation de pièces légères, peut atteindre une cadence de travail deux à trois fois supérieur à celle des robots les plus rapides actuellement sur le marché à titre d'exemple, il peut assurer le transfert de 3 pièces par seconde par 3 mouvements aller et retours complets.

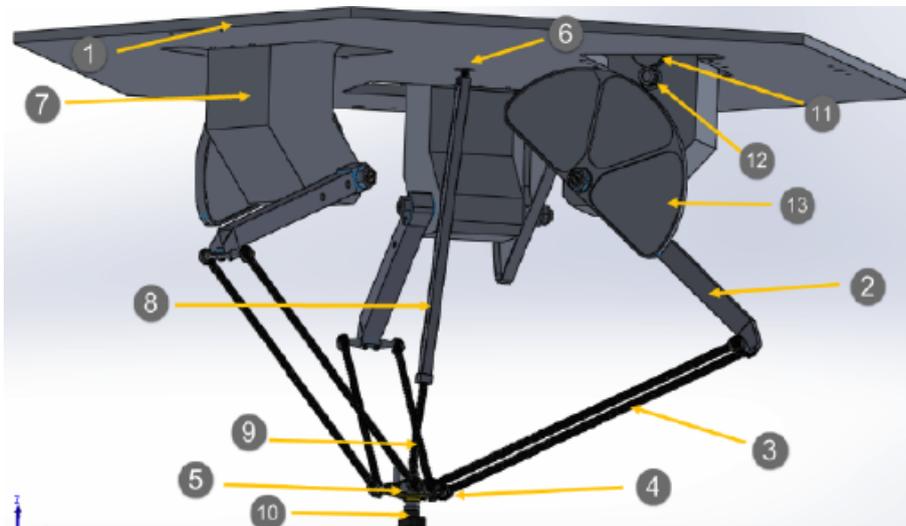


FIG. 1.15 : Robot Delta ISIR88

TAB. 1.2 : Paramètres physiques du robot

Paramètre	Valeur
Longueur du bras	0.205 m
Longueur de l'avant-bras	0.380 m
Masse de la nacelle plate	0.042 kg
Masse de bras	0.098 kg
Masse de barres parallèles	0.028 kg
Masse de coude	0.015 kg

La figure 1.15 nous servira pour la description du robot Delta-4 et de son fonctionnement. Ce robot a 4 degrés de liberté est principalement constitué d'une "base fixe" (1) solidaire d'un bâti support de l'installation (non représenté) et d'une plaque mobile (S), nommé "nacelle". La liaison entre la base fixe (1) et la nacelle (5) est assurée par 3 chaînes cinématiques, chacune d'elles est formée d'un "bras" (2) monté en articulation pivot sur la base fixe et de 2 "barres parallèles (avant-bras)" (3) munies chacune d'une articulation (4) à chaque extrémité l'ensemble formé de 2 barres parallèles et des 2 éléments de liaison au bras et à la nacelle sera nommé "parallélogramme". Chaque bras (2) est entraîné par un "moteur de bras" (7) qui aura le plus souvent la forme d'un ensemble moteur-réducteur-capteur. Le "préhenseur" (10) sera monté en pivotement sur la nacelle et entraîné par le moteur (6) par l'intermédiaire de l'arbre télescopique" (8) muni d'une articulation de type cardan (9) à chacune de ses extrémités. L'orientation de la nacelle est constamment assurée par les 3 parallélogrammes comportant chacun 2 petits tétés et 2 grands côtés formés par les barres parallèles, chaque petit côté solidaire de l'extrémité d'un bras reste constamment parallèle à l'axe de rotation du bras considéré les 3 paires de barres parallèles ne garantissent que les 3 petits côtés solidaires de la nacelle restent parallèles à ceux solidaires de l'extrémité des bras et donc parallèles aux axes de rotation des bras qui, par construction, sont situés dans un même plan. Les articulations aux extrémités des barres parallèles sont de type rotule. Chaque barre peut donc tourner autour de son axe longitudinal. Cette rotation ne perturbe pas le comportement de cette structure articulée formant le parallélogramme de l'espace. Une liaison par des ressorts et des étriers entre les 2 barres parallèles permet de simplifier la construction des rotules et limite les ébats en rotation des barres parallèles. La transmission entre l'un des moteurs et le bras correspondant n'est pas direct, en effet, la transmission est assurée par un système de réduction constitué par une poulie (11), pignon (12) et une roue (13). Afin d'obtenir un rapport de réduction égale 12.

1.6 Conclusion

Au cours de ce premier chapitre, nous avons donné une classification des robots existants selon le travail effectué et la forme générale, ensuite, puis les définitions de base des différents compartiments d'un manipulateur industriel. Ensuite, nous nous sommes focalisés sur les manipulateur parallèle car notre projet concerne un robot spécifique qui est le robot Delta, ou nous avons donné un bref historique sur ce robot, une description, ces points forts et faible, ainsi que ses différents domaines d'utilisation. Nous avons clôturé le chapitre par la description du robot Delta ISIR88 disponible au laboratoire de commande des processus (LCP) de l'ENP.

Chapitre 2

Modélisation du robot Delta

2.1 Introduction

Toute application pratique impliquant l'utilisation du robot Delta nécessite pouvoir le faire bouger d'un point à un autre ou lui faire suivre une trajectoire désirée. Afin d'atteindre ces objectifs, il est indispensable d'élaborer une loi de commande. Cette dernière se base sur les relations entre les différents éléments du robot et leur influence mutuelle. Ses relations sont formalisées mathématiquement sous le nom de modèle géométrique, cinématique et dynamique du robot.

Il est aussi nécessaire de noter que tout robot parallèle possède des points ou même zones de singularité où le robot ne peut pas maintenir ses positions précises ou perd un degré de liberté à cause de sa géométrie parallèle.

2.2 Modélisation géométrique

La modélisation géométrique d'une structure robotisée, consiste simplement en la mise sous forme d'équations de relations liant la position de l'organe terminal (effectuant la tâche) aux différentes coordonnées articulaires du manipulateur.

2.2.1 Modèle géométrique direct

Le modèle géométrique direct (MGD) du robot Delta permet d'exprimer la position translationnelle (x,y,z) de sa nacelle en fonction des trois angles articulaires (q_1, q_2, q_3) .

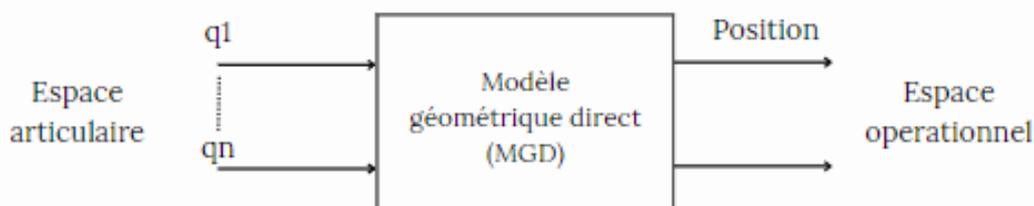


FIG. 2.1 : Modèle géométrique direct du robot Delta

Le calcul du MGD d'une structure parallèle telle que le robot Delta n'est pas systématique comme c'est le cas pour les structures série (transformations homogènes avec formulation de DH ou Dombre [10]).

La symétrie ternaire que présente le Delta nous permet de travailler sur un tiers du robot, c'est à dire un seul bras.

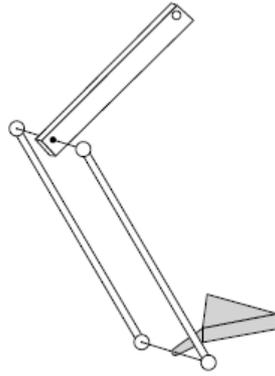


FIG. 2.2 : Bras du Delta

Un repère R_i est lié à chaque bras du Delta comme l'indique la figure 2.3 qui représente la vue de dessus de la base. Le travail se fera ensuite d'une façon indépendante sur chaque repère.

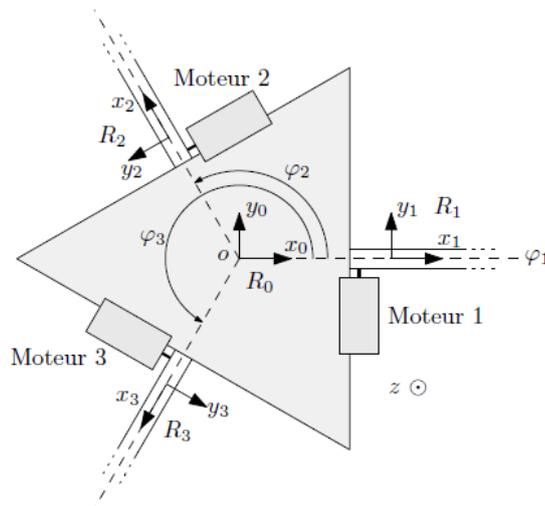


FIG. 2.3 : Vue du dessus de la base du Delta

L'objectif est de trouver une relation entre les angles articulaires α_i (correspondants à R_i) et la position de la nacelle $(x, y, z)_{R_0}$ dans un sens ou dans l'autre (modèle direct ou inverse).

Le passage entre le repère absolu R_0 et les repères intermédiaires se fera par le biais des deux relations matricielles suivantes :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{R_0} = \begin{pmatrix} \cos \varphi_i & -\sin \varphi_i & 0 \\ \sin \varphi_i & \cos \varphi_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{R_i}$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{R_i} = \begin{pmatrix} \cos \varphi_i & \sin \varphi_i & 0 \\ -\sin \varphi_i & \cos \varphi_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{R_0}$$

Les valeurs pris par les ϕ_i sont $\phi_1 = 0$, $\phi_2 = \frac{2\pi}{3}$ et $\phi_3 = \frac{4\pi}{3}$ selon que $i = 1 ; 2$ ou 3

La figure 2.4 donne le schéma annoté d'un bras du Delta en considérant les deux barres parallèles comme une seule (ce qui ne gêne pas pour le développement du modèle géométrique du moment que la nature des articulations est laissé comme telle).

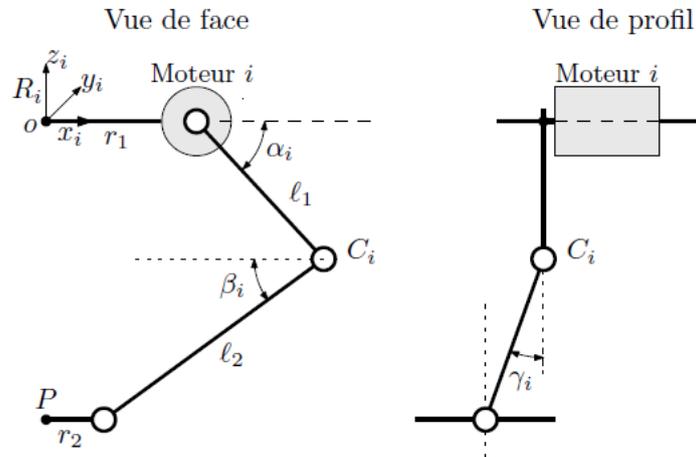


FIG. 2.4 : Schéma d'un bras du Delta

Étant donné que la nacelle reste toujours horizontale et donc parallèle à la base (mouvement translationnel seulement), on peut la considérer comme étant un point matériel (l'orientation n'a pas d'importance puisque de toute façon, elle n'est pas possible), situé au niveau du centre de gravité de la nacelle (c'est l'endroit où est placé l'organe terminal). Le schéma simplifié obtenu est présenté sur la figure 2.5.

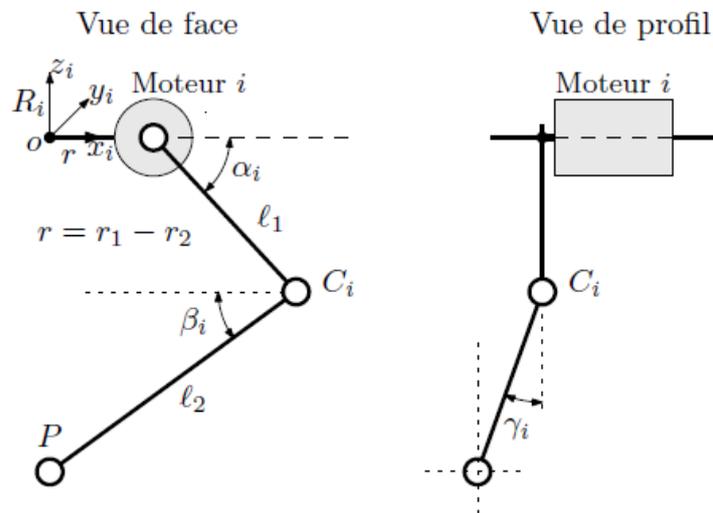


FIG. 2.5 : Schéma simplifié d'un bras du Delta

Les points C_i vont servir de points intermédiaires dans le développement des modèles géométriques direct et inverse, ils ont comme coordonnées les quantités suivantes :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_i} = \begin{pmatrix} r + l_1 \cos \alpha_i \\ 0 \\ -l_1 \sin \alpha_i \end{pmatrix} \quad (2.1)$$

Avec $i = 1, 2, 3$. Les points C_i appartiennent à des cercles de rayon l_1 centrés sur l'axe du moteur. Des équations (4.11) en appliquant la rotation de repère :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_0} = \begin{pmatrix} \cos \varphi_i & -\sin \varphi_i & 0 \\ \sin \varphi_i & \cos \varphi_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r + l_1 \cos \alpha_i \\ 0 \\ -l_1 \sin \alpha_i \end{pmatrix} \quad (2.2)$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_0} = \begin{pmatrix} (r + l_1 \cos \alpha_i) \cos \varphi_i \\ (r + l_1 \cos \alpha_i) \sin \varphi_i \\ -l_1 \sin \alpha_i \end{pmatrix}$$

La jonction au niveau du point P étant une liaison rotule, le point C_i appartient à une sphère de rayon l_2 et de centre P(x, y, z)_{R₀} (voir figure 2.6). Cette sphère est définie dans le repère absolu R₀ par l'équation :

$$(X - x)^2 + (Y - y)^2 + (Z - z)^2 = l_2^2 \quad (2.3)$$

Où X, Y et Z sont les coordonnées d'un point appartenant à la sphère et x, y et z sont les coordonnées du centre de cette sphère (point P).

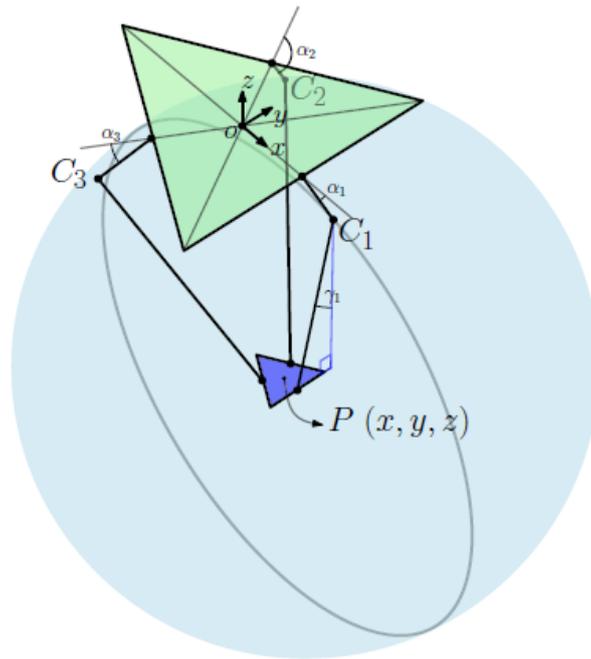


FIG. 2.6 : Sphère de centre P et de rayon l_2 passant par C_1

Les équations des cercles passant par C_i de rayon l_1 centrés sur l'axe des moteurs combinés aux équations de sphère centrée en P de rayon l_2 passant par les C_i permettent de trouver directement le MGD (les points C_i sont à l'intersection de la sphère et des cercle). La méthode présentée est celle développée par Clavel dans [11].

En remplaçant l'équation (2.2) dans l'équation de la sphère (2.3) on trouve :

$$\begin{aligned} & x^2 - 2x(r + l_1 \cos \alpha_i) \cos \varphi_i + (r + l_1 \cos \alpha_i)^2 \cos^2 \varphi_i + \\ & y^2 - 2y(r + l_1 \cos \alpha_i) \sin \varphi_i + (r + l_1 \cos \alpha_i)^2 \sin^2 \varphi_i + \\ & z^2 + 2z l_1 \sin \alpha_i + l_1^2 \sin^2 \alpha_i = l_2^2 \end{aligned} \quad (2.4)$$

Ce qui donne :

$$\begin{aligned} x^2 - 2x(r + \ell_1 \cos \alpha_i) \cos \varphi_i + y^2 - 2y(r + \ell_1 \cos \alpha_i) \sin \varphi_i + \\ z^2 + 2z\ell_1 \sin \alpha_i = \ell_2^2 - \ell_1^2 - r^2 - 2r\ell_1 \cos \alpha_i \end{aligned} \quad (2.5)$$

avec : $i = 1, 2, 3$

En posant :

$$\begin{cases} D_i = -\ell_2^2 + \ell_1^2 + r^2 + 2r\ell_1 \cos \alpha_i \\ E_i = 2(r + \ell_1 \cos \alpha_i) \cos \varphi_i \\ F_i = 2(r + \ell_1 \cos \alpha_i) \sin \varphi_i \\ G_i = -2\ell_1 \sin \alpha_i \end{cases}$$

Donc nous avons 3 équations à 3 inconnues à résoudre :

$$x^2 - E_1x + y^2 - F_1y + z^2 - G_1z = -D_1 \quad (2.6)$$

$$x^2 - E_2x + y^2 - F_2y + z^2 - G_2z = -D_2 \quad (2.7)$$

$$x^2 - E_3x + y^2 - F_3y + z^2 - G_3z = -D_3 \quad (2.8)$$

La résolution de ces équations en (x, y, z) nous donne le modèle géométrique direct.

En effectuant [(2.6)-(2.7)] et [(2.7) - (2.8)] :

$$(E_2 - E_1)x + (F_2 - F_1)y + (G_2 - G_1)z = D_2 - D_1 \quad (2.9)$$

$$(E_3 - E_2)x + (F_3 - F_2)y + (G_3 - G_2)z = D_3 - D_2 \quad (2.10)$$

De l'équation (2.9) on obtient :

$$x = -\frac{F_2 - F_1}{E_2 - E_1}y - \frac{G_2 - G_1}{E_2 - E_1}z + \frac{D_2 - D_1}{E_2 - E_1} \quad (2.11)$$

En remplaçant l'équation (2.11) dans l'équation (2.10) :

$$\begin{aligned} y = \left[\frac{-(E_3 - E_1)(G_2 - G_1) + (G_3 - G_1)(E_2 - E_1)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] z \\ + \left[\frac{-(D_3 - D_1)(E_2 - E_1) + (E_3 - E_1)(D_2 - D_1)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] \end{aligned} \quad (2.12)$$

Donc :

$$y = \frac{H_1}{H_2}z + \frac{H_3}{H_2} \quad \text{Avec : } \begin{cases} H_1 = -(E_3 - E_1)(G_2 - G_1) + (G_3 - G_1)(E_2 - E_1) \\ H_2 = -(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1) \\ H_3 = -(D_3 - D_1)(E_2 - E_1) + (E_3 - E_1)(D_2 - D_1) \end{cases} \quad (2.13)$$

De l'équation (2.9) on obtient aussi :

$$y = \frac{E_1 - E_2}{F_2 - F_1}x + \frac{G_1 - G_2}{F_2 - F_1}z + \frac{D_2 - D_1}{F_2 - F_1} \quad (2.14)$$

En remplaçant toujours dans l'équation (2.10) :

$$\begin{aligned} x = \left[\frac{-(G_3 - G_1)(F_2 - F_1) - (F_3 - F_1)(G_1 - G_2)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] \\ z + \left[\frac{(D_3 - D_1)(F_2 - F_1) - (F_3 - F_1)(D_1 - D_2)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] \end{aligned} \quad (2.15)$$

Donc :

$$x = \frac{H_5}{H_2}z + \frac{H_4}{H_2} \quad \text{Avec : } \begin{cases} H_2 = -(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1) \\ H_4 = (D_3 - D_1)(F_2 - F_1) - (F_3 - F_1)(D_1 - D_2) \\ H_5 = -(G_3 - G_1)(F_2 - F_1) - (F_3 - F_1)(G_1 - G_2) \end{cases} \quad (2.16)$$

Il nous reste à calculer le z; en remplaçant l'expression de x (2.16) et y (2.13) dans (2.6) on aura :

$$\underbrace{z^2 \left(1 + \frac{H_5^2 + H_1^2}{H_2^2}\right)}_L - z \underbrace{\left(-\frac{2(H_5H_4 + H_1H_3)}{H_2^2} + \frac{E_1H_5 + F_1H_1}{H_2} + G_1\right)}_M + \underbrace{\left(D_1 + \frac{H_4^2 + H_3^2}{H_2^2} - \frac{E_1H_4 + F_1H_3}{H_2}\right)}_N = 0 \quad (2.17)$$

Finalement :

$$z = \frac{M - \sqrt{M^2 - 4LN}}{2L} \quad (2.18)$$

Sachant que :

$$\begin{cases} L = 1 + \frac{H_5^2 + H_1^2}{H_2^2} \\ M = -\frac{2(H_5H_4 + H_1H_3)}{H_2^2} + \frac{E_1H_5 + F_1H_1}{H_2} + G_1 \\ N = D_1 + \frac{H_4^2 + H_3^2}{H_2^2} - \frac{E_1H_4 + F_1H_3}{H_2} \end{cases}$$

Remarque 1 : La résolution de l'équation du second degré en z donne en réalité deux solutions, chacune d'entre-elles correspondant au centre d'une sphère passant par les trois points C_i , les centres de ces deux sphères (qui sont donc les solutions de l'équation) sont symétriques par rapport au plan constitué par les trois points C_i , la solution qui correspond à la position réelle de la nacelle sera donc celle qui est située en dessous de ce plan (en concordance avec l'espace de travail) et donc, celle avec un signe « - », l'autre solution est rejetée.

2.2.2 Modèle géométrique inverse

Le modèle géométrique inverse du robot Delta permet de trouver les trois angles articulaires $(\alpha_1, \alpha_2, \alpha_3)$ afin d'avoir une position (x, y, z) souhaitée de sa nacelle.

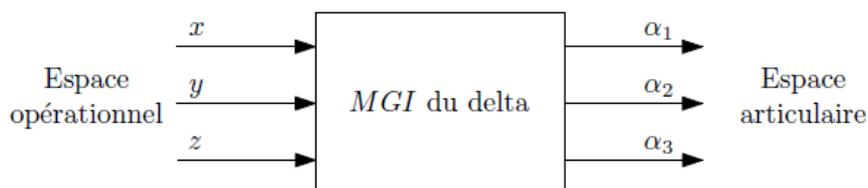


FIG. 2.7 : Modèle géométrique inverse du robot Delta

Plusieurs formulations ont été proposées pour le calcul du modèle géométrique inverse du robot Delta [11], [12].

On peut résoudre les équations (2.6) en $(\alpha_1, \alpha_2, \alpha_3)$ et trouver le MGI. Cette solution a été proposée dans [11].

En reprenant l'équation (2.5), on sépare les termes en $(\cos\alpha_i)$ et en $(\sin\alpha_i)$:

$$(2\ell_1 z) \sin \alpha_i + (2r\ell_1 - 2\ell_1 x \cos \varphi_i - 2\ell_1 y \sin \varphi_i) \cos \alpha_i = (\ell_2^2 - \ell_1^2 - r^2 + 2ry \sin \varphi_i - z^2 - y^2 - x^2 + 2rx \cos \varphi_i) \quad \text{Avec : } i = 1, 2, 3 \quad (2.19)$$

On pose :

$$\begin{cases} A_i = -2\ell_1 z \\ B_i = 2r\ell_1 - 2\ell_1 x \cos \varphi_i - 2\ell_1 y \sin \varphi_i \\ C_i = \ell_2^2 - \ell_1^2 - r^2 + 2ry \sin \varphi_i - z^2 - y^2 - x^2 + 2rx \cos \varphi_i \end{cases}$$

Donc l'équation (2.19) devient :

$$-A_i \sin \alpha_i + B_i \cos \alpha_i = C_i$$

En posant :

$$t = \tan\left(\frac{\alpha_i}{2}\right) \quad \Rightarrow \quad \cos(\alpha_i) = \frac{1-t^2}{1+t^2} \quad \text{et} \quad \sin(\alpha_i) = \frac{2t}{1+t^2}$$

On aura :

$$(B_i + C_i)t^2 + 2At + (C_i - B_i) = 0 \quad \Rightarrow \quad t = \frac{-A_i \pm \sqrt{A_i^2 - (C_i + B_i)(C_i - B_i)}}{B_i + C_i}$$

En revenant au $[\tan(\alpha_i/2)]$:

$$\tan\left(\frac{\alpha_i}{2}\right) = \frac{-A_i \pm \sqrt{A_i^2 + B_i^2 - C_i^2}}{B_i + C_i} \quad (2.20)$$

La solution est donnée par :

$$\alpha_i = 2 \arctan\left(\frac{-A_i + \sqrt{A_i^2 + B_i^2 - C_i^2}}{B_i + C_i}\right) \quad \text{Avec : } i = 1, 2, 3 \quad (2.21)$$

Remarque 2 : La résolution de l'équation en $(\cos\alpha_i)$ et en $(\sin\alpha_i)$ donne deux solutions, chacune d'entre elles correspondent à un point d'intersection du cercle de C_i avec la sphère de centre P(x, y, z). La solution qui correspond à l'angle réel du bras (en concordance avec l'espace de travail) sera celle qui donne l'angle de plus faible valeur, donc celle avec un signe « + » (2.21) l'autre solution est rejetée.

2.3 Modélisation cinématique

La modélisation cinématique permet d'établir la relation entre les vitesses articulaires et les vitesses de l'organe terminal.

2.3.1 Modèle cinématique direct

Le modèle cinématique direct calcule les vitesses des coordonnées de l'outil final en fonction des vitesses articulaires, il est donné par la relation suivante :

$$\dot{P} = J(\alpha)\dot{\alpha} \quad (2.22)$$

Avec $J(\alpha)$, la matrice Jacobienne de dimension (mn) du mécanisme, n étant le nombre d'articulations du robot et m le nombre de ddl. Pour calculer la matrice Jacobienne du robot Delta, on peut utiliser l'ensemble des contraintes reliant l'espace opérationnel avec l'espace articulaire. Les trois équations de contraintes pour le robot Delta peuvent être choisies comme suit :

$$\|C_i P\|_2^2 - l_2^2 = 0 \quad i = 1, 2, 3 \quad (2.23)$$

Où la longueur de l'avant-bras l_2 est supposée constante. Soit s_i dénoté le vecteur $C_i P$. Comme la norme euclidienne peut être écrite comme $s_i^T s_i$, ainsi d'après la figure 2.5 le vecteur s_i peut être écrit comme suit [13] :

$$s_i = \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} - R_{0/i} \left(\begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} l_1 \cos(\alpha_i) \\ 0 \\ l_1 \sin(\alpha_i) \end{bmatrix} \right) \quad (2.24)$$

Où

$$R_{0/i} = \begin{pmatrix} \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 \\ \sin(\phi_i) & \cos(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{pmatrix} \quad (2.25)$$

On dérive l'équation (2.24) par rapport au temps :

La dérivée de s_i est donnée par :

$$\dot{s}_i = \begin{bmatrix} \dot{x}_n \\ \dot{y}_n \\ \dot{z}_n \end{bmatrix} - R_{0/i} \begin{bmatrix} -l_1 \sin(\alpha_i) \\ 0 \\ l_1 \cos(\alpha_i) \end{bmatrix} \dot{\alpha}_i = \dot{P} - b_i \dot{\alpha}_i \quad i = 1, 2, 3 \quad (2.26)$$

Avec :

$$b_i = R_{0/i} \begin{bmatrix} -l_1 \sin(\alpha_i) \\ 0 \\ l_1 \cos(\alpha_i) \end{bmatrix} \quad (2.27)$$

Ainsi :

$$s_i^T \begin{bmatrix} \dot{x}_n \\ \dot{y}_n \\ \dot{z}_n \end{bmatrix} - s_i^T b_i \dot{\alpha}_i = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.28)$$

Qui peut être exprimé sous forme de matrice pour les trois bras de robot comme :

$$\begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix} \dot{P} - \begin{bmatrix} s_1^T b_1 & 0 & 0 \\ 0 & s_2^T b_2 & 0 \\ 0 & 0 & s_3^T b_3 \end{bmatrix} \dot{\alpha} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.29)$$

Maintenant on peut tirer la matrice jacobienne d'un robot Delta à partir de l'équation (2.29) :

$$\dot{P} = J\dot{\alpha} \quad (2.30)$$

Avec :

$$J = \begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix}^{-1} \begin{bmatrix} s_1^T b_1 & 0 & 0 \\ 0 & s_2^T b_2 & 0 \\ 0 & 0 & s_3^T b_3 \end{bmatrix} \quad (2.31)$$

À noter que la matrice jacobienne J ne dépend pas seulement de α , comme c'est généralement le cas pour les robots sériels, mais elle est aussi une fonction des coordonnées opérationnelles P , qui peut être calculée avec le modèle géométrique direct du robot Delta.

2.3.2 Modèle cinématique inverse

Le modèle cinématique inverse permet le passage des vitesses de la nacelle \dot{P} vers les vitesses articulaires $\dot{\alpha}$. On reprend l'équation (2.22) en multipliant ses deux membres par J^{-1} , nous obtenons :

$$\dot{\alpha} = J^{-1}(\alpha)\dot{P} \quad (2.32)$$

Cette relation impose l'inversibilité de la matrice J ($\det(J) \neq 0$). En fait, les points pour lesquels la matrice J n'est pas inversible (i.e ($\det(J) = 0$)) représentent des singularités.

2.4 Modèle des accélérations

Le modèle des accélérations spécifie une cartographie de l'accélération dans l'espace articulaire à l'accélération dans l'espace cartésien [13]. Pour calculer ce dernier, on peut dériver l'équation (2.29) trouvée précédemment par rapport au temps :

$$\begin{aligned} & \begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix} \ddot{P} + \begin{bmatrix} \dot{s}_1^T \\ \dot{s}_2^T \\ \dot{s}_3^T \end{bmatrix} \dot{P} - \left(\begin{bmatrix} s_1^T b_1 & 0 & 0 \\ 0 & s_2^T b_2 & 0 \\ 0 & 0 & s_3^T b_3 \end{bmatrix} \ddot{\alpha} \right. \\ & \left. + \begin{bmatrix} \dot{s}_1^T b_1 + s_1^T \dot{b}_1 & 0 & 0 \\ 0 & \dot{s}_2^T b_2 + s_2^T \dot{b}_2 & 0 \\ 0 & 0 & \dot{s}_3^T b_3 + s_3^T \dot{b}_3 \end{bmatrix} \dot{\alpha} \right) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (2.33)$$

Compte tenu de l'équation (2.31), on peut écrire :

$$\ddot{P} = \begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix}^{-1} \left(- \begin{bmatrix} \dot{s}_1^T \\ \dot{s}_2^T \\ \dot{s}_3^T \end{bmatrix} J + \begin{bmatrix} \dot{s}_1^T b_1 + s_1^T \dot{b}_1 & 0 & 0 \\ 0 & \dot{s}_2^T b_2 + s_2^T \dot{b}_2 & 0 \\ 0 & 0 & \dot{s}_3^T b_3 + s_3^T \dot{b}_3 \end{bmatrix} \right) \dot{\alpha} + J\ddot{\alpha} \quad (2.34)$$

De cette relation on peut constater que l'accélération de la nacelle \ddot{P} est reliée à l'accélération articulaire $\ddot{\alpha}$ comme suit :

$$\ddot{P} = \dot{J}\dot{\alpha} + J\ddot{\alpha} \quad (2.35)$$

2.5 La dynamique de l'actionneur

Dans la plupart des robots industriels, un réducteur est introduit entre chaque actionneur et bras articulaire (voir figure 2.8).

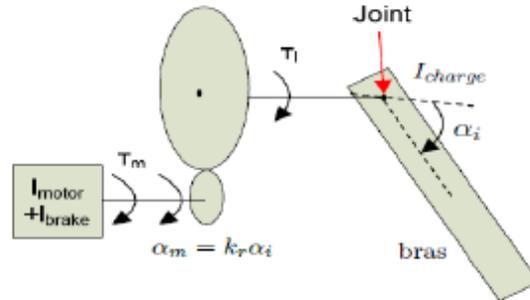


FIG. 2.8 : Actionneur avec réducteur

avec :

$$\alpha_m = k_r \alpha_i \quad (2.36)$$

où α_m est le déplacement angulaire de l'arbre du moteur, α_i est le déplacement angulaire de bras de robot, tandis que la constante k_r est le rapport de réduction.

Si τ_i est le couple de charge à l'axe du robot et τ_m est le couple produit par l'actionneur à l'axe de l'arbre de moteur et compte tenu de l'équation (2.36) on peut écrire :

$$\tau_m = \frac{\tau_i}{k_r} \quad (2.37)$$

Le modèle dynamique inverse d'un robot parallèle peut être calculer en utilisant différentes méthodes (ex : Lagrange ou Newton-Euler).

Pour le robot Delta, la complexité de la dynamique du modèle se pose principalement en raison du mouvement de l'avant-bras. Ce problème peut être simplifié si leurs inerties de rotation sont négligées. Ainsi, la force entre la nacelle et le bras est dans une direction donnée par l'orientation de l'avant-bras. Le modèle pour le robot Delta est calculé avec la méthode Newton-Euler avec les hypothèses simplificatrices suivantes :

- Le moment d'inertie des avant-bras est négligé.
- La masse de l'avant bras m_{ab} est répartie à ses deux extrémités : $r m_{ab}$ sur sa partie supérieure (coude), et $(1 - r) m_{ab}$ sur sa partie inférieure (nacelle) ($0 < r < 1$)[14].
- Les effets de l'élasticité et de frottement sont négligés.

2.6 Paramètres dynamiques

Les paramètres suivants sont utilisés pour calculer le modèle dynamique du robot Delta. La masse totale agissant sur la nacelle est la somme de la masse de la nacelle m_n ,

la masse de la charge m_{charge} et les 3 masses qu'ont contribué chacune des 3 avant-bras $3(1-r)m_{fb}$

$$m_{nt} = m_n + m_{\text{charge}} + 3(1-r)m_{fb} \quad (2.38)$$

r est le rapport de la masse de l'avant-bras. La valeur de r est choisie égale à $2/3$ [14]. La position du centre de masse pour chacune des parties supérieures de bras est calculée à l'aide de l'équation du centre de masse.

$$r_{Gb} = l_1 \frac{\frac{1}{2}m_b + m_c + rm_{fb}}{m_{br}} \quad (2.39)$$

avec

$$m_{br} = m_b + m_c + rm_{fb} \quad (2.40)$$

où m_b est la masse de la partie supérieure du bras, m_c est la masse du coude, m_{fb} est la masse de l'avant-bras.

L'inertie agit sur la partie supérieure du bras est la somme de l'inertie du moteur I_m , l'inertie de frein moteur I_{brake} et l'inertie du bras de I_{bc} . L'inertie du moteur et du frein moteur peut être exprimée comme l'inertie au niveau articulaire multiplié avec la quadratique du rapport de réduction. L' inertie au niveau articulaire est alors donnée par :

$$I_{bi} = I_m k r^2 + I_{brake} k r^2 + I_{bc} \quad (2.41)$$

où I_{bc} est la somme de l'inertie créée à partir de la partie supérieure du bras et de l'inertie de masse créée du point final du bras supérieur, ce qui est donné par :

$$I_{bc} = \frac{m_b}{3} l_1^2 + l_1^2 (m_c + rm_{fb}) = l_1^2 \left(\frac{m_b}{3} + m_c + rm_{fb} \right) \quad (2.42)$$

2.7 Calcul du modèle dynamique en utilisant le principe du travail virtuel

Il existe deux forces agissent sur la nacelle. La force de gravité G_n et la force inertielle (d'Alembert) F_n . Ses expressions sont données par :

$$G_n = m_{nt} \begin{pmatrix} 0 & 0 & -g \end{pmatrix}^T \quad F_n = m_{nt} \ddot{P}_n \quad (2.43)$$

La contribution de ses deux forces dans l'espace articulaire peut être calculer avec la transposée de la matrice Jacobienne :

$$\tau_n = J^T F_n = J^T m_{nt} \ddot{P}_n \quad (2.44)$$

$$\tau_{Gn} = J^T G_n = J^T m_{nt} \begin{pmatrix} 0 & 0 & -g \end{pmatrix}^T \quad (2.45)$$

Deux types de couples agissants sur les bras actionnés. Le couple τ_{Gb} produit par la force gravitationnelle de chaque bras et le couple τ_b produite à partir de la force inertielle

(d'Alembert) agissant sur chaque bras. La contribution gravitationnelle peut être calculée comme la force qui agit perpendiculairement à la partie supérieure du bras à travers le centre de masse comme illustré à la figure 2.9.

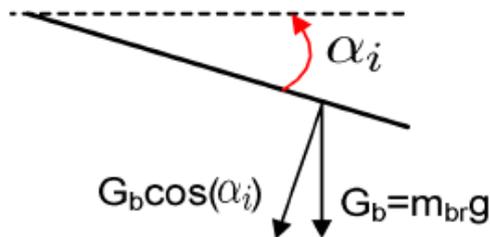


FIG. 2.9 : Force gravitationnelle agissant sur la partie supérieure du bras d'un robot Delta

Nous donnons :

$$\tau_{Gb} = r_{Gb} G_b (\cos \alpha_1 \quad \cos \alpha_2 \quad \cos \alpha_3)^T \quad (2.46)$$

où G_b est la force gravitationnelle agissant sur le centre de masse de chaque bras. La contribution de chaque couple de la partie supérieure du bras sur les articulations peuvent être exprimée par :

$$\tau_b = I_b \alpha \quad (2.47)$$

Où I_b est la matrice d'inertie du bras dans l'espace articulaire et elle est donnée par :

$$I_b = \begin{bmatrix} I_{b1} & 0 & 0 \\ 0 & I_{b2} & 0 \\ 0 & 0 & I_{b3} \end{bmatrix} \quad (2.48)$$

Selon le principe d'Alembert la contribution de toutes les forces d'inertie doit être égale à la contribution de toutes les forces non inertielle. L'application de ce principe au niveau de l'articulation nous permettons de trouver :

$$\tau + \tau_{Gn} + \tau_{Gb} = \tau_b + \tau_n \quad (2.49)$$

où τ est le vecteur des couples appliqués aux articulations. τ_n contenir le terme \ddot{P} ce qui peut être exprimé dans l'espace articulaire par l'équation (2.35), cela donne :

$$\tau = (I_b + m_{nt} J^T J) \ddot{\alpha} + J^T m_{nt} J \dot{\alpha} - \tau_{Gn} - \tau_{Gb} \quad (2.50)$$

Ce qui peut être écrit comme suit :

$$\tau = M(\alpha) \ddot{\alpha} + C(\alpha, \dot{\alpha}) \dot{\alpha} + G(\alpha) \quad (2.51)$$

où $M(\alpha) \in \mathbb{R}^{3 \times 3}$ est la matrice d'inertie, $C(\alpha, \dot{\alpha}) \dot{\alpha} \in \mathbb{R}^3$ représente les forces de Coriolis et centrifuges, et $G(\alpha) \in \mathbb{R}^3$ est le vecteur des forces de gravitation agissant sur le manipulateur.

2.8 Singularités du robot Delta

Dans un manipulateur sériel, les positions singulières découlent de la perte de degrés de liberté alors que dans un manipulateur parallèle, elles résultent du gain d'un ou plusieurs degrés de liberté. La plupart du temps, le passage par une singularité provoque un hyperguidage pour ensuite passer par la perte de contrôle d'un ou plusieurs degrés de liberté.

En général, la résolution du déterminant de la matrice Jacobienne permet de déterminer facilement les points singuliers. En outre, la complexité de la matrice jacobienne de robot Delta a poussé le constructeur [11], [12] à faire une analyse des positions de l'avant-bras (barres parallèles) les unes par rapport aux autres. Ceci a permis de définir les configurations singulières représentées dans la figure 2.10 [15] :

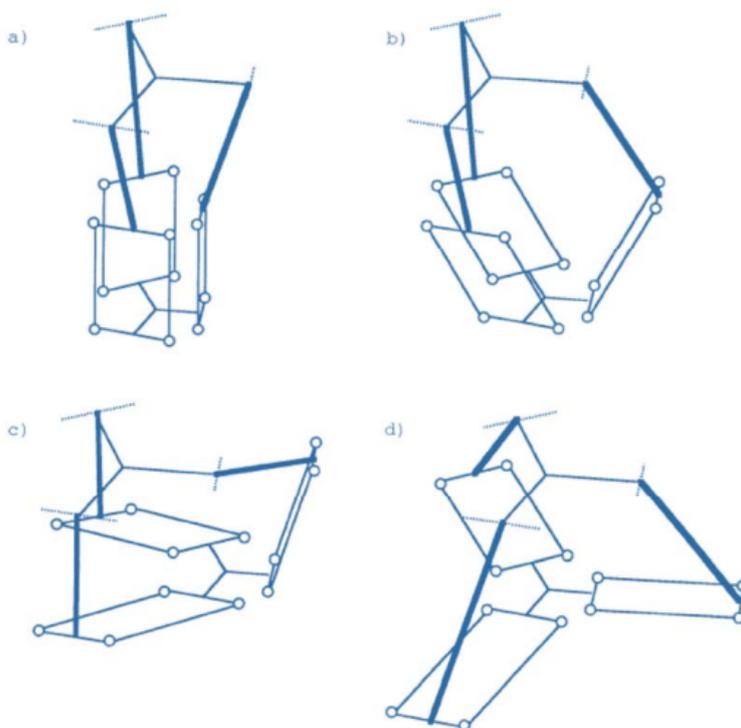


FIG. 2.10 : Représentation des 4 types de singularités du robot Delta

Les singularités de la structure considérée ont été déterminées à partir de l'analyse de la position relative des 6 barres solidaires de la nacelle ; nous considérons que chaque barre transmet uniquement une force dont la ligne d'action est confondue avec la droite reliant les 2 articulations à ses extrémités.

La position de la nacelle est totalement définie et sa stabilité assurée si les conditions suivantes sont respectées :

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

Ces conditions mettent en évidence les 4 types de singularités schématisées à la figure 2.10.

1. Lorsque les 6 barres (3 paires) sont toutes parallèles à une même direction 2.10.a, la nacelle conserve 3 degrés de liberté, soit : déplacement sur une portion de sphère et rotation autour d'un axe perpendiculaire à la nacelle ;
2. Si 4 barres (2 paires) sont parallèles 2.10.b, la nacelle conserve un degré de liberté, soit : déplacement sur un arc de cercle ;
3. Si 4 barres (2 paires) sont dans un même plan ou dans 2 plans parallèles 2.10.c, a nacelle conserve un degré de liberté, soit : rotation définie par l'axe reliant les deux articulations situées entre la nacelle et les deux barres non considérées ;
4. Lorsque les 3 parallélogrammes sont situés dans 3 plans parallèles ou confondus 2.10.d, la nacelle conserve 3 degrés de liberté, soit : 2 rotations d'axes appartenant au plan de la nacelle (axes horizontaux) et translation de faible amplitude perpendiculairement au même plan.

2.9 Volume de travail

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

En prenant compte des contraintes de construction et des singularités, le volume de travail du Delta qui sera retenu et qui sera utilisé dans notre travail sera un cylindre ayant les caractéristiques suivantes :

- Diamètre : D
- Hauteur : H
- Position sur l'axe z de la surface supérieure : Z_h

Les valeurs de ces dimensions, qui dépendent des constantes géométrique du robot : ℓ_1, ℓ_2 et $r = r_1 - r_2$, sont les suivantes [7], [11] :

$$D = 2 \cdot \ell_2 \cdot \sin \gamma_{\max}$$

$$H = \ell_1 \sqrt{\left(1 + \frac{\ell_2}{\ell_1}\right)^2 - \left(\frac{r}{\ell_1} + 2 \frac{\ell_2 \sin \gamma_{\max}}{\ell_1 \sqrt{3}}\right)^2} - \ell_1 \sqrt{1 + \left(\frac{\ell_2}{\ell_1}\right)^2 - \frac{\ell_2}{\ell_1} \sqrt{3}}$$

$$Z_h = -\ell_1 \sqrt{1 + \left(\frac{\ell_2}{\ell_1}\right)^2 - 2 \frac{\ell_2}{\ell_1} \cos(\alpha_i + \beta_i)_{\min}}$$

La figure 2.11 montre le volume de travail du Delta considéré.

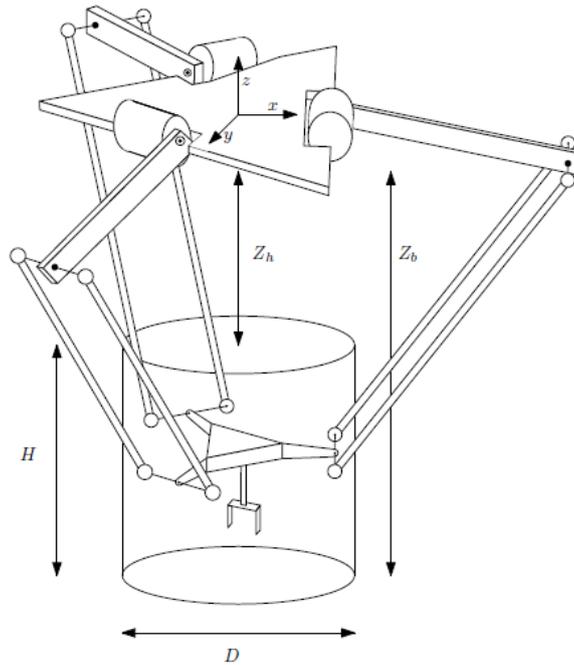


FIG. 2.11 : Volume de travail considéré du Delta

2.10 Génération de mouvement et de trajectoire

Dans le contexte de la robotique, la partie chargée de contrôler le robot a pour but de générer les instructions nécessaires en fonction des positions, vitesses et accélérations que l'on souhaite donner à l'organe terminal pour que ce dernier suive une trajectoire souhaitée, constituée d'une séquence ou d'un certain nombre de points de l'espace. Plusieurs types de trajectoires peuvent être envisagés en fonction des tâches à accomplir par le robot, des contraintes environnementales et du type de robot :

- Mouvement entre deux points sans suivi de trajectoire ;
- Mouvement entre deux points avec suivi de trajectoire ;
- Mouvement avec suivi de trajectoire à travers de nombreux points intermédiaires ;

On aura alors deux types de commandes :

a. Commande de positionnement

Dans ce cas-là, on travaille en commande point à point. La génération de trajectoire se fait dans l'espace articulaire. On peut alors avoir une référence en échelon de position ou bien utiliser une loi de mouvement afin de contrôler le temps de parcours et synchroniser les mouvements articulaires entre eux fig 2.12.

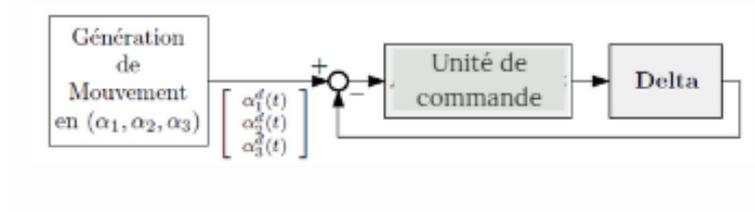


FIG. 2.12 : Génération de mouvement dans l'espace articulaire

b. Commande de suivi de trajectoire

Pour ce deuxième type de commande, la génération de trajectoires se fait dans l'espace opérationnel. Une loi de mouvement doit cependant être greffée à la trajectoire de référence fig 2.13.

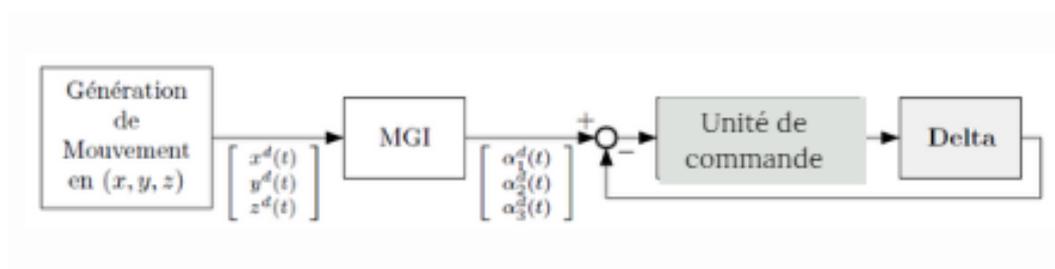


FIG. 2.13 : Génération de mouvement dans l'espace opérationnel

2.10.1 Lois de mouvement

Pour rendre compte de la composante temporelle du mouvement, une loi de mouvement doit être greffée à la trajectoire, qu'il s'agisse de la génération de trajectoires dans l'espace articulaire ou dans l'espace opérationnel. Cette loi de mouvement peut être appliquée aux variables articulaires directement ou à la trajectoire opérationnelle (par le biais de la variable curviligne s).

Contrairement à la commande en échelon, nous pouvons contrôler les vitesses et les accélérations du robot en manipulant le mouvement du robot dans le domaine temporel.

2.10.1.1 Loi Bang-Bang

Cette loi de mouvement nous permet de contrôler l'évolution de la position, vitesse et accélération. Cependant, cette loi de mouvement n'est pas optimale car si on impose le temps et la distance à parcourir, on ne peut plus imposer la vitesse et l'accélération maximales. La figure 2.14 représente la position, la vitesse et l'accélération correspondant à une telle loi de mouvement.

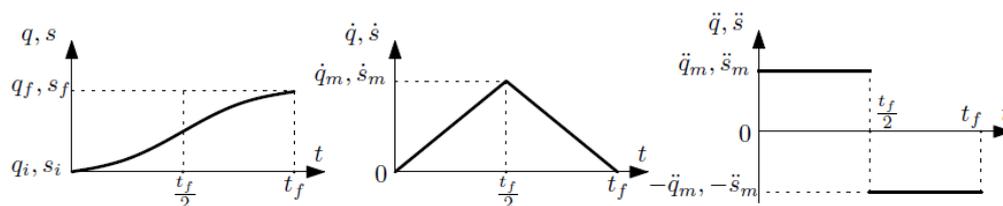


FIG. 2.14 : Loi de mouvement Bang-bang

2.10.1.2 Loi Trapèze

La loi Trapézoïdale est une loi Bang-Bang modifiée, un palier à vitesse fixe est ajouté entre les deux étapes d'accélération et de décélération, ceci nous permet de contrôler la vitesse maximale même si le temps de parcours et la distance sont fixés (en modulant la durée du palier à vitesse constante). La loi en Trapèze s'impose alors comme un choix judicieux.

Aussi, le mouvement est plus uniforme (changement de la consigne d'accélération moins brusque que la loi Bang-bang) et le temps de parcours est optimisé, car on peut saturer la vitesse et l'accélération à leurs maximums possibles en utilisant donc au mieux les possibilités du moteur.

La figure 2.15 représente la position, vitesse et accélération correspondant à une loi de mouvement en trapèze.

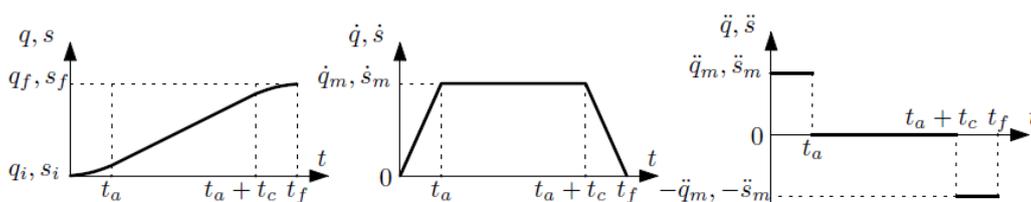


FIG. 2.15 : Loi de mouvement en Trapèze

Où :

- + t_a : temps de la phase d'accélération ;
 - + t_c : temps de la phase à vitesse constante ;
 - + t_f : temps total du parcours ;
 - + q_i, s_i : position initiale (articulaire ou curviligne pour la trajectoire opérationnelle) ;
 - + q_f, s_f : position finale (articulaire ou curviligne pour la trajectoire opérationnelle) nous noterons D la distance parcourue ;
 - + \dot{q}_m, \dot{s}_m : vitesse maximale (palier de vitesse) pour la loi articulaire et opérationnelle respectivement ;
 - + \ddot{q}_m, \ddot{s}_m : accélération maximale de la phase d'accélération et de décélération ($-\ddot{q}_m, -\ddot{s}_m$)
- Connaissant la distance à parcourir D , la loi de mouvement selon s est donnée par les

équations suivantes :

$$\begin{cases} t_f = 2t_a + t_c \\ \dot{s}_m = \ddot{s}_m t_a \\ s_f = s_i + D \\ s(t) = s_i + \ddot{s}_m \cdot \left(\frac{t^2}{2}\right) & \text{pour } : t \leq t_a \\ s(t) = s_i + \ddot{s}_m \frac{t_a^2}{2} + \dot{s}_m \cdot (t - t_a) & \text{pour } : t_a < t \leq t_a + t_c \\ s(t) = s_i + \ddot{s}_m \frac{t_a^2}{2} + \dot{s}_m (t - t_a) - \ddot{s}_m \cdot \frac{(t - t_a - t_c)^2}{2} & \text{pour } : t_a + t_c < t \leq t_f \end{cases} \quad (2.52)$$

Ces équations sont aussi valables pour la variable articulaire q .

Ces équations nous permettent de contrôler plusieurs aspects (la distance du parcours, le temps de parcours de la distance, la durée de la phase d'accélération et de vitesse constantes et l'accélération et la vitesse maximale) en fixant les paramètres de la loi de mouvement. Cette loi sera donc utilisée dans la génération des différentes trajectoires.

2.10.2 Trajectoire

La génération de trajectoires dans l'espace articulaire n'est pas envisageable pour le robot Delta, on ne peut pas se permettre de laisser la trajectoire de l'organe terminal sans contrôle entre les points de départ et d'arrivée. La génération de trajectoire dans l'espace opérationnel s'impose donc comme une nécessité.

La génération de trajectoire dans l'espace opérationnel passe par la connaissance de l'évolution de la coordonnée curviligne $s(t)$ à partir de laquelle l'évolution des coordonnées articulaires $x(t)$, $y(t)$ et $z(t)$ est obtenu par projection (figure 2.16)[7]. Le modèle géométrique inverse est ensuite appliqué pour avoir les références articulaires à injecter aux actionneurs.

Nous avons implémenté plusieurs trajectoires allant de la plus simple à la plus complexe avec comme but final la construction d'une trajectoire de prise-dépose adaptée au Delta.

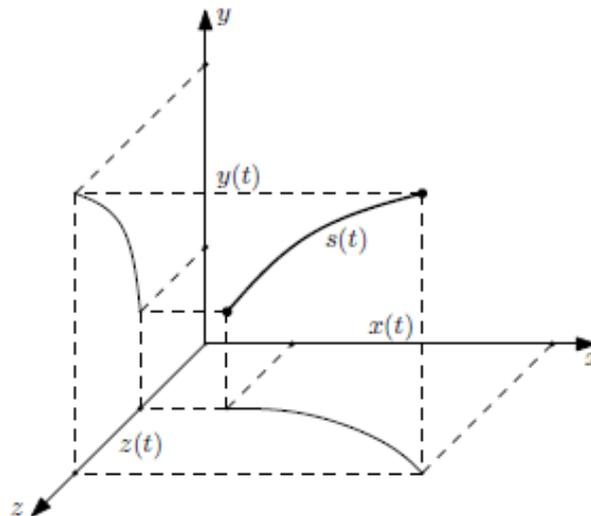


FIG. 2.16 : Projection de la coordonnée curviligne $s(t)$ donnant $x(t)$; $y(t)$; $z(t)$

a. Trajectoire rectiligne

La génération de trajectoires rectilignes prend comme argument le point de départ et le point d'arrivée.

Dans notre cas, nous avons pris une trajectoire rectiligne selon l'axe z et selon l'axe x . Ces deux trajectoire nous permettrons par la suite de composer une séquence de prise-dépose par une combinaison adéquate.

La projection se fait selon les équations suivantes :

Trajectoire selon l'axe x et z respectivement :

$$\begin{cases} x(t) = x_i \pm s(t) \\ y(t) = y_i \\ z(t) = z_i \end{cases} \quad \begin{cases} x(t) = x_i \\ y(t) = y_i \\ z(t) = z_i \pm s(t) \end{cases}$$

x_i, y_i , et z_i étant les coordonnées du point de départ (point initial).

b. Trajectoire ellipsoïde

Une trajectoire en demi-ellipse peut-être utilisée pour une tâche de prise-dépose (figure 3.3).

En prenant le cas d'une ellipsoïde selon l'axe x nous avons les équations suivantes :

$$\begin{cases} x(t) = x_i \pm A \left[1 - \cos \left(\frac{\pi S(t)}{D} \right) \right] \\ y(t) = y_i \\ z(t) = z_i + B \sin \left(\frac{\pi S(t)}{D} \right) \end{cases}$$

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

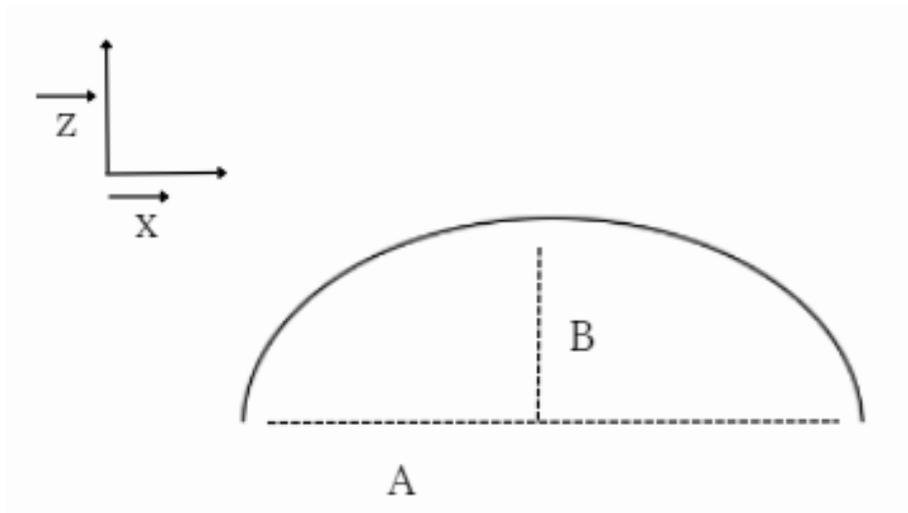


FIG. 2.17 : Trajectoire ellipsoïde

2.11 Conclusion

Dans ce chapitre, nous avons représenté le robot Delta *ISIR*₈₈ avec différents modèles :

Le modèle géométrique direct et inverse (MGD/MGI), le modèle cinématique de vitesse direct et inverse et le modèle cinématique d'accélération (MDD). Tous ses modèles nous ont permis d'élaborer une loi de commande qui prend en compte non seulement l'aspect géométrique du robot (la relation entre les angles des barres parallèle et la position cartésienne de l'effecteur final) mais aussi l'aspect dynamique (la masse et l'inertie du robot). Nous avons aussi mis en évidence les singularités du robot, ainsi que la loi de mouvement et la trajectoire que le robot doit suivre.

Chapitre 3

Commande PD d'ordre fractionnaire

3.1 Introduction

Le régulateur PID est la technique de régulation la plus utilisée dans les applications industrielles. Elle a été présentée pour la première fois par Elmer Sperry en 1911 [16].

Un des développements, intéressants des contrôleurs PID ces dernières années consiste en l'introduction du calcul fractionnaire dans des actions intégrales et dérivées.

Dans ce chapitre, nous commençons par une introduction mathématique aux opérateurs fractionnaires puis nous mènerons une étude comparative entre les régulateurs PD d'ordre entier et d'ordre fractionnaire appliqués au robot Delta.

3.2 Opérateurs d'ordre fractionnaire

Le calcul d'ordre fractionnaire est une généralisation des opérateurs de l'intégration et de la différentiation classiques aux ordres non entiers. La notation utilisée pour les opérateurs d'ordre fractionnaire est ${}_aD_t^\alpha$ qui est défini comme suit [17] :

$${}_aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \Re(\alpha) > 0, \\ 1 & \Re(\alpha) = 0, \\ \int_a^t (d\tau)^{-\alpha} & \Re(\alpha) < 0, \end{cases} \quad (3.1)$$

où α est l'ordre de l'opération, généralement $\alpha \in \mathbb{C}$ et les nombres réels a et t sont des limites de l'opération intégral-différentielle.

3.2.1 Définitions

Il existe plusieurs définitions mathématiques pour l'intégration et la dérivation d'ordre fractionnaire. Les définitions les plus utilisées sont celles de Grünwald Leitnikov, de Riemann-Liouville et de Caputo.

3.2.1.1 Définition de Grünwald-Leitnikov

Une des définitions les plus rencontrées de la dérivée d'ordre fractionnaire dans la littérature est celle de Grünwald-Leitnikov qui est donnée par [17] :

$${}_aD_t^\alpha f(t) = \frac{{}_a d_t^\alpha f(t)}{dt^\alpha} = \lim_{h \rightarrow 0} \left(\frac{1}{h^\alpha} \right) \sum_{k=0}^N (-1)^k \binom{\alpha}{k} f(t - kh) \quad (3.2)$$

$$\text{où } \alpha \in \mathbb{C}, \binom{\alpha}{0} = 1, \binom{\alpha}{k} = \frac{\alpha(\alpha-1)\dots(\alpha-k+1)}{k!}, \quad a < t, N = \left(\frac{t-a}{h} \right).$$

Sous la condition que les dérivées $f^{(k)}(t)$ ($k = 1, 2, \dots, (m+1)$) sont continues dans l'intervalle $[a, t]$, la définition de Grünwald-Letnikov peut être aussi donnée par :

$${}_aD_t^\alpha f(t) = \sum_{k=0}^m \frac{f^{(k)}(a)(t-a)^{-\alpha+k}}{\Gamma(-\alpha+k+1)} + \frac{1}{\Gamma(-\alpha+m+1)} \int_a^t (t-\tau)^{m-\alpha} f^{(m+1)}(\tau) d\tau \quad (3.3)$$

Où m est un nombre entier tel que $m < \Re(\alpha) < (m + 1)$ et $\Gamma(\alpha)$ est la fonction Gamma définie par :

$$\Gamma(\alpha) = \int_0^{\infty} e^{-x} x^{\alpha-1} dx \quad (3.4)$$

3.2.1.2 Définition de Riemann-Liouville

La définition de Riemann-Liouville de la dérivée ou de l'intégrale d'ordre fractionnaire α d'une $f(t)$ est donnée par [17] :

$${}_a D_t^{-\alpha} f(t) = \begin{cases} D^m [{}_a D_t^{\alpha-m} f(x)], & \text{si } \Re(\alpha) > 0 \\ \frac{1}{\Gamma(-\alpha)} \int_a^t (t-x)^{-\alpha-1} f(x) dx, & \text{si } \Re(\alpha) < 0 \\ f(t), & \text{si } \Re(\alpha) = 0 \end{cases} \quad (3.5)$$

où $\alpha \in \mathbb{C}$, $m < \Re(\alpha) < (m + 1)$ et $f(\cdot)$ une fonction localement intégrable définie sur $[a, \infty[$.

3.2.1.3 Définition de Caputo

Caputo a proposé une nouvelle définition de la dérivée d'ordre fractionnaire qui porte d'ailleurs son nom et qui incorpore les conditions initiales de la fonction à traiter en termes de ses dérivées d'ordre entier. La dérivée d'ordre fractionnaire $\alpha > 0$ d'une fonction $f(t)$ définie sur $[a, +\infty[$ est donnée comme suit [17] :

$${}_a D_t^{\alpha} f(t) = \frac{1}{\Gamma(m-\alpha)} \int_a^t (t-\tau)^{m-\alpha-1} f^{(m)}(\tau) d\tau \quad (3.6)$$

Où m est un entier tel que $m < \alpha < (m + 1)$ et $f^{(m)}(t)$ est la $m^{\text{ième}}$ dérivée de la fonction $f(t)$.

3.2.2 Propriétés des opérateurs d'ordre fractionnaire

Les principales propriétés des dérivées et intégrales d'ordre fractionnaire sont les suivantes [5] :

-Deux paramètres, " α " et " a " sont nécessaires pour définir un opérateur d'ordre fractionnaire.

-La définition de la dérivation d'ordre fractionnaire étant basée sur celle d'une intégration d'ordre fractionnaire, une dérivation d'ordre fractionnaire revêt un caractère global contrairement à une dérivation entière. Il s'avère en effet que la dérivée d'ordre fractionnaire d'une fonction f nécessite la connaissance de $f(t)$ sur l'intervalle $[a, t]$, alors que dans le cas entier, seule la connaissance locale de f autour de t est nécessaire. Cette propriété permet d'interpréter les systèmes d'ordre fractionnaire comme des systèmes à mémoire longue, les systèmes entiers étant alors interprétables comme des systèmes à mémoire courte.

- La définition de Grünwald Leitnikov décrit l'unification des deux notions, l'intégral et la dérivée d'ordre fractionnaire.

- La dérivée d'ordre fractionnaire de l'intégrale du même ordre d'une fonction temporelle $f(t)$ donne : ${}_a D_t^\alpha \circ {}_a I_t^\alpha f(t) = f(t)$ avec $\Re(\alpha) > 0$. Cette relation n'étant pas toujours vraie pour $\Re(\alpha) < 0$.

- Les opérateurs d'intégration d'ordre fractionnaire vérifient la propriété de semi-groupe, soit :

$${}_a I_t^{n_1} \circ {}_a I_t^{n_2} f(t) = {}_a I_t^{n_1+n_2} f(t) \quad \text{avec} : \begin{cases} \Re(n_1) > 0 \\ \Re(n_2) > 0 \end{cases} \quad (3.7)$$

- Contrairement aux opérateurs d'intégration d'ordre fractionnaire, les opérateurs de dérivation d'ordre fractionnaire (réel ou complexe), ne vérifient la propriété de semi-groupe que sous certaines conditions, soit :

$${}_a D_t^r ({}_a D_t^n f(t)) = {}_a D_t^{r+n} f(t) \quad \text{avec} : \begin{cases} r & \text{entier positif} \\ n & \text{arbitraire} \end{cases} \quad (3.8)$$

- L'opérateur de dérivation d'ordre fractionnaire (réel ou complexe) est linéaire. Ainsi, si y et g sont deux fonctions continues et (λ, α) réelles, on aura :

$$D^{(\alpha)}(\lambda.y + \alpha.g) = \lambda.D^{(\alpha)}(y) + \alpha.D^{(\alpha)}(g). \quad (3.9)$$

- Pour $\alpha = 0$, $D^{(\alpha)}y(t)$ est l'opérateur identité $D^{(0)}y(t) = y(t)$.

3.2.3 Transformée de Laplace des opérateurs d'ordre fractionnaire

Les définitions temporelles des opérateurs différentiels non entiers souffrent d'un formalisme mathématique quelque peu compliqué, mais leurs expressions dans le domaine de Laplace relève d'une simplicité remarquable, en particulier dans le cas de l'étude de systèmes relaxés à $t = 0$. Dans ce qui suit, on s'intéresse qu'au cas où l'ordre m est réel.

3.2.3.1 Transformée de Laplace de la dérivée d'ordre fractionnaire

La transformée de Laplace de la dérivée d'ordre fractionnaire α , $(n-1) \leq \alpha < n$, selon la définition de Grünwald-Leitnikov est donnée par [5], [17] :

$$L \{D^\alpha f(t)\} = s^\alpha F(s) \quad (3.10)$$

La transformée de Laplace de la dérivée d'ordre fractionnaire α , $(n-1) \leq \alpha < n$, selon la définition de Riemann-Liouville est donnée par :

$$L \{D^\alpha f(t)\} = s^\alpha F(s) - \sum_{k=0}^{n-1} s^k [D^{\alpha-k-1} f(t)]_{t=0} \quad (3.11)$$

L'applicabilité de cette transformée en pratique est limitée à cause de l'absence d'interprétation physique des valeurs initiales $[D^{\alpha-k-1} f(t)]_{t=0}$, pour $k = 0, 1, \dots, (n-1)$.

Selon la définition de Caputo, la transformée de Laplace de la dérivée d'ordre fractionnaire α , $(n - 1) \leq \alpha < n$, est donnée par :

$$L \{D^\alpha f(t)\} = s^\alpha F(s) - \sum_{k=0}^{n-1} s^{\alpha-k-1} f^k(0) \quad (3.12)$$

Cette transformée est très utilisée en pratique à cause des valeurs initiales classique $f^k(0)$, pour $k = 0, 1, \dots, (n - 1)$.

3.2.3.2 Transformée de Laplace de l'intégrale d'ordre fractionnaire

La transformé de Laplace de l'intégration d'ordre fractionnaire α , $(n - 1) \leq \alpha < n$, pour toutes les définitions est donnée par :

$$L \{I^\alpha f(t)\} = s^{-\alpha} F(s) \quad (3.13)$$

3.2.4 Approximation analogique des opérateurs d'ordre fractionnaire

D'habitude les simulations sont effectuées avec un logiciel préparé pour traiter seulement les puissances d'ordre entier de s . Alors il est très important de trouver des approximations d'ordre entier pour des fonctions de transfert d'ordre fractionnaire. Autrement dit, lorsque les simulations doivent être exécutées ou les correcteurs doivent être implémentés, les fonctions de transfert d'ordre fractionnaire sont remplacées par des fonctions de transfert d'ordre entier, avec un comportement assez identique à celles désirées, mais beaucoup plus facile à manipuler. Dans la littérature il existe plusieurs méthodes d'approximation analogique des systèmes d'ordre fractionnaire, et les plus utilisées sont :

- Méthode EFC (Expansion Fractionnaire Continue)
- Méthode de Charef
- Méthode d'Oustaloup
- Méthode de Carlson

La méthode d'Oustaloup est plus élaborée et plus pratique pour les approximations des fonctions de transferts d'ordre fractionnaire, c'est pourquoi on a choisi de la présenter en détail dans la suite.

Approximation d'Oustaloup

Approximation du filtre d'Oustaloup à un différenciateur d'ordre fractionnaire $G(s) = s^\alpha$ est largement utilisée dans les applications. Le filtre Oustaloup peut être conçu :

$$G_l(s) = K \prod_{i=1}^N \left\{ \frac{s + w'_i}{s + w_i} \right\} \quad (3.14)$$

Où :

$$w'_i = w_b w_u^{(2i-1-\alpha)/N} K = w_h^\alpha w_u = \sqrt{\frac{w_h}{w_b}} \quad (3.15)$$

L'approximation est conçue pour les fréquences $w \in [w_b, w_h]$ et N est l'ordre de l'approximation. Comme on peut le voir son représentation prend la forme d'un produit d'une série des systèmes stables linéaires de premier ordre. Comme on peut l'observer, le choix d'une large bande des résultats d'approximation et de grands w_u et un ordre N élevé donne un espacement des pôles de près de w_h à ceux très proches de w_b . Cet espacement est logarithmique avec un groupement proche de $-w_b$ et pose des problèmes de discrétisation. Le choix d'une large bande d'approximation est d'autre part souhaitable, parce que l'approximation se comporte le mieux à l'intérieur de l'intervalle et pas à sa limite, donc certaines marges doivent être maintenues [5].

3.3 Régulateur PID

Les régulateurs PID fonctionnent en boucle fermée, ils ciblent l'erreur entre la réponse du système et l'entrée de référence souhaitée à laquelle ils attribuent un gain proportionnel de l'erreur k_p (contrôleur P), un gain à l'intégrale de l'erreur k_i (contrôleur I) et un gain à la dérivée de l'erreur k_d (contrôleur D). Ces actions jouent sur les caractéristiques de la réponse en boucle fermée, à savoir le temps de réponse, le dépassement, les oscillations et l'état d'équilibre, oscillations et l'erreur en régime permanent.

Ce régulateur est défini mathématiquement par l'équation suivante :

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (3.16)$$

$e(t)$ étant l'erreur de réglage.

La fonction de transfert d'un régulateur PID standard, avec filtrage de la dérivée est comme suit :

$$U(s) = K_p + \frac{1}{T_i s} + \frac{T_d s}{1 + \frac{T_d}{N} s} \quad (3.17)$$

avec : $T_i = \frac{1}{K_i}$ et $T_d = K_d$
 N : fréquence de filtrage

Le tableau ci-dessous résume l'effet de chaque action sur la réponse du système :

TAB. 3.1 : Tableau d'effet de chaque action

Action	P	I	D
Action statique	Diminue l'erreur	Élimine l'erreur	Aucune
Action dynamique	Augmente la rapidité	Diminue la rapidité et l'amortissement	Augmente la rapidité et l'amortissement

3.4 Commande par PD

Elle est l'une des commandes classiques les plus utilisées en asservissement de position des robots industriels vu sa maîtrise, simplicité d'implémentation et son faible coût de calcul. Sa loi se donne par :

$$u(t) = k_p e(t) + k_d \dot{e}(t) \quad (3.18)$$

Où :

$u(t) \in \mathbb{R}^3$: représente la commande ;

$e(t) \in \mathbb{R}^3$: l'erreur de poursuite (écart entre position désirée et position mesurée) ;

$K_p, K_d \in \mathbb{R}^{3 \times 3}$: sont les paramètres du régulateur, déterminés de façon à satisfaire un cahier de charges.

La structure de commande est illustrée dans la figure 3.1.

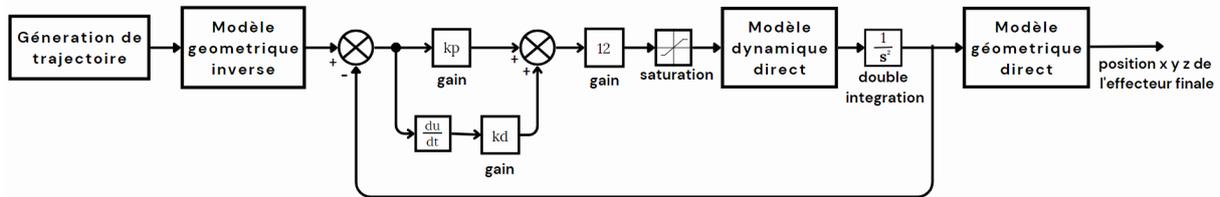


FIG. 3.1 : Schéma de commande du robot Delta avec régulateur PD

3.5 Commande par PD d'ordre fractionnaire

Ce régulateur, qui est un cas particulier des régulateurs PID d'ordre fractionnaire proposés par Podlubny, est une généralisation du régulateur PD, sa loi de commande est définie ainsi

$$u(t) = K_p e(t) + K_d D^r e(t) \tag{3.19}$$

Avec $0 \leq r \leq 2$: l'ordre de dérivation. Sa fonction de transfert est :

$$C(s) = k_p + k_d s^r \tag{3.20}$$

La structure d'un régulateur PD d'ordre fractionnaire est illustrée dans la figure 3.2.

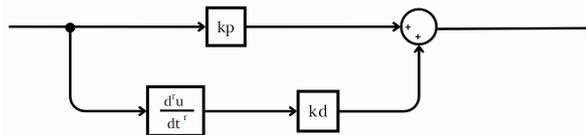


FIG. 3.2 : Structure d'un régulateur PD d'ordre fractionnaire

Pour la conception des systèmes de commande à ordre fractionnaire, nous avons utilisé la toolbox **FOMCON**. Cette boîte à outils est basée sur la théorie des systèmes à ordre fractionnaire, qui permet de modéliser et de contrôler des systèmes dynamiques dont les comportements ne peuvent pas être entièrement décrits par des équations différentielles d'ordre entier.

La boîte à outils FOMCON fournit des fonctionnalités pour la modélisation, l'identification, la simulation et la conception de régulateurs pour les systèmes à ordre fractionnaire

Le calcul des coefficients laisse le choix entre plusieurs méthodes de difficulté croissante. D'une part, des méthodes expérimentales sont faciles à mettre en place. Par ailleurs, il existe des méthodes mathématiques avancées permettant d'obtenir les coefficients adéquats pour un système donné.

Dans notre cas, nous avons utilisé l'**algorithme d'optimisation génétique**.

Dans le paragraphe suivant, nous donnons quelques notions de base sur ces algorithmes

3.6 L'algorithme génétique

Les algorithmes génétiques sont des processus d'optimisation itératifs. Ils permettent d'approcher la solution d'un problème d'optimisation (pour nous, la recherche du minimum d'une fonction coût) en minimisant les mécanismes de l'évolution conduite par le génie génétique. L'idée est de faire évoluer un ensemble de solutions à un problème donnée, dans l'optique de trouver les meilleurs résultats. Ce sont des algorithmes dits stochastiques, car ils utilisent itérativement des processus aléatoires.

La phase initiale consiste à choisir une population quelconque de n_{pop} individus candidats à devenir la solution optimale du problème. Prenons à titre d'exemple, $n_{pop} = 100$, chaque individu sera identifié à son **génotype**, c'est-à-dire à un triplet d'entiers (K_p, r, K_d) .

Pour passer de la k - ième à la $(k + 1)$ - ième génération d'individus, on évalue chacun des n_{pop} génotypes de la k -ième génération et on ne retient que les 20 meilleurs, c'est-à-dire ceux dont le coût est le moins élevé. On définit ensuite 80 nouveaux individus par croisements et mutations des 20 meilleurs génotypes. De la sorte, génération après génération, le nombre d'individus reste constant et les caractéristiques de la population s'améliorent (au sens où les valeurs de la fonction coût tendent à diminuer).

On répète le cycle **sélection-croisement-mutation** un certain nombre de fois. De manière un peu arbitraire. Le meilleur génotype de la dernière génération est alors utilisé pour fixer les valeurs de (K_p, r, K_d) [18].

Population initiale

Chaque individu possède trois gènes : les paramètres (K_p, r, K_d) . On associe à chaque paramètre x_i dans l'intervalle $[x_i^{\min}, x_i^{\max}]$ une chaîne binaire $b_0 b_1 b_2 \dots b_{l-1}$ définie par l bits. A cette chaîne correspond une valeur entière naturelle :

$$N(x_i) = \sum_{i=0}^{l-1} 2^{l-1-i} b_i \quad (3.21)$$

Le paramètre réel x_i de l'espace de recherche relatif à $N(x_i)$ est obtenu par mise à l'échelle linéaire :

$$x_i = \frac{x_i^{\max} - x_i^{\min}}{2^l - 1} N(x_i) + x_i^{\min} \quad (3.22)$$

Chaque individu est donc représenté par 3 paramètres de l bits.

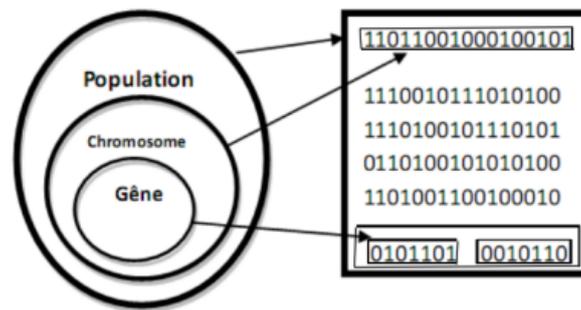


FIG. 3.3 : Hiérarchie dans l'AG

Sélection des individus

Pour chaque génération, il faut retenir les 20 meilleurs génotypes pour créer la génération suivante. Parmi ces génotypes, les 5 meilleurs seront enregistrés dans une base de donnée.

Génie génétique [19]

- Croisements : consiste à obtenir deux nouveaux individus enfants en combinant les chromosomes d'une paire d'individus parents.

-La mutation : Afin de limiter le risque d'apparition de clones dans la $(k + 1)$ - ième génération, chaque génotype obtenu par croisement de deux génotypes de la k - ième génération subit une mutation sur l'un ou plusieurs de ses gènes.

-Nouvelle génération : On a choisi de conserver les 20 meilleurs génotypes de la k - ième génération pour constituer la $(k + 1) - i$ ème génération. Ces génotypes sont regroupés dans une liste L.

$$L = [c_0, \dots, c_{19}] \quad (3.23)$$

Les 80 autres génotypes de la $(k + 1)$ - ième génération sont créés en deux temps.

- Vingt génotypes de la $(k + 1)$ - ième génération sont issus de croisements du meilleur génotype de la k - ième génération (c'est-à-dire c_0) avec 10 génotypes choisis aléatoirement parmi c_1, \dots, c_{19} .

- Soixante autres génotypes de la $(k + 1)$ - ième génération sont issus par croisements de 30 couples formés à partir des génotypes c_1, \dots, c_{19} .

Différent critères d'arrêt peuvent être choisis, l'AG est souvent arrêté lorsque l'on atteint un certain nombre de générations fixé a priori ou lorsque la population n'évolue plus assez rapidement.

L'algorithme se résume donc comme suit :

Construction et évaluation d'une **population initiale** ;
Jusqu'à atteindre un **critère d'arrêt** :
sélection d'une partie de la population,
reproduction des individus sélectionnés,
mutation de la descendance,
évaluation du degré d'adaptation de chaque individu,
remplacement de la population initiale par une nouvelle population.

3.7 Simulation

Dans cette section nous présenterons les résultats de la simulation des deux régulateurs sur le robot ISIR88 à l'aide de "Matlab SIMULINK", tel que les paramètres de chacun sont donnés comme suit :

- **Régulateur PD :**
 $K_p = 1.1, K_d = 0.001.$
- **Régulateur PD d'ordre fractionnaire :**
 $K_p = 1.1, K_d = 0.001, \mu = 1.15.$

3.7.1 Test de poursuite

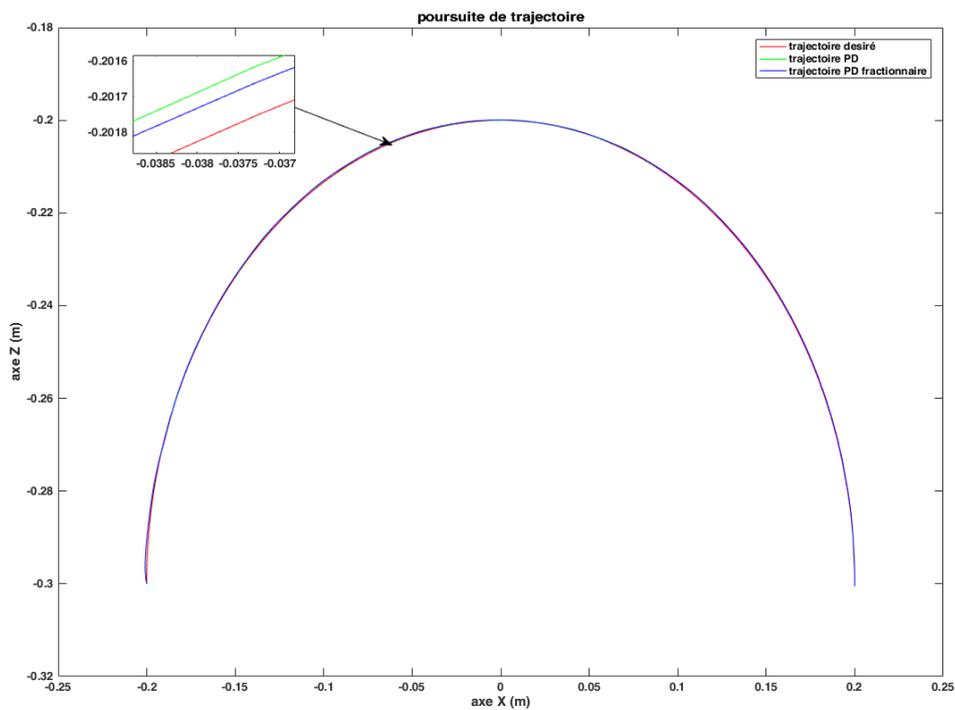


FIG. 3.4 : Poursuite de trajectoire dans l'espace opérationnel

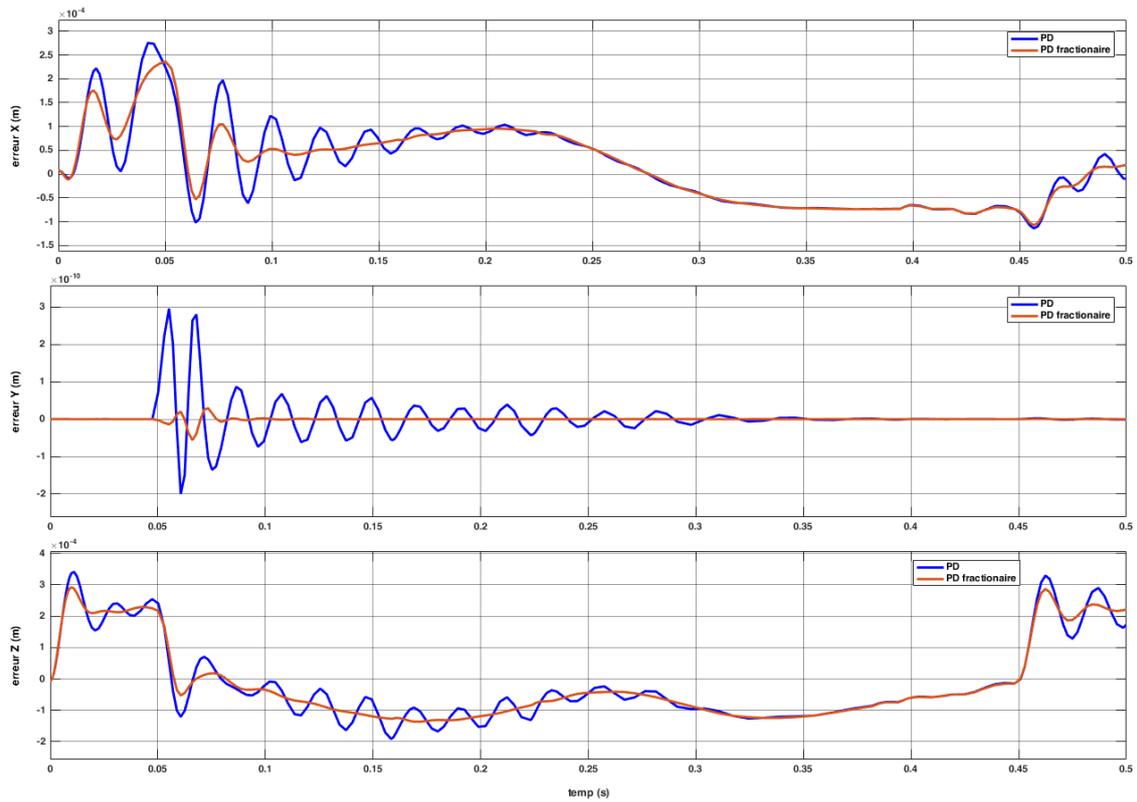


FIG. 3.5 : Erreurs de position opérationnelle obtenus par les deux régulateurs

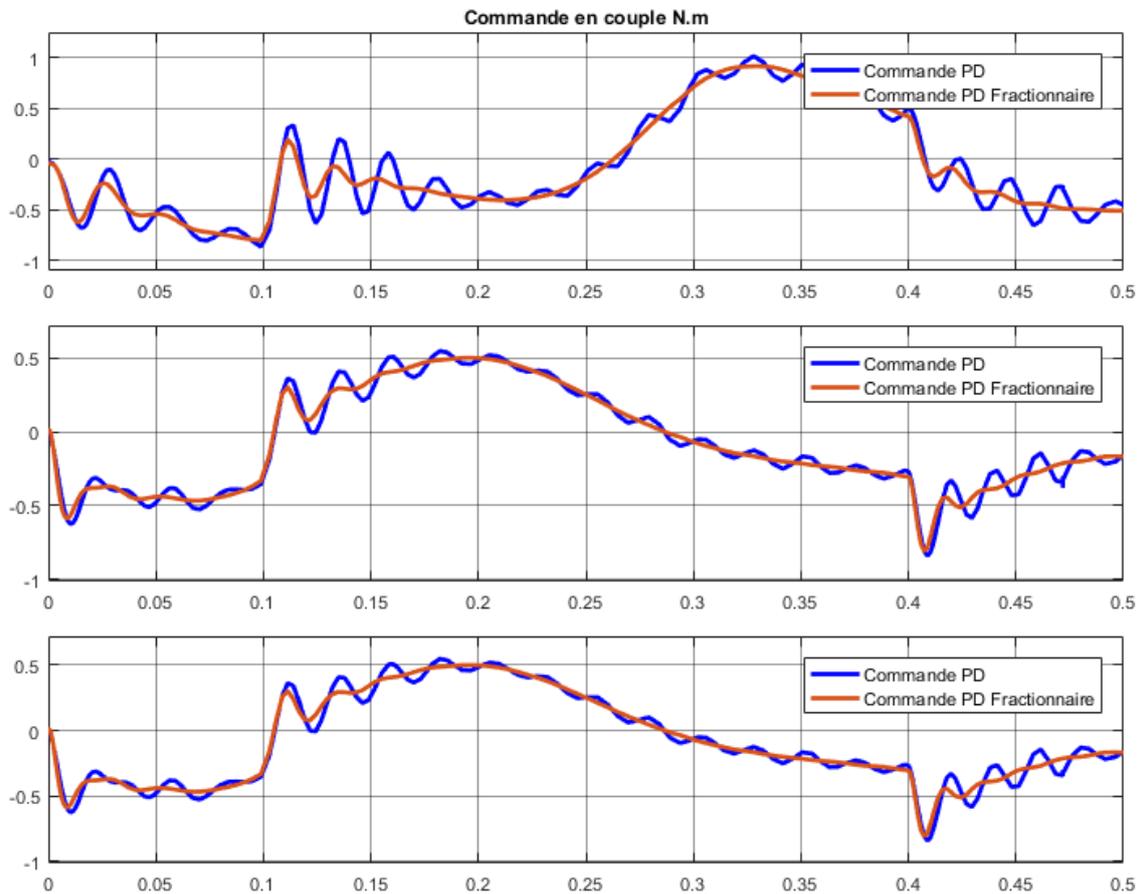


FIG. 3.6 : Couples obtenus par la simulation des deux régulateurs

3.7.2 Test de robustesse

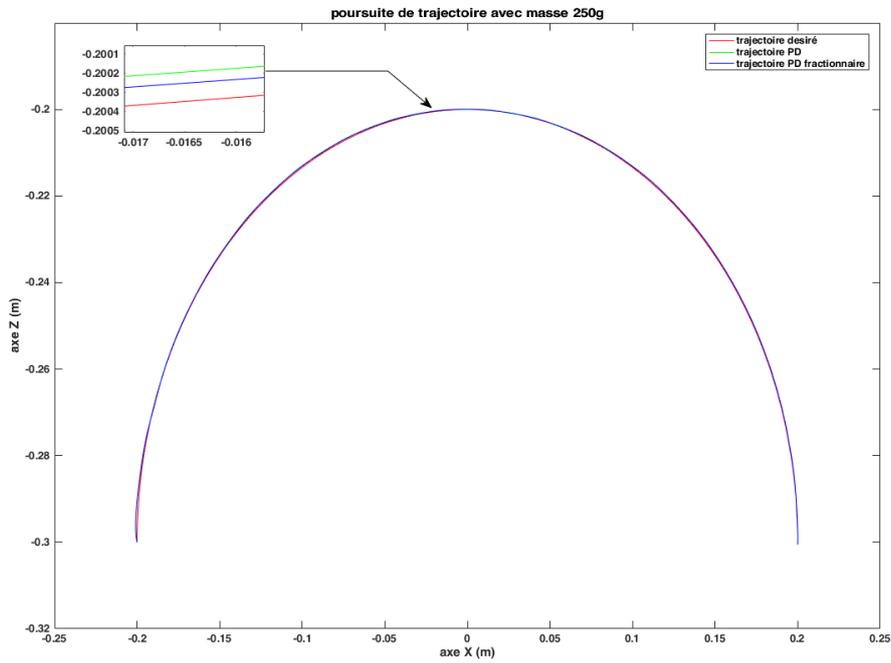


FIG. 3.7 : Poursuite de trajectoire dans l'espace opérationnel avec une masse de 250g

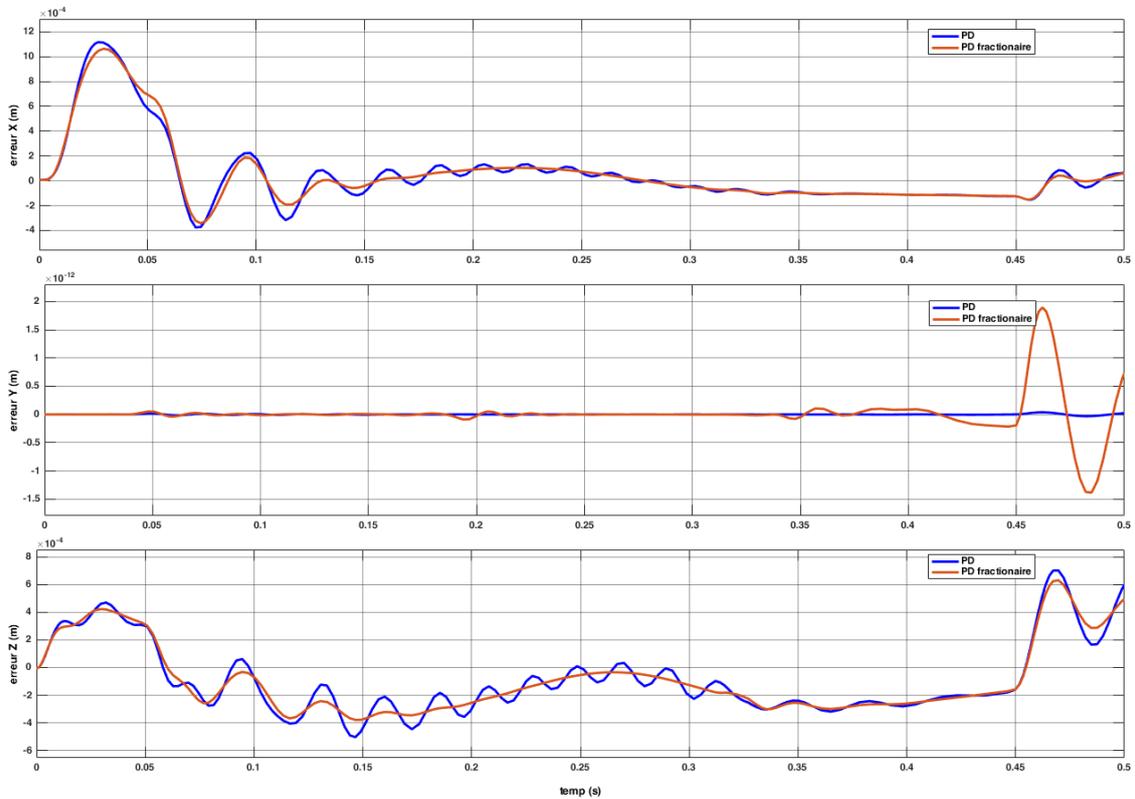


FIG. 3.8 : Erreurs de position opérationnelle obtenus par les deux régulateurs avec une masse de 250g

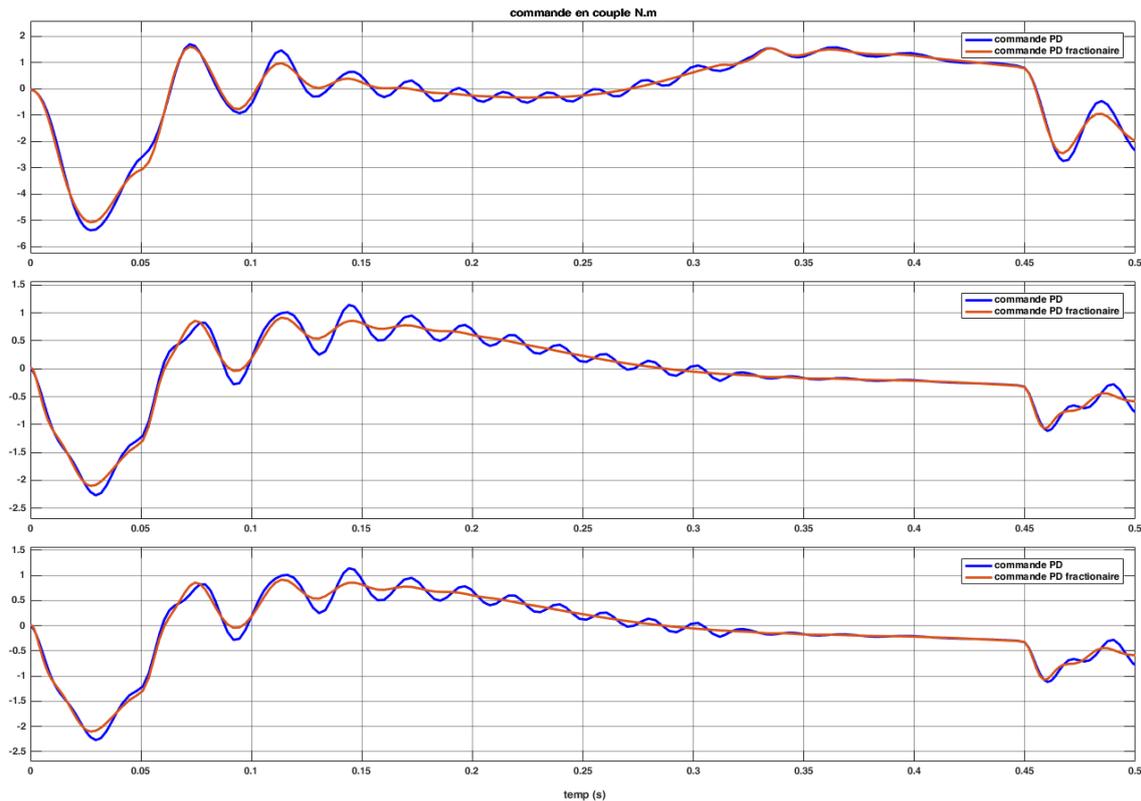


FIG. 3.9 : Couples obtenus par la simulation des deux régulateurs avec une masse de 250g

3.7.3 Comparaison et discussion

Afin d'évaluer les performances de poursuite et de robustesse, le critère de l'erreur quadratique moyenne (the Root Mean Square Error (RMSE)) sera utilisé. Leur expression est donnée comme suit :

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y_{d_i})^2} \quad (3.24)$$

où y_d est la trajectoire désirée, y_i est la trajectoire actuelle et n désigne le nombre d'échantillons dans une itération.

La figure 3.4 présente la poursuite de trajectoire dans l'espace opérationnel sous la commande proposée (la commande PD et PD fractionnaire) et les figures 3.5, 3.6 montre l'erreur de poursuite des positions opérationnelle et les couples de commande respectivement, On peut voir que la trajectoire effectuée suit la trajectoire désirée et que le régulateur PD avec ordre fractionnaire aboutis a des résultats meilleurs que ceux du régulateur PD (en vibration et en amplitude de l'erreur). Cette amélioration peut aussi être observée à travers le critère RMSE. le tableau 3.2 résume les performances obtenues.

TAB. 3.2 : Performances à travers le critère RMSE

Régulateur	PD	PD fractionnaire
RMSE axe-x (m)	$1.013 \cdot 10^{-4}$	$1.975 \cdot 10^{-5}$
RMSE axe-y (m)	$7.152 \cdot 10^{-13}$	$5.006 \cdot 10^{-14}$
RMSE axe-z (m)	$1.729 \cdot 10^{-4}$	$2.22 \cdot 10^{-4}$

Pour le test de robustesse, une charge supplémentaire de 250 g est ajoutée à la nacelle, afin de tester la poursuite de trajectoire en présence des charges utiles, les résultats de simulation sont présentés dans les figures, 3.8 3.9. On constate, une légère dégradation au niveau de l'erreur ce qui est logique, car les régulateurs ont été calculés pour un robot sans charge. On remarque aussi que le régulateur PD avec ordre fractionnaire reste toujours meilleur que le PD classique, ceci est visible à travers le calcul du RMSE figurant dans le tableau en dessous.

TAB. 3.3 : Performances de poursuite pour une charge de 250 g à travers le critère RMSE

Régulateur	PD	PD fractionnaire
RMSE axe-x (m)	$6.347 \cdot 10^{-5}$	$6.127 \cdot 10^{-5}$
RMSE axe-y (m)	$7.35 \cdot 10^{-14}$	$2.247 \cdot 10^{-14}$
RMSE axe-z (m)	$5.988 \cdot 10^{-4}$	$4.948 \cdot 10^{-4}$

3.8 Conclusion

Dans ce chapitre, nous avons élaboré des régulateurs PD et PD d'ordre fractionnaire de robot Delta à l'aide d'un algorithme d'optimisation génétique, un algorithme méta-heuristique qui estime les gains et les ordres fractionnaires.

Nous avons également comparé les résultats obtenus à partir des erreurs de position dans les régulateurs PD et PD d'ordre fractionnaire, dans lesquels le PD d'ordre fractionnaire a montré des résultats meilleur en termes de la robustesse.

Chapitre 4

La commande "Time Delay Control"

4.1 Introduction

Dans la majorité des théories de commande, on traite des systèmes bien définis mathématiquement et relativement simple.

Cependant, dans de nombreuses situations pratiques, les paramètres du système sont mal connus, complexes, ou dépendent de l'environnement ce qui crée de grandes variations imprévisibles des paramètres du système et des perturbations inattendues. Les véhicules sous-marins, les robots manipulateurs et les systèmes autonomes en sont quelques exemples. Plusieurs méthodes de commandes ont été développées afin de contrer ce problème, la première étant la commande adaptative qui met à jour les gains du contrôleur soit à l'aide d'une estimation récursive des paramètres du système afin que la sortie du système suive la réponse souhaitée. Une méthode appelée time delay control (TDC).

Dans ce chapitre, nous allons synthétiser théoriquement la commande (TDC) simple que nous allons, par la suite appliquer sur le robot Delta ISIR88, TDC avec mode glissant d'ordre fractionnaire et une commande adaptative avec mode glissement basée sur la TDC.

4.2 Time Delay Control

L'objectif principal de la TDC est d'utiliser l'observation passée de la réponse du système comme entrée de commande à l'instant actuel pour modifier immédiatement les actions de commande au lieu d'identifier les paramètres ou d'ajuster le gain du régulateur du système de commande, ce qui conduit à un régulateur indépendant de modèle, c'est une compensation sans aucune utilisation de modèle dynamique.

Le modèle dynamique du robot est donné par l'équation :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + d(t) = \tau \quad (4.1)$$

où $M(q) \in \mathbb{R}^{n \times n}$ est la matrice d'inertie, $C(q, \dot{q})\dot{q} \in \mathbb{R}^n$ est un vecteur résultant des forces de Coriolis et centrifuges, $G(q) \in \mathbb{R}^n$ est le vecteur du couple de gravité, $d(t) \in \mathbb{R}^n$ est le vecteur des perturbations externes, et $\tau \in \mathbb{R}^n$ est le vecteur des couples des articulations.

L'équation (4.1) peut être réécrite comme suit :

$$(M(q) - \bar{M})\ddot{q} + \bar{M}\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + d(t) = \tau \quad (4.2)$$

où \bar{M} est une matrice diagonale constante.

Une représentation compacte de (4.2) est donnée par :

$$\ddot{q}_t = N_t + \bar{M}^{-1}\tau_t \quad (4.3)$$

où $N_t = -\bar{M}^{-1}[C(q_t, \dot{q}_t)\dot{q}_t + G(q_t) + d(t) + (M(q_t) - \bar{M})\ddot{q}_t]$.

L'indice t indique l'instant actuel.

Pour obtenir une estimation de N_t , on utilise un bloc TDE (Time Delay Estimator) qui utilise les mesures passées (retardées avec un pas d'exécution) afin de calculé \hat{N}_t :

$$\hat{N}_t = N_{t-L} = \ddot{q}_{t-L} - \bar{M}^{-1}\tau_{t-L} \quad (4.4)$$

dans lequel L est le temps différé et \hat{N}_t est l'estimation de N_t .

En général, L est choisi comme le temps d'échantillonnage de la mise en œuvre pratique [20].

Comme le montre (4.4), le principal avantage de la TDE réside dans sa structure simple, où il est beaucoup plus facile de calculer $\ddot{q}_{t-L} - \bar{M}^{-1}\tau_{t-L}$ qui se constitue de signaux mesurables, plutôt que de calculer N_t , tel que N_t est exprimé par des fonctions non linéaires et des perturbations.

Pour la mise en œuvre, l'accélération passée \ddot{q}_{t-L} est calculée numériquement à l'aide de l'équation suivante :

$$\ddot{q}_{t-L} = \frac{q_t - 2q_{t-L} + q_{t-2L}}{L^2}. \quad (4.5)$$

Les erreurs de suivi de position et de vitesse sont définies par :

$$\tilde{q} = q_d - q, \quad \dot{\tilde{q}} = \dot{q}_d - \dot{q} \quad (4.6)$$

où q_d est le vecteur de la trajectoire désirée dans l'espace articulaire.

Le régulateur TDC est donné comme suit [20] :

$$\tau_t = \bar{M} (\ddot{q}_d + K_d \dot{\tilde{q}} + K_p \tilde{q}) + \tau_{t-L} - \bar{M} \ddot{q}_{t-L}. \quad (4.7)$$

En substituant (4.4) et (4.7) dans (4.3), on obtient le vecteur d'erreurs TDE suivant H^t :

$$\ddot{\tilde{q}}_t + K_d \dot{\tilde{q}}_t + K_p \tilde{q}_t = -\bar{M} (\hat{N}_t - N_t) = -\bar{M} H^t. \quad (4.8)$$

Ces erreurs TDE sont bornées [20], si $\|I - M^{-1}(q_t) \bar{M}\| < 1$.

Formellement, $t \geq 0$; aussi, l'erreur d'estimation du TDE est bornées par la constante H^* . C'est-à-dire $|\hat{N}_t - N_t| \leq H^*$. Ça implique que les gains de commande doivent être sélectionnés pour garantir cette limitation. Ainsi, en général, des petits gains de contrôle sont utilisés pour satisfaire l'inégalité. En outre, si les gains de commande sont incorrectement petits, la performance se dégrade. Contrairement, si ceux-ci deviennent trop grands pour une réponse rapide, ils ont tendance à produire une instabilité du système.

La structure de commande TDC est représentée par la figure 4.6 :

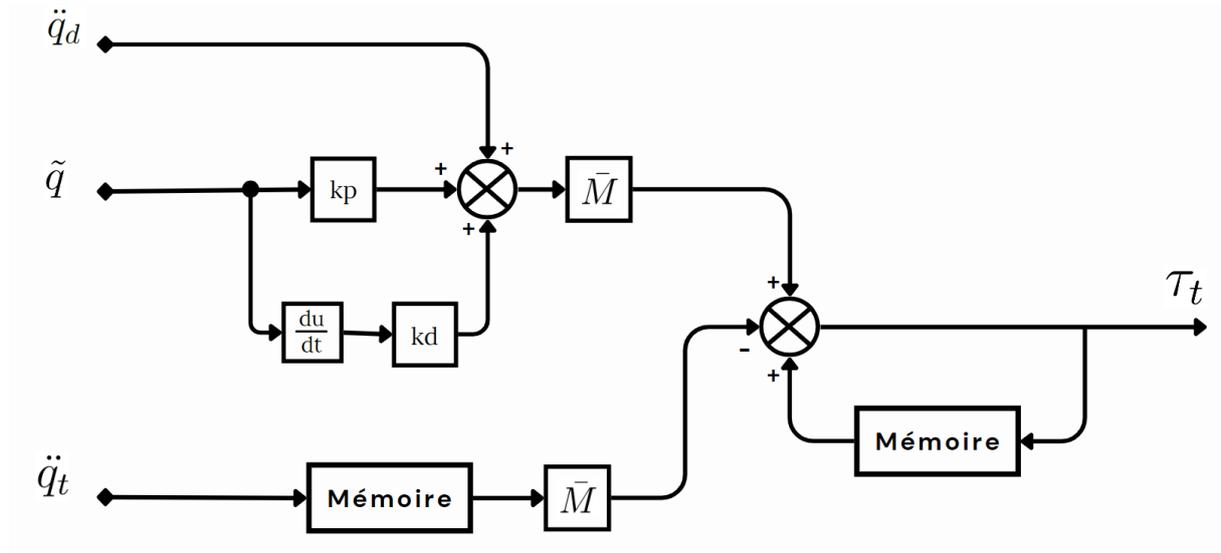


FIG. 4.1 : Schéma du régulateur TDC

4.3 La TDC par mode glissement d'ordre fractionnaire

Afin de compenser l'effet des erreurs TDE, éliminer les non-linéarités et augmenter la robustesse du régulateur tout en assurant un temps de convergence fini, un nouveau terme est généralement ajouté à la commande [21].

4.3.1 Le mode glissant et le phénomène de Chattering

L'approche de mode glissant consiste à amener la variable qu'on désire annuler vers la surface de glissement. Lorsqu'on atteint la surface de glissement, la convergence de l'erreur est faite à l'aide d'une logique de commutation appropriée autour de celle-ci jusqu'au point d'équilibre, d'où le phénomène de glissement. Cette commutation est due à l'introduction de la fonction $\text{sign}(s)$ dans la loi de commande tel que :

$$\text{sign}(s) = \begin{cases} -1 & \text{si } s < 0 \\ 0 & \text{si } s = 0 \\ 1 & \text{si } s > 0 \end{cases} \quad (4.9)$$

La commutation très rapide de la commande par mode glissant provoque un phénomène appelé "le phénomène de Chattering". Ce dernier est indésirable car il ajoute au spectre de la commande des composantes de haute fréquence. Le broutement (phénomène de Chattering) peut être réduit en remplaçant la fonction "sign" par une fonction de saturation adéquate qui filtre les hautes fréquences. On donne ci-dessous la fonction qu'on a utilisée [5] :

$$\text{sat}(s) = \begin{cases} -1 & \text{si } s < \varepsilon \\ \frac{s}{\varepsilon} & \text{si } |s| < \varepsilon \\ 1 & \text{si } s > \varepsilon \end{cases} \quad (4.10)$$

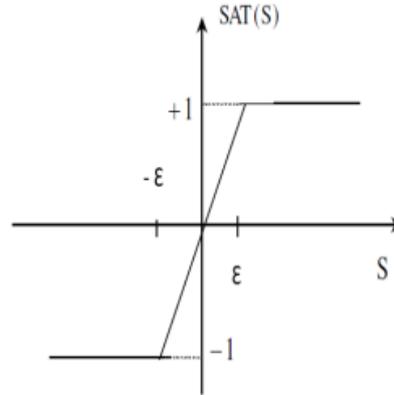


FIG. 4.2 : Fonction "saturation"

4.3.2 Critère de choix de la surface

Afin d'améliorer les performances déjà atteintes par [5], on utilise quatre surfaces de glissement la première est proposée par [22] :

$$S_1 = \dot{e} + k_1 D^{\alpha_2} [\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2-1} [\text{sig}(e)^{b_2}] \quad (4.11)$$

Trois autres surfaces ont été construites à base de cette dernière :

$$S_2 = \dot{e} + k_1 D^{\alpha_1} [\text{sig}(e)^{b_1}] \quad (4.12)$$

Avec $0 < \alpha_1 < 1, \quad 0 < b_1 < 1$

$$S_3 = \dot{e} + k_1 D^{\alpha_1} [\text{sig}(e)^{b_2}] \quad (4.13)$$

Avec $0 < \alpha_1 < 1, \quad 1 < b_2 < 2$

$$S_4 = \dot{e} + k_1 D^{\alpha_1} [\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2} [\text{sig}(e)^{b_2}] \quad (4.14)$$

Avec $0 < \alpha_1, \alpha_2 < 1 \quad 0 < b_1 < 1 \quad 1 < b_2 < 2$

Tel que : $\text{Sig}(x) = |x|. \text{sign}(x)$

et : $\text{Sig}(x)^b = |x|^b. \text{sign}(x)$

Après maintes simulations, nous avons constaté que la surface S_1 donne de meilleurs résultats d'où son utilisation dans la synthèse des lois de commandes

La surface choisie se compose de deux parties similaires qui sont :

$$k_1 D^{\alpha_2} [\text{sig}(e)^{b_1}], k_2 D^{\alpha_2-1} [\text{sig}(e)^{b_2}]$$

Pour visualiser l'effet de chaque paramètre, on se propose d'utiliser la première partie de notre surface avec $S_1 = 0$

$$S_1 = \dot{e} + k_1 D^{\alpha} [\text{sig}(e)^{b_1}]$$

Pour $S_1 = 0$ on aura :

$$\dot{e} = -k_1 D^{\alpha} [\text{sig}(e)^{b_1}]$$

a. Effet du paramètre b_1 sur la convergence

On fixe tous les paramètres tel que : $k_1 = 35$, $\alpha = 0.2$
Puis on simule pour différentes valeurs de b_1 .

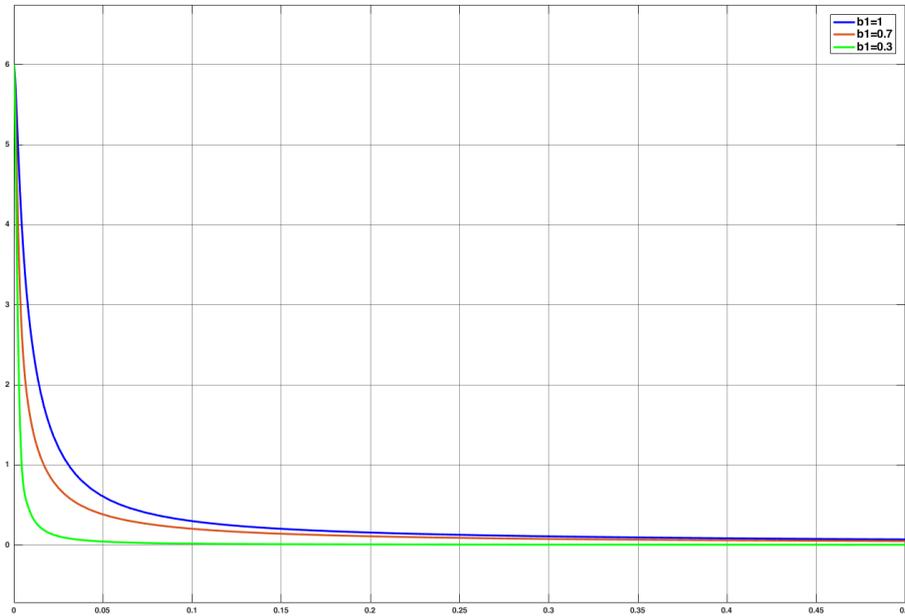


FIG. 4.3 : Effet de b_1 sur la convergence de l'erreur

On remarque que le temps de convergence est inversement proportionnel avec le valeur de b_1 .

b. Effet du paramètre α sur la convergence

On fixe tous les paramètres tel que : $k_1=5$, $b_1 = 1$
puis on simule pour différentes valeurs de α

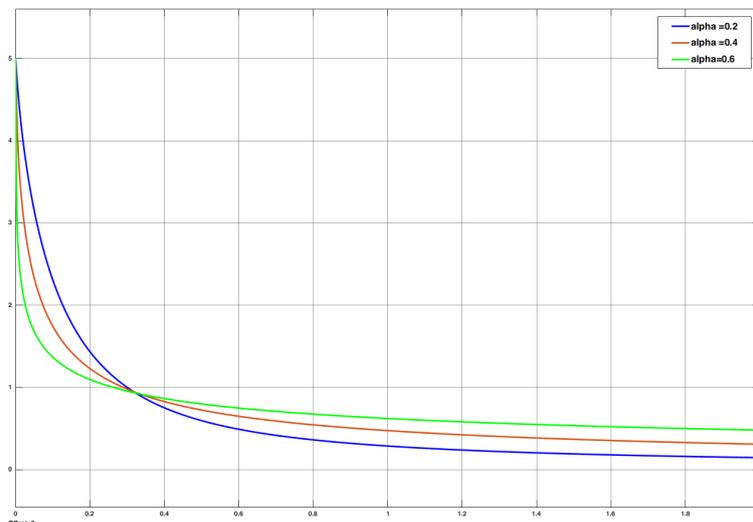


FIG. 4.4 : Effet de α sur la convergence de l'erreur

On remarque que l'effet de α se divise en deux : en régime transitoire un α élevée résulte une convergence rapide, mais en régime permanent la convergence est lente.

Pour un α faible le résultat est contraire i.e : la convergence commence lente en régime transitoire puis accélère en régime permanent.

En combinant les avantages des deux paramètre α et $b1$ et on utilisant l'algorithme génétique on obtient des valeurs optimales pour chaque paramètre ce qui induira une poursuite presque parfaite de la trajectoire désirée.

4.3.3 Conception de la commande et analyse de la converge

La loi de commande proposée est la suivante :

$$\tau_t = \tau_{t-l} - \bar{M}\ddot{q}_{t-l} + \bar{M} [\ddot{q}_d + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}] + K \text{sign}(S_1)] \quad (4.15)$$

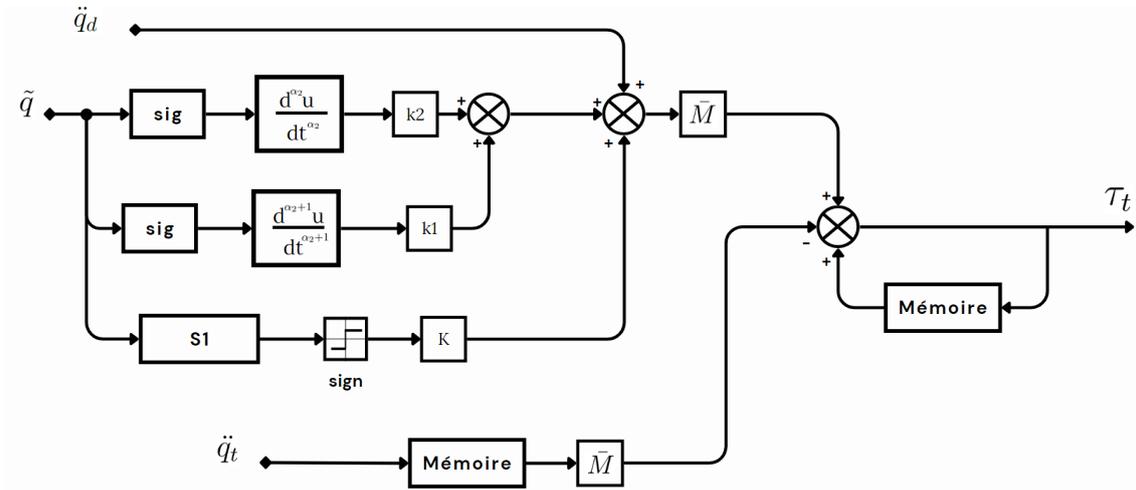


FIG. 4.5 : Structure de commande TDC S_1

La loi (4.15) doit garantir la convergence asymptotique de l'erreur de poursuite et sa dérivée par rapport au temps. Pour cela, on utilise le théorème de Lyapunov, avec la fonction de Lyapunov est donnée par :

$$V_1 = \frac{1}{2} S_1^T S_1 \quad (4.16)$$

Sa dérivée par rapport au temps est donnée par :

$$\dot{V}_1 = S_1^T \dot{S}_1 \quad (4.17)$$

En introduisant les équations (4.11) et (4.17) :

$$\begin{aligned} \dot{V}_1 &= S_1^T (\ddot{e}_t + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}]) \\ \dot{V}_1 &= S_1^T (\ddot{q}_{ref,t} - \ddot{q}_t + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}]) \end{aligned} \quad (4.18)$$

On remplace \ddot{q}_t par son expression donnée par l'équation (4.3), on obtient :

$$\dot{V}_1 = S_1^T (\ddot{q}_{ref,t} + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}] - \bar{M}^{-1}\tau_t - N_t) \quad (4.19)$$

En appliquant la loi de commande donnée par l'équation (4.15) on obtient :

$$\begin{aligned} \dot{V}_1 = S_1^T & [\ddot{q}_{ref,t} + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}] - N_t \\ - \bar{M}^{-1} (\tau_{t-L} - \bar{M} \ddot{q}_{t-L} + \bar{M} (\ddot{q}_{ref,t} & + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}] + K \text{sign}(S_1))] \end{aligned} \quad (4.20)$$

Donc :

$$\dot{V}_1 = S_1^T (-N_t - \bar{M}^{-1} (\tau_{t-L} - \bar{M} \ddot{q}_{t-L}) - K \text{sign}(S_1)) \quad (4.21)$$

D'après l'équation (4.3), on peut obtenir la suivante :

$$\dot{V}_1 = S_1^T (-N_t + \hat{N} - K \text{sign}(S_1)) \quad (4.22)$$

Comme $\|N_t - \hat{N}\| \leq H^*$, donc :

$$\dot{V}_1 \leq \sum_1^3 |S_i| (H^* - K_m) \quad (4.23)$$

Tel que K_m est la valeur propre minimale de la matrice K. Alors, si on choisit $K_m > N^*$, donc devient définie négative. Par conséquent, à partir de la définition de V_1 , nous pouvons conclure que $S_{t,1}$ converge asymptotiquement vers zéro.

4.4 Commande adaptative avec mode glissement basée sur la TDC

Pour compenser les diverses incertitudes, y compris les erreurs TDE induites par la TDC, de nouvelles lois adaptatives ont été développées comme par exemple la commande adaptatif par mode glissement (ASMC) [23] [20].

L'ASMC proposée basée sur la TDC. On conçoit une commande adaptative afin une loi adaptative afin de réduire le phénomène de chattering et les erreurs de poursuite. Cette méthode modifie l'expression classique de TDE avec une expression plus précise et l'introduction d'un gain adaptatif.

4.4.1 Conception de la commande

On reprend le modèle dynamique du robot donné par l'équation (4.1) :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + d(t) = \tau \quad (4.24)$$

et on choisit la surface (4.11).

A partir de l'approche TDC et (4.15) :

$$\begin{aligned} \tau_t^{TDC} &= -\hat{M}\ddot{q}_{t-L} + \tau_{t-L}^{TDC} + \hat{M}(\ddot{q}_d + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}] + K \text{sign}(S_1)) \\ &= -\hat{M} \underbrace{\hat{N}_{tt}}_{\text{TDE}} + \hat{M} (\ddot{q}_d + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}] + K \text{sign}(S_1)) \end{aligned} \quad (4.25)$$

Le schéma TDC proposé génère le couple d'entrée suivant τ_t qui utilise l'erreur TDE précédente.

$$\tau_t = \tau_t^{TDC} - \alpha \bar{M} \eta_{t-L} \quad (4.26)$$

où α est le paramètre d'ajustement et η_{t-L} est l'erreur TDE précédente qui est dérivée comme suit :

$$\eta_{t-L} = N_{t-L} - N_{t-2L}, = \hat{N}_t - \hat{N}_{t-L} \quad (4.27)$$

D'après (4.27), le τ_t^{TDC} dans (4.26) est représentée par :

$$\begin{aligned} \tau_t^{TDC} &= -(1 + \alpha) \bar{M} \ddot{q}_{t-L} + \alpha \hat{M} \ddot{q}_{t-2L} + (1 + \alpha) \tau_{t-L}^{TDC} - \alpha \tau_{t-2L}^{TDC} \\ &+ \bar{M} \left(\ddot{q}_d + k_1 D^{\alpha_2+1} [\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2} [\text{sig}(e)^{b_2}] + \hat{K} \text{sign}(S_1) + \lambda S_1 \right) \\ &= -(1 + \alpha) \bar{M} \hat{N}_t + \alpha \hat{M} \hat{N}_{t-L} + \hat{M} \left(\ddot{q}_d + k_1 D^{\alpha_2+1} [\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2} [\text{sig}(e)^{b_2}] + \hat{K} \text{sign}(S_1) + \lambda S_1 \right) \\ &= -\hat{M} \underbrace{\{ \hat{N}_t + \alpha \eta_{t-L} \}}_{\text{TDE}} + \hat{M} \left(\ddot{q}_d + k_1 D^{\alpha_2+1} [\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2} [\text{sig}(e)^{b_2}] + \hat{K} \text{sign}(S_1) + \lambda S_1 \right) \end{aligned} \quad (4.28)$$

La structure de commande ASMC basée sur la TDC est représentée par la figure 4.6 :

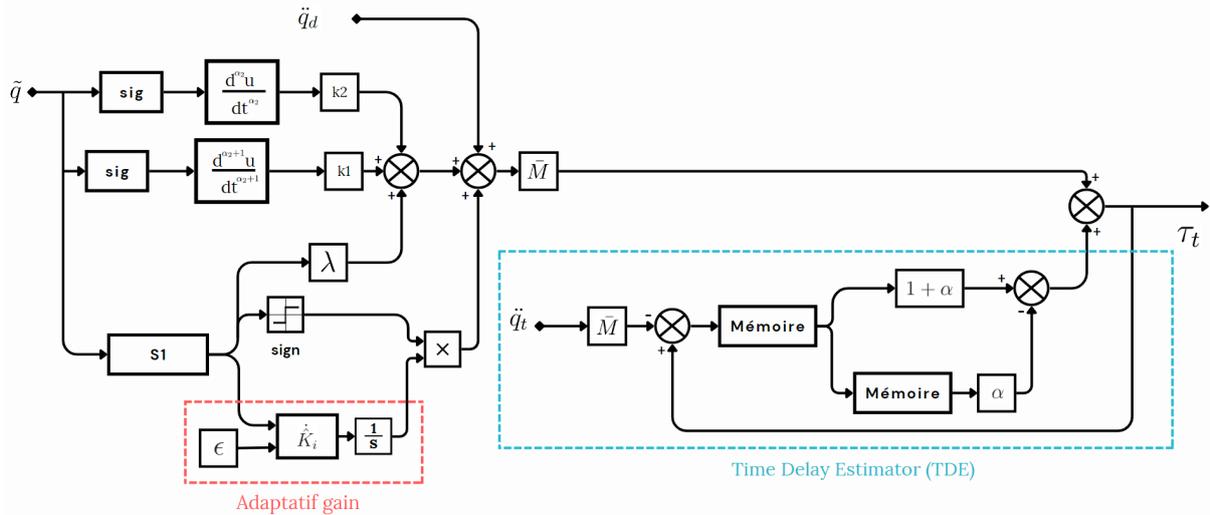


FIG. 4.6 : Structure de commande ASMC

4.4.2 Analyse de la convergence

Soit la fonction de Lyapunov suivante :

$$V(t) = \frac{1}{2} S_1^T S_1 + \frac{1}{2\phi} \sum_{i=1}^n \tilde{K}_i^2. \quad (4.29)$$

La dérivée temporelle de $V(t)$ est donnée par

$$\dot{V} = S_1^T \dot{S}_1 + \sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\tilde{K}}_i \quad (4.30)$$

Avec :

$$\begin{aligned} \dot{\hat{K}}_i &= \begin{cases} \phi |S_{i,1}|, & \text{si } \|S_1\|_\infty \geq \epsilon \\ -\beta\phi |S_{i,1}|, & \text{si } \|S_1\|_\infty < \epsilon \end{cases} \\ \hat{K}_i &= \begin{cases} K^*, & \text{si } \hat{K}_i \geq K^* \\ -K^*, & \text{si } \hat{K}_i < -K^* \end{cases} \end{aligned} \quad (4.31)$$

avec β, ϕ des constance positive et $\beta > 1$

la variable \tilde{K}_i definie comme suit :

$$\tilde{K}_i = \hat{K}_i - H_i^*$$

et comme η_t est bornée alors $\eta_t + \alpha\eta_{t-L}$ est aussi bornée par H_i^*

Où

$$H_i^t = \eta_t + \alpha\eta_{t-L}$$

$$H_i^* \geq |H_i^t|$$

Ainsi

$$\begin{aligned} \dot{V} &= S_1^T (\ddot{e}_t + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}]) + \sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\hat{K}}_i. \\ \dot{V} &= S_1^T (\ddot{q}_{d,t} - \ddot{q}_t + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}]) + \sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\hat{K}}_i. \end{aligned} \quad (4.32)$$

En substituant (4.3) dans (4.32) , on obtient :

$$\dot{V} = S_1^T (\ddot{q}_d - \bar{M}^{-1}\tau - N_t + k_1 D^{\alpha_2+1}[\text{sig}(e)^{b_1}] + k_2 D^{\alpha_2}[\text{sig}(e)^{b_2}]) + \sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\hat{K}}_i.$$

En remplaçant par la commande trouvée en (4.28) on obtient

$$\begin{aligned} \dot{V} &= S_1^T (\hat{N}_t + \alpha\eta_{t-L} - \hat{K} \text{sign}(S_1) - \lambda S_1 - N_t) + \sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\hat{K}}_i. \\ \dot{V} &= S_1^T (\eta_t + \alpha\eta_{t-L} - \hat{K} \text{sign}(S_1) - \lambda S_1) + \sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\hat{K}}_i. \end{aligned}$$

Si $\|S_1\|_\infty \geq \epsilon$, alors à partir de (4.30), et (4.31), nous avons

$$\sum_{i=1}^n \frac{1}{\phi} \tilde{K}_i \dot{\hat{K}}_i = - \sum_{i=1}^n |S_{i,1}| (\tilde{K}_i).$$

On a également

$$S_1^T (H^t - \hat{K} \text{sign}(S_1)) \leq \sum_{i=1}^n |S_{i,1}| (H_i^* - \tilde{K}_i).$$

La démonstration de cette inégalité peut être trouvée dans [24].

Enfin, on aura

$$\dot{V} \leq -\lambda S_{i,1}^T \quad (4.33)$$

Par conséquent, $V(t)$ diminue, ce qui permet à la variable glissante S_1 d'entrer dans un ensemble, $S = \{S_1 \mid \|S_1\|_\infty < \epsilon\}$. Puisque \dot{V} est indéfini dans ce collecteur glissant, la variable S_1 peut sortir de l'ensemble S , ce qui signifie que $\|S_1\|_\infty \geq \epsilon$. Dans ce cas, $\dot{V}(t)$ redevient négatif comme le montre (4.33) et la variable S_1 retourne dans l'ensemble S . La borne supérieure de S_1 peut être obtenue comme suit :

$$\frac{1}{2}\|S_1\|^2 \leq V \leq \frac{1}{2}\|S_1\|^2 + \frac{1}{2\phi} \sum_{i=1}^n \tilde{K}_i^2.$$

D'où,

$$\frac{1}{2}\|S_1\|^2 < \sum_{i=1}^n \left(\frac{1}{2}\epsilon^2 + \frac{1}{2\phi} \tilde{K}_i^2 \right)$$

4.5 Simulation

Dans cette section, nous présenterons les résultats de la simulation de la surface de glissement d'ordre fractionnaire (4.11) et l'ASMC basée sur la TDC sur le robot ISIR88 à l'aide de "Matlab SIMULINK". Les paramètres sont calculés avec l'algorithme génétique.

- **La commande TDC avec la surface S_1 :**

$$\alpha_2 = 0.4150, b_1 = 0.9, b_2 = 0.8, k_1 = 33.7295, k_2 = 86.9654, K = 383.8247, \bar{M} = 1.957810^{-6}.$$

- **L'ASMC basée sur la TDC :**

$$\alpha_2 = 0.4150, b_1 = 0.9, b_2 = 0.8, k_1 = 33.7295, k_2 = 86.9654, K = 383.8247, \bar{M} = 1.957810^{-6}, \alpha = 0.287, \Phi = 287.9118, \epsilon = 0.0377, \beta = 1.0569, \lambda = 49.92.$$

4.5.1 Test de poursuite

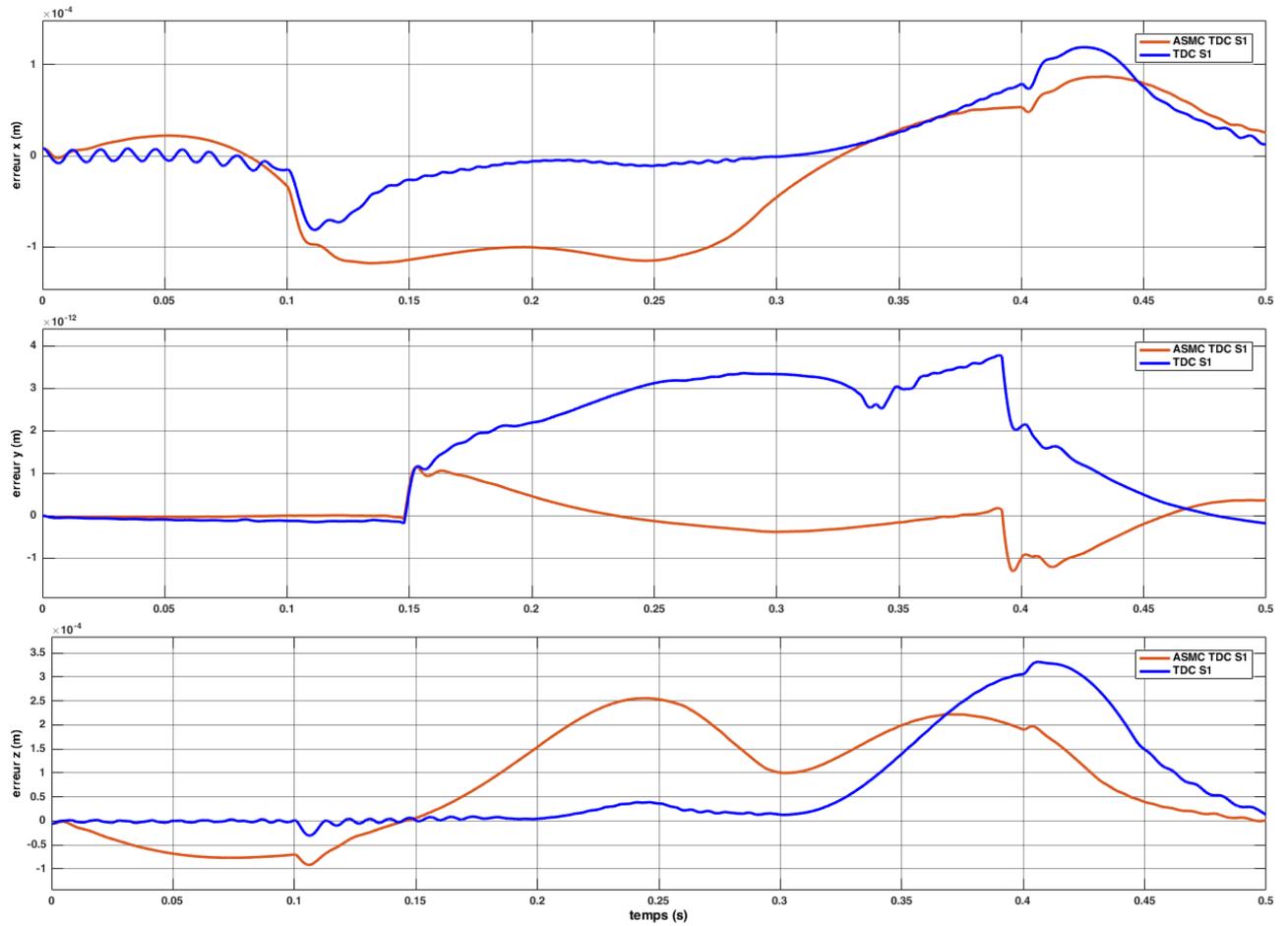


FIG. 4.7 : Erreurs de position opérationnelle obtenus en utilisant les deux méthodes TDC avec la fonction sign

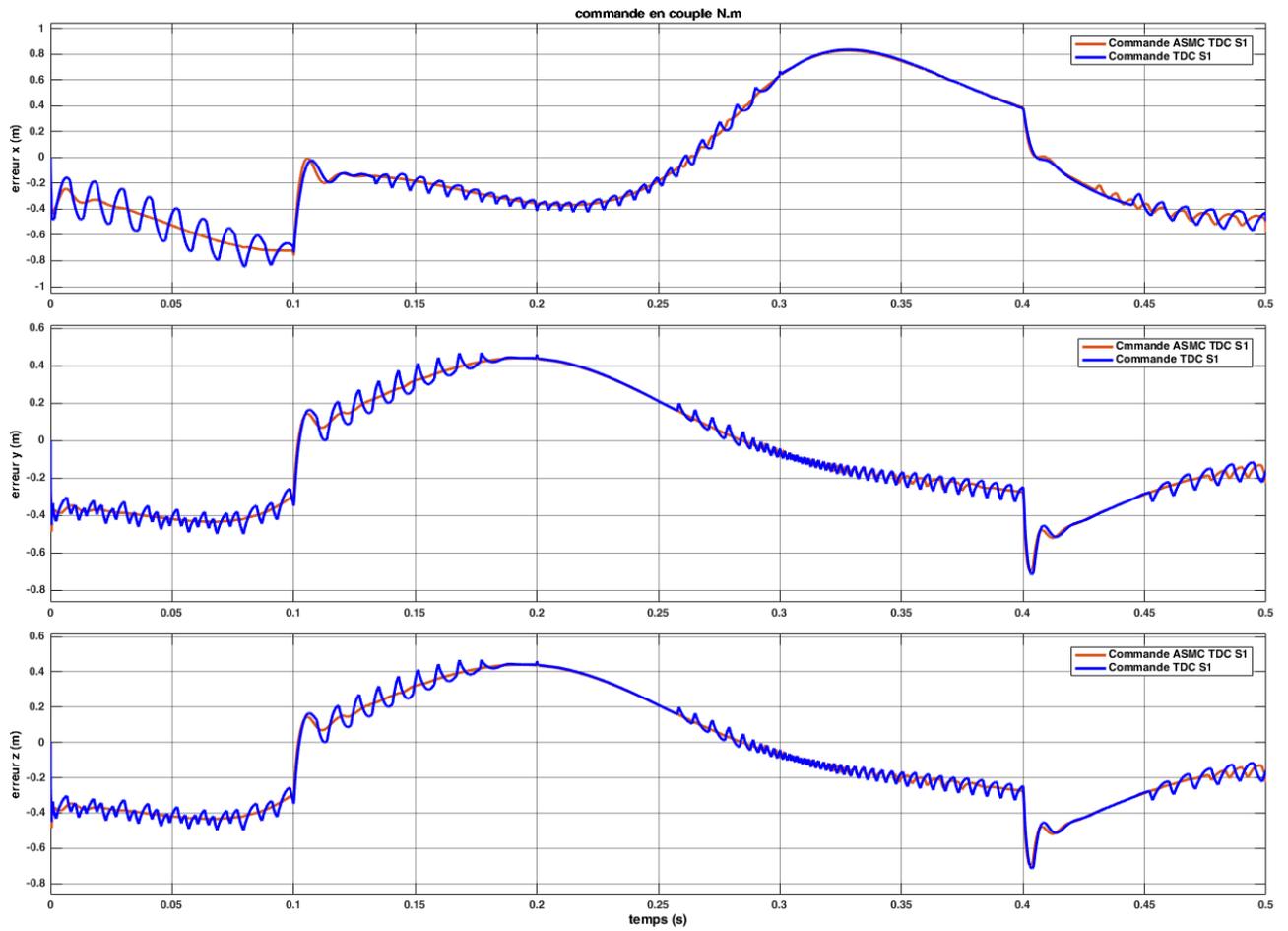


FIG. 4.8 : Couples de commande obtenus par les deux methodes TDC avec la fonction sign

On remarque que l'utilisation de la fonction **sign(s)** dans la commande génère beaucoup de vibration. C'est pour cela que dans toute les simulations qui suivent on remplace la fonction **sign(s)** par la fonction **tanh(s)**, afin de lisser la commande.

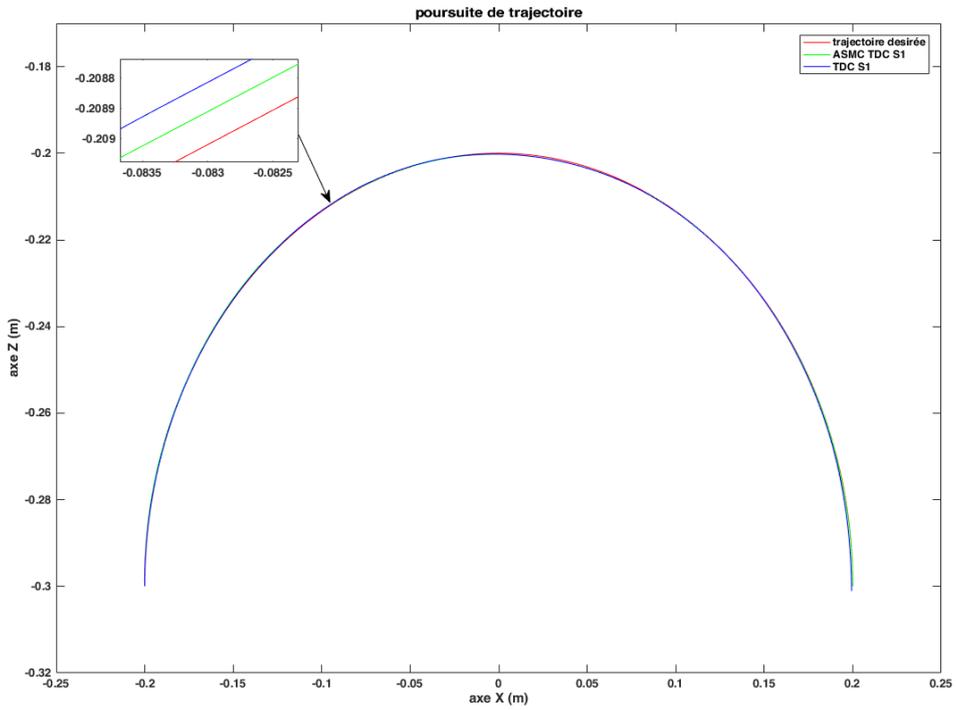


FIG. 4.9 : Poursuite de trajectoire dans l'espace opérationnel

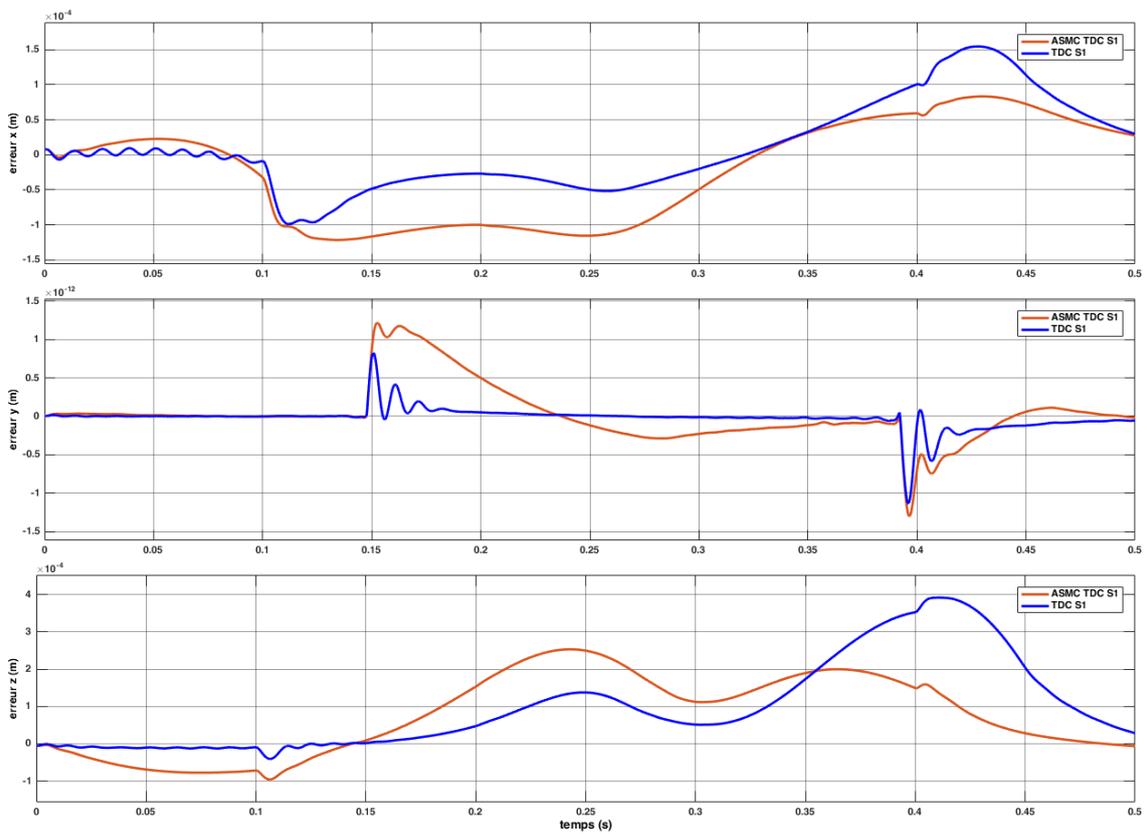


FIG. 4.10 : Erreurs de position opérationnelle obtenus en utilisant les deux méthodes TDC

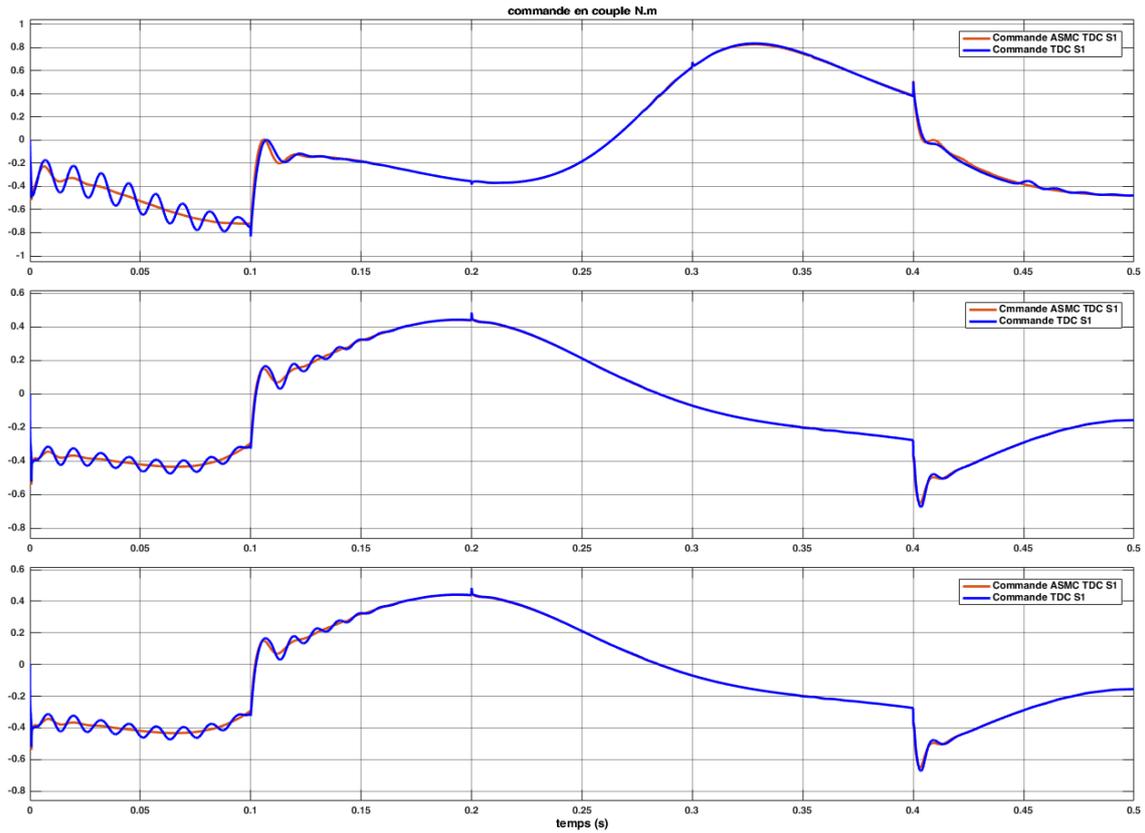


FIG. 4.11 : Couples de commande obtenus par les deux methodes TDC

4.5.2 Test de robustesse

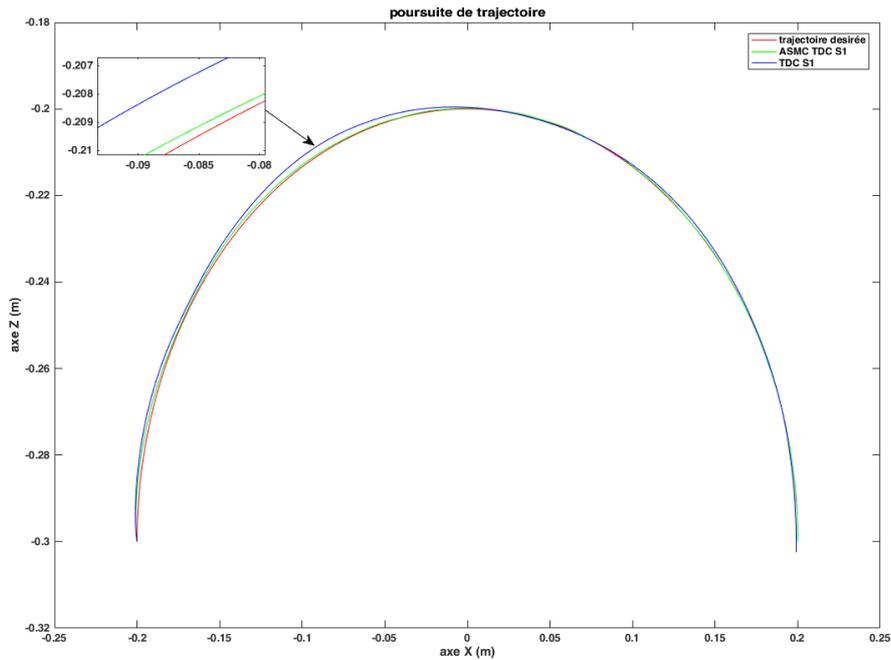


FIG. 4.12 : Poursuite de trajectoire dans l'espace opérationnel avec une masse de 250g et une perturbation

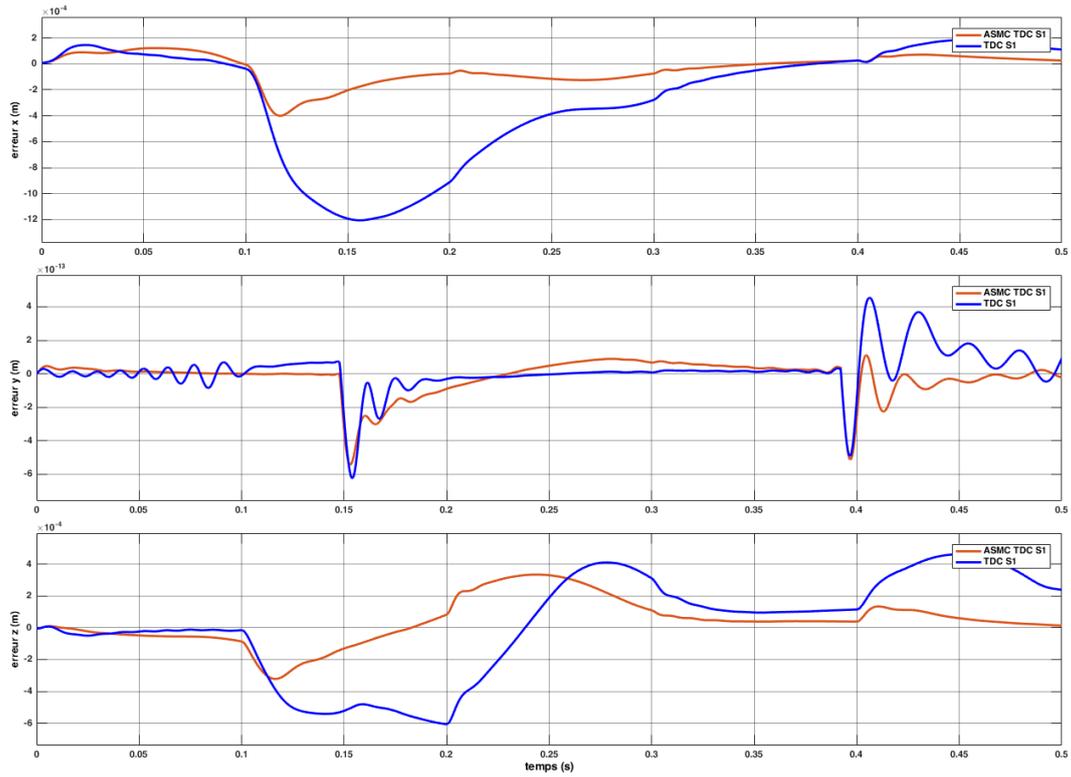


FIG. 4.13 : Erreurs de position opérationnelle obtenus par les deux méthodes TDC avec une masse de 250g

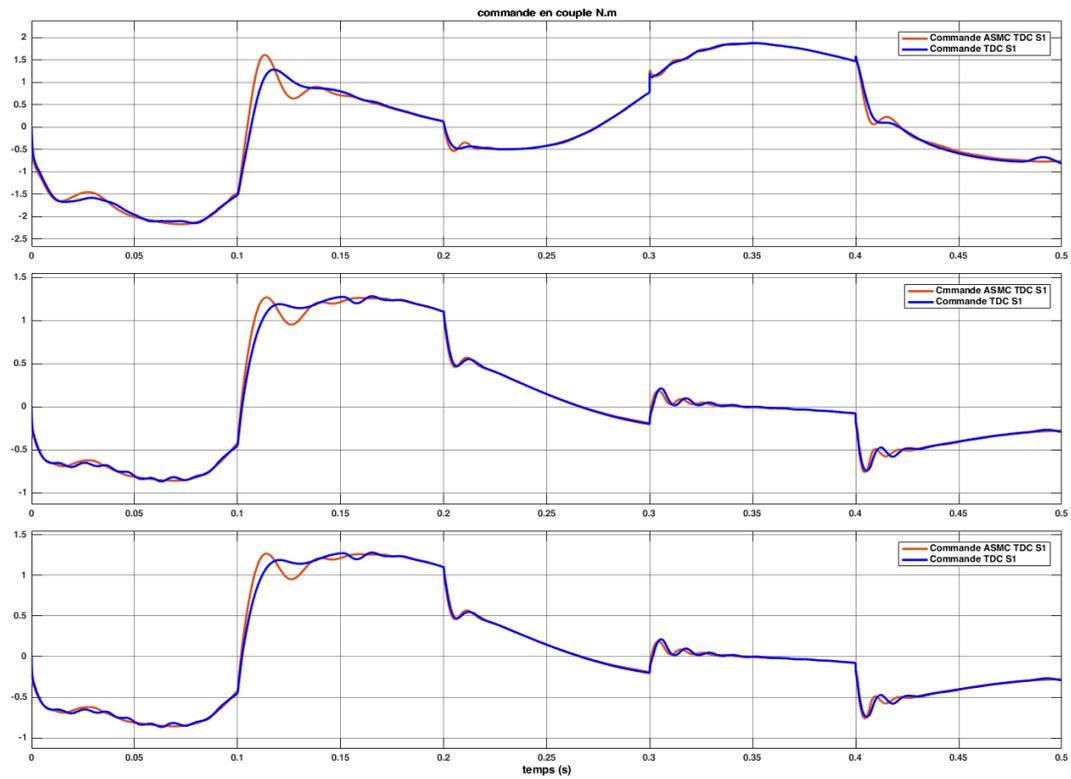


FIG. 4.14 : Couples obtenus par la simulation des deux méthodes TDC avec une masse de 250g et une perturbation

4.5.3 Comparaison et discussion

La figure 4.9 présente la poursuite de trajectoire dans l'espace opérationnel en utilisant les deux nouvelles méthodes (TDC avec S_1 et ASMC TDC avec S_1) et les figure 4.10, 4.11 montrent les erreurs de poursuite de position et les couples de commandes respectivement. Il est clair que le régulateur ASMC TDC S_1 génère une trajectoire plus proche de la référence que celle du régulateur TDC S_1 conventionnel. Cette différence peut aussi être observée dans la figure 4.10, mais cette différence reste petite. Le tableau 4.1 du critère RMSE nous permis de comparer les résultats.

TAB. 4.1 : Performances selon le critère RMSE

Régulateur	ASMC TDC S_1	TDC S_1
RMSE axe-x (m)	$2.75 \cdot 10^{-5}$	$2.97 \cdot 10^{-5}$
RMSE axe-y (m)	$1.482 \cdot 10^{-14}$	$5.444 \cdot 10^{-14}$
RMSE axe-z (m)	$6.323 \cdot 10^{-6}$	$2.816 \cdot 10^{-5}$

Le test de robustesse est l'ou les régulateurs TDC excellent, car il prend en considération la dynamique du robot et les perturbations grâce a l'estimateur TDE. Dans notre cas, nous avons simulé le suivi d'une trajectoire avec une charge utile de 250g et une perturbation. La figure 4.12 représente le mouvement du robot dans l'espace opérationnel et les figures 4.13, 4.14 montrent les erreurs de poursuite de position et les couples de commandes respectivement. L'effet de la masse et de la perturbation et clairement apparent dans la figure 4.12 où on peut voir son effet (la trajectoire devie légèrement au moment de l'application de la perturbation puis re-converge). Mais malgré les deviation causé par les perturbations, l'erreur reste acceptable et la commande reste très faisable.

On note aussi que le régulateur ASMC TDC S_1 donne de meilleurs résultats que ceux du TDC S_1 simple, et cela revient au fait qu'il contient un estimateur (TDE) plus évolué et un terme d'adaptation au niveau du régulateur TDE qui lui permet de mieux compenser les dynamiques et les perturbations inconnues. Ceci est visible à travers le calcul du RMSE figurant dans le tableau ci-dessous.

TAB. 4.2 : Performances de poursuite pour une charge de 250 g et une perturbation à travers le critère RMSE

Régulateur	ASMC TDC S_1	TDC S_1
RMSE axe-x (m)	$2.525 \cdot 10^{-5}$	$1.098 \cdot 10^{-4}$
RMSE axe-y (m)	$2.178 \cdot 10^{-14}$	$9.337 \cdot 10^{-14}$
RMSE axe-z (m)	$1.327 \cdot 10^{-5}$	$2.377 \cdot 10^{-4}$

4.6 Conclusion

Dans ce chapitre, nous avons élaboré deux régulateurs TDC avec mode glissant, ou nous avons choisi une surface fractionnaire unique ce qui nous a permis d'aboutir à de très bons résultats. Ensuite nous avons amélioré ce dernier en introduisant deux modifications, la première étant d'améliorer l'estimateur TDE, la deuxième est l'introduction d'un gain \hat{K} adaptatif au niveau de la commande, ce qui a permis d'améliorer les résultats, en réduisant l'erreur et en augmentant la robustesse.

Conclusion Générale

Dans ce mémoire, nous nous sommes intéressés à la modélisation et à la commande du robot Delta ISIR 88 disponible au sein du Laboratoire de Commande de Processus de l'École Nationale Polytechnique.

Pour ce qui est de modélisation, la caractérisation du robot a englobé plusieurs niveaux de modélisation. Nous nous sommes principalement intéressés aux modèles géométrique, cinématique et dynamique de la structure.

Cette modélisation nous a permis de mettre en évidence les particularités du robot parallèle Delta et ceci sur plusieurs aspects :

- La modélisation géométrique ne fait pas appel à des méthodes systématiques comme c'est le cas pour les robots sériels, mais à l'utilisation des formes géométriques caractérisant les liaisons passives et actionnées du robot.

- Pour la modélisation cinématique, nous avons utilisé un ensemble d'équations de contraintes reliées les coordonnées articulaires aux coordonnées cartésiennes ; et par dérivation, nous avons pu obtenir le modèle des accélérations.

- Une modélisation dynamique réaliste est très importante pour avoir des simulations crédibles et pour l'amélioration de la précision des lois de commande. Pour se faire nous avons utilisé le principe du travail virtuel. On divise le mouvement du robot en petites étapes, calculant les travaux effectués par les forces et moments sur chaque élément. Ces travaux sont sommés pour obtenir le travail total et prédire le comportement et les forces internes du robot.

Dans la partie commande, nous nous sommes concentrés en premier lieu sur la commande PD. Deux versions de ce derniers ont été conçues : un PD classique d'ordre entier et un PD d'ordre fractionnaire nous a donné plus de flexibilité grâce à l'opérateur fractionnaire qui nous a permis non seulement améliorer les performances, mais de minimiser les vibrations dans le signal de commande et sortie.

Dans un deuxième temps, nous avons défini le régulateur TDC qui est un régulateur indépendant des équations dynamiques du robot "model free regulator". A la base de ce dernier nous avons développé deux regulateurs. Le premier étant TDC avec mode glissant surface fractionnaire. Le deuxième étant une version améliorée du premier où on introduit deux changements. Le premier au niveau de l'estimateur de la dynamique et perturbation du robot (TDE) ou on tient compte dans notre estimation du terme plus antérieure que

ceux du premier estimateur,. La deuxième modification est l'introduction d'un gain K adaptatif au niveau de la commande. Ces modifications ont permis de diminuer l'erreur en général et d'augmenter la robustesse du système. Ces résultats sont confirmés par les simulations.

Pour toutes les lois de commande, nous avons utilisé un algorithme génétique, qui est un algorithme métaheuristique afin d'optimiser les paramètres de chaque régulateur afin d'aboutir à des résultats optimaux.

Concernant la partie expérimentation qui consiste à implémenter les commandes synthétisées dans le chapitre trois et quatre au robot réel qui se trouvent au niveau du laboratoire de commande des processus au sein de l'école nationale polytechnique, malheureusement un des encodeur du robot a lâcher ce qui a rendu la commande en boucle fermée de son moteur impossible. Par conséquent, l'implémentation n'a pas pu avoir lieu.

Pour la continuité et l'amélioration de ce travail, nous proposons quelques perspectives à savoir :

- L'utilisation d'un gain adaptatif \hat{M} au lieu de \bar{M} dans la TDC.
- L'implémentation de l'ASMC TDC sur le robot Delta ISIR88.
- La génération de la trajectoire dynamique à l'aide d'une caméra.

Bibliographie

- [1] L. MATIGNON, *Introduction à la robotique*. Cour, Université de Caen, 2012.
- [2] *roboticbeast*, <https://www.roboticbeast.com/les-differents-types-de-robot-1/>, avril 2023.
- [3] T.-D.-H. HAMIED et A.-E. BAKKOUCHE, “Conception et réalisation d’un robot parallèle à trois degrés de liberté”, Mémoire de fin d’étude, Université Echahid Hamma Lakhdar d’El-Oued, 2018.
- [4] J.-P. MERLET et D. DANEY, “Dimensional Synthesis of Parallel Robots with a Guaranteed Given Accuracy over a Specific Workspace”, in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, p. 942-947.
- [5] M. BENRABAH et A. BENALI, “Modélisation et Commande d’Ordre Fractionnaire du Robot Parallèle Delta ISIR 88. Simulation et Expérimentation”, Mémoire de projet de fin d’études, Ecole Nationale Polytechnique, 2021.
- [6] C. E. BOUDJEDIR, “Contribution à la Commande par Apprentissage Itératif des Robots Parallèles de Type Delta Théorie, Simulation et Expérimentation”, Thèse de doctorat, Ecole Nationale Polytechnique, 2019.
- [7] O. KHELIFI et L. AZZOUG, “modélisation et commande d’un robot parallèle de type Delta”, Mémoire de projet de fin d’études : Automatique, Ecole Nationale Polytechnique, 2010.
- [8] A. BENSALD et C. MOKRAOUI, “Modélisation et Commande d’un Robot de Type Delta de la Simulation à l’Expérimentation.”, Projet de fin d’étude : Automatique, Ecole Nationale Polytechnique, 2019.
- [9] *filament2print*, https://filament2print.com/fr/blog/158_imprimantes-3d-Delta.html, mai 2023.
- [10] W. KHALIL et E. DOMBRE, *Bases de la modélisation et la commande des robots manipulateurs de type série*. Cour, Université Numérique Ingénierie Technologie – Robotique, 2012.
- [11] R. CLAVEL, “Conception d’un robot parallèle rapide à 4 degré de liberté”, [Doctoral Thesis], Swiss Federal Institute of Technology Lausanne (EPFL), 1991.
- [12] A. CODOUREY, “Contribution à la commande des robots rapides et précis - Application au robot Delta à entraînement direct”, Thèse de doctorat, Ecole Polytechnique Federale de Lausanne - EPFL, 1991.
- [13] A. OLSSON, “Modeling and control of a Delta-3 robot”, MASTER THESIS, Lund University, 2009.

- [14] A. CODOUREY, *Dynamic modelling and mass matrix evaluation of the Delta parallel robot for axes decoupling control*. the International Conference on Intelligent Robots et Systems, vol 3, pp. 1211-1218, 1996.
- [15] M. BOURI et R. CLAVEL, *Bases de la robotique*. Ecole Polytechnique Federale de Lausanne - EPFL, 2018.
- [16] S. BENNETT, *The past of pid controllers*. Annual Reviews in Control, vol. 25, pp. 43–53, 2001.
- [17] B. BOUROUBA, “Contribution à la Commande Adaptative Optimale d’Ordre Fractionnaire”, Thèse de doctorat, Université Des Freres Mentouri – Constantine-1, 2010.
- [18] *igm.univ-mlv*, http://igm.univ-mlv.fr/~dr/XPOSE2013/tleroux_genetic_algorithm/fonctionnement.html, mai 2023.
- [19] M. HAZERCHI et M. L. TAZIR, “Planification de la trajectoire d’un robot mobile autonome dans un environnement statique et/ou dynamique”, Projet de fin d’étude : Automatique, Ecole Nationale Polytechnique, 2012.
- [20] C. E. BOUDJEDIR, D. BOUKHETALA et M. BOURI, “An Enhanced Adaptive Time Delay Control-Based Integral Sliding Mode for Trajectory Tracking of Robot Manipulators”, in *IEEE Transactions on Control Systems Technology*, vol. 31, no. 3, pp. 1042– 1050, 2023.
- [21] M. JIN, S. H. KANG et P. H. CHANG, *Robust compliant motion control of robot with nonlinear friction using time-delay estimation*. IEEE Transactions on Industrial Electronics, 55(1) :258–269, 2008.
- [22] W. YAORYAO, J. SURONG, C.BAI et W.HONGTAO, “A new continuous fractional-order nonsingular terminal sliding mode control for cable-driven manipulators”, *Advances in Engineering Software*, 2018.
- [23] J.PARK, W.KWON et P.PARK, “An Improved Adaptive Sliding Mode Control based on Time Delay Control for Robot Manipulators”, *IEEE Transactions on Industrial Electronics*, 2022.
- [24] J. BAEK, S. CHO et S. HAN, *Practical time-delay control with adaptive 501 gains for trajectory tracking of robot manipulators*. IEEE Trans. Ind. 502 Electron., vol. 65, no. 7, pp. 5682–5692, 2018.
- [25] C. E. BOUDJEDIR, D. BOUKHETALA et M. BOURI, “Iterative learning control of multivariable uncertain nonlinear systems with nonrepetitive trajectory”, *Nonlinear Dyn*, vol. 95, no. 3, pp. 2197–2208, 2019.
- [26] C. E. BOUDJEDIR, D. BOUKHETALA et M. BOURI, “Nonlinear PD plus sliding mode control with Application to a parallel Delta robot”, *Journal of Electrical Engineering*, vol. 69, no. 05, pp 329-336, 2018.
- [27] C. E. BOUDJEDIR, D. BOUKHETALA et M. BOURI, “Iterative Learning Control for Trajectory Tracking of a Parallel Delta Robot”, *at - Automatisierungstechnik*, vol. 67, no. 2, pp. 145156, 2019.
- [28] Y. WANG, F. YAN, F. J. J. CHEN et B. CHEN, “A New Adaptive Time-Delay Control Scheme for Cable-Driven Manipulators”, *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3469-3481, 2019.

- [29] B. C. Y. WANG S. Jiang et H. WU, “A new continuous fractionalorder nonsingular terminal sliding mode control for cable-driven manipulators”, *Advances in Engineering Software*, vol. 119, pp. 21-29, 2018.
- [30] S. J. Y. WANG F. Yan et B. CHEN, “Time delay control of cable driven manipulators with adaptive fractional-order nonsingular terminal sliding mode”, *Advances in Engineering Software*, vol. 121, pp. 13-25, 2018.
- [31] C. E. BOUDJEDIR et D. BOUKHETALA, “An improved control strategy based sliding mode approach for high-order systems with mismatched disturbances”, *Journal of Electrical Engineering*, vol. 73, no. 1, pp. 11-18, 2022.