



MODÉLISATION ET COMMANDE D'UN ROBOT PARALLÈLE DE TYPE DELTA

Projet de fin d'études

présenté et soutenu publiquement le 29 Juin 2010

pour l'obtention du titre de

Ingénieur d'état de l'École Nationale Polytechnique d'Alger

Spécialité : Automatique - Électronique

par

KHELIFI Oualid (Automatique)

AZZOUG Larbi (Électronique)

Encadreurs : Pr D . BOUKHETALA – ENP

Dr M . BOURI – EPFL

Remerciements

En préambule à ce projet de fin d'études, nous souhaiterions adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué de près ou de loin à l'élaboration de ce travail.

Nous tenons à remercier spécialement nos promoteurs Mr Bouri et Mr Boukhetala pour nous avoir encadré, pour les précieux conseils qu'ils nous ont donné, pour leur aide et pour leur présence tout au long de ce projet.

Nous remercions également tout nos enseignants de l'école nationale polytechnique qui nous ont tant donné tout au long de notre cursus. Nos profonds remerciements à Mr Chekireb.

Enfin, nous souhaiterions remercier nos familles, plus particulièrement nos parents qui nous ont apporté un soutien sans failles et sans lesquels nous ne serions jamais arrivés à ce stade.

Dédicaces

*À mes parents et à mon
petit frère*

À ma chère fiancée

OUALID

Dédicaces

À mes chers parents

À toute ma famille

LARBI

Table des Matières

Table des Matières	i
Liste des figures	iv
Liste des tableaux	vi
Codes sources	vi
Introduction	1
I Les robots parallèles	2
1 Introduction	2
2 Caractéristiques des robots parallèles	3
3 Structures parallèles particulières	4
3.1 Mécanisme de Pollard	4
3.2 Plate-forme de Stewart	5
3.3 Robot parallèles plan	6
3.4 Le robot Delta	6
4 Conclusion	7
II Le Robot Delta	8
1 Introduction	8
2 Les caractéristiques du Delta	9
3 Types de robot Delta	10
3.1 Delta linéaire	10
3.2 Delta à actionnement rotatif	11
4 Constituants du robot Delta	12
5 Le Delta à actionnement Direct	13
5.1 L'actionnement direct - Direct drive	13
5.2 Cas du robot Delta	14
III Construction du Delta sous SimMechanics	15
1 Introduction	15
2 SimMechanics	16
3 Modèle SimMechanics du robot Delta	16
4 Conclusion	21
IV Modélisation du robot Delta	22
1 Introduction	22
2 Modèle géométrique	23
2.1 Modèle géométrique Direct	27
2.2 Modèle géométrique inverse	29
2.3 Validation du MGD et du MGI	32

2.4	Volume de travail	33
2.5	Singularités du Delta	35
3	Modèle cinématique	37
3.1	Modèle cinématique direct	39
3.2	Modèle cinématique inverse	40
3.3	Validation du MCD et du MCI	41
4	Modèle Dynamique	42
4.1	Développement des équations dynamiques	43
4.1.1	Rappel sur la méthode de Newton-Euler	44
4.1.2	Application au Delta	45
4.2	Modèle dynamique direct	49
4.3	Validation du MDD	52
4.4	Modèle dynamique inverse	53
4.5	Validation du MDI	53
V	Commande du robot Delta	55
1	Introduction	55
2	Objectifs	56
3	Génération de mouvement et de trajectoire	56
3.1	Lois de mouvement	57
3.2	Trajectoires	60
4	Commande en boucle ouverte avec MDI à priori	63
5	Synthèse de lois de commande	66
5.1	Architectures de commandes	66
5.1.1	Commande centralisées	66
5.1.2	Commande décentralisées	67
5.2	Conditions d'instrumentation et d'acquisition	68
5.3	Commande PID	69
5.4	Double boucle PID vitesse - position	73
5.5	Introduction du modèle dynamique	74
6	Tests de performances	75
6.1	Cadence de prise-dépose	76
6.2	Trajectoire de prise dépose	80
6.3	Robustesse par rapport à la charge transportée	84
6.4	Robustesse par rapport aux perturbations sur le positionnement	86
6.5	Robustesse des performances dans le volume de travail	86
7	Commande par apprentissage itératif (ILC)	90
7.1	Présentation de la commande	90
7.1.1	Topologies de la commande par apprentissage itératif	91
7.1.2	Conditions de fonctionnement de la commande ILC	91
7.2	Application de la commande sur le Delta	92
7.3	Commentaires et conclusion	97
	Conclusion	98

Annexes	100
Annexe A	100
Annexe B	102
Bibliographie	104

Liste des figures

I.1	Fermeture d'une chaîne cinématique	2
I.2	Effet de la fermeture d'une chaîne cinématique sur le volume de travail	3
I.3	Mécanisme de Pollard	4
I.4	Plate-forme de Stewart	5
I.5	Simulateur de Vol : application de la Plate-forme de Stewart	5
I.6	Robot parallèle plan à articulations prismatiques	6
I.7	Robot Delta de l'EPFL	7
II.1	Le FlexPicker de ABB	9
II.2	Le SurgiScope	9
II.3	Chaîne d'assemblage à base de micro Delta	9
II.4	Les barres parallèles du robot Delta	10
II.5	Le Delta linéaire	11
II.6	Le Delta à actionnement rotatif	11
II.7	Schéma descriptif des constituants du robot Delta	12
III.1	Représentation schématique du robot Delta	15
III.2	Un bras du Delta sous SimMechanics	17
III.3	Un bras du Delta correspondant au schéma SimMechanics	17
III.4	Repère SimMechanics/Repère initial	18
III.5	Bloc bras sous Simulink	19
III.6	Représentation du Delta sous SimMechanics	20
III.7	Modèle du Delta sous Simulink	20
III.8	Modèle 3-D du Delta générée par SimMechanics pour $\alpha^0 = \pi/4$	21
IV.1	Modèle géométrique	23
IV.2	Un bras du Delta	24
IV.3	Vue du dessus de la base du Delta	24
IV.4	Schéma d'un bras du delta	25
IV.5	Schéma simplifié d'un bras du Delta	25
IV.6	Sphère de centre P et de rayon ℓ_2 passant par C_1	26
IV.7	Modèle géométrique direct du robot Delta	27
IV.8	Modèle géométrique inverse du robot Delta	29
IV.9	Modèle SimMechanics avec capture de la position de la nacelle	32
IV.10	Validation du MGI et du MGD	32
IV.11	Volume de travail considéré du Delta	34
IV.12	Les singularités du robot Delta	36
IV.13	Modèle cinématique du robot Delta	37
IV.14	Validation du MCI et du MCD	42
IV.15	Modèle dynamique	43
IV.16	Isolation de la nacelle du Delta	45
IV.17	Force de liaison \vec{F}_i	46
IV.18	Isolation de la barre représentant les barres parallèles	47
IV.19	Isolation d'un des trois bras du Delta	48
IV.20	MDD du delta	50
IV.21	Schéma de simulation du MDD	51

IV.22	Simulation du modèle analytique du Delta sous simulink	51
IV.23	Détails du bloc Accélération du MDD précédent	51
IV.24	Schéma Simulink utilisé pour la validation du MDD	52
IV.25	Résultats de validation du MDD	52
IV.26	Schéma Simulink utilisé pour la validation du MDD	53
IV.27	Implémentation du MDI du delta sous Simulink	53
IV.28	Schéma Simulink utilisé pour la validation du MDI	54
IV.29	Résultats Validation du MDI	54
V.1	Génération de mouvement	57
V.2	Loi de mouvement Bang-bang	58
V.3	Loi de mouvement en Trapèze	59
V.4	Projection de la coordonnée curviligne $s(t)$ donnant $x(t), y(t), z(t)$	60
V.5	Trajectoire ellipsoïde	61
V.6	Trajectoire en segments de droites	61
V.7	Clothoïde	62
V.8	Trajectoire de prise-dépose utilisant les clothoïdes	62
V.9	Commande en boucle ouverte du robot Delta	63
V.10	Simulation commande en boucle ouverte du robot Delta	64
V.11	Résultats de la commande en boucle ouvert	64
V.12	Résultats avec la commande en boucle ouverte avec charge	65
V.13	Commande centralisé du robot Delta	67
V.14	Commande décentralisé du robot Delta	67
V.15	Commande du Delta avec des régulateurs PID décentralisés	69
V.16	Régulateur PID implémenté	69
V.17	Résultat de positionnement par Step - Régulateur PID	70
V.18	Consigne en loi de mouvement trapézoïdale	71
V.19	Poursuite de trajectoire opérationnelle - PID	72
V.20	Double boucle PID position-vitesse	73
V.21	Poursuite de trajectoire opérationnelle - Double boucle	73
V.22	Double boucle PID position-vitesse avec à priori de couple	74
V.23	Poursuite de trajectoire opérationnelle - Double boucle + MDI	75
V.24	Implantation réelle du régulateur	75
V.25	Trajectoire de prise-dépose en segments de droites	76
V.26	Prise dépose à 1 Hz avec les segments de droites	77
V.27	Prise dépose à 2 Hz avec les segments de droites	78
V.28	Prise dépose à 4 Hz avec les segments de droites	79
V.29	Trajectoire de prise-dépose en clothoïdes	80
V.30	Prise dépose à 1 Hz avec la clothoïde	81
V.31	Prise dépose à 2 Hz avec la clothoïde	82
V.32	Prise dépose à 4 Hz avec la trajectoire en clothoïdes	83
V.33	Dimensionnement des moteurs selon les couples souhaités	85
V.34	Application d'une perturbation sur le positionnement de la nacelle	86
V.35	Points de test pour le réglage des paramètres du régulateur	87
V.36	Symétrie hexanaire dans le comportement du robot Delta	88
V.37	Évolution de l'erreur moyenne dans l'espace de travail	89
V.38	Principe de fonctionnement de la commande itérative	91

V.39	Schéma de l'implémentation de la commande ILC sur le Delta	93
V.40	Erreur en fonction du temps (15 itérations) - trajectoire selon x	94
V.41	Test de la commande ILC (15 itérations) - trajectoire prise-dépose à 2Hz	95
V.42	Test de la commande ILC (15 itérations) - trajectoire prise-dépose à 4Hz	96
VI.1	Tests pour différentes charges à une cadence de 2 Hz	102
VI.2	Tests pour différentes charges à une cadence de 4 Hz	103

Liste des tableaux

III.1	Constantes de simulation du Delta	17
V.1	Caractéristiques de l'instrumentation et de l'acquisition	68
VI.1	Évolution des erreurs suivant les itérations (ILC) - cadence de 2 Hz	100
VI.2	Évolution des erreurs suivant les itérations (ILC) - cadence de 4 Hz	100
VI.3	Évolution des erreurs suivant les points de tests (volume de travail)	101

Codes sources

III.1	Initialisation simulation SimMechanics	18
-------	--	----

Introduction

La robotique en général et les bras manipulateurs en particuliers ont connue un grand essor ces dernières années, notamment dans l'industrie ou leur utilisation s'est largement démocratisée. De plus en plus de tâches sont dévolues aux robots qui sont de plus en plus complexes.

Les robots manipulateurs sont utilisés dans de multiples fonctions qui vont du soudage à la manutention en passant par l'usinage. Il est évident que chaque tâche requiert un type particulier de manipulateur suivant les contraintes à respecter.

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.

Notre travail a débuté par un état de l'art sur la robotique parallèle, s'en est suivi une présentation du robot Delta et de ses spécificités (chapitre I et II).

L'étude s'est ensuite poursuivie par la mise en place d'un modèle de simulation du Delta sous SimeMechanics MATLAB (chapitre III). Nous nous sommes par la suite intéressés plus en détail au robot Delta, différents aspects de sa modélisation ont été abordés (modèle géométrique, cinématique et dynamique, volume de travail, singularités. . .) (chapitre IV).

Cela nous a finalement amené à la commande du robot (chapitre V). Les tests en boucle ouverte et le développement de commandes adaptées au Delta permettra de cerner toutes les particularités de la structure étudiée.

Les robots parallèles

I

1	Introduction	2
2	Caractéristiques des robots parallèles	3
3	Structures parallèles particulières	4
3.1	Mécanisme de Pollard	4
3.2	Plate-forme de Stewart	5
3.3	Robot parallèles plan	6
3.4	Le robot Delta	6
4	Conclusion	7

1 Introduction

Les architectures parallèles sont apparues il y a environ 50 ans avec les premiers simulateurs de vol. Elles sont utilisées depuis dans d'autres applications nécessitant la manipulation d'objets lourds avec de grandes accélérations ou l'assemblage dans les applications qui requièrent une grande précision et la maîtrise des efforts de contact ou encore la prise-dépose d'objets à grande cadence pour les applications d'emballage. Le concept a été repris récemment dans le domaine de la machine-outil pour des applications d'usinage à grande vitesse et à haute précision.

Un robot parallèle est un mécanisme en chaîne cinématique fermée bouclé (figure I.1), constitué d'un organe terminal mobile et d'une base fixe, reliés 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 fixés sur la base.

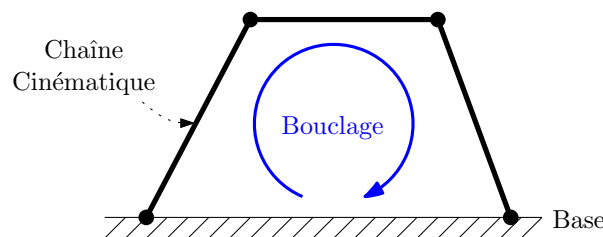


FIGURE I.1: Fermeture d'une chaîne cinématique

2 Caractéristiques des robots parallèles

Outre la fermeture de la chaîne cinématique, les robots parallèles présentent d'autres caractéristiques particulières. Plusieurs aspects qui différencient les robots parallèles et les robots séries classiques peuvent être cités. Ces aspects permettent de mettre en évidence les caractéristiques des structures parallèles, leurs avantages et leurs inconvénients.

- **Rigidité** : un robot parallèle est un robot à chaîne cinématique bouclé. Cela lui confère une plus grande rigidité de positionnement, permettant ainsi d'améliorer la précision par rapport aux robots séries.
- **Volume de travail** : C'est le volume qui peut être atteint par l'organe terminal. Dans le cas d'un robot parallèle ce volume ou capacité d'accessibilité est restreint par rapport aux robots séries. La fermeture d'une boucle comme c'est le cas pour une structure parallèle engendre plus de contraintes aux mouvements de l'organe terminal. La figure I.2 illustre cela.

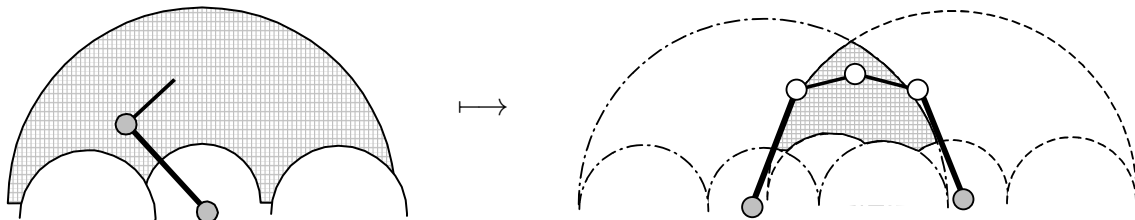


FIGURE I.2: Effet de la fermeture d'une chaîne cinématique sur le volume de travail

- **Rapport masse transportable / masse du robot** : Dans l'architecture série classique, l'organe terminal se trouve à l'extrémité de la chaîne articulée. Chaque actionneur doit avoir la puissance nécessaire pour mettre en mouvement tous les corps et actionneurs en aval. Dans les cas des robots parallèles, les actionneurs se situent le plus souvent au voisinage des points d'articulation de la base, si ce n'est sur la base elle-même, ce qui donne un rapport masse transportable / masse du robot beaucoup plus élevé.
- **Précision et répétabilité** : Pour les robots série une erreur de positionnement sur une ou plusieurs articulations a une répercussion sur la position finale de l'organe terminal. Ce problème d'accumulation des erreurs ne se présente pas pour les robots parallèles puisque leur morphologie leur assure une rigidité remarquable et une moindre dépendance des articulations en termes d'erreurs de positionnement
- **Comportement dynamique** : Compte tenu de leur structure, les robots série ont des performances dynamiques limitées. Avec les robots parallèles, le fait de diminuer l'inertie du robot contribue à l'augmentation de la rapidité et de la cadence de travail.
- **Technologie de construction** : L'un des avantages des manipulateurs parallèles est l'utilisation fréquente de plusieurs chaînes cinématiques identiques sur un robot unique (rentabilité, pièces de rechanges, ...). Un autre avantage de certains manipulateurs parallèles (avec actionneurs fixe) est la plus grande liberté dans le choix des actionneurs : rotoïde, prismatique, à entraînement direct ou indirect, à moteur électrique ou non électrique. De plus cela a pour effet de faciliter l'entretien et la

réparation des moteurs. Le branchement de l'instrumentation (capteurs) et de l'alimentation est aussi plus aisé et enfin, la détérioration des fils de branchement est moindre pour les robots parallèles car on n'est pas contraint de les faire passer à travers toute la chaîne cinématique comme c'est le cas pour les robots séries.

Il existe une dualité entre robot séries et robots parallèles. En termes de modélisation, contrairement aux robots séries, les modèles inverses des robots parallèles sont en général plus faciles à établir que les modèles directs. Comme pour les robots séries, les robots parallèles ont des positions singulières particulières. Pour le cas des robots séries une singularité résulte d'une perte en degrés de libertés alors que pour les robots parallèles, elle résulte d'un gain en degrés de libertés (perte de rigidité).

Enfin il faut noter que la modélisation des robots parallèles est particulière à chaque structure contrairement aux robots séries, ou des méthodes systématiques ont été développées.

3 Structures parallèles particulières

3.1 Mécanisme de Pollard

C'est l'une des premières structures parallèles connues. Cette structure a été décrite par Pollard en 1942 et était destinée à la peinture automobile. La structure est constituée de trois bras orientés avec trois actionneurs, cela à pour effet de positionner l'organe terminal à travers trois segments articulés passifs (voir figure I.3). Cette structure est restée au stade de mécanisme car la technologie de commande disponible ne permettait pas d'en faire un véritable robot précis et fiable.

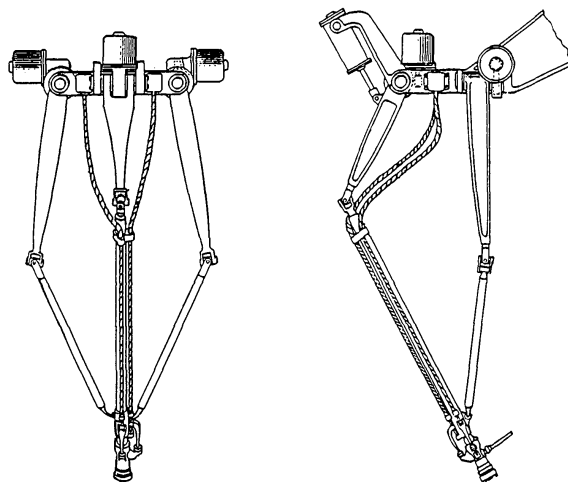


FIGURE I.3: Mécanisme de Pollard

3.2 Plate-forme de Stewart

Il s'agit d'un manipulateur à six degrés de liberté, dont la plate-forme est déplacée par l'élongation de six actionneurs linéaires. La structure possède six segments dont les sommets sont reliés par des rotules à la plate-forme mobile. L'autre extrémité du segment est connecté à la base en utilisant un cardan. L'actionneur linéaire permet de changer la longueur totale du segment, il en résulte la possibilité de commander tout les positionnement et les orientations possibles de la plate-forme. Son application la plus connue est le simulateur de vol. La raison de cette utilisation est la masse élevée de la cabine à laquelle on doit faire subir des mouvements selon les six degrés de liberté (3 translations et 3 rotations).

Cette structure peut être aussi utilisée afin de positionner des pièces destinées à être usinés.

La figure I.4 représente la configuration de la plate-forme de Stewart, la figure I.5 elle représente l'application usuelle de la plate-forme de Stewart : le simulateur de vol.

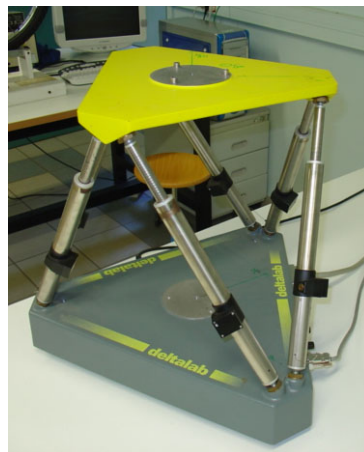
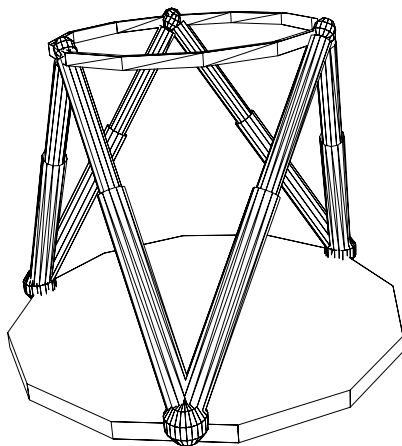


FIGURE I.4: *Plate-forme de Stewart*



FIGURE I.5: *Simulateur de Vol : application de la Plate-forme de Stewart*

3.3 Robot parallèles plan

Il existe plusieurs configurations possibles pour une telle structure. De tels robots permettent un positionnement dans le plan et une rotation. Ils sont donc à 3 degrés de libertés. Les bras de la structure peuvent être mis en mouvement soit par des actionneurs prismatiques ou des actionneurs rotoïde + avant bras.

La figure I.6 montre un exemple d'une telle structure.

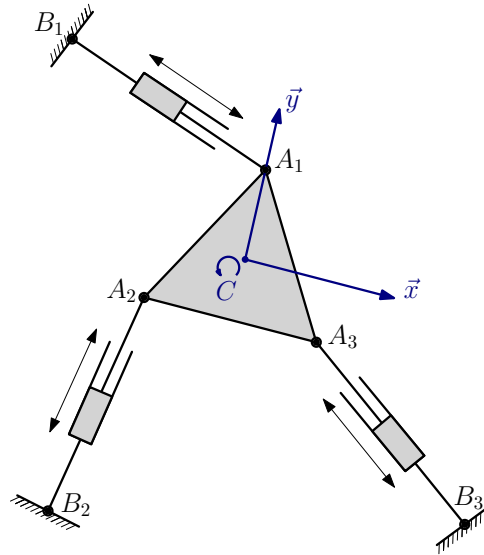


FIGURE I.6: Robot parallèle plan à articulations prismatiques

Cette structure est dite de type $R\underline{P}R$ (jonctions $R - P - R$). La jonction soulignée est celle qui est actionnée.

Ce genre de structure peut être utilisé pour le positionnement précis de pièces afin que des tâches de perçage y soient effectuées par exemple.

3.4 Le robot Delta

C'est un robot parallèle à trois degrés de libertés en translation. Il possède trois bras de manipulation formé de 3 parallélogrammes.

Il peut être actionné de différentes manières, allant des vérins linéaires aux actionneurs de rotation comme celui montré dans la figure I.7.

Il a été mis au point en 1985, par Raymond Clavel, doctorant à l'École polytechnique fédérale de Lausanne (EPFL) au Laboratoire de système robotiques (LSRO).



FIGURE I.7: *Robot Delta de l'EPFL*

Cette structure est certainement le plus grand succès industriel des robots parallèles. Le robot Delta sera l'objet de notre étude dans le cadre de ce travail.

4 Conclusion

La caractéristique fondamentale qui différencie les robots parallèles des robots séries usuelles est la fermeture de la chaîne cinématique, ces robots sont constitués d'une ou plusieurs boucles.

Toutes les caractéristiques générales des robots parallèles citées précédemment s'appliquent bien évidemment au cas du robot Delta. Une étude plus détaillée des caractéristiques du robot Delta sera faite dans le chapitre suivant (II).

Chapitre

II

Le Robot Delta

1	Introduction	8
2	Les caractéristiques du Delta	9
3	Types de robot Delta	10
3.1	Delta linéaire	10
3.2	Delta à actionnement rotatif	11
4	Constituants du robot Delta	12
5	Le Delta à actionnement Direct	13
5.1	L'actionnement direct - Direct drive	13
5.2	Cas du robot Delta	14

1 Introduction

Certainement, l'architecture parallèle la plus connue, outre la plate-forme de Stewart, est celle du robot Delta. Au début des années 1980 le Pr. Reymond Clavel de l'École Polytechnique Fédérale de Lausanne invente et fait breveter l'idée d'utiliser trois parallélogrammes pour construire un robot dont la plate-forme mobile ne se déplace que selon trois translations.

Une licence pour la construction de tels robots a été initialement octroyée à Demaurex, mais suite à des restructurations commerciales, d'autres compagnies ont eu le droit de construire des Deltas. Plus de 4000 robots Delta ont été vendus depuis.

Aujourd'hui, le brevet principal du robot Delta est expiré et une panoplie de compagnies fabriquent des Deltas. ABB continue de produire son FlexPicker (figure II.1), alors que Bosch (le nouveau propriétaire de Demaurex) fabrique au moins deux autres modèles de robots delta. La principale utilisation de ces robots là est l'emballage et la manutention dans le domaine agroalimentaire.

De plus, le Centre Suisse d'Electronique et de Microtechnique offre deux modèles de micro-robot Delta qui sont utilisés généralement pour l'industrie horlogère, l'assemblage de micro-systèmes (capteurs, moteurs) et de composants montés en surface (CMS) (figure II.3).

L'entreprise française ISIS continue d'offrir le SurgiScope, un système porte-outils plafonnier pour les chirurgiens (figure II.2). La compagnie américaine Delta Tau vient de lancer un robot Delta pour l'assemblage. Le principe du robot Delta est également

utilisé par d'autres compagnies qui fabriquent des machines outils, notamment, Renault, Hitachi, Reichenbacher, Fatronik.



FIGURE II.1: *Le FlexPicker de ABB*



FIGURE II.2: *Le SurgiScope*



FIGURE II.3: *Chaîne d'assemblage à base de micro Delta*

2 Les caractéristiques du Delta

Le robot Delta possède principalement trois caractéristique le différenciant des robots industriels conventionnels (robots série) :

- Comme on l'a déjà vu dans le chapitre I, le Delta est un robot parallèle, ceci veut dire que la liaison entre sa base d'amarrage et son organe terminal se fait par le biais de plusieurs chaînes cinématiques (trois chaînes cinématiques identiques pour le Delta conventionnel), diminuant ainsi l'espace de travail et augmentant la rigidité et la précision du robot.
- La nacelle du robot (structure portant l'organe terminal) possède la propriété de garder constamment une configuration horizontale et ce, d'une manière totalement

- passive, c'est-à-dire sans aucun asservissement ou contrôle de l'orientation de la nacelle (propriété obtenue par construction mécanique du robot).
- Les moteurs chargés d'actionner le robot sont fixés au niveau de la base d'amarrage du robot, ce qui a pour effet de diminuer considérablement les masses mobiles du robot et donc leurs inerties, permettant ainsi de grandes vitesses et accélérations dans l'utilisation de ce robot.
 - Les barres parallèles qui constituent un parallélogramme. C'est cette particularité permet à la nacelle du robot Delta d'avoir des mouvements translationnels seulement.

3 Types de robot Delta

Le robot Delta est un robot à trois degrés de libertés possédant trois chaînes cinématiques indépendantes identiques. La caractéristique essentielle qui est identiques pour tous les robots Delta est l'utilisation de barres parallèles. Ces barres sont fixés entre eux à une petite barre de maintien par des liaisons rotules (voir figure II.4). C'est cette particularité qui permet à la nacelle de rester parallèle à la base et d'avoir ainsi des mouvements translationnels seulement.

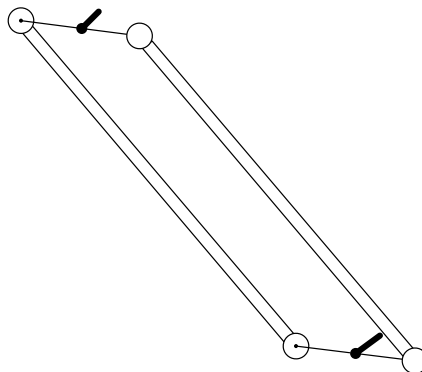


FIGURE II.4: Les barres parallèles du robot Delta

Ces chaînes peuvent être actionnées de différentes façons. Le but de l'actionnement est de transmettre une force de traction/poussée sur les trois couples de barres parallèles. Cela peut être obtenu de différentes façons.

Il existe deux types particuliers de robot Delta selon l'actionneur utilisé, le robot Delta linéaire dont les trois actionneurs des trois chaînes cinématiques sont de type prismatiques et le robot Delta à actionnement rotatif dont les actionneurs sont rotoïdes.

3.1 Delta linéaire

Ce type de robot Delta utilise trois actionneurs prismatiques, ces actionneurs sont fixés directement aux barres parallèles car ce type d'actionneur produit directement une force et peut donc transmettre une force de traction/poussée aux barres parallèles.

Ces actionneurs prismatiques peuvent être de type pneumatique ou hydraulique (vérins), ou alors de type moteurs électrique couplé à une vis sans fin, une crémaillère ou une courroie. La figure II.5 montre une telle structure.

Ce type de robot Delta est cependant peu utilisé à cause des actionneurs prismatiques qui sont encombrant pour le cas des systèmes moteurs/crémaillères, et contraignant pour le cas des vérins.



FIGURE II.5: *Le Delta linéaire*

3.2 Delta à actionnement rotatif

Ce type de robot Delta utilise trois actionneurs rotoïdes, ces actionneurs ne peuvent pas être fixés directement aux barres parallèles comme c'est le cas dans le Delta linéaire, car ces actionneurs produisent un couple et donc pour transmettre une force de traction/poussée aux barres parallèles on doit prévoir un bras couplé à l'axe de l'actionneur.

Cet avant bras est fixé à l'arbre du moteur d'un côté et à la barre de fixation des barres parallèles de l'autre (voir indication sur la figure II.6). Cela permet d'avoir une configuration équivalente au cas du robot Delta linéaire.



FIGURE II.6: *Le Delta à actionnement rotatif*

4 Constituants du robot Delta

Dans le présent projet, notre travail s'est axé sur le robot Delta à actionnement rotatif, ce type de robot représente en effet la plus grosse part des Delta utilisés en environnement industriel.

La figure II.7 montre le schéma descriptif des constituants du robot Delta sur lequel nous allons travailler.

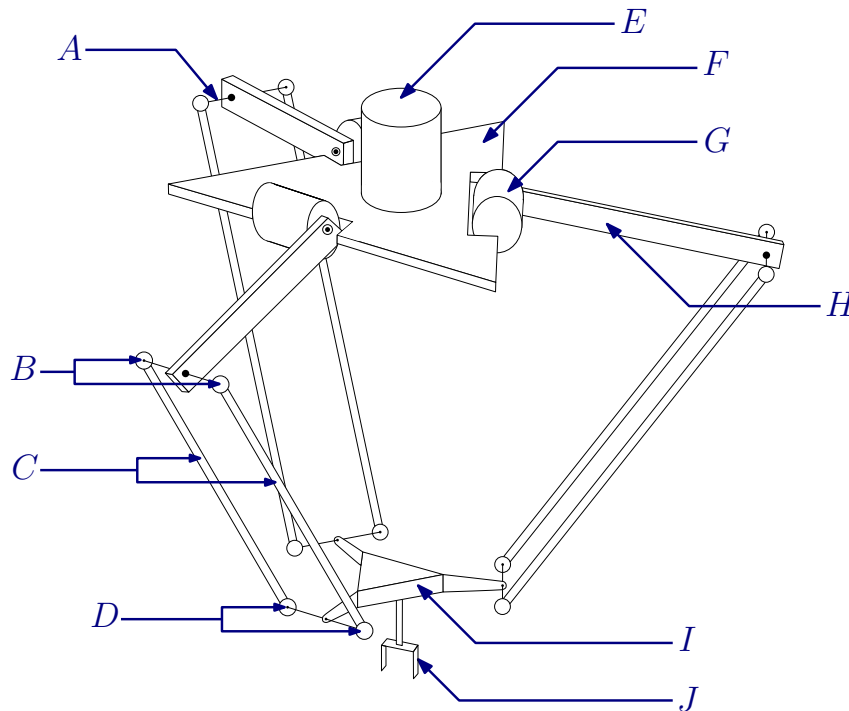


FIGURE II.7: Schéma descriptif des constituants du robot Delta

Avec :

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

B : Rotules supérieures.

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

D : Rotules inférieures.

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

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

- G** : Actionneur d'une chaîne cinématique (articulation active). Cet actionneur constitue le seul élément du robot accessible directement par une commande ou pour des mesures (par le biais de sa instrumentation). Celui-ci est fixé sur la base et actionne un seul bras (H).
- H** : Bras d'une chaîne cinématique du Delta. Transmet le mouvement du moteur (G) vers la nacelle (I) à travers les barres parallèles (A).
- I** : Nacelle du robot Delta. La nacelle est portée par trois chaînes cinématiques identiques. La nacelle constitue un support pour l'organe terminal (J).
- J** : Organe terminal du robot Delta (Pince, ...).

A noter que des barres de maintiens sont ajoutées afin d'éliminer la rotation de chaque barre parallèle autour d'elle-même. Cette rotation est due au fait que la structure théorique du robot Delta prévoit l'utilisation de cardans au lieu de rotules.

Pour des raisons de simplification de la construction mécanique on préfère utiliser des liaisons rotules et rajouter ces barres de maintiens (à ressorts).

5 Le Delta à actionnement Direct

5.1 L'actionnement direct - Direct drive

Dans les moteurs à entraînement direct le mouvement du rotor est transmis directement au bras à actionner. Cela permet de simplifier grandement la mécanique et la complexité des articulations actionnées par l'élimination des systèmes de transmission nécessaires aux traditionnels actionneurs. En outre, ils permettent un contrôle précis des couples en éliminant les jeux mécaniques et les frictions résultantes des transmissions à engrenages. Ce genre d'actionnement est de plus en plus utilisé, spécialement dans le domaine de la robotique.

Ces moteurs n'utilisant pas de réducteurs, ils doivent être capables de fournir un couple important. Cela requiert l'utilisation de bobinages volumineux. Un moteur à actionnement direct est donc plus volumineux et plus encombrant.

Ces moteurs possèdent aussi une inductance importante due à la taille du bobinage. Afin que la commande du moteur soit la plus efficace possible, l'amplificateur doit être capable de faire varier le courant (commande en couple) de façon adéquate malgré la forte inductance. L'utilisation de tensions d'alimentations importantes est alors nécessaire.

De plus, l'élimination du réducteur impose l'utilisation de capteurs de position très performants. La résolution des capteurs n'étant plus multipliée par le rapport de réduction, elle doit être suffisante pour conserver la même précision en sortie de l'arbre moteur. Les encodeurs optiques standards ne peuvent plus être utilisés et d'autres solutions doivent être trouvées. On utilise alors des systèmes de capture plus complexes à base d'interféro-

mètres laser ou de résolveurs intégrés¹. On peut atteindre alors des valeurs de 1 000 000 d'incrémentes par tour.

Concernant l'utilisation de tels moteurs sur des bras manipulateurs les considérations suivantes sont posées :

- Dans une structure de robot sériel conventionnel, les moteurs sont situés en général tout le long de la chaîne cinématique. Ceci impose donc une contrainte sur le poids des moteurs. Afin d'utiliser des moteurs à entraînement direct, une solution doit être trouvée en adoptant des structures mécaniques différentes (structure parallèle par exemple).
- L'utilisation de réducteurs diminue les effets de non linéarités de la structure vue de l'arbre du moteur. Les couplages et les effets dynamiques non linéaires sont ainsi atténués. La commande peut dans ce cas là ne pas tenir compte de ces effets complexes et simplifier le régulateur implémenté.

L'utilisation de moteur à entraînement direct amène des avantages indéniables mais requiert la prise en compte de certaines contraintes. Afin d'explicitier cela, l'exemple du robot Delta à actionnement direct sera abordé dans la prochaine section.

5.2 Cas du robot Delta

Nous verrons plus tard que pour le cas du robot Delta, il existe un fort couplage entre les trois chaînes cinématiques. Pour avoir de bons résultats en termes de rapidité et de précision, une modélisation dynamique efficace et précise doit être faite.

Le problème avec l'utilisation de réducteurs est la difficulté de modéliser certains phénomènes comme les frottements, les jeux mécaniques et le graissage des engrenages. Cela a pour effet de diminuer la connaissance du modèle du robot. De plus, la présence d'engrenage (jeux, graissage) augmente la compliance c'est à dire la flexibilité de la structure, effet indésirable pour un robot parallèle dont la principale caractéristique est la rigidité et la précision.

L'utilisation d'un moteur type à actionnement direct peut avoir l'avantage d'éliminer les problèmes cités précédemment. De plus, les moteurs étant situés sur la base fixe, leurs masses et leurs encombrements ne présentent aucun problème.

Dans le cas de l'utilisation de réducteurs, nous avons vu que les effets de couplage et des non linéarités étaient atténués. Ce n'est plus le cas avec l'actionnement direct, la commande doit donc prendre en compte ces effets en intégrant le modèle dynamique du robot dans sa structure.

La figure I.7 montre un robot Delta à actionnement direct.

1. Un résolveur est un transducteur électromagnétique qui transforme la valeur angulaire d'un rotor en une valeur électrique, basé sur le principe d'un transformateur dont le couplage varie avec l'angle. L'étude du signal de sortie du transformateur donne l'angle exact.

Construction du Delta sous SimMechanics

III

1	Introduction	15
2	SimMechanics	16
3	Modèle SimMechanics du robot Delta	16
4	Conclusion	21

1 Introduction

Dans ce chapitre nous allons essayer de construire un modèle de simulations dynamique du robot Delta. Le modèle de simulation a été réalisé par construction mécanique (construction virtuelle) de la structure du robot sous SimMechanics.

La figure III.1 donne la représentation schématique de la structure sur laquelle nous allons travailler. À noter que c'est le cas du **Delta à actionnement direct** qui sera traité.

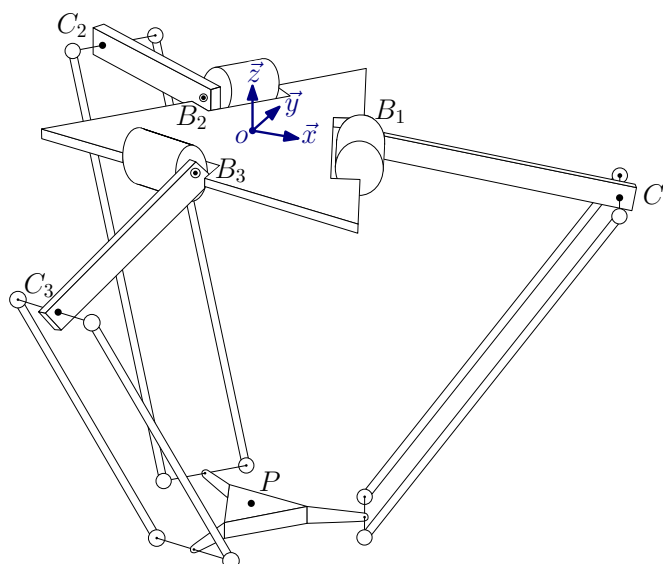


FIGURE III.1: Représentation schématique du robot Delta

2 SimMechanics

SimMechanics est un Toolbox de MATLAB, celui-ci offre la possibilité de créer le modèle physique d'une structure mécanique possédant plusieurs éléments assemblés par des liaisons connues (liaison pivot, rotule, encastrement . . .), ce modèle est ensuite totalement utilisable comme un bloc SIMULINK. SimMechanics fournit un ensemble complet de bibliothèques de blocs permettant la modélisation d'une structure mécanique. Les pièces figurant dans ces bibliothèques sont utilisées sous forme d'un bloc SIMULINK, on peut ainsi profiter de la puissance de SIMULINK afin de simuler le système seul, avec une commande ou encore en interaction avec d'autres types de systèmes (électriques, hydrauliques, . . .).

SimMechanics est basé sur le logiciel Simscape, cette plateforme produite pour la modélisation de systèmes physiques sous SIMULINK permet une modélisation et une conception de systèmes en fonction de principes physiques de base.

Contrairement à d'autres blocs SIMULINK, qui représentent des opérations mathématiques opérés sur des signaux, les blocs Simscape représentent directement des éléments ou des relations physiques.

Le système de conception SimMechanics est ainsi très particulier, la construction d'un système mécanique se base sur la connaissance physique de sa structure et non sur la connaissance des équations analytiques qui le décrivent, ce qui facilite grandement la modélisation du système. Avec un tel outil on peut modéliser un système de façon très proche de la réalité car nous n'avons pas à faire d'hypothèses simplificatrices, hypothèses nécessaires dans le cas du développement du modèle analytique du système.

Afin de construire le modèle sous SimMechanics, on doit spécifier les éléments de base constituant le système (corps solides ou liaisons mécaniques). Leurs propriétés (masse, inertie, géométrie), leurs mouvements possibles (types d'articulations, . . .), les contraintes cinématiques (actionneurs, . . .) et le système de coordonnées auquel est référencé le solide. On doit aussi spécifier la position initiale de la structure.

SimMechanics intègre aussi des outils de visualisation qui permettent d'afficher et d'animer le comportement dynamique d'un modèle 3-D de la structure, avant et pendant la simulation.

3 Modèle SimMechanics du robot Delta

Il s'agit simplement de transcrire la structure du robot Delta en schéma SimMechanics (à base d'éléments solides et de liaisons). Nous ajouterons aussi des éléments capteurs et actionneurs aux niveaux des articulations motorisées du Robot.

Dans ce modèle nous avons choisi de négliger les effets de flexion des barres et les frottements présents dans la structure.

Les constantes du système¹ avec lesquelles nous allons travailler sont données sur le tableau III.1.

Constante	Désignation	Valeur
ℓ_1	Longueur bras	260 mm
ℓ_2	Longueur barre parallèle	520 mm
$r = r_1 - r_2$	Différence de Rayon base/nacelle	$0,634 \times \ell_1$
r_1	Rayon de la base	$r + r_2$
m_n	Masse de la nacelle	160 g
m_b	Masse de la barre avant bras	80 g
m_p	Masse d'une barre parallèle	44 g
m_r	Masse des jonctions bras/barres parallèles	20 g

TABLE III.1: Constantes de simulation du Delta

En premier lieu, nous allons modéliser chaque bras du Delta indépendamment. La figure III.2 donne le schéma SimMechanics commun aux 3 bras du robot et la figure III.3 donne la structure mécanique réelle du bras correspondante.

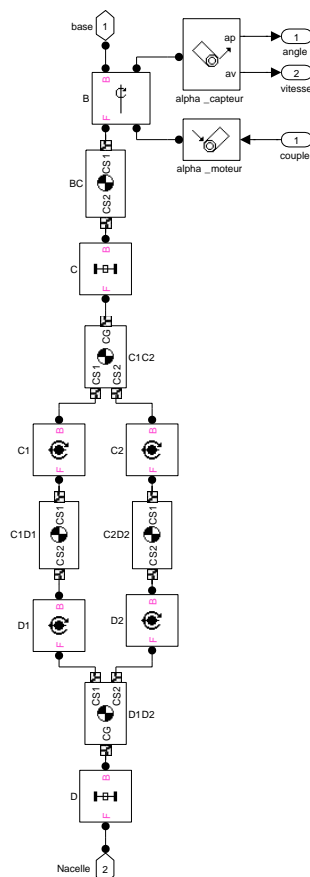


FIGURE III.2: Un bras du Delta sous SimMechanics

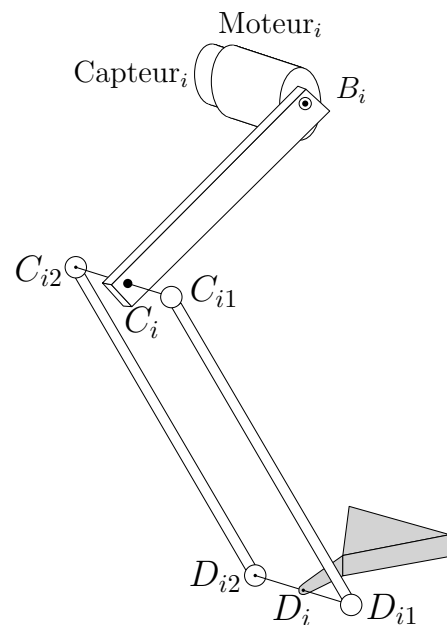


FIGURE III.3: Un bras du Delta correspondant au schéma SimMechanics

1. Données transmises par l'EPFL.

Le script MATLAB donné dans le listing III.1 réalise la définition des constantes mécaniques du système et met en place les positions initiales des éléments de la structure suivant les angles α_i (angles au niveau des moteurs) initiaux souhaités. A noter que nous avons choisis que les 3 angles initiaux soit égaux $\alpha_1 = \alpha_2 = \alpha_3 = \alpha^0$, on définira donc simplement α^0 de façon cohérente avec le volume de travail.

Le script réalise aussi la rotation de R_i à R_0 (notée T dans le script) qui permet de placer les modèles des 3 bras du Delta (même bras espacé d'un angle de $3\pi/2$).

Remarque Pour la compréhension du script il est important de noter que le placement du repère (x, y, z) est particulier sous SimMechanics et se présente comme ceci :

- x vers la droite
- y vers le haut (la gravité dans la direction de $-y$)
- z sortant en trois dimensions

La figure III.4 montre le placement des repère sous SimMechanics par rapport au placement initial pris lors des développements précédents.

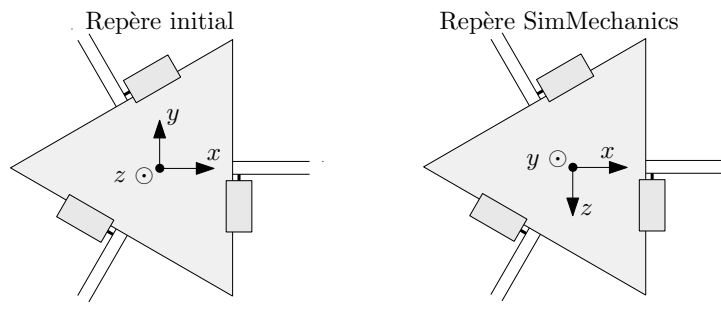


FIGURE III.4: Repère SimMechanics/Repère initial

```

% Définition des constantes géométrique
% initiale de la simulation SimMechanics
% du robot Delta

clear;
% constantes
l1 = 0.260;
l2 = 0.540;
r = 0.634*l1;
r2 = 0.033;
r1=r+r2;
a=0.05; % distance entre les deux barres
R=0.005; % rayon des barres

% changement repère entres barre
phi = [0;2*pi/3;4*pi/3];

% position initiale alpha_1=alpha_2=alpha_3
alpha=0;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% rev_B et BC
rev_B=[0 0 1];
B=[r1 0 0];

C=[r1+l1*cos(alpha) -l1*sin(alpha) 0];
G_BC=[r1+l1/2*cos(alpha) -l1/2*sin(alpha) 0];

% masse / inertie
T_BC=[cos(alpha) sin(alpha) 0; -sin(alpha) cos(alpha)
0; 0 0 1]; % placer le repère : rotation axe
M_BC=0.08;
I_BC=I_barre(M_BC,R,l1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% C1C2
C1=C+[0 0 a/2];
C2=C+[0 0 -a/2];
G_C1C2=C;

% masse / inertie
T_CC=[cos(pi/2) 0 sin(pi/2) ; 0 1 0; -sin(pi/2) 0 cos(pi
/2)]; % placer le repère : rotation axe
M_C1C2=0.02;
I_C1C2=I_barre(M_C1C2,R,a);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% D et D1D2
y_D=-l1*sin(alpha)-(sqrt(l2^2-(r+l1*cos(alpha))^2));

```

```
G_D1D2=[r2 y_D 0];
D=G_D1D2;
D1=D+[0 0 a/2];
D2=D+[0 0 -a/2];

% masse / inertie
T_DD=[cos(pi/2) 0 sin(pi/2) ;0 1 0; -sin(pi/2) 0 cos(pi
/2)]; % placer le repère : rotation axe
M_D1D2=0.02;
I_D1D2=I_barre(M_D1D2,R,a);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% C1D1 C2D2
y_CD_G=-(sqrt(12^2-(r+l1*cos(alpha))^2))/2;
x_CD_G=-(r+l1*cos(alpha))/2;

G_C1D1=C1+[(x_CD_G) (y_CD_G) 0];
G_C2D2=C2+[(x_CD_G) (y_CD_G) 0];

% masse / inertie
beta=asin(-2*y_CD_G/12);
T_CD=[cos(beta) -sin(beta) 0;sin(beta) cos(beta) 0;0 0
1]; % placer le repère : rotation axe

M_C1D1=0.044;
I_C1D1=I_barre(M_C1D1,R,12);

M_C2D2=M_C1D1;
I_C2D2=I_C1D1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

T=zeros(3,3,3);
% rotation générale du R_i au repère R_0
for i=1:3
a=cos(phi(i));
b=sin(phi(i));
T(:,:,i)=[a 0 b;0 1 0;-b 0 a];
end

T_1=zeros(3,3,3);
% son inverse
for i=1:3
a=cos(phi(i));
b=sin(phi(i));
T_1(:,:,i)=[a 0 -b;0 1 0;b 0 a];
end

% nacelle
M_N=0.16;
I_N=[M_N*r2^2/4 0 0;0 M_N*r2^2/2 0;0 0 M_N*r2^2/4];
% nacelle ronde

%base
% géométrie de la base (masse, inertie sans importance
pour la dynamique)
M_B=10;
I_B=[M_B*r1^2/4 0 0;0 M_B*r1^2/2 0;0 0 M_B*r1^2/4];
% base ronde
```

Listing III.1: Initialisation simulation SimMechanics

Le bloc Simulink du modèle du bras ainsi réalisé est représenté dans la figure III.5. La fenêtre « Block Parameters » permet de définir le numéro du bras définissant ainsi la rotation qui sera appliquée à ce bras la.

$$\begin{cases} bras = 1 & \longrightarrow & rotation = 0 \\ bras = 2 & \longrightarrow & rotation = 2\pi/3 \\ bras = 3 & \longrightarrow & rotation = 4\pi/3 \end{cases}$$

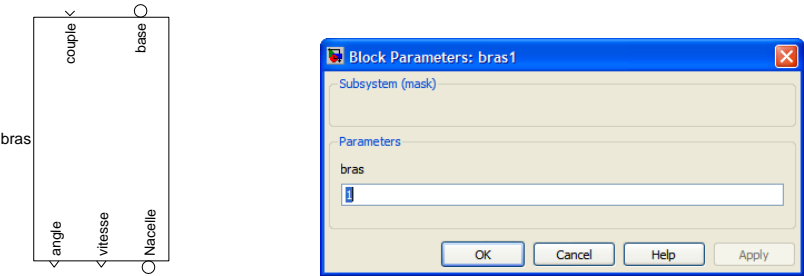


FIGURE III.5: Bloc bras sous Simulink

L'étape suivante consiste à assembler les trois bras du Delta et à ajouter la nacelle et la base du robot. Le schéma SimMechanics du robot complet est donné sur la figure III.6

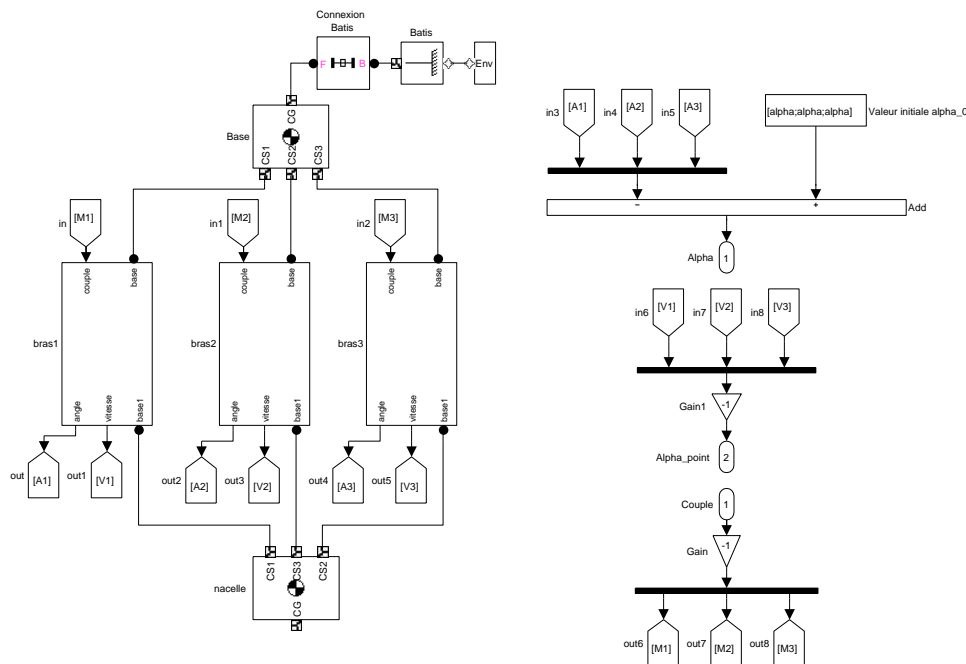


FIGURE III.6: Représentation du Delta sous SimMechanics

Le schéma précédent montre que nous avons aussi rajouté quelque éléments (partie de droite de la figure III.6) qui prennent en charge le traitement des mesures de positions (prise en compte de la position initiale notamment) de vitesse et des couples moteur.

Le schéma précédent (figure III.6) est ensuite formaté sous la forme d'un bloc prêt à l'emploi, présenté dans la figure III.7.

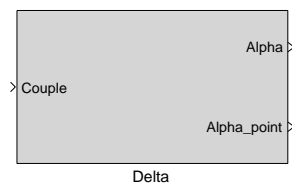


FIGURE III.7: Modèle du Delta sous Simulink

Avec :

$$\begin{cases} \text{Alpha} = [\alpha_1 & \alpha_2 & \alpha_3] \\ \text{Alpha_point} = [\dot{\alpha}_1 & \dot{\alpha}_2 & \dot{\alpha}_3] \\ \text{Couple} = [C_1 & C_2 & C_3] \end{cases}$$

La figure III.8 montre la représentation 3-D du modèle SimMechanics que nous avons réaliser, cela nous permet au passage de valider la justesse de la structure réalisée par rapport à la structure réelle (cf figure III.1).

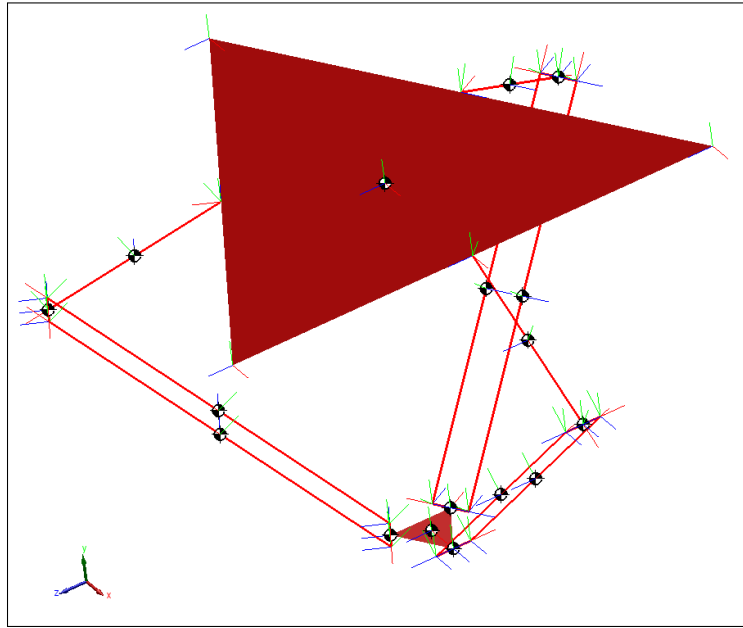


FIGURE III.8: *Modèle 3-D du Delta générée par SimMechanics pour $\alpha^0 = \pi/4$*

4 Conclusion

Dans cette partie de notre travail, nous avons pu réaliser un modèle de simulation complet, présenté sous la forme d'un bloc Simulink pouvant être utilisé directement dans des schémas de simulation, en l'interconnectant tout simplement (comme n'importe quel autre bloc Simulink) à d'autres éléments de la librairie de Simulink. Le modèle qui a été réalisé sera dorénavant utilisé comme plateforme de test, sur laquelle seront appliquées les commandes et avec laquelle sera validée la modélisation analytique du robot (Modèle Géométrique, Modèle Cinématique et Modèle Dynamique) qui seront abordés dans le chapitre suivant.

Chapitre

IV

Modélisation du robot Delta

1	Introduction	22
2	Modèle géométrique	23
2.1	Modèle géométrique Direct	27
2.2	Modèle géométrique inverse	29
2.3	Validation du MGD et du MGI	32
2.4	Volume de travail	33
2.5	Singularités du Delta	35
3	Modèle cinématique	37
3.1	Modèle cinématique direct	39
3.2	Modèle cinématique inverse	40
3.3	Validation du MCD et du MCI	41
4	Modèle Dynamique	42
4.1	Développement des équations dynamiques	43
4.2	Modèle dynamique direct	49
4.3	Validation du MDD	52
4.4	Modèle dynamique inverse	53
4.5	Validation du MDI	53

1 Introduction

Après avoir construit le modèle de simulation du robot Delta et avant de pouvoir appliquer des lois de commandes sur le Delta, il est à présent primordial de le modéliser afin d'avoir les outils nécessaires à sa description et afin de formaliser les relations mathématiques qui existent entre l'entrée du robot (variable articulaire) et la sortie de celui-ci (variable opérationnelle).

Plusieurs niveaux de modélisation sont possibles, dépendant des spécifications et de la nature de l'application envisagée : modèle géométrique, cinématique et dynamique.

Cette modélisation représente les transformations (ou les relations) entre l'espace opérationnel (dans lequel on définit la situation de l'organe terminal) et l'espace articulaire (dans lequel on définit la configuration des articulations du robot). On distingue :

- Les modèles géométriques direct et inverse qui expriment la situation de l'organe terminal en fonction des variables articulaires du mécanisme et inversement.
- Les modèles cinématiques direct et inverse qui expriment la vitesse de l'organe terminal en fonction des vitesses articulaires et inversement.

- Les modèles dynamiques définissant les équations du mouvement du robot permettent d'établir les relations entre les couples ou forces exercées par les actionneurs et les positions, vitesses et accélérations des articulations.

2 Modèle géométrique

La modélisation géométrique d'une structure robotisée consiste simplement en la mise sous forme d'équations des relations liants la position de l'organe terminal (effectuant la tâche) aux différentes coordonnées articulaires du manipulateur (figure IV.1).

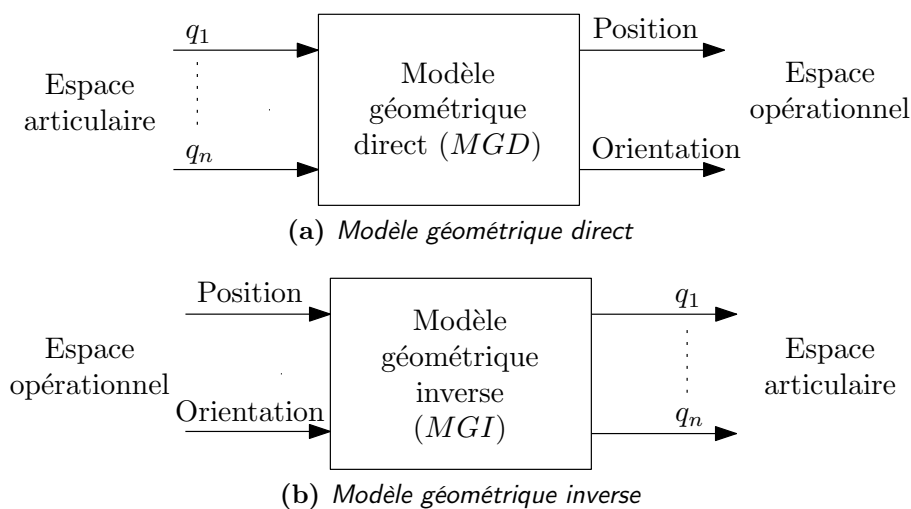


FIGURE IV.1: *Modèle géométrique*

Pour le cas du robot Delta étudié (robot à 3 degrés de liberté à actionnement rotatif) le modèle géométrique permet de lier la position de la nacelle (trois translations) aux angles des 3 articulations rotoïdes du robot. Donc :

$$\left. \begin{array}{l} \text{Modèle} \\ \text{géométrique} \\ \text{du Delta} \end{array} \right\} 3 \text{ positionnements } (x, y, z) \longleftrightarrow 3 \text{ variables articulaires rotoïdes } (q_1, q_2, q_3)$$

La symétrie ternaire que présente le Delta (voir les figures III.1 et IV.3) nous permet de travailler sur un tiers du robot, c'est à dire un seul bras (figure IV.2).

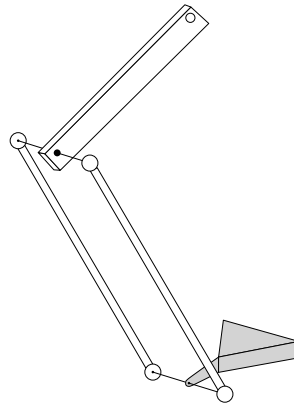


FIGURE IV.2: Un bras du Delta

Un repère R_i est lié à chaque bras du Delta comme l'indique la figure IV.3 qui représente la vue de dessus de la base. Le travail se fera ensuite d'une façon indépendante sur chaque repère.

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 :

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

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

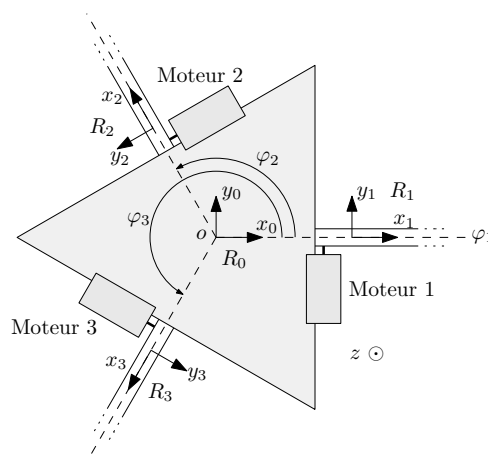


FIGURE IV.3: Vue du dessus de la base du Delta

Les valeurs prises par les φ_i sont $\varphi_1 = 0$, $\varphi_2 = 2\pi/3$ et $\varphi_3 = 4\pi/3$ selon que $i = 1, 2$ ou 3

La figure IV.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 laissée comme telle).

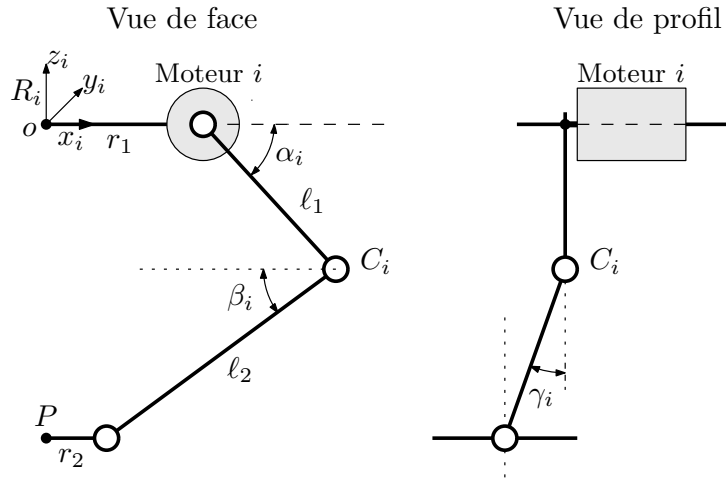


FIGURE IV.4: Schéma d'un bras du delta

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 IV.5.

A noter que les coordonnées articulaire q_i sont représentées ici par les angles α_i .

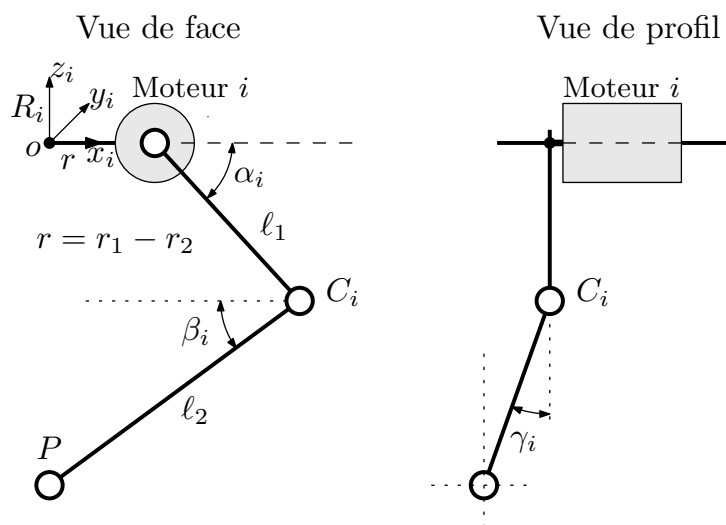


FIGURE IV.5: Schéma simplifié d'un bras du Delta

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 :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_i} = \begin{pmatrix} r + \ell_1 \cos \alpha_i \\ 0 \\ -\ell_1 \sin \alpha_i \end{pmatrix} \quad (\text{IV.2})$$

Avec $i = 1, 2$ ou 3 . Les points C_i appartiennent à des cercles de rayon ℓ_1 centrés sur l'axe du moteur.

Des équations (IV.2) en appliquant la rotation de repère :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_0} = \begin{pmatrix} \cos \varphi_i & -\sin \varphi_i & 0 \\ \sin \varphi_i & \cos \varphi_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r + \ell_1 \cos \alpha_i \\ 0 \\ -\ell_1 \sin \alpha_i \end{pmatrix} \quad (\text{IV.3})$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_0} = \begin{pmatrix} (r + \ell_1 \cos \alpha_i) \cos \varphi_i \\ (r + \ell_1 \cos \alpha_i) \sin \varphi_i \\ -\ell_1 \sin \alpha_i \end{pmatrix}$$

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 $P(x, y, z)_{R_0}$ (voir figure IV.6). Cette sphère est définie dans le repère absolu R_0 par l'équation :

$$(X - x)^2 + (Y - y)^2 + (Z - z)^2 = \ell_2^2 \quad (\text{IV.4})$$

ou 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).

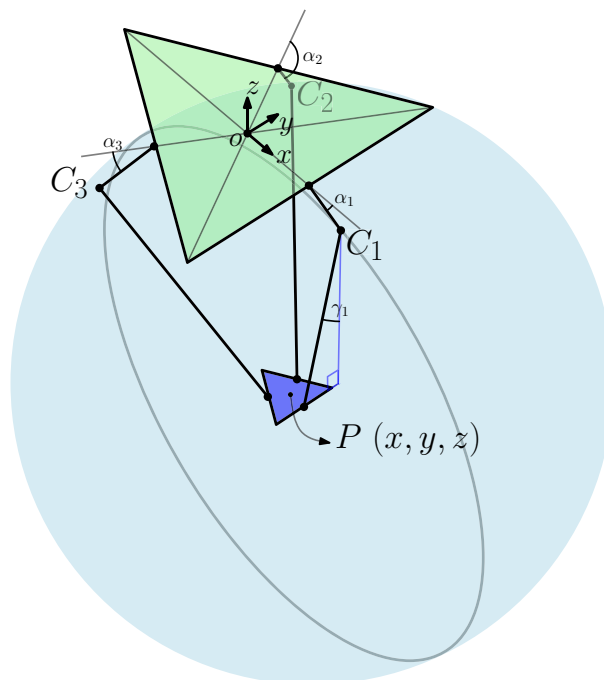


FIGURE IV.6: Sphère de centre P et de rayon ℓ_2 passant par C_1

Le développement précédent (équations IV.2, IV.3 et IV.4) nous permettra de trouver les modèles géométriques direct et inverse du robot Delta.

2.1 Modèle géométrique Direct

Le modèle géométrique direct du robot Delta permet d'exprimer la position translationnelle (x, y, z) de sa nacelle en fonction des trois angles articulaires $(\alpha_1, \alpha_2, \alpha_3)$.



FIGURE IV.7: *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).

Ce calcul est particulier à chaque structure. Pour le cas du robot Delta, 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 [Cla91].

En remplaçant l'équation (IV.3) dans l'équation de la sphère (IV.4) on trouve :

$$\begin{aligned} x^2 - 2x(r + \ell_1 \cos \alpha_i) \cos \varphi_i + (r + \ell_1 \cos \alpha_i)^2 \cos^2 \varphi_i + \\ y^2 - 2y(r + \ell_1 \cos \alpha_i) \sin \varphi_i + (r + \ell_1 \cos \alpha_i)^2 \sin^2 \varphi_i + \\ z^2 + 2z\ell_1 \sin \alpha_i + \ell_1^2 \sin^2 \alpha_i = \ell_2^2 \end{aligned} \quad (\text{IV.5})$$

Ce qui donne :

$$\begin{aligned} x^2 - 2x(r + \ell_1 \cos \alpha_i) \cos \varphi_i + y^2 - 2y(r + \ell_1 \cos \alpha_i) \sin \varphi_i + \\ z^2 + 2z\ell_1 \sin \alpha_i = \ell_2^2 - \ell_1^2 - r^2 - 2r\ell_1 \cos \alpha_i \quad \text{Avec : } i = 1, 2, 3 \end{aligned} \quad (\text{IV.6})$$

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 :

$$\begin{cases} x^2 - E_1x + y^2 - F_1y + z^2 - G_1z = -D_1 & (\text{IV.7a}) \\ x^2 - E_2x + y^2 - F_2y + z^2 - G_2z = -D_2 & (\text{IV.7b}) \\ x^2 - E_3x + y^2 - F_3y + z^2 - G_3z = -D_3 & (\text{IV.7c}) \end{cases}$$

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

En effectuant [(IV.7a)–(IV.7b)] et [(IV.7b)–(IV.7c)] :

$$\begin{cases} (E_2 - E_1)x + (F_2 - F_1)y + (G_2 - G_1)z = D_2 - D_1 & \text{(IV.8a)} \\ (E_3 - E_2)x + (F_3 - F_2)y + (G_3 - G_2)z = D_3 - D_2 & \text{(IV.8b)} \end{cases}$$

De l'équation (IV.8a) on obtient :

$$x = -\frac{F_2 - F_1}{E_2 - E_1}y - \frac{G_2 - G_1}{E_2 - E_1}z + \frac{D_2 - D_1}{E_2 - E_1} \quad \text{(IV.9)}$$

En remplaçant cela dans l'équation (IV.8b) :

$$y = \left[\frac{-(E_3 - E_1)(G_2 - G_1) + (G_3 - G_1)(E_2 - E_1)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] z + \left[\frac{-(D_3 - D_1)(E_2 - E_1) + (E_3 - E_1)(D_2 - D_1)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] \quad \text{(IV.10)}$$

Donc :

$$y = \frac{H_1}{H_2}z + \frac{H_3}{H_2} \quad \text{Avec : } \begin{cases} H_1 = -(E_3 - E_1)(G_2 - G_1) + (G_3 - G_1)(E_2 - E_1) \\ H_2 = -(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1) \\ H_3 = -(D_3 - D_1)(E_2 - E_1) + (E_3 - E_1)(D_2 - D_1) \end{cases} \quad \text{(IV.11)}$$

De l'équation (IV.8a) on obtient aussi :

$$y = \frac{E_1 - E_2}{F_2 - F_1}x + \frac{G_1 - G_2}{F_2 - F_1}z + \frac{D_2 - D_1}{F_2 - F_1} \quad \text{(IV.12)}$$

En remplaçant toujours dans l'équation (IV.8b) :

$$x = \left[\frac{-(G_3 - G_1)(F_2 - F_1) - (F_3 - F_1)(G_1 - G_2)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] z + \left[\frac{(D_3 - D_1)(F_2 - F_1) - (F_3 - F_1)(D_1 - D_2)}{-(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1)} \right] \quad \text{(IV.13)}$$

Donc :

$$x = \frac{H_5}{H_2}z + \frac{H_4}{H_2} \quad \text{Avec : } \begin{cases} H_2 = -(F_3 - F_1)(E_2 - E_1) + (E_3 - E_1)(F_2 - F_1) \\ H_4 = (D_3 - D_1)(F_2 - F_1) - (F_3 - F_1)(D_1 - D_2) \\ H_5 = -(G_3 - G_1)(F_2 - F_1) - (F_3 - F_1)(G_1 - G_2) \end{cases} \quad \text{(IV.14)}$$

Il nous reste à calculer le z ; en remplaçant l'expression de x (IV.14) et y (IV.11) dans (IV.7a) on aura :

$$z^2 \underbrace{\left(1 + \frac{H_5^2 + H_1^2}{H_2^2}\right)}_L - z \underbrace{\left(-\frac{2(H_5H_4 + H_1H_3)}{H_2^2} + \frac{E_1H_5 + F_1H_1}{H_2} + G_1\right)}_M + \underbrace{\left(D_1 + \frac{H_4^2 + H_3^2}{H_2^2} - \frac{E_1H_4 + F_1H_3}{H_2}\right)}_N = 0 \quad \text{(IV.15)}$$

Finalement :

$$z = \frac{M - \sqrt{M^2 - 4LN}}{2L} \quad \text{(IV.16)}$$

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 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 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.

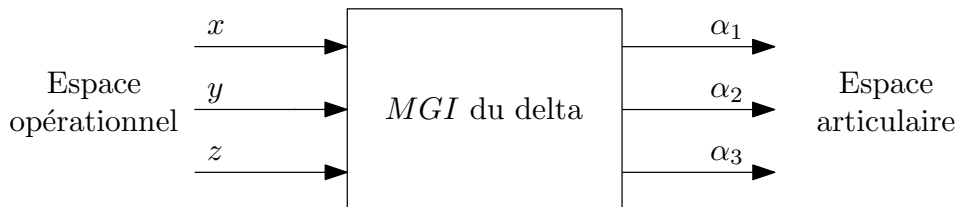


FIGURE IV.8: *Modèle géométrique inverse du robot Delta*

Plusieurs formulations ont été proposés pour le calcul du modèle géométrique inverse du robot Delta [Cla91, Cod91].

On peut résoudre les équations (IV.7) en $(\alpha_1, \alpha_2, \alpha_3)$ et trouver le MGI. Cette solution a été proposée dans [Cla91].

En reprenant l'équation (IV.6), on sépare les termes en $(\cos \alpha_i)$ et en $(\sin \alpha_i)$:

$$\begin{aligned} (2\ell_1 z) \sin \alpha_i + (2r\ell_1 - 2\ell_1 x \cos \varphi_i - 2\ell_1 y \sin \varphi_i) \cos \alpha_i &= (\ell_2^2 - \ell_1^2 - r^2 + \\ 2ry \sin \varphi_i - z^2 - y^2 - x^2 + 2rx \cos \varphi_i) &\quad \text{Avec : } i = 1, 2, 3 \end{aligned} \quad (\text{IV.17})$$

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 (IV.17) devient :

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

En posant :

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

On aura :

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

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

$$\tan\left(\frac{\alpha_i}{2}\right) = \frac{-A_i \pm \sqrt{A_i^2 + B_i^2 - C_i^2}}{B_i + C_i} \quad (\text{IV.18})$$

La solution est donnée par :

$$\alpha_i = 2 \arctan\left(\frac{-A_i + \sqrt{A_i^2 + B_i^2 - C_i^2}}{B_i + C_i}\right) \quad \text{Avec : } i = 1, 2, 3 \quad (\text{IV.19})$$

Remarque La résolution de l'équation en $(\cos \alpha_i)$ et en $(\sin \alpha_i)$ donne deux solutions, chacune d'entre elles correspondant 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 le l'espace de travail) sera celle qui donne l'angle de plus faible valeur, donc celle avec un signe « + » (équation IV.19), l'autre solution est rejetée.

Cette solution présente un problème de singularité mathématique de type (0/0). En effet dans le cas ou $(B_i + C_i) = 0$ on voit bien que le numérateur s'annule aussi. On montre que cette singularité est présente sur les surfaces sphériques de centres $((r - \ell_1), 0, 0)$ dans chaque repère intermédiaire (repère relatif à un des bras du robot) et de rayon ℓ_2

Une autre formulation du MGI qui règle ce problème, a été proposée par Codourey dans [Cod91].

Dans le repère R_i on a :

$$\begin{cases} (x_{c_i} - x)^2 + (y_{c_i} - y)^2 + (z_{c_i} - z)^2 = \ell_2^2 \\ \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{C_i/R_i} = \begin{pmatrix} r + \ell_1 \cos \alpha_i \\ 0 \\ -\ell_1 \sin \alpha_i \end{pmatrix} \end{cases} \quad (\text{IV.20})$$

ou x, y et z sont les coordonnées de la nacelle Sachant que $y_{c_i} = 0$ (le point C_i étant dans le plan $\{x_i, z_i\}$), la première équation devient :

$$x_{c_i}^2 - 2xx_{c_i} + x^2 + y^2 + z_{c_i}^2 - 2zz_{c_i} + z^2 = \ell_2^2 \quad (\text{IV.21})$$

En éliminant les $\cos \alpha_i$ et $\sin \alpha_i$ de la seconde équation du système IV.20, nous obtenons :

$$\begin{aligned} (x_{c_i} - r)^2 + z_{c_i}^2 &= \ell_1^2 \\ \Rightarrow x_{c_i}^2 - 2rx_{c_i} + r^2 + z_{c_i}^2 &= \ell_1^2 \end{aligned} \quad (\text{IV.22})$$

On tire alors les coordonnées des points intermédiaires $C_i(x_{c_i}, y_{c_i}, z_{c_i})$ en effectuant [(IV.22)–(IV.21)] :

$$2x_{c_i}(x - r) + 2zz_{c_i} = \ell_1^2 - \ell_2^2 - r^2 + x^2 + y^2 + z^2$$

qui est de la forme :

$$Ax_{c_i} + 2zz_{c_i} = B \quad \text{Avec :} \quad \begin{cases} A = 2(x - r) \\ B = \ell_1^2 - \ell_2^2 - r^2 + x^2 + y^2 + z^2 \end{cases} \quad (\text{IV.23})$$

Donc :

$$x_{c_i} = \frac{B - 2zz_{c_i}}{A} \quad (\text{IV.24})$$

En remplaçant cela dans l'équation (IV.22) on trouve :

$$\begin{aligned} (A^2 + 4z^2)z_{c_i}^2 + 4z(rA - B)z_{c_i} + (B^2 - 2rAB - A^2\ell_1^2) &= 0 \\ az_{c_i}^2 - 2bz_{c_i} + c &= 0 \end{aligned} \quad (\text{IV.25})$$

Finalement :

$$z_{c_i} = \frac{b - \sqrt{b^2 - ac}}{a} \quad (\text{IV.26})$$

Sachant que :

$$\begin{cases} a = A^2 + 4z^2 \\ b = -2z(rA - B) \\ c = B^2 - 2rAB - A^2\ell_1^2 + A^2r^2 \end{cases} \quad \text{et} \quad \begin{cases} A = 2(x - r) \\ B = \ell_1^2 - \ell_2^2 - r^2 + x^2 + y^2 + z^2 \end{cases}$$

L'autre solution de z_{c_i} étant rejetée car en dehors du volume de travail, pour les mêmes raisons que celles cités précédemment.

On peut maintenant remonter à l'angle α_i avec la relation (qui est tirée de la figure IV.5) :

$$\alpha_i = \arcsin \left(\frac{-z_{c_i}}{\ell_1} \right) \quad (\text{IV.27})$$

Cette solution a l'avantage de ne pas présenter de singularité par rapport a une solution en $[\arctan]$. Cependant une correction doit être apportée à α_i si le bras n'es pas dans $[-\pi/2, \pi/2]$ (selon la valeur de x_{c_i}).

Le calcul du MGI du Delta se fait donc selon les étapes suivantes :

1. Ayant (x, y, z) souhaité dans R_0 pour la nacelle on applique la transformation de R_0 à R_i (équation IV.1) on aura (x, y, z) dans R_i .
2. Grâce aux équations (IV.24) et (IV.26) et aux résultats précédents on trouve les coordonnées x_{c_i} et z_{c_i} des C_i .
3. On calcule avec cela (et on corrige si nécessaire) l'angle α_i de l'équation (IV.27).
4. Ces étapes sont à refaire pour $i = 0, 1, 2$ afin d'obtenir tous les angles $\alpha_1, \alpha_2, \alpha_3$.

2.3 Validation du MGD et du MGI

En ayant la visualisation 3D de la structure et la possibilité de déterminer les coordonnées articulaires et opérationnelles (nacelle) des mouvements du modèle SimMechanics, nous pouvons valider les MGI et MGD développé précédemment.

Afin de valider le modèle géométrique nous allons utiliser la possibilité donnée par SimMechanics d'ajouter un bloc capteur (virtuel) permettant de donner la position (x, y, z) de la nacelle. Cela est montré sur la figure IV.9.

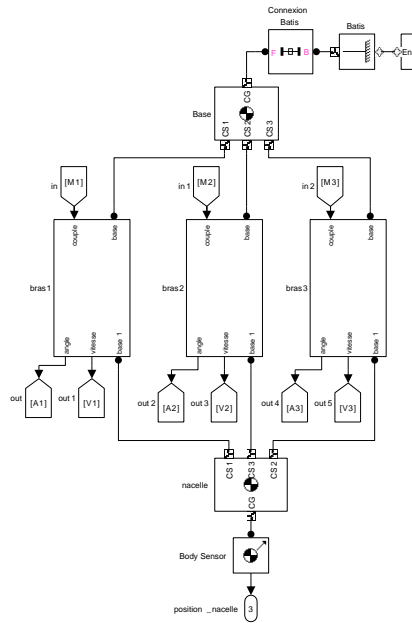


FIGURE IV.9: Modèle SimMechanics avec capture de la position de la nacelle

Aussi, afin de contrôler les mouvements du robot et pouvoir ainsi faire la validation, un régulateur PID simple à été rajouté au modèle du robot Delta.

Pour la validation nous allons comparer la variations de $(\alpha_1, \alpha_2, \alpha_3)$ avec le $MGI(x, y, z)$ et la variation de (x, y, z) avec le $MGD(\alpha_1, \alpha_2, \alpha_3)$ (figure IV.10).

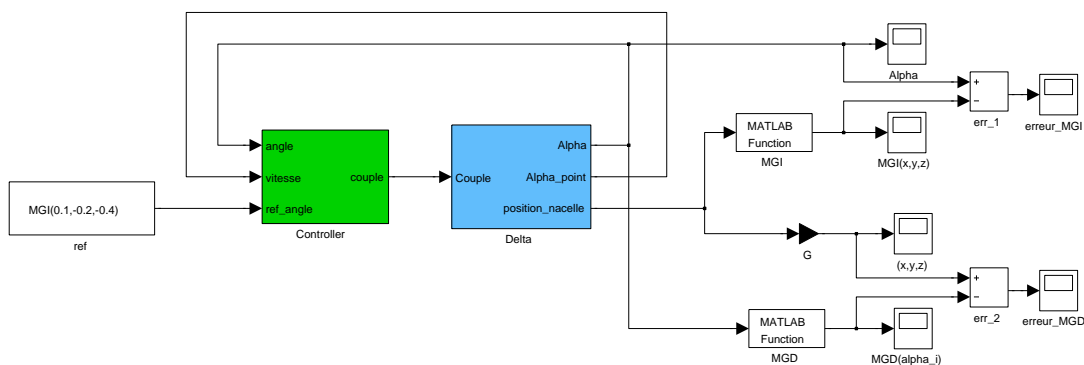


FIGURE IV.10: Validation du MGI et du MGD

Les erreurs relatives sont de l'ordre du 10^{-12} . Les MGD et MGI développés sont donc corrects.

2.4 Volume de travail

Le volume de travail du robot Delta est la portion de l'espace pouvant être atteinte par son organe terminal situé sur sa nacelle. Il est déterminé par l'intersection des volumes atteignables par les trois chaînes cinématiques constituant le Delta prises indépendamment. L'espace théorique obtenu précédemment est réduit à un volume plus restreint en prenant en compte certaines limitations mécaniques, inhérentes à la mécanique du Delta et ses singularités :

- Course limitée des rotules (angle d'ouverture des rotules) présentes aux articulations des avant-bras, ceci se manifeste par une limitation des angles γ_i (voir IV.4 et IV.5) et par un cloisonnement de l'espace théorique dans un prisme hexagonal [Cla91].
- Éviter l'interaction entre les bras et les barres parallèles.
- Éviter l'interaction entre les deux barres parallèles et le moteur du bras sur lequel ils y sont rattachés. Cette limitation et celle qui la précède sont respectées par la minoration de la quantité $\alpha_i + \beta_i$
- Les points singuliers (singularités) doivent être enlevés du volume de travail (c.f 2.5), cela se fait par une majoration de la quantité $\alpha_i + \beta_i$ pour la singularité (a) et par un choix judicieux des dimensions caractéristiques du Delta pour les trois autres singularités [Cla91].

En considérant les limites précédentes, l'espace de travail prend la forme d'un prisme hexagonal terminé par un corps de révolution (pseudo conique) dans sa partie inférieure. Aussi, pour un fonctionnement optimal du robot Delta et pour une efficacité de la commande, on évitera d'utiliser le Delta sur les limites extrêmes de son volume de travail pour les raisons suivantes :

- Diminution de rigidité dans certains points excentrés du volume de travail ;
- Rapprochement des points singuliers ;
- Diminution de la résolution opérationnelle (pour une résolution articulaire donnée des capteurs) quand les chaînes cinématiques du robot sont plus ou moins déployés ;
- Diminution de la masse transportable quand le robot est complètement déployé.

La partie basse du volume de travail (calotte pseudo conique) est très peu intéressante et ce pour les mêmes raisons que celles citées précédemment, étant donné que le Delta travaille dans des positions extrêmes dans ce cas de figure. Cette partie du volume de travail sera donc ignorée, sa suppression ne constitue d'ailleurs pas une perte substantielle dans le volume de travail global.

En définitif, le volume de travail du Delta qui sera retenu et qui sera utilisé dans notre travail sera un cylindre (cylindre inscrit dans le prisme hexagonal cité auparavant) 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 géométriques du robot : ℓ_1 , ℓ_2 et $r = r_1 - r_2$, sont les suivantes [Cla91] :

$$\begin{aligned}
 D &= 2 \cdot \ell_2 \cdot \sin \gamma_{max} \\
 H &= \ell_1 \sqrt{\left(1 + \frac{\ell_2}{\ell_1}\right)^2 - \left(\frac{r}{\ell_1} + 2 \frac{\ell_2 \sin \gamma_{max}}{\ell_1 \sqrt{3}}\right)^2} - \ell_1 \sqrt{1 + \left(\frac{\ell_2}{\ell_1}\right)^2 - \frac{\ell_2}{\ell_1} \sqrt{3}} \\
 Z_h &= -\ell_1 \sqrt{1 + \left(\frac{\ell_2}{\ell_1}\right)^2 - 2 \frac{\ell_2}{\ell_1} \cos(\alpha_i + \beta_i)_{min}}
 \end{aligned} \tag{IV.28}$$

La figure IV.11 montre le volume de travail du Delta considéré.

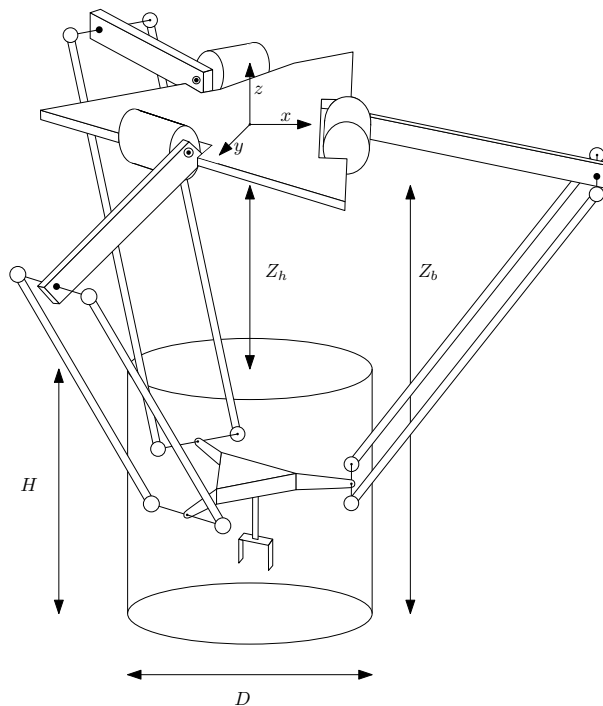


FIGURE IV.11: Volume de travail considéré du Delta

Les valeurs caractéristiques du robot (tableau III.1) nous permettent de déterminer les dimensions du volume de travail du robot, en appliquant les relations (IV.28) avec : $\gamma_{max} = 40^\circ$ et $(\alpha + \beta)_{min} = 30^\circ$. Ces dimensions sont les suivantes (figure IV.11) :

- Diamètre du cylindre contenant le volume de travail : $D = 668mm$
- Hauteur du cylindre contenant le volume de travail : $H = 232mm$
- Limite supérieure du volume de travail selon l'axe z : $Z_h = -320mm$
- Limite inférieure du volume de travail selon le même axe z : $Z_b = -552mm$

2.5 Singularités du Delta

Pour le cas des robots parallèles les points singuliers résultent du gain en degrés de libertés. Contrairement aux robots séries où ils résultent d'une perte en degrés de libertés.

Le gain en degrés de liberté pour une structure comme le robot Delta est un effet indésirable. La rigidité du robot permet entre autres de garder la nacelle parfaitement parallèle à la base, les orientations étant bloqués passivement par la forme de la structure.

Les moteurs doivent être capables de contrôler parfaitement les mouvements de la nacelle pour assurer une commandabilité optimale du robot. Or, le passage par un point singulier peut avoir pour conséquence la rotation non désirée de la nacelle ou un mouvement de translation non souhaité, la position de la nacelle n'est à ce moment-là plus contrôlable et tout le système de commande est remis en cause.

L'analyse de modèle statique du Delta fait ressortir un autre effet du rapprochement du robot par ses positions singulières, l'équation du modèle statique qui est la suivante :

$$F = J^{-T}\Gamma \quad (\text{IV.29})$$

ou J^{-T} est la Jacobienne inverse transposée.

Pour un torseur statique F appliqué sur la nacelle (charge), les couples Γ fournies par les moteurs sont inversement proportionnels au déterminant de J^{-T} (résolution du système d'équations), cela veut dire que les couples fournis par les moteurs doivent être d'autant plus importants que le robot se rapproche d'une singularité (car $\det(J^{-T})$ se rapproche de la valeur nulle) et ce pour une même charge (torseur F constant). Le robot perd donc de son efficacité en plus des sollicitations importantes que subissent les moteurs dans ce genre de configurations.

Il est donc primordial que ces positions singulières soient éliminées de l'espace de travail. Pour ce faire, il nous faut d'abord déterminer les configurations géométriques qui donnent ces points singuliers. La relation entre les vitesses articulaires et les vitesses opérationnelles est, rappelons le, la suivante :

$$\dot{Q} = J^{-1}\dot{X} \quad (\text{IV.30})$$

ou J^{-1} est la matrice Jacobienne. Une configuration singulière, dans le cas d'un robot parallèle, se manifeste par la mobilité de la nacelle ($\dot{X} \neq 0$), malgré le blocage des articulations ($\dot{Q} = 0$), cela ne peut avoir lieu que lorsque la matrice Jacobienne J^{-1} est dégénérée (singulière) c'est à dire lorsque son déterminant est nul.

En général, la résolution du déterminant de la matrice Jacobienne permet de déterminer facilement les singularités. Comme nous le verrons dans la prochaine section, le modèle cinématique du robot Delta est difficile à expliciter analytiquement et on préférera un calcul numérique de celui-ci.

La résolution du déterminant de la jacobienne pour le Delta relèverait donc de la quasi impossibilité. D'autres méthodes doivent être alors utilisées. [Mer97] Expose un certain nombre de méthodes, basées sur des analyses géométriques de la structure robotique. On

citera notamment celle qui est basée sur la géométrie de GRASSMAN, utilisant la notion de vecteurs de PLUCKER pour déterminer les points singuliers de la Jacobienne.

Comme présenté dans [Cla91], l'analyse des positions des barres parallèles (parallélogrammes) les une par rapport aux autres permet de définir les configurations pour laquelle la nacelle gagne en degrés de libertés. Les configurations singulières suivantes sont énoncés :

1. Les trois parallélogrammes sont parallèles (figure IV.12a) ;
2. Les trois parallélogrammes sont sur le même plan horizontal (figure IV.12b) ;
3. Deux des trois parallélogrammes ont leurs barres sur le même plan (figure IV.12c) ;
4. Deux des trois parallélogrammes ont leurs barres parallèles (figure IV.12d).

Configurations singulières du Delta

Les configurations singulières citées précédemment sont illustrées ci-après :

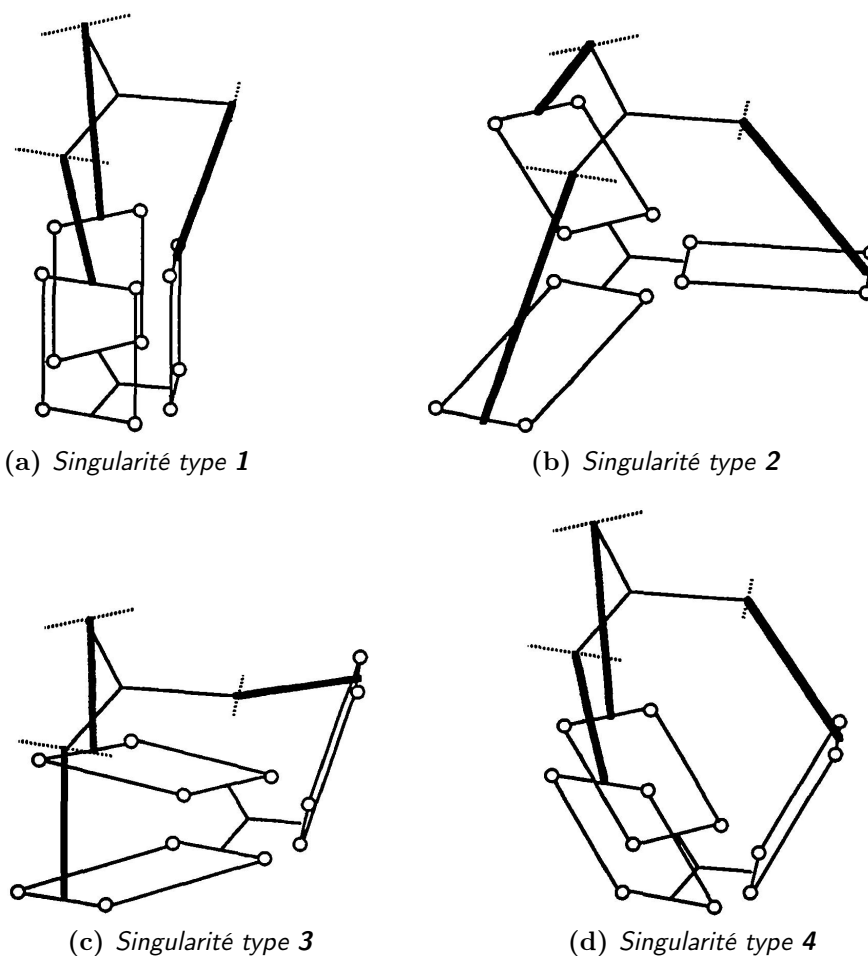


FIGURE IV.12: Les singularités du robot Delta

3 Modèle cinématique

La modélisation cinématique d'un robot manipulateur en général et du robot Delta en particulier, permet d'établir les relations entre les vitesses articulaires (généralisées) \dot{q}_i et les vitesses opérationnelles de l'organe terminal.

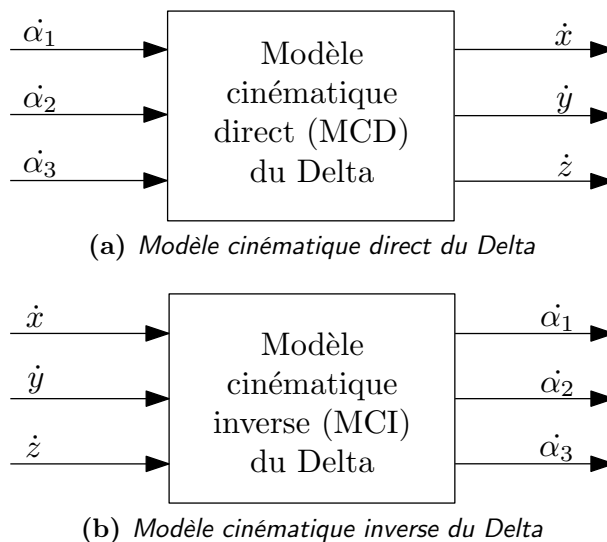


FIGURE IV.13: *Modèle cinématique du robot Delta*

L'établissement de ce modèle est nécessaire, notamment pour la commande est la régulation des tâches effectuées par le robot. En effet, le but de la régulation est d'agir sur les variables opérationnelles (\dot{x} , \dot{y} , \dot{z}) pour avoir un comportement désiré. Cependant, l'accès à ces variables, en vue de les contrôler, ne peut se faire que par le biais des variables articulaires ($\dot{\alpha}_1$, $\dot{\alpha}_2$, $\dot{\alpha}_3$ directement accessibles par les actionneurs), une relations entre ces deux groupes de vitesses (articulaires et opérationnelles) nous permettra alors de passer des unes vers les autres.

Le modèle cinématique du robot Delta nous permet donc de lier les vitesses généralisées (articulaires) aux vitesses opérationnelles (nacelle). Dans sa version directe (voir figure IV.13a), le MCD (Modèle Cinématique Direct) relie les vitesses opérationnelles aux vitesses généralisées des articulations à travers la matrice JACOBIENNE (ou JACOBIENNE tout court) selon la relation suivante :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = J \begin{pmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \end{pmatrix} \quad (\text{IV.31})$$

Le modèle cinématique inverse (MCI) établie quand à lui, le lien entre les vitesses généralisées (vitesses des articulations) et les vitesses opérationnelles (vitesses de la nacelle)

dans cet ordre, selon la relation suivante :

$$\begin{pmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \end{pmatrix} = J^{-1} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \quad (\text{IV.32})$$

Il suffit donc, d'après les deux relations précédentes, de déterminer la matrice JACOBIENNE (ou JACOBIENNE inverse, le passage entre les deux se fait par une simple inversion matricielle) pour que le modèle cinématique du robot soit entièrement défini.

La manière la plus simple (néanmoins la plus directe) de déterminer la matrice JACOBIENNE consiste à utiliser les relations établies dans le cadre du modèle géométrique, relations qui relient les coordonnées articulaires (α_1 , α_2 et α_3) aux coordonnées opérationnelles (x , y et z) :

$$\begin{cases} x = f_1(\alpha_i) \\ y = f_2(\alpha_i) \\ z = f_3(\alpha_i) \end{cases} \quad (\text{IV.33})$$

En dérivant ces relations de part et d'autre, par rapport à la variable temporelle t , nous obtenons le résultat suivant :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} \frac{\partial f_1}{\partial \alpha_1} & \frac{\partial f_1}{\partial \alpha_2} & \frac{\partial f_1}{\partial \alpha_3} \\ \frac{\partial f_2}{\partial \alpha_1} & \frac{\partial f_2}{\partial \alpha_2} & \frac{\partial f_2}{\partial \alpha_3} \\ \frac{\partial f_3}{\partial \alpha_1} & \frac{\partial f_3}{\partial \alpha_2} & \frac{\partial f_3}{\partial \alpha_3} \end{pmatrix} \begin{pmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \end{pmatrix} \quad (\text{IV.34})$$

On peut donc conclure, d'après les relations (IV.31) et (IV.34) que la matrice JACOBIENNE s'écrit de la façon suivante :

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial \alpha_1} & \frac{\partial f_1}{\partial \alpha_2} & \frac{\partial f_1}{\partial \alpha_3} \\ \frac{\partial f_2}{\partial \alpha_1} & \frac{\partial f_2}{\partial \alpha_2} & \frac{\partial f_2}{\partial \alpha_3} \\ \frac{\partial f_3}{\partial \alpha_1} & \frac{\partial f_3}{\partial \alpha_2} & \frac{\partial f_3}{\partial \alpha_3} \end{pmatrix} \quad (\text{IV.35})$$

De la même manière, si l'on écrit les relations établies lors de l'étude géométrique du robot Delta qui décrivent les angles α_i en fonctions des variables opérationnelles (x , y et z) :

$$\begin{cases} \alpha_1 = g_1(x, y, z) \\ \alpha_2 = g_2(x, y, z) \\ \alpha_3 = g_3(x, y, z) \end{cases} \quad (\text{IV.36})$$

on aboutit à un résultat analogue à la relation (IV.35), donnant cette fois-ci l'expression de la JACOBIENNE inverse :

$$J^{-1} = \begin{pmatrix} \frac{\partial g_1}{\partial x} & \frac{\partial g_1}{\partial y} & \frac{\partial g_1}{\partial z} \\ \frac{\partial g_2}{\partial x} & \frac{\partial g_2}{\partial y} & \frac{\partial g_2}{\partial z} \\ \frac{\partial g_3}{\partial x} & \frac{\partial g_3}{\partial y} & \frac{\partial g_3}{\partial z} \end{pmatrix} \quad (\text{IV.37})$$

Il faut noter que ce même résultat peut être obtenu par une simple inversion de la matrice JACOBIENNE de la relation (IV.35) :

$$\mathcal{J}^{-1} = \begin{pmatrix} \frac{\partial g_1}{\partial x} & \frac{\partial g_1}{\partial y} & \frac{\partial g_1}{\partial z} \\ \frac{\partial g_2}{\partial x} & \frac{\partial g_2}{\partial y} & \frac{\partial g_2}{\partial z} \\ \frac{\partial g_3}{\partial x} & \frac{\partial g_3}{\partial y} & \frac{\partial g_3}{\partial z} \end{pmatrix} = \begin{pmatrix} \frac{\partial f_1}{\partial \alpha_1} & \frac{\partial f_1}{\partial \alpha_2} & \frac{\partial f_1}{\partial \alpha_3} \\ \frac{\partial f_2}{\partial \alpha_1} & \frac{\partial f_2}{\partial \alpha_2} & \frac{\partial f_2}{\partial \alpha_3} \\ \frac{\partial f_3}{\partial \alpha_1} & \frac{\partial f_3}{\partial \alpha_2} & \frac{\partial f_3}{\partial \alpha_3} \end{pmatrix}^{-1} \quad (\text{IV.38})$$

Finalement, l'élaboration du MCI et du MCD du robot Delta se résume à une simple dérivation des relations (IV.11, IV.14 et IV.16 pour le MCD et IV.26 pour le MCI) déjà établies pour le modèle géométrique du Delta.

La complexité des relations citées précédemment rend l'opération de dérivation assez fastidieuse. Nous avons donc utilisé le logiciel MAPLE afin d'effectuer la tâche de dérivation. Les résultats obtenus sont les éléments de la matrice JACOBIENNE (ou de la JACOBIENNE inverse) écrits sous forme analytique, nous constatons au passage que ces éléments sont d'une taille assez importante, ce qui a pour effet d'augmenter leur temps de calcul, chose qui n'est certainement pas souhaitable pour une utilisation en temps réel.

Partant de ce constat, nous avons opté pour l'utilisation d'une autre méthode, moins gourmande en temps de calcul, qui consiste tout simplement à calculer les dérivées d'une manière numérique (selon la définition de la dérivée).

3.1 Modèle cinématique direct

Pour établir le modèle cinématique direct, nous utilisons les formules (IV.11), (IV.14) et (IV.16) de la méthode établie dans [Cla91], nous obtenons les éléments de la JACOBIENNE par dérivation numérique de ces formules, comme expliqué dans le paragraphe précédent en adoptant la formulation suivante :

$$\left\{ \begin{array}{l} \frac{\partial f_i}{\partial \alpha_1} = \frac{f_i(\alpha_1 + \Delta, \alpha_2, \alpha_3) - f_i(\alpha_1, \alpha_2, \alpha_3)}{\Delta} \\ \frac{\partial f_i}{\partial \alpha_2} = \frac{f_i(\alpha_1, \alpha_2 + \Delta, \alpha_3) - f_i(\alpha_1, \alpha_2, \alpha_3)}{\Delta} \\ \frac{\partial f_i}{\partial \alpha_3} = \frac{f_i(\alpha_1, \alpha_2, \alpha_3 + \Delta) - f_i(\alpha_1, \alpha_2, \alpha_3)}{\Delta} \\ [f_1 \quad f_2 \quad f_3] \equiv MGD \end{array} \right. \quad (\text{IV.39})$$

Ou : Δ est le pas d'intégration et i allant de 1 à 3.

On obtient alors bien les 9 éléments de la matrice Jacobienne.

3.2 Modèle cinématique inverse

Comme pour le modèle cinématique direct, le modèle cinématique inverse se base sur la dérivation numérique des relations du modèle géométrique inverse. En adoptant le même principe que précédemment on aura :

$$\left\{ \begin{array}{l} \frac{\partial g_i}{\partial x} = \frac{g_i(x + \Delta, y, z) - g_i(x, y, z)}{\Delta} \\ \frac{\partial g_i}{\partial y} = \frac{g_i(x, y + \Delta, z) - g_i(x, y, z)}{\Delta} \\ \frac{\partial g_i}{\partial z} = \frac{g_i(x, y, z + \Delta) - g_i(x, y, z)}{\Delta} \\ [g_1 \quad g_2 \quad g_3] \equiv MGI \end{array} \right. \quad (\text{IV.40})$$

Ou : Δ est le pas d'intégration et i allant de 1 à 3.

On obtient alors bien les 9 éléments de la matrice Jacobienne inverse.

Dans le cadre de la commande du robot Delta, le modèle cinématique inverse est utilisé d'une manière intensive, une formulation précise et rapide du modèle cinématique inverse est ainsi nécessaire pour assurer un fonctionnement optimal du robot. Pour faire concorder ces deux critères (précision et rapidité), nous avons comparé les différentes méthodes disponibles, notamment celles disponibles dans les travaux de [Cla91] et de [Cod91]. Les critères de comparaison ont été la rapidité de calcul (temps de calcul de la JACOBIENNE inverse de chaque méthode pour la même situation) et la précision de la méthode (par rapport aux résultats des formules analytiques des éléments de la JACOBIENNE inverse).

Les résultats de cette comparaison amène un certain nombre de remarques :

La précision La précision de chacune des deux méthodes est quasi identique, de plus, les résultats obtenus par ces deux méthodes numériques ne diffère pratiquement pas des résultats analytique, l'écart dépend essentiellement du choix du pas de dérivation. On pourra même rajouter que la précision obtenue avec la méthode analytique peu être égalée, ceci est due au fait que même si les formules utilisées sont exactes (formules obtenues analytiquement), la représentation des nombres réel sur une machine de calcul (représentation à virgule flottante) possède une précision limitée (quelque soit la puissance de la machine de traitement), précision qui peut être atteinte par un choix approprié du pas de dérivation de la méthode numérique.

La rapidité La comparaison entre les deux méthodes numériques nous montre aussi que la méthode issue du modèle géométrique inverse de [Cla91] est nettement plus rapide que celle issue du modèle géométrique inverse de [Cod91]. Cependant, la méthode de [Cla91] peut aboutir à des singularités de calcul, ces singularités sont présentes sur des surfaces

sphériques de centres $((r - \ell_1), 0, 0)$ dans chaque repère intermédiaire (repère relatif à un des bras du robot) et de rayon ℓ_2 .

Afin d'optimiser le MCI du Delta, nous avons eu l'idée de combiner les deux méthodes numérique précédente, en tirant profit de leurs avantages respectifs. La méthode que nous utilisons consiste à faire un test du point de calcul de la JACOBIENNE inverse, si ce dernier appartient à l'une des trois sphères de singularité, la méthode de [Cod91] est utilisée, sinon, la méthode de [Cla91] (plus rapide) est utilisée. Enfin, et pour valider cette dernière méthode, nous avons procédé à des tests comparatifs avec les 2 méthodes précédentes, comme prévu, notre dernière méthode est plus rapide que celle de [Cod91] (temps de calcul divisée par 2 en moyenne), mais elle reste cependant moins rapide que celle de [Cla91] sauf que le problème des singularités de calcul est éliminé, ce qui constitue donc un bon compromis entre les méthodes de [Cla91] et de [Cod91].

Remarque Le modèle cinématique donnant les vitesses opérationnelles en fonction des vitesses articulaires peut être interprété comme un modèle différentiel. La relation :

$$\begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = J \begin{pmatrix} \Delta \alpha_1 \\ \Delta \alpha_2 \\ \Delta \alpha_3 \end{pmatrix}$$

permettrait alors de déterminer la résolution pouvant être obtenue dans l'espace opérationnel connaissant la résolution dans l'espace articulaire (résolution des capteurs de position des actionneurs).

3.3 Validation du MCD et du MCI

Afin de valider le modèle cinématique nous allons ajouter au modèle SimMechanics précédent (schéma IV.9) un capteur de vitesse (virtuel) placé sur la nacelle, le capteur donnera donc un vecteur [position vitesse] de la nacelle.

Nous allons comparer la variations de $(\dot{\alpha}_1, \dot{\alpha}_2, \dot{\alpha}_3)$ avec le $MCI(x, y, z, \dot{x}, \dot{y}, \dot{z})$ et la variation de $(\dot{x}, \dot{y}, \dot{z})$ avec le $MCD(\alpha_1, \alpha_2, \alpha_3, \dot{\alpha}_1, \dot{\alpha}_2, \dot{\alpha}_3)$.

Le schéma de simulation de la figure IV.14 nous permettra de valider les MCD et MCI développés précédemment.

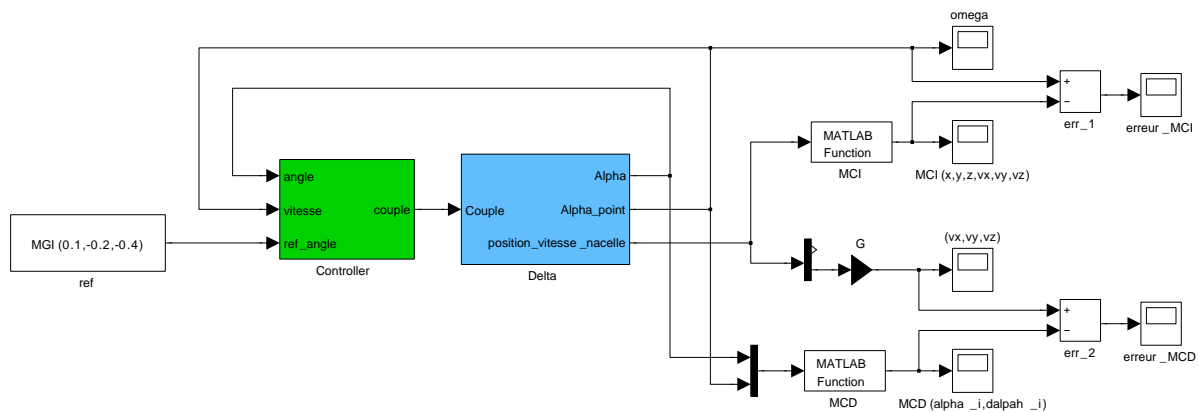


FIGURE IV.14: Validation du MCI et du MCD

Les erreurs relatives sont de l'ordre du 10^{-8} . Les MCD et MCI développés sont donc corrects.

4 Modèle Dynamique

Dans tout système physique, un bon modèle dynamique peut suffire à caractériser et à représenter complètement le fonctionnement de celui-ci. Pour une structure robotisée, l'utilisation du modèle dynamique est primordiale dans les étapes de conception et de commande du robot.

Dans le cas des robots manipulateurs, les équations dynamiques ou modèle dynamique donnent la relation entre l'effort des actionneurs, les accélérations et les trajectoires du mouvement résultant de l'organe terminal.

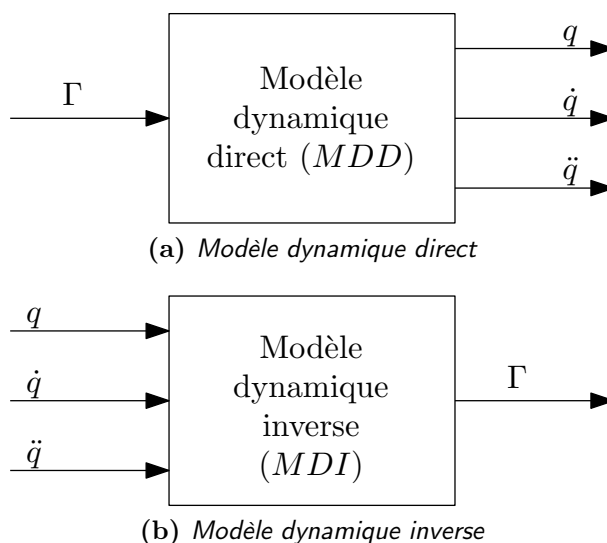
Le modèle dynamique d'un robot exprime cette relation dans un sens ou dans l'autre. C'est pour cela que nous avons deux cas : Modèle dynamique direct et Modèle dynamique inverse.

1. Modèle dynamique direct (figure IV.15a)

Exprime la variation des positions, vitesses et accélérations articulaires ($\alpha, \dot{\alpha}, \ddot{\alpha}$) résultants d'un certain effort exercé sur les actionneurs Γ . Autrement dit, le MDD permet de prévoir le comportement du système (mouvement du robot) pour une force ou couple donnée exercée.

2. Modèle dynamique inverse (figure IV.15b)

Exprime la valeur de l'effort Γ à exercer sur les actionneurs afin d'obtenir un certain profil sur les positions, vitesses et accélérations articulaires ($\alpha, \dot{\alpha}, \ddot{\alpha}$). Le MDI permet donc de trouver la force ou le couple à exercer pour avoir une trajectoire désirée.

FIGURE IV.15: *Modèle dynamique*

Le modèle dynamique d'un robot manipulateur est utilisé dans différentes tâches et étapes de la conception à l'implémentation d'une commande.

Ce modèle permet ainsi dans une étape de conception de dimensionner la structure mécanique, les actionneurs, les capteurs... , il permet aussi de simuler le comportement de la structure avant sa construction et d'y appliquer une commande, enfin plusieurs algorithmes de commandes pouvant être appliqués se base sur ce modèle, le modèle dynamique permet ainsi de développer la commande.

Le MDD est utilisé afin d'introduire un modèle de simulation du système, ce modèle remplace ainsi le système réel et permet de faire différents tests en simulation prévoyant le comportement du robot.

Quand au MDI, il est souvent utilisé lors de la commande de robots rapides dont fait partie le Delta (méthodes de commande basés sur l'utilisation de l'à priori de couple) le MDI permet de prévoir à partir de la trajectoire désirée $[q_d, \dot{q}_d, \ddot{q}_d]$ les couples articulaires nécessaires. Cette utilisation sera illustrée dans le chapitre relatif à la synthèse de lois de commande (V).

4.1 Développement des équations dynamiques

Le développement du modèle dynamique d'une structure mécanique en général et du robot Delta en particulier requiert l'utilisation d'un formalisme de calcul adéquat.

Dans la plupart des cas on adopte soit le formalisme Lagrangien ou bien celui de Newton-Euler.

Formalisme Lagrangien Décrit le comportement d'un système en termes de travail et d'énergie. Ce formalisme est basé sur une conservation énergétique du système au cours du mouvement et se calcule dans l'espace des paramètres.

Formalisme de Newton-Euler Décrit le comportement d'un système en termes de forces et de moments. Ce formalisme est basé sur les lois de la dynamique appliquée à chaque solide du système séparément.

A noter qu'un modèle dynamique basé sur la méthode des travaux virtuel (Théorème de d'Alembert) a été développé par Clavel [Cla91] pour la modélisation dynamique du robot Delta.

Pour notre part nous allons nous baser sur le formalisme de Newton-Euler afin de développer le modèle dynamique du robot Delta à 3 degrés de libertés à actionnement rotatif. Avant cela, nous présentons un petit rappel théorique sur cette méthode.

4.1.1 Rappel sur la méthode de Newton-Euler

La formulation de Newton-Euler est basée sur les lois gouvernant la dynamique des corps rigides. La procédure est la suivante :

1. Ecrire les équations définissant les vitesses et accélérations angulaires et linéaires de chaque segment.
2. Ecrire les équations décrivant les forces et les moments de force exercés sur les segments successifs.

Les équations de Newton-Euler permettent donc de décrire la dynamique combinée de translation et de rotation d'un corps rigide. Si on prend un repère dont l'origine coïncide avec le centre de masse du corps, ces équations peuvent être exprimées sous la forme matricielle suivante :

$$\begin{pmatrix} \vec{F} \\ \vec{\tau} \end{pmatrix} = \begin{pmatrix} mI & 0 \\ 0 & J_c \end{pmatrix} \begin{pmatrix} \vec{\gamma} \\ \vec{\omega} \end{pmatrix} + \begin{pmatrix} 0 \\ \vec{\omega} \wedge J_c \vec{\omega} \end{pmatrix} \quad (\text{IV.41})$$

Avec :

\vec{F} = Force totale agissant sur le centre de masse

m = Masse du solide

I = Matrice identité

$\vec{\gamma}$ = Accélération du centre de masse

$\vec{\tau}$ = Moment agissant sur le centre de masse

J_c = Matrice d'inertie par rapport au centre de masse

$\vec{\omega}$ = Vecteur vitesse angulaire du solide

La méthode de Newton-Euler doit être appliquée sur chaque solide composant le système. Cela est fait par l'isolation du solide et la prise en compte des efforts de liaison ce qui permet de garder au final la validité du système complet.

4.1.2 Application au Delta

Nous avons vu précédemment que l'utilisation de la méthode de Newton-Euler requiert l'isolation de chaque solide qui compose le système. Cela se fait en introduisant les forces de liaisons nécessaires entre les solides. On leur appliquera ensuite les équations de Newton-Euler (IV.41).

La structure du Delta est décomposable en 3 parties :

- La nacelle
- Les avant bras du Delta : barre relié à l'axe de l'actionneur (il y'en a 3 qui seront traités de façon identique).
- Les bras ou couple de deux barres parallèles reliant les avant bras à la nacelle (il y'en a 3 qui seront traités de façon identique).

A noter que nous négligerons dans notre modélisation les effets de frottement et les éléments du robot sont considérés comme parfaitement rigides.

a. Newton-Euler sur la nacelle

En isolant la nacelle on fait apparaître les forces de liaison avec les couples de barres parallèles $[F_1, F_2, F_3]$. C'est ce qui est représenté dans la figure IV.16.

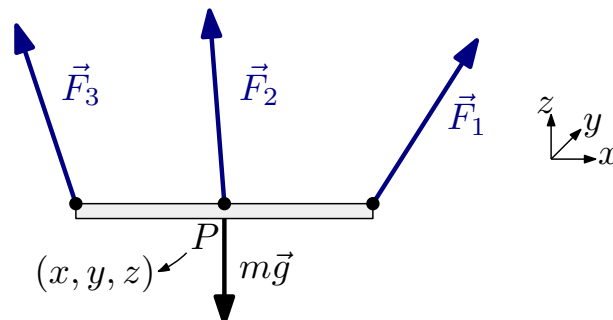


FIGURE IV.16: Isolation de la nacelle du Delta

De part la conception du robot Delta (système de barres parallèles) la nacelle ne peut faire que des mouvements de translation dans l'espace. Les équations de Newton-Euler se réduisent donc aux équations de la dynamique de translation (équation de Newton), en se rapportant au schéma IV.16 on a :

$$\vec{F}_1 + \vec{F}_2 + \vec{F}_3 + m\vec{g} = m\vec{\gamma} \quad (\text{IV.42})$$

Avec :

$$\vec{\gamma} = \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} \quad m\vec{g} = \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} \quad \vec{F}_i = \begin{pmatrix} F_i^x \\ F_i^y \\ F_i^z \end{pmatrix} = F_i \vec{u}_{F_i} \quad i = 1, 2, 3$$

Nous allons maintenant tirer les coordonnées des vecteurs directeurs \vec{u}_{F_i} des forces \vec{F}_i . La figure IV.17 présente l'orientation de la force \vec{F}_i par rapport au bras du Delta.

A noter que les deux barres parallèles sont représentées comme étant une seule barre située au milieu.

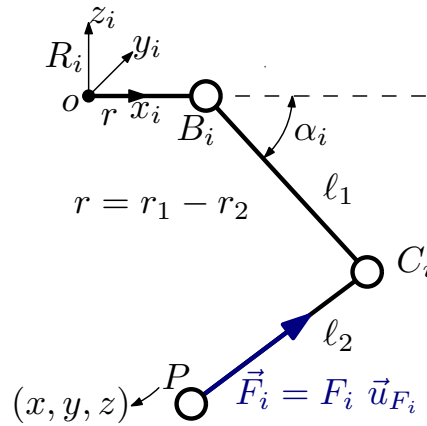


FIGURE IV.17: Force de liaison \vec{F}_i

On a :

$$\vec{F}_i = F_i \frac{\overrightarrow{PC_i}}{|\overrightarrow{PC_i}|} = F_i \vec{u}_{F_i} \quad \vec{u}_{F_i} = \frac{\overrightarrow{PC_i}}{|\overrightarrow{PC_i}|} = \begin{pmatrix} u_i^x \\ u_i^y \\ u_i^z \end{pmatrix} \quad (\text{IV.43})$$

Dans le repère R_0 en prenant en compte l'équation (IV.3) :

$$\overrightarrow{PC_i} = \begin{pmatrix} x_{c_i} - x \\ y_{c_i} - y \\ z_{c_i} - z \end{pmatrix} = \begin{pmatrix} (r + l_1 \cos \alpha_i) \cos \varphi_i - x \\ (r + l_1 \cos \alpha_i) \sin \varphi_i - y \\ -l_1 \sin \alpha_i - z \end{pmatrix} \quad (\text{IV.44})$$

Avec :

$$|\overrightarrow{PC_i}| = l_2$$

Finalement :

$$\vec{u}_{F_i} = \begin{pmatrix} u_i^x \\ u_i^y \\ u_i^z \end{pmatrix} = \begin{pmatrix} \frac{1}{l_2} [(r + l_1 \cos \alpha_i) \cos \varphi_i - x] \\ \frac{1}{l_2} [(r + l_1 \cos \alpha_i) \sin \varphi_i - y] \\ -\frac{1}{l_2} [l_1 \sin \alpha_i + z] \end{pmatrix} \quad (\text{IV.45})$$

En reprenant l'équation de Newton-Euler (IV.42) on a :

$$F_1 \begin{pmatrix} u_1^x \\ u_1^y \\ u_1^z \end{pmatrix} + F_2 \begin{pmatrix} u_2^x \\ u_2^y \\ u_2^z \end{pmatrix} + F_3 \begin{pmatrix} u_3^x \\ u_3^y \\ u_3^z \end{pmatrix} = \begin{pmatrix} m\ddot{x} \\ m\ddot{y} \\ m(\ddot{z} + g) \end{pmatrix} \quad (\text{IV.46})$$

La direction des force étant complétement définie par l'équation (IV.43), il nous reste à définir les modules F_1 , F_2 et F_3 . De l'équation précédente (IV.43) :

$$U \begin{pmatrix} F_1 \\ F_2 \\ F_3 \end{pmatrix} = \begin{pmatrix} m\ddot{x} \\ m\ddot{y} \\ m(\ddot{z} + g) \end{pmatrix} \quad \text{Avec : } U = \begin{pmatrix} u_1^x & u_2^x & u_3^x \\ u_1^y & u_2^y & u_3^y \\ u_1^z & u_2^z & u_3^z \end{pmatrix} \quad (\text{IV.47})$$

Finalement :

$$\begin{pmatrix} F_1 \\ F_2 \\ F_3 \end{pmatrix} = U^{-1} \begin{pmatrix} m\ddot{x} \\ m\ddot{y} \\ m(\ddot{z} + g) \end{pmatrix} \quad (\text{IV.48})$$

A noter que Les accélérations $(\ddot{x}, \ddot{y}, \ddot{z})$ sont obtenue par dérivation numérique.

On passe maintenant à l'application des équations de Newton-Euler sur le couple de barres parallèles.

Remarque A la masse (m) de la nacelle on doit ajouter une partie de la masse des barres parallèles, cela sera expliqué dans le paragraphe suivant.

b. Newton-Euler sur les barres parallèles

Nous avons considéré une seule barre prise au milieu des deux barres parallèle, en isolant cette barre on obtient le schéma donné sur la figure IV.18.

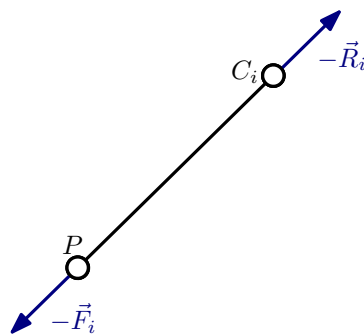


FIGURE IV.18: Isolation de la barre représentant les barres parallèles

Le modèle dynamique que nous avons utilisé en premier lieu est celui proposé par [Cod91], celui-ci admet quelques hypothèses simplificatrices :

- Les barres parallèles sont considérées de masse négligeable. Cependant, la masse réelle de ces deux barres est répartie sur les deux points P et C_i (une partie sur la nacelle et l'autre sur le bras).
- Le moment d'inertie des avant bras est considéré comme nul.

Les simplifications précédentes nous permettent d'écrire :

$$\vec{F}_i = -\vec{R}_i \quad (\text{IV.49})$$

c. Newton-Euler sur le Bras

La figure IV.19 montre l'isolation du bras rattaché à l'arbre de l'actionneur.

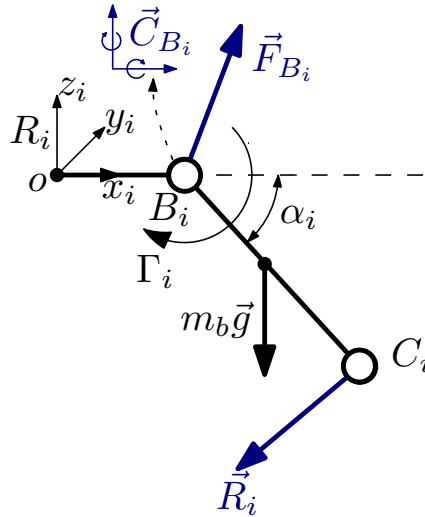


FIGURE IV.19: Isolation d'un des trois bras du Delta

Il nous reste à présent à mettre en relation le couple moteur Γ_i avec les autres variables du modèle dynamique (α_i , $\dot{\alpha}_i$ et $\ddot{\alpha}_i$), il suffit pour cela d'appliquer l'équation d'Euler sur le bras du robot. Le fait que nous ayons une rotation simple autour de y_i nous permet d'appliquer l'équation d'Euler directement en B_i sans passer par le centre de gravité :

$$\overrightarrow{B_i C_i} \wedge \vec{R}_i + \overrightarrow{B_i G_i} \wedge m_b \vec{g} + \vec{\Gamma}_i + \vec{C}_{B_i} = J_{B_i} \vec{\alpha}_i + \vec{\alpha}_i \wedge J_{B_i} \vec{\alpha}_i \quad (\text{IV.50})$$

avec :

$$\overrightarrow{B_i C_i} = \begin{pmatrix} \ell_1 \cos \alpha_i \\ 0 \\ -\ell_1 \sin \alpha_i \end{pmatrix} \quad \overrightarrow{B_i G_i} = \begin{pmatrix} \frac{\ell_1}{2} \cos \alpha_i \\ 0 \\ -\frac{\ell_1}{2} \sin \alpha_i \end{pmatrix}$$

$$\vec{R}_i = -\vec{F}_i = -F_i \begin{pmatrix} \frac{1}{\ell_2} [r + \ell_1 \cos \alpha_i - x_{R_i}] \\ -\frac{y_{R_i}}{\ell_2} \\ -\frac{1}{\ell_2} [\ell_1 \sin \alpha_i + z_{R_i}] \end{pmatrix} \quad (\text{IV.51})$$

$$J_{B_i} = \begin{pmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & \frac{m_b \ell_1^2}{3} & 0 \\ -I_{xz} & 0 & I_{zz} \end{pmatrix}$$

$$\vec{\Gamma}_i = \begin{pmatrix} 0 \\ \Gamma_i \\ 0 \end{pmatrix} \quad \vec{C}_{B_i} = \begin{pmatrix} C_{B_i}^x \\ 0 \\ C_{B_i}^z \end{pmatrix}$$

En tenant compte de (IV.51), l'équation (IV.50) devient :

$$\begin{pmatrix} F_i \frac{\ell_1}{\ell_2} y_{R_i} \sin \alpha_i + C_{B_i}^x \\ -F_i \frac{\ell_1}{\ell_2} z_{R_i} \cos \alpha_i - F_i \frac{\ell_1}{\ell_2} (x_{R_i} - r) \sin \alpha_i + m_b g \ell_g \cos \alpha_i + \Gamma_i \\ F_i \frac{\ell_1}{\ell_2} y_{R_i} \cos \alpha_i + C_{B_i}^z \end{pmatrix} = \begin{pmatrix} 0 \\ \ddot{\alpha}_i I_{yy}^{B_i} \\ 0 \end{pmatrix} \quad (\text{IV.52})$$

ou :

$$I_{yy}^{B_i} = \frac{m_b \ell_1^2}{3} = I_b$$

Le deuxième élément de l'équation (IV.52) est le seul élément utile, il nous permettra de tirer les modèles dynamiques direct et inverse, selon que l'on cherche à déterminer le profil de mouvement du Delta ($\ddot{\alpha}_i$, $\dot{\alpha}_i$ et α_i) ou les couples Γ_i à fournir par les moteurs :

$$-F_i \frac{\ell_1}{\ell_2} [z_{R_i} \cos \alpha_i + (x_{R_i} - r) \sin \alpha_i] + m_b g \ell_g \cos \alpha_i + \Gamma_i = \ddot{\alpha}_i I_b \quad i = 1, 2, 3 \quad (\text{IV.53})$$

Remarques

1. L'équation (IV.50) est écrite dans le repère intermédiaire R_i , cela a pour effet de simplifier les calculs.
2. Les coordonnées de la nacelle dans ce repère sont donc $(x_{R_i}, y_{R_i}, z_{R_i})$.
3. L'équation de Newton n'a pas été développée car elle n'a aucune utilité dans notre cas. On ne s'intéresse pas aux efforts de liaison en B (F_{B_i} et C_{B_i}). Comme c'est le cas pour les deux équations non considérées dans (IV.52).
4. Le moment d'inertie I_b doit être corrigé par l'ajout d'un terme représentant la contribution d'une partie de la masse des deux barres parallèles (et de la masse de l'articulation C_i) selon la simplification citée précédemment dans le cadre de l'application de l'équation de Newton-Euler sur ces barres.
5. La longueur ℓ_g est la distance entre le point B_i et le centre de gravité de l'ensemble bras + masses additionnelles (une partie de la masse des avant bras en plus de la masse de l'articulation C_i).

Nous passons maintenant à l'utilisation des équations dynamiques (IV.53) afin de trouver le modèle dynamique direct et inverse du robot Delta.

4.2 Modèle dynamique direct

Dans notre cas le développement du modèle dynamique direct permet de lier chaque couple moteur Γ_i aux accélérations articulaire ($\ddot{\alpha}_i$) tout en prenant en compte les vitesses et les positions ($\dot{\alpha}_i, \alpha_i$). La figure IV.20 illustre la relation que l'on doit trouver.



FIGURE IV.20: MDD du delta

On note que la seule connaissance des Γ_i permet de trouver les variables $(\ddot{\alpha}_i, \dot{\alpha}_i, \alpha_i)$. En fait le modèle dynamique direct du robot delta permet d'écrire la relation :

$$\ddot{\alpha}_i = f_i [\dot{\alpha} , \alpha , \Gamma] \quad i = 1, 2, 3 \quad (\text{IV.54})$$

Avec :

$$\begin{cases} \alpha = [\alpha_1 \ \alpha_2 \ \alpha_3]^T \\ \dot{\alpha} = [\dot{\alpha}_1 \ \dot{\alpha}_2 \ \dot{\alpha}_3]^T \\ \Gamma = [\Gamma_1 \ \Gamma_2 \ \Gamma_3]^T \end{cases}$$

Connaissant les conditions initiales du système, on peut démarrer l'évolution de la dynamique du robot.

L'intégration successive de la formule (IV.54) et la connaissance de la variation de la commande en couple Γ_i permet de définir donc complètement le profil de trajectoire $[\ddot{\alpha}_i, \dot{\alpha}_i, \alpha_i]$ produit.

Nous avons donc :

$$\begin{cases} \ddot{\alpha}_1 = f_1 [\dot{\alpha} , \alpha , \Gamma] \\ \ddot{\alpha}_2 = f_2 [\dot{\alpha} , \alpha , \Gamma] \\ \ddot{\alpha}_3 = f_3 [\dot{\alpha} , \alpha , \Gamma] \end{cases} \quad \text{avec :} \quad \dot{\alpha}_i = \dot{\alpha}_i^0 + \int \ddot{\alpha}_i \quad \text{et} \quad \alpha_i = \alpha_i^0 + \int \dot{\alpha}_i \quad (\text{IV.55})$$

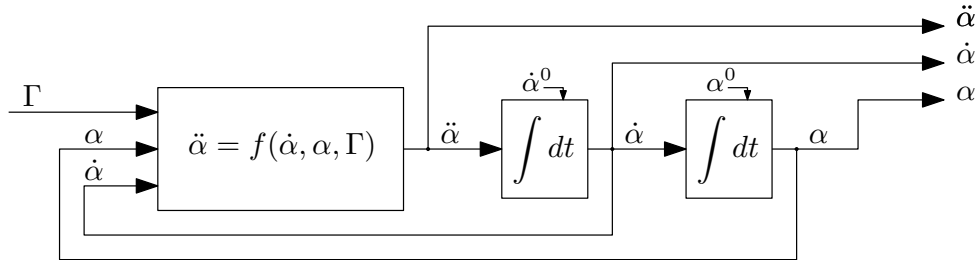
Des équations dynamiques (IV.53) on trouve le modèle dynamique direct donnée par les 3 équations différentielles d'ordre 2 suivantes :

$$\begin{cases} \ddot{\alpha}_1 = \frac{1}{I_b} \left(-F_1 \frac{\ell_1}{\ell_2} [z_{R_1} \cos \alpha_1 + (x_{R_1} - r) \sin \alpha_1] + m_b g \ell_g \cos \alpha_1 + \Gamma_1 \right) \\ \ddot{\alpha}_2 = \frac{1}{I_b} \left(-F_2 \frac{\ell_1}{\ell_2} [z_{R_2} \cos \alpha_3 + (x_{R_2} - r) \sin \alpha_2] + m_b g \ell_g \cos \alpha_3 + \Gamma_2 \right) \\ \ddot{\alpha}_3 = \frac{1}{I_b} \left(-F_3 \frac{\ell_1}{\ell_2} [z_{R_3} \cos \alpha_5 + (x_{R_3} - r) \sin \alpha_3] + m_b g \ell_g \cos \alpha_5 + \Gamma_3 \right) \end{cases} \quad (\text{IV.56})$$

Il faut cependant noter que :

- Les F_i sont fonctions des variables articulaires $[\ddot{\alpha}_i, \dot{\alpha}_i, \alpha_i]$. Cette fonction est non explicite car sa formule analytique est difficile à trouver. On préfère un calcul numérique en utilisant l'équation (IV.48) (contient une dérivation numérique).
- Les x_{R_i} et z_{R_i} sont aussi fonctions des variables articulaires et plus précisément des coordonnées articulaires α_i . On les obtient facilement à partir du MGD (en plus d'une rotation vers R_i).

Le modèle développé précédemment (équations IV.56) représente les équations du modèle dynamique direct du robot Delta, il peut être facilement implémenté comme modèle de simulation du robot (voir le schéma IV.21), c'est d'ailleurs ce qui est fait dans la plupart des cas (ce n'est cependant pas le cas pour nous).



$$\alpha = [\alpha_1 \alpha_2 \alpha_3]^T \quad \Gamma = [\Gamma_1 \Gamma_2 \Gamma_3]^T \quad f = [f_1 f_2 f_3]^T \quad (\alpha^0, \dot{\alpha}^0) \text{ valeur initiale de } (\alpha, \dot{\alpha})$$

FIGURE IV.21: Schéma de simulation du MDD

L'implémentation de ce modèle a été faite sous Simulink, en utilisant les fonctions du MCD, du MGD et deux autres fonctions MATLAB (l'une pour le calcul des forces F_i et l'autre pour le calcul des accélérations $\ddot{\alpha}_i = f_i[\dot{\alpha}, \alpha, \Gamma]$). La figure IV.22 illustre le schéma Simulink réalisé.

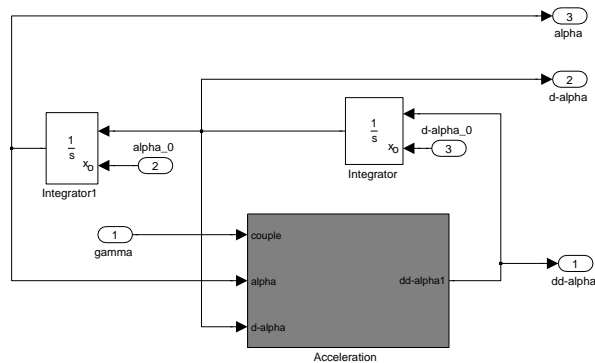


FIGURE IV.22: Simulation du modèle analytique du Delta sous simulink

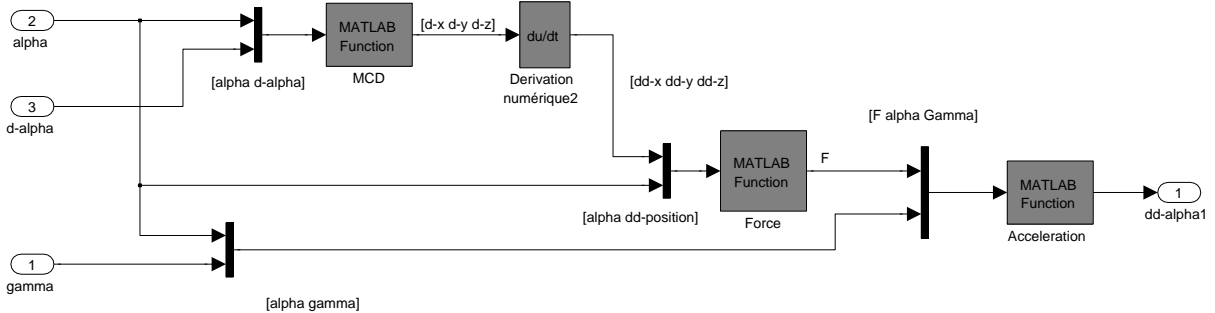


FIGURE IV.23: Détails du bloc Accélération du MDD précédent

4.3 Validation du MDD

Pour la validation du MDD, nous avons comparé les résultats qu'il donnait avec les résultats du modèle SimMechanics du Delta, les consignes articulaires sont bien évidemment les mêmes pour les deux blocs (correspondants à une trajectoire en segment de droite selon l'axe x), le schéma Simulink utilisé pour cette comparaison est donné sur la figure IV.24.

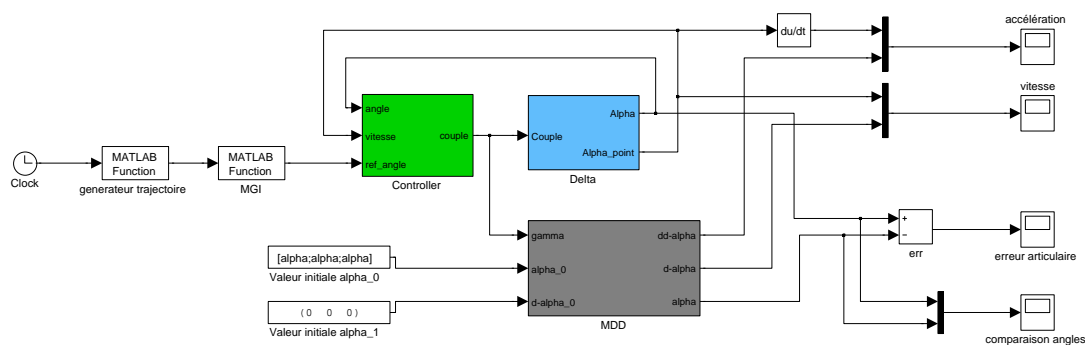


FIGURE IV.24: Schéma Simulink utilisé pour la validation du MDD

Les résultats de validation sont illustrés dans la figure IV.25.

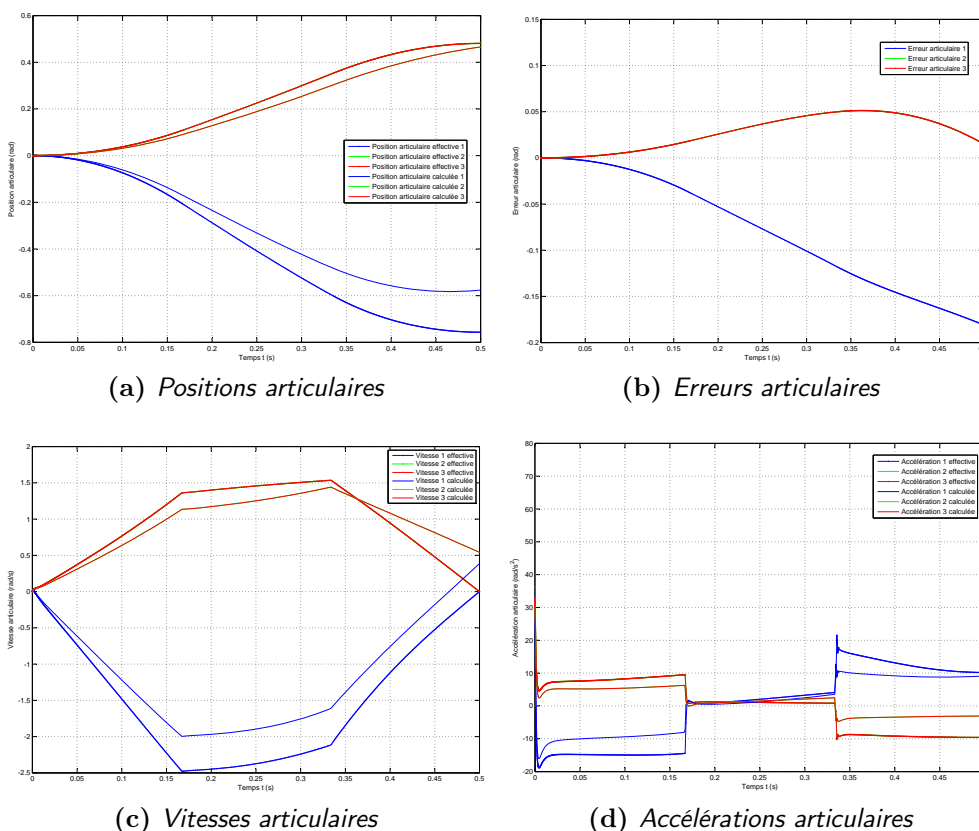


FIGURE IV.25: Résultats de validation du MDD

4.4 Modèle dynamique inverse

Dans le modèle dynamique inverse du Delta, on cherche à trouver la commande en couple Γ_i nécessaire pour obtenir un profil de déplacement souhaité, défini par les variables $[\ddot{\alpha}_i, \dot{\alpha}_i, \alpha_i]$. La figure IV.26 illustre le rôle du MDI.

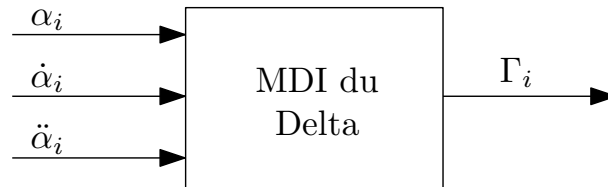


FIGURE IV.26: Schéma Simulink utilisé pour la validation du MDD

Les couples à fournir Γ_i se déduisent directement des relations (IV.53) :

$$\Gamma_i = \ddot{\alpha}_i I_b + F_i \frac{\ell_1}{\ell_2} [z_{R_i} \cos \alpha_i + (x_{R_i} - r) \sin \alpha_i] - m_b g \ell_g \cos \alpha_i \quad i = 1, 2, 3 \quad (IV.57)$$

La même remarque est à faire concernant le calcul des F_i , des x_{R_i} et des z_{R_i} (c.f section 4.2).

L'équation (IV.57) caractérise donc entièrement le modèle dynamique inverse.

L'implémentation de ce modèle a été faite sous Simulink, en utilisant les fonctions du MCD, du MGD et deux autres fonctions MATLAB (l'une pour le calcul des forces F_i et l'autre pour le calcul des couples Γ_i). La figure IV.27 illustre le schéma Simulink réalisé.

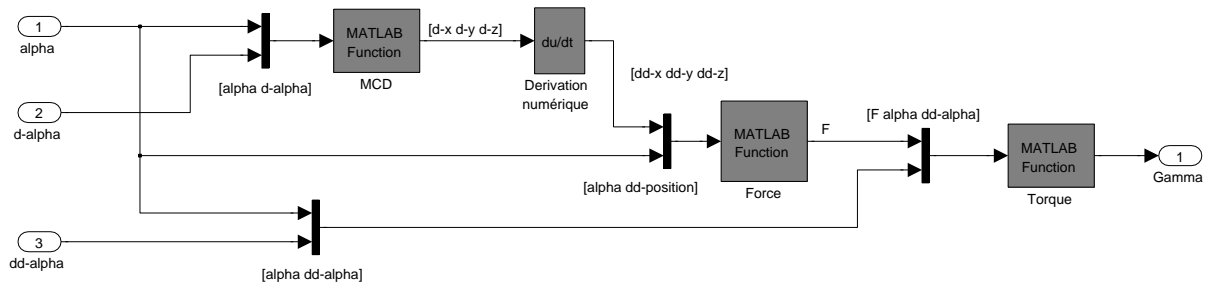


FIGURE IV.27: Implémentation du MDI du delta sous Simulink

4.5 Validation du MDI

Afin de valider le modèle dynamique inverse trouvé, le schéma Simulink donné sur la figure IV.28 est adopté.

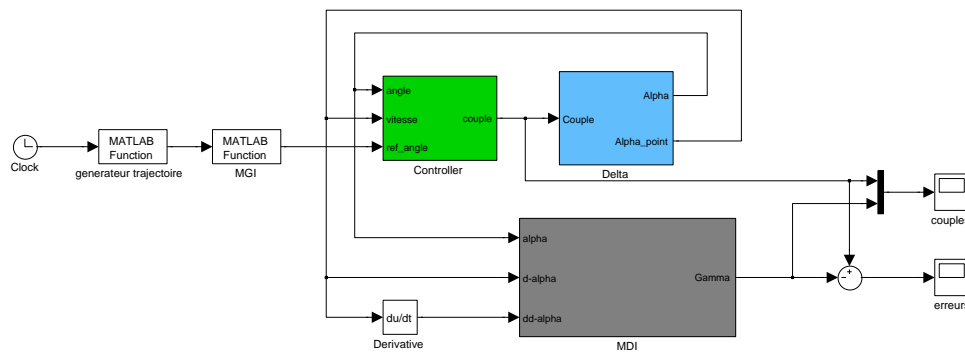


FIGURE IV.28: Schéma Simulink utilisé pour la validation du MDI

Dans ce schéma de validation nous comparons les couples injectés comme consignes au Delta (modèle SimMechanics) avec les couples calculés par le modèle dynamique inverse à partir des mesures de trajectoire sur le modèle SimMechanics.

Les couples mis en œuvre et les couples calculés sont donnés sur les tracés de la figure IV.29.

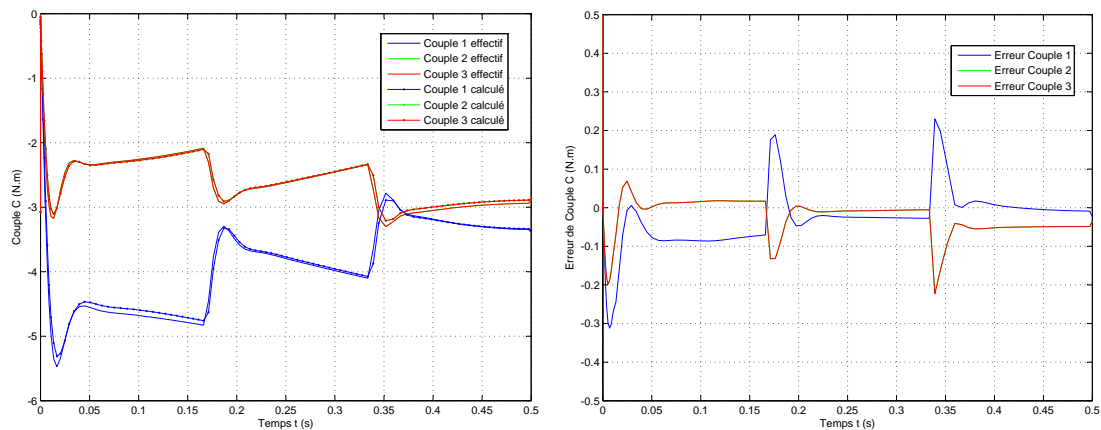


FIGURE IV.29: Résultats Validation du MDI

L'erreur maximale est de l'ordre de 10,8 %¹. Cet erreur paraît tout à fait acceptable vue les approximations que nous avons prises lors de la modélisation dynamique du Delta.

Nous avons négligé les inerties et les masses des barres parallèles (avant bras), cette approximation a eu pour conséquence de simplifier grandement les calculs. Ces masses ont été reportés sur la nacelle (deux tiers de la masse des avant bras) et sur les bras (un tiers de la masse des avant bras) comme cela a été fait dans [Cod91], ces résultats nous permettent de nous rendre compte du degrés de validité de cette hypothèse.

1. Erreur RMS relative par rapport à la valeur RMS des couples.

Commande du robot Delta

V

1	Introduction	55
2	Objectifs	56
3	Génération de mouvement et de trajectoire	56
3.1	Lois de mouvement	57
3.2	Trajectoires	60
4	Commande en boucle ouverte avec MDI à priori	63
5	Synthèse de lois de commande	66
5.1	Architectures de commandes	66
5.2	Conditions d'instrumentation et d'acquisition	68
5.3	Commande PID	69
5.4	Double boucle PID vitesse - position	73
5.5	Introduction du modèle dynamique	74
6	Tests de performances	75
6.1	Cadence de prise-dépose	76
6.2	Trajectoire de prise dépose	80
6.3	Robustesse par rapport à la charge transportée	84
6.4	Robustesse par rapport aux perturbations sur le positionnement	86
6.5	Robustesse des performances dans le volume de travail	86
7	Commande par apprentissage itératif (ILC)	90
7.1	Présentation de la commande	90
7.2	Application de la commande sur le Delta	92
7.3	Commentaires et conclusion	97

1 Introduction

Dans le cadre d'un usage industriel du robot Delta, l'élaboration d'une commande performante reste certainement la partie la plus importante dans la programmation d'une tâche donnée, cela est d'autant plus vrai pour le cas d'un robot rapide et précis tel que le Delta. La viabilité de la commande est conditionnée par une modélisation précise du robot, cela explique le fait que l'on ait consacré une partie assez importante de notre travail à la modélisation du robot. A présent, nous allons mettre à contribution les modèles élaborés dans le cahpitre précédent pour la mise au point de commande adaptées aux applications du Delta.

2 Objectifs

Le concept du robot Delta est né d'un besoin industriel très précis : la prise et le dépôt d'objets légers à des cadences élevés. Ce cahier de charge indique clairement que les objectifs de la commande sont au nombre de deux :

- Un positionnement précis lors des étapes de prise et de dépose ;
- La poursuite d'une trajectoire définie, lors de l'étape de manutention et déplacement de la pièce, tout en restant dans l'espace de travail du robot et en prenant en compte les contraintes environnementales (obstacles, plancher...).

L'étude et l'implémentation des commandes qui suivront, se fera donc en ayant comme objectifs ces deux aspects, le positionnement et la poursuite d'une trajectoire.

3 Génération de mouvement et de trajectoire

Dans le cadre de la robotique, la partie chargée de commander le robot a pour objectif de générer les consignes nécessaires (en terme de couple pour chaque moteur) en fonction des positions, vitesses et accélérations que l'on souhaite donner à l'organe terminal, pour que ce dernier (situé sur la nacelle pour le cas du Delta) suive une trajectoire désirée, constituée par une suite ou un certain nombre de points spatiaux. Suivant les tâches que doit accomplir le robot, les contraintes environnementales et le type du robot, plusieurs types de trajectoires peuvent être envisagés :

- Mouvement entre deux points sans asservissement de trajectoire ;
- Mouvement entre deux points avec suivi de trajectoire ;
- Mouvement passant par plusieurs points intermédiaire avec suivi de trajectoire ;

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 (exemple trapézoïdale) afin de contrôler le temps de parcours et synchroniser les mouvements articulaires entre eux (figure [V.1a](#)).

Le problème avec ce genre de commande est que l'on ne contrôle pas la trajectoire de l'organe terminal. Son mouvement n'est qu'une conséquence des mouvements articulaires commandés. L'utilisation d'une telle commande est donc impossible pour des tâches de prise-dépose, souvent faites dans des environnements industriels encombrés. Cela est accentué par le fait que le robot Delta, avec la dépendance qui existe entre ses trois chaînes cinématiques, possèdent des mouvements difficilement prévisibles (en connaissant les mouvements articulaires) au contraire des robots séries.

La commande de positionnement a pour avantage de s'affranchir du calcul en ligne des transformations de coordonnées et de contrôler parfaitement la dynamique des actionneurs.

Aussi, nous utiliserons une telle commande comme étape préliminaire au développement du second type de commande plus élaborée et plus adapté au cas du Delta.

b. Commande de suivi de trajectoire

Pour ce deuxième type de commande, la génération de trajectoire se fait dans l'espace opérationnel. Une loi de mouvement doit cependant être greffée à la trajectoire de référence (figure V.1b)

Cette commande à l'avantage indéniable de commander le robot dans l'espace opérationnel, le mouvement et le positionnement de l'organe terminal est alors parfaitement connu et directement commandé.

Une telle commande nécessite néanmoins le calcul en ligne des transformations de coordonnées. Outre cela, les dynamiques de la commande des actionneurs ne sont plus contrôlées, leurs évolutions ne sera qu'une conséquence du mouvement qu'on impose à l'organe terminal (forme de la trajectoire, loi de mouvement, accélération maximale, position de la nacelle). On veillera néanmoins à vérifier cette dynamique afin qu'elle soit admissible par les actionneurs.

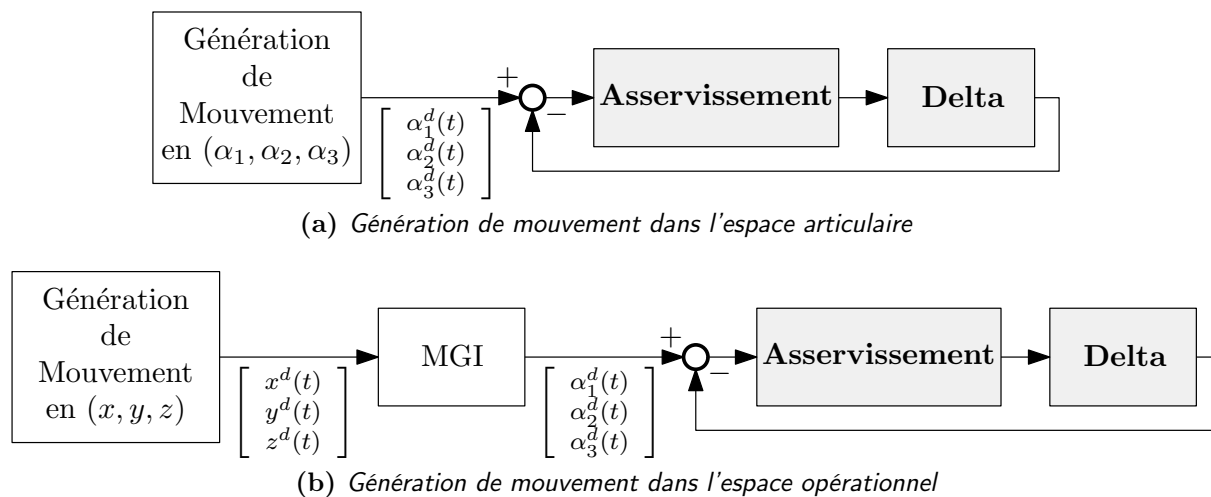


FIGURE V.1: Génération de mouvement

3.1 Lois de mouvement

Que ça soit pour la génération de trajectoire dans l'espace articulaire ou opérationnel, afin de prendre en compte l'aspect temporel du mouvement (évolution de la trajectoire en fonction du temps), une loi de mouvement doit être greffée à la trajectoire. Cette loi de mouvement peut s'appliquer directement sur les variables articulaires (cas de la génération

de mouvement dans l'espace articulaire) ou alors sur la trajectoire opérationnelle (par le biais de la variable curviligne s).

En contrôlant le mouvement du robot dans le domaine temporel, on peut contrôler les vitesses et accélérations de celui-ci, au contraire d'une commande en échelon.

a. Loi Bang-Bang

Cette loi de mouvement présente une accélération constante dans un intervalle $t_f/2$ puis une décélération constante pour la même durée (t_f temps de parcours).

La figure V.2 représente la position, vitesse et accélération correspondant à une telle loi de mouvement.

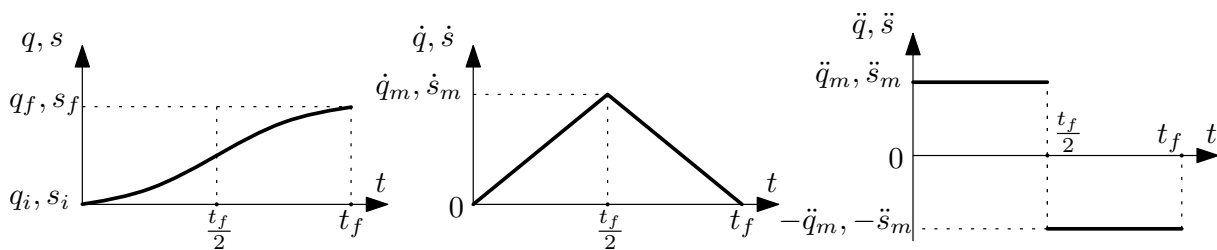


FIGURE V.2: Loi de mouvement Bang-bang

Cette loi de mouvement n'est cependant pas optimale en termes de temps de parcours et d'exploitation des possibilités des actionneurs car en imposant le temps et la distance de parcours, on n'a plus la possibilité d'imposer la vitesse et accélération maximales, or, il est préférable d'utiliser les moteurs au maximum de leur capacité (couple nominal). Nous ne détaillerons donc pas plus cette loi de mouvement.

b. 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 V.3 représente la position, vitesse et accélération correspondant à une loi de mouvement en trapèze.

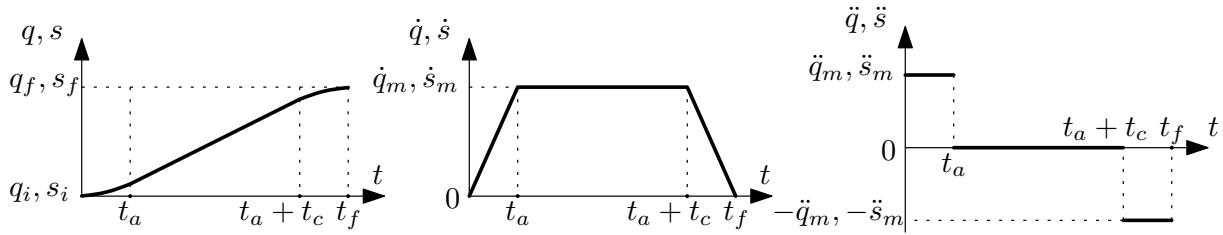


FIGURE V.3: Loi de mouvement en Trapèze

Ou :

- 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 parcouru ;
- \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 :

$$\left\{ \begin{array}{l} t_f = 2t_a + t_c \\ \dot{s}_m = \ddot{s}_m t_a \\ s_f = s_i + D \\ s(t) = s_i + \ddot{s}_m \cdot \left(\frac{t^2}{2}\right) \quad \text{pour : } t \leq t_a \\ s(t) = s_i + \ddot{s}_m \frac{t_a^2}{2} + \dot{s}_m \cdot (t - t_a) \quad \text{pour : } t_a < t \leq t_a + t_c \\ s(t) = s_i + \ddot{s}_m \frac{t_a^2}{2} + \dot{s}_m (t - t_a) - \ddot{s}_m \cdot \frac{(t - t_a - t_c)^2}{2} \quad \text{pour : } t_a + t_c < t \leq t_f \end{array} \right. \quad (\text{V.1})$$

Ces équations sont aussi valables pour la variable articulaire q .

Les équations (V.1) nous permettent d'avoir un libre choix de fixer les paramètres de la loi de mouvement selon :

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

Pour notre part nous utiliserons la loi de mouvement en Trapèze comme base aux différentes trajectoires que nous allons implémenter.

3.2 Trajectoires

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 (figure V.1a). 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 V.4). Le modèle géométrique inverse est ensuite appliqué pour avoir les références articulaires à injecter aux actionneurs (figure V.1b) .

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.

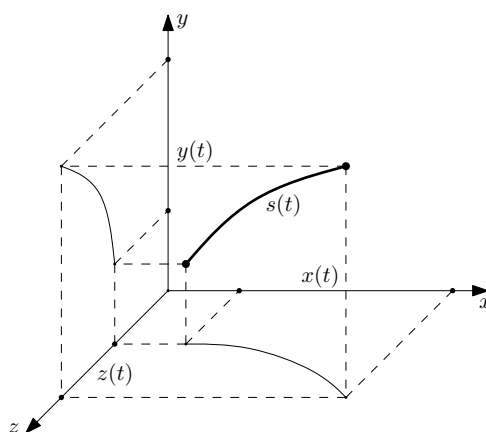


FIGURE V.4: Projection de la coordonnée curviligne $s(t)$ donnant $x(t)$, $y(t)$, $z(t)$

a. Trajectoire rectiligne

La génération de trajectoire rectiligne 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 trajectoires nous permettront 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 V.5).

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).

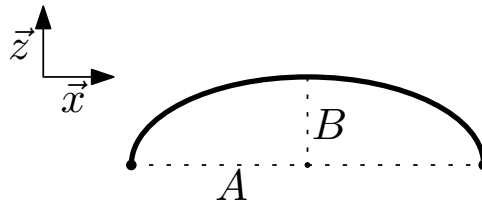


FIGURE V.5: Trajectoire ellipsoïde

Une telle trajectoire ne peut pas être utilisée pour une tâche de prise-dépose possédant des hauteurs différentes entre le point de départ et le point d'arrivée, des segments de droites peuvent être rajoutés aux deux bouts de la demi ellipse pour remédier à ça, mais cela introduirait une variation de courbure importante aux points de jonction. Néanmoins, cette trajectoire reste très intéressante d'un point de vue dynamique puisqu'elle ne présente pas de discontinuité de courbure.

c. Trajectoire de prise-dépose en segments de droites

Trois segments de droite sont utilisés afin de construire cette trajectoire de prise-dépose. Deux segments selon l'axe z (H_1, H_2) et un segments intermédiaire selon l'axe x (D) (figure V.6).

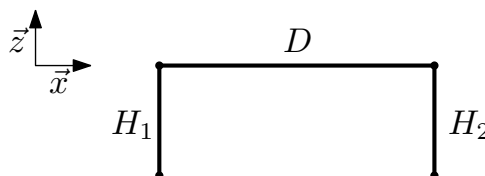


FIGURE V.6: Trajectoire en segments de droites

Cette trajectoire peut être utilisée pour une prise-dépose à hauteur différente, par contre elle présente des discontinuités de courbure qui nous amène à arrêter le robot

au niveau des points de jonction, ce qui fait perdre le robot en rapidité et rend cette trajectoire non optimale.

d. Trajectoire de prise-dépose en Clothoïdes

La clothoïde est une trajectoire caractérisée par une variation de courbure proportionnelle à l'abscisse curviligne (figure V.7).

La caractéristique principale de cette trajectoire est que l'accélération que subit un corps la parcourant à vitesse constante est continue. Une telle trajectoire est très intéressante pour les systèmes mécaniques devant changer de direction (exemple : véhicules) et est très utilisés dans le dessin des routes, des chemins de fer...

L'équation paramétrique définissant une telle trajectoire est :

$$2R(t)s(t) = A^2$$

Où R est le rayon de courbure, s l'abscisse curviligne et A le facteur de proportionnalité.

Elle est définie explicitement par rapport à la coordonnée curviligne $s(t)$ avec les intégrales de Fresnel :

$$\begin{cases} x(s) = A \int_0^s \frac{\cos v}{\sqrt{v}} dv \\ z(s) = A \int_0^s \frac{\sin v}{\sqrt{v}} dv \end{cases}$$

Ces intégrales n'ont pas de solution analytique, une solution utilisant le développement en série de Taylor est utilisé pour leur calcul numérique.

Pour une trajectoire de prise-dépose, la clothoïde va nous permettre de joindre les segments de droite d'une manière plus appropriée, l'accélération normale ne présente aucune discontinuité, ceci rend cette trajectoire très intéressante d'un point de vue dynamique. La trajectoire utilisée est illustrée sur la figure V.8.

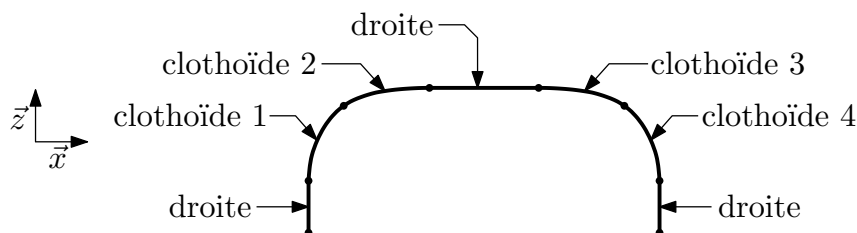


FIGURE V.8: Trajectoire de prise-dépose utilisant les clothoïdes

Cette dernière trajectoire permet une prise-dépose à hauteur différent et présente une continuité de la courbure ce qui améliorera la dynamique et la rapidité du robot.

4 Commande en boucle ouverte avec MDI à priori

La commande en boucle ouverte du robot nous permettra de :

- Valider encore une fois le modèle dynamique trouvé et mettre en évidence ses limites.
- Justifier la nécessité d'une commande en boucle fermée.

La commande du système en boucle ouverte nécessite l'utilisation du modèle dynamique inverse (MDI), le schéma de commande de la figure V.9 est alors adopté.

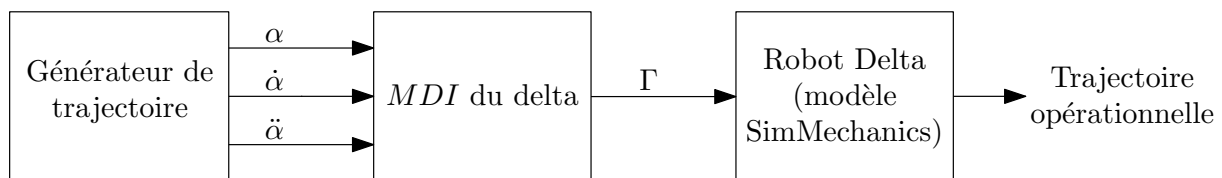


FIGURE V.9: Commande en boucle ouverte du robot Delta

Théoriquement si le modèle dynamique était parfait, le suivi de trajectoire devrait l'être aussi car le modèle dynamique direct du robot exprime les équations du mouvement naturel du robot :

$$\ddot{\alpha} = f(\Gamma, \alpha, \dot{\alpha})$$

Quand au modèle inverse, il est donné en théorie par :

$$\Gamma = f^{-1}(\ddot{\alpha}, \dot{\alpha}, \alpha)$$

La trajectoire donnée par le générateur devrait donc, dans le cas où le MDI était parfait, se retrouver à la sortie du robot, les deux systèmes MDI et le robot Delta étant l'inverse l'un de l'autre.

Le problème est que le modèle dynamique du robot (aussi réaliste soit-il) ne peut jamais être parfait, et ce pour plusieurs raisons :

- L'hypothèse simplificatrice prise lors du calcul du modèle dynamique analytique ;
- La mauvaise connaissance de certains phénomènes (qui n'ont pas été modélisés) comme les frottements présents au niveau des différentes liaisons ;
- Les perturbations qui peuvent agir sur le robot, comme la variation de masse de la charge à transporter.

Nous n'aurons donc pas un suivi optimal de trajectoire, de plus l'effet d'une perturbation ne pourra être contré. Afin d'explicitier ce qui a été dit, le schéma de simulation utilisé pour ce test est illustré par la figure V.10.

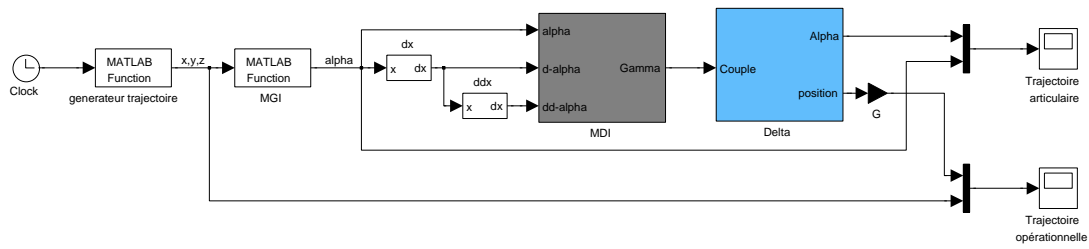


FIGURE V.10: Simulation commande en boucle ouverte du robot Delta

Le générateur de trajectoire a été configuré pour une trajectoire linéaire selon l'axe x (longueur de 15 cm réalisée en 0,15 s), avec un profil trapézoïdal en vitesse curviligne (dans l'espace opérationnel).

Les résultats de simulations sont donnés sur la figure V.12.

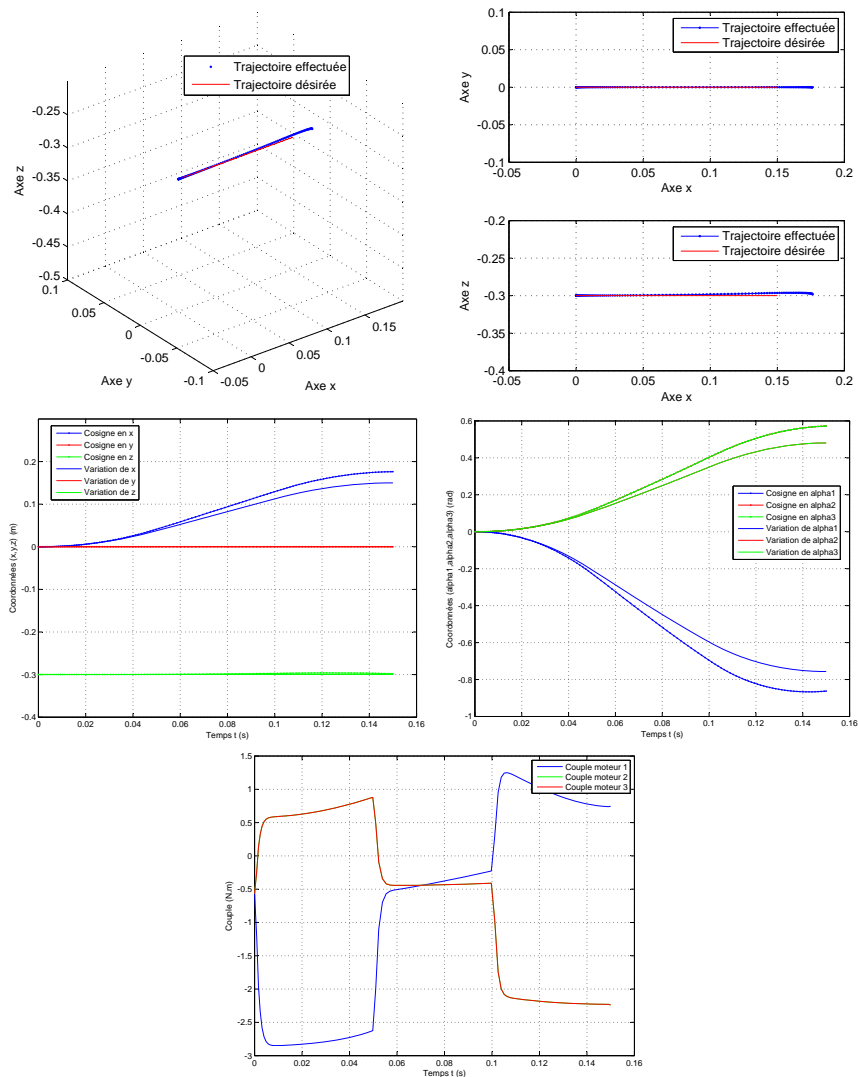


FIGURE V.11: Résultats de la commande en boucle ouvert

On voit clairement que le MDI se comporte plutôt bien, malgré les approximations prises lors de son calcul. Cela nous permet de valider encore plus notre modèle dynamique.

Si ces résultats sont suffisants pour une validation du modèle, ils ne le sont pas pour des objectifs de commande du robot. En effet, on constate la présence d'un dépassement important selon l'axe x . Ces résultats montrent que l'utilisation du robot Delta ne peut se faire en boucle ouverte.

Une rétroaction est donc nécessaire afin d'obtenir un meilleur suivi de trajectoire.

Aussi, et afin de confirmer cette nécessité, nous ajoutons une perturbation à notre système. Cette perturbation se présente sous la forme d'une charge additionnelle à la nacelle (représentant une pièce transportée de 200 g) non prise en compte dans le MDI, cette masse est introduite dans le modèle SimMechanics seulement.

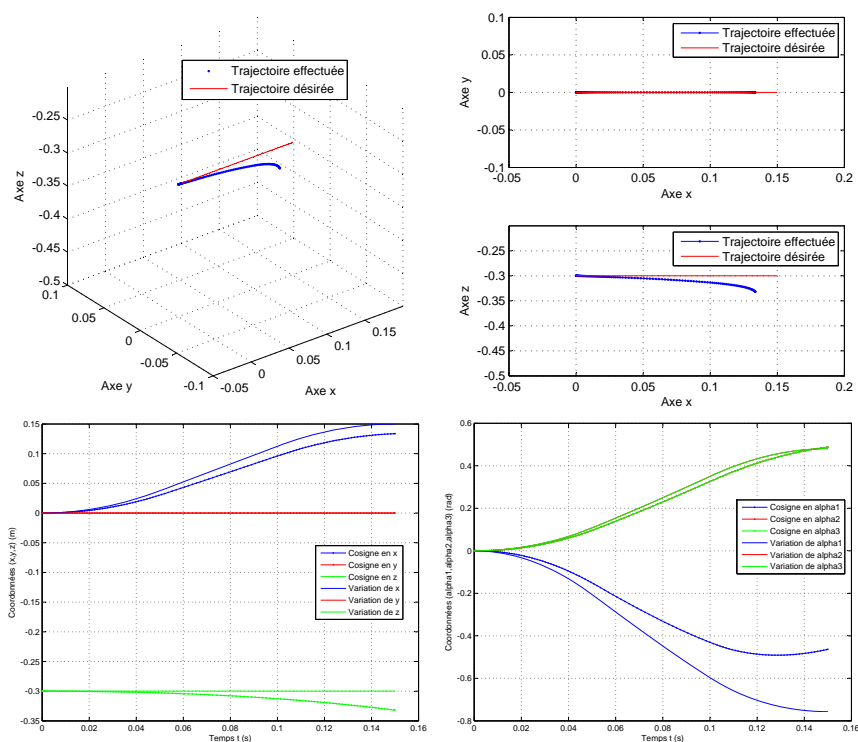


FIGURE V.12: Résultats avec la commande en boucle ouverte avec charge

La consigne de couple donnée au système n'a évidemment pas changée. La figure V.12 montre la dégradation du suivi de trajectoire.

Le test précédent n'était en fait qu'une simulation des conditions réelles dans lesquelles le robot sera utilisé. En effet, la masse de la charge à transporter n'est jamais connue avec précision de plus, celle-ci varie en fonction du temps. Dans de telles conditions, le robot nécessite donc bien d'être commandé en boucle fermée.

5 Synthèse de lois de commande

Dans cette partie nous allons essayer de présenter le cheminement qui va nous conduire à synthétiser un régulateur satisfaisant nos objectifs de commandes.

Cette synthèse se fera de façon progressive avec des améliorations graduelle du régulateur utilisé jusqu'à obtenir les résultats désirés.

5.1 Architectures de commandes

Dans le paragraphe précédent, nous avons mis en évidence la nécessité de commander le robot Delta en boucle fermée. Pour le cas des robots manipulateurs en général et du robot Delta en particulier, la présence de plusieurs actionneurs fonctionnant en concurrence implique que la commande doit agir sur chaque moteur (articulations actives du Delta), cette action du système de commande sur les moteurs peut se faire selon deux schémas :

- Schéma de commande de type centralisée ;
- Schéma de commande de type décentralisée.

Afin de commander notre robot, un choix de schéma de commande s'impose. Ce choix sera fait à la fin de cette section. Avant cela nous allons présenter les particularités de ces deux stratégies de commande.

5.1.1 Commande centralisées

Le robot Delta présente un fort couplage dynamique entre ses trois articulations. On pourrait alors se dire qu'une commande du type centralisée serait un choix judicieux afin de prendre en compte ces couplages. Dans ce cas là, la commande des articulations est appliquée en considérant un système MIMO.

La difficulté réside dans la synthèse de lois de commandes sur de tels systèmes. De plus, comme vu dans le chapitre sur la modélisation, le modèle dynamique du robot Delta est assez complexe (couplages, non linéarités).

Une telle commande serait alors très difficile à synthétiser et à implémenter pour un calcul temps réel.

La figure [V.13](#) montre le schéma d'une commande centralisée appliquée au robot Delta.

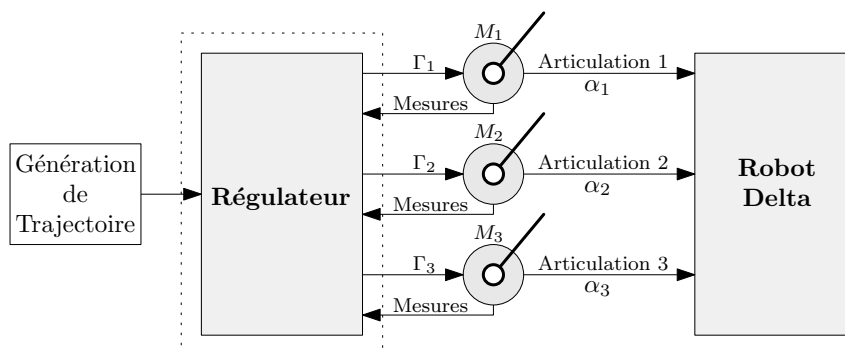


FIGURE V.13: Commande centralisé du robot Delta

On voit, bien que le régulateur prend en compte toutes les captures de positions articulaires et renvoie les trois commandes aux actionneurs. On retrouve bien une commande caractéristique d'un système MIMO.

5.1.2 Commande décentralisées

Plutôt que de prendre notre système en bloc, il serait intéressant de considérer trois sous systèmes SISO découplés. Quand aux couplages existants, ils seraient considérés comme des perturbations vis-à-vis de la commande la commande.

La synthèse se trouve alors très simplifiée et de ce fait l'implémentation aussi.

La figure V.14 montre le schéma d'une commande décentralisée appliquée au robot Delta.

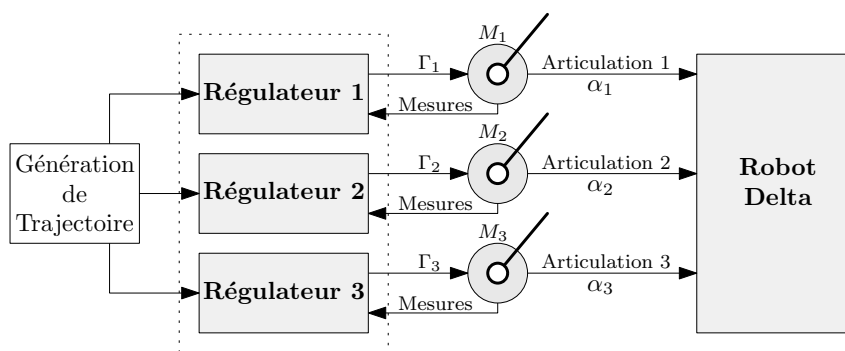


FIGURE V.14: Commande décentralisé du robot Delta

Dans le cas d'une commande décentralisée, il existe trois régulateurs indépendants, chacun prend en charge la capture des positions de l'articulation à laquelle il est rattaché, et calcule de façon complètement indépendante la commande à appliquer à son actionneur.

Ce schéma présente l'avantage d'un calcul simplifié de la commande. Nous adopterons donc cette architecture pour la synthèse des lois de commande du Delta. Cependant, une limitation réside dans le fait que le robot Delta est un robot rapide, possédant des paramètres dynamiques variables en fonction de sa position dans l'espace. Les couplages seraient

donc amplifiés et ne pourrait plus être négligés. Tout en gardant un schéma de commande décentralisé, l'introduction du modèle dynamique pourra alors régler ce problème (c.f. 5).

5.2 Conditions d'instrumentation et d'acquisition

Afin de se rapprocher le plus possible du régulateur implémenté en pratique, nous devons prendre en compte les considérations suivantes :

- **Pour l'instrumentation** : les moteurs sont pourvus d'encodeurs incrémentaux. Ces encodeurs ont une résolution fini qui est représentée par le nombre d'incrément par tour. Les données du capteur sont donc quantifiées avec un pas de quantification égal à la valeur de l'incrément. De plus l'acquisition des données se fait de façon échantillonnée.
Pour ce qui est de la vitesse, nous avons supposé précédemment que cette information est directement disponible de l'instrumentation. Pour le cas des encodeurs incrémentaux, cette vitesse est déduite de la position par dérivation numérique.
- **Pour l'actionneur** : la commande des moteurs est du type commande en couple. Un servomoteur est chargé de réaliser la consigne en couple introduite. Cette consigne doit être échantillonnée et quantifiée sur un nombre fini de bits.
De plus, le couple maximum des moteurs étant fini, une saturation de celui-ci doit être introduite.

Pour nos simulations, nous avons utilisé les données¹ représentées sur le tableau V.1.

Donnée	Valeur
Fréquence d'échantillonnage de l'acquisition	1 <i>kHz</i>
Précision des capteurs incrémentaux	100 000 incréments/tour
Saturation de la commande (couple)	25 <i>N.m</i>
Fréquence d'échantillonnage de la commande	1 <i>kHz</i>
Quantification de la commande	Sur 12 bits

TABLE V.1: *Caractéristiques de l'instrumentation et de l'acquisition*

Problèmes liés à l'action intégrale

L'action intégrale a pour effet d'augmenter la précision d'un système de régulation. Le problème est posé dans la pratique à cause de la saturation de l'actionneur.

En effet, lorsque l'erreur est importante ou persistante (cas de la poursuite) l'actionneur est vite saturée et donc l'erreur ne peut être comblée rapidement. L'action intégrale

1. Données transmises par l'EPFL.

par son effet mémoire aura accumulé une énergie importante qui augmentera encore l'erreur quand celle-ci commencera à diminuer. Cet effet est appelé emballement de l'action intégrale ou Integral wind-up.

Cet effet peut être contrôlé de plusieurs manières. On citera les deux solutions suivantes :

- Saturation de l'action intégrale à une valeur adéquate qui n'atteint pas la valeur de saturation de l'actionneur.
- Désactiver l'action intégrale quand l'erreur est importante et l'enclencher quand celle-ci commence à diminuer. Pour le cas de la poursuite, il suffirait d'en faire de même avec un réenclenchement de l'intégrateur au moment où le positionnement est nécessaire.

Pour notre part nous avons adopté la première solution.

5.3 Commande PID

Afin de commencer notre synthèse de commandes, nous allons utiliser le régulateur le plus classique et le plus simple : le régulateur PID. Le régulateur implémenté est un PID de position décentralisé. Le régulateur prend en compte les considérations cités précédemment (c.f. 5.2).

Comme première étape, nous allons tester le régulateur PID pour une commande en point à point, l'ajuster et commenter les résultats. Il sera par la suite appliqué pour le suivi de trajectoire.

Le schéma de régulation est donné sur la figure V.15.

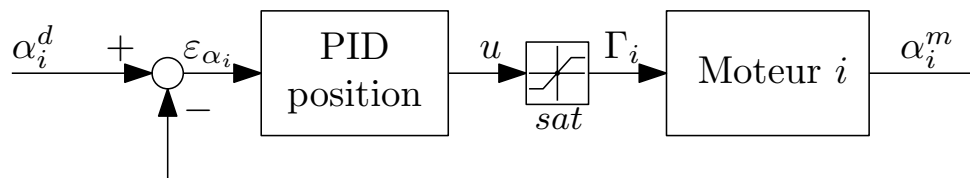


FIGURE V.15: Commande du Delta avec des régulateurs PID décentralisés

Le régulateur PID implémenté est donné sur la figure V.16.

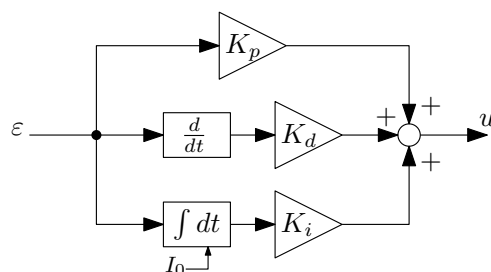


FIGURE V.16: Régulateur PID implémenté

Ce régulateur est identifié par ses trois constantes K_p , K_i et K_d et par la valeur initiale de l'action intégrale I_0 . La saturation de la sortie doit être mise en place afin que la commande soit admissible par les moteurs.

Le réglage des constantes du régulateur PID s'est fait par tuning en ayant en tête les effets des différentes actions : proportionnelle, intégrale et dérivative.

Nous effectuerons sur ce système de commande des tests différents, en appliquant plusieurs types de références articulaires :

1. Commande en point à point :
 - Échelon (Step) ;
 - Profil trapézoïdal de vitesse (en articulaire).
2. Commande en poursuite de trajectoire

a. Consigne en échelon

Pour la commande en point à point, la première étape consiste à appliquer des échelons articulaires suivant la position qu'on souhaite atteindre (utilisation du MGI).

Les résultats de ce test pour un point de départ situé en $(0, 0, -0.3)$ et pour une arrivée située en $(-0.1, -0.1, -0.5)$ sont montrés sur la figure V.17

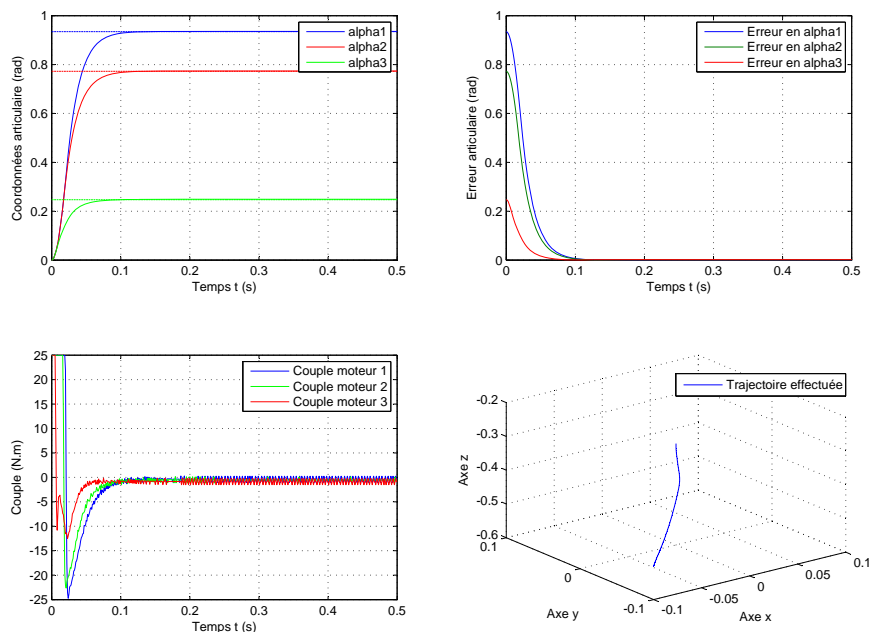


FIGURE V.17: Résultat de positionnement par Step - Régulateur PID

Les remarques suivantes sont à faire concernant le cas de la consigne en échelon :

- L'accélération et la vitesse maximale sont incontrôlables, de même que le temps de parcours qui dépend de la référence désirée (paramètres du PID fixes), la dynamique

du mouvement est donc mal maîtrisée. Dans notre cas nous avons une vitesse maximale de 6.5 m/s et une accélération de 48 g . Ce qui est beaucoup même pour le cas d'un robot rapide comme le Delta.

- L'effet de la gravité n'est pas pris en compte dans la commande, il se trouve alors que le dépassement n'est pas identique sur une séquence de montée ou de descente par exemple.
- De plus, le signal de commande (couple) prend des valeurs importantes en début de cycle. Cela est dû à la commande en elle-même, qui n'a pas de connaissance sur les effets de la dynamique du système (gravité, inertie).

Nous avons essayé de contrer cet effet de deux façons : en initialisant l'action intégrale I_0 à une certaine valeur et en ajoutant un a priori prenant en compte le modèle statique du robot. Ces deux méthodes ont montré cependant leurs limites. La première pour un manque évident de flexibilité (changer l'initialisation selon le cas), la seconde méthode quand à elle ne prend pas en compte les variations dynamiques du robot, elle ne sera donc pas suffisante pour la phase dynamique du mouvement.

- La trajectoire suivie par l'organe terminal n'est pas contrôlée (caractéristique même d'une commande point-à-point).

b. Consignes avec loi mouvement

Nous essayerons à présent d'utiliser une loi de mouvement dans la génération des consignes. Cela nous permettra de contrôler les accélérations, vitesses et temps de parcours du robot. Cette loi donne lieu à un positionnement point-à-point quand elle est appliquée dans l'espace articulaire. Dans notre cas la loi en trapèze est adoptée.

Les résultats obtenus pour cette méthode sont présentés sur la figure V.18

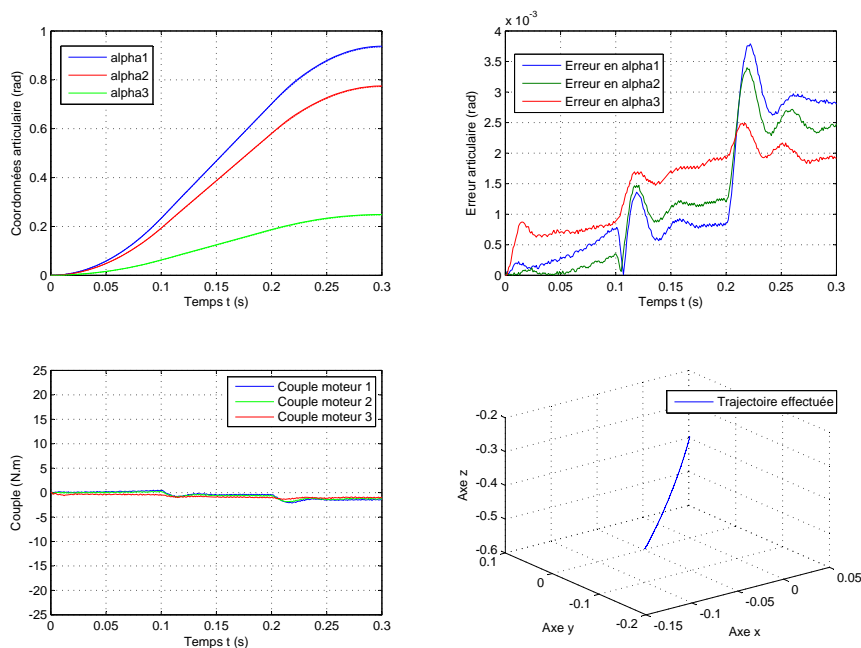


FIGURE V.18: Consigne en loi de mouvement trapézoïdale

Par rapport à la consigne précédente (en échelon) on voit bien que les couples de commande sont nettement plus faibles (alors que les paramètres du régulateur sont les mêmes), ce qui engendre une accélération beaucoup plus faible ($5.9 g$ contre $48 g$ pour le cas précédent) et ce pour des erreurs du même ordre de grandeur (erreur de positionnement final).

Ce résultat justifie amplement l'utilisation d'une loi de mouvement dans la génération des consignes puisque la dynamique du robot et des moteurs est entièrement contrôlée et de ce fait très bien optimisée. Pour tous nos prochains tests nous adopterons de façon systématique une loi de mouvement trapézoïdale.

Cependant, le mouvement non contrôlé de la nacelle rend cette commande très inadaptée au cas du Delta. (c.f § b. section 3).

Dorénavant, tous les tests de commande effectués seront réalisés pour des poursuites de trajectoires dans l'espace opérationnel.

c. Génération de consignes opérationnelles

Nous avons pu nous rendre compte dans les paragraphes précédents de la nécessité de commander le robot dans le domaine opérationnel, de plus, nous avons déjà mentionné la nécessité de garder le contrôle de la trajectoire suivie par l'organe terminal du robot pour des applications de prise-dépose en environnement industriel.

Le test du régulateur précédent s'est fait en suivant le protocole suivant : trajectoire selon x d'une longueur de 10 cm parcourue en $0,1\text{ s}$.

les résultats sont présentés dans la figure V.19.

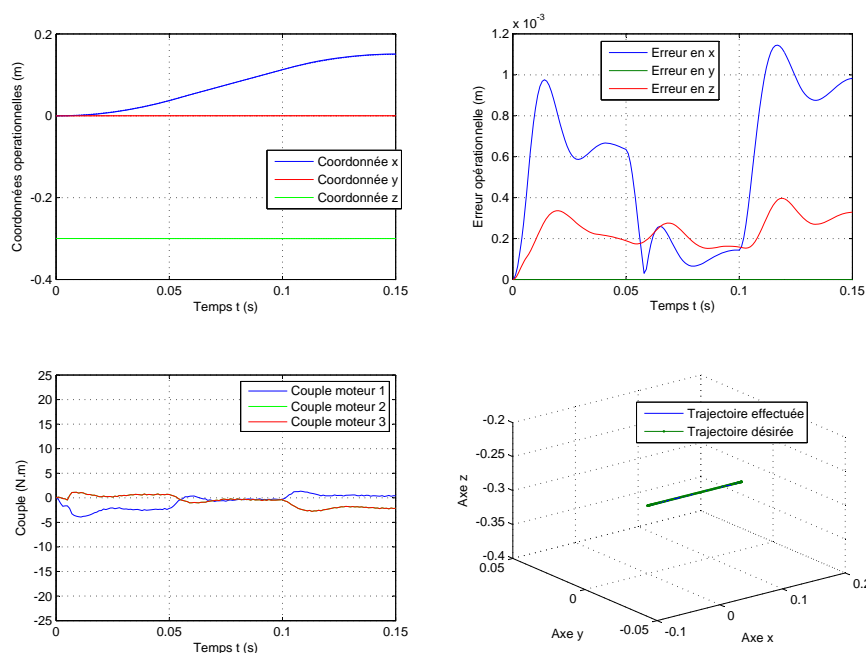


FIGURE V.19: Poursuite de trajectoire opérationnelle - PID

A noter que le régulateur utilisé possède toujours les mêmes paramètres.

Il faut aussi signaler que la trajectoire suivie par l'organe terminal est parfaitement connue (figure V.19).

Les résultats précédents nous montrent que les erreurs sont maximales dans les phases dynamiques du mouvement (accélération et décélération), cela impose la nécessité d'utiliser une commande plus élaborée qui doit prendre en compte la dynamique du Delta.

5.4 Double boucle PID vitesse - position

Afin d'améliorer les performances on utilise souvent un double bouclage. Un sur la position et un autre sur la vitesse. Le schéma de commande devient - figure V.20.

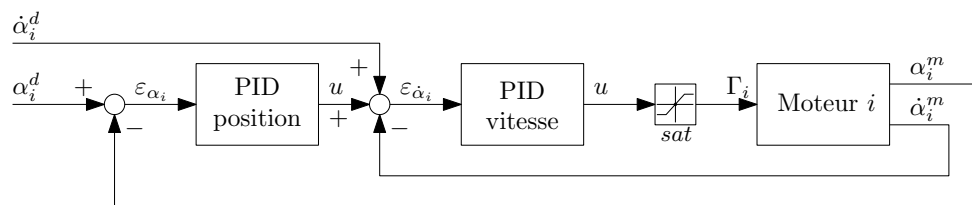


FIGURE V.20: Double boucle PID position-vitesse

Cette commande permet de prendre en compte la variation souvent rapide de la vitesse du Delta et d'améliorer ainsi les performances de suivi de trajectoire.

En utilisant le protocole de test mis en place précédemment, on obtient les résultats donnés sur la figure V.21.

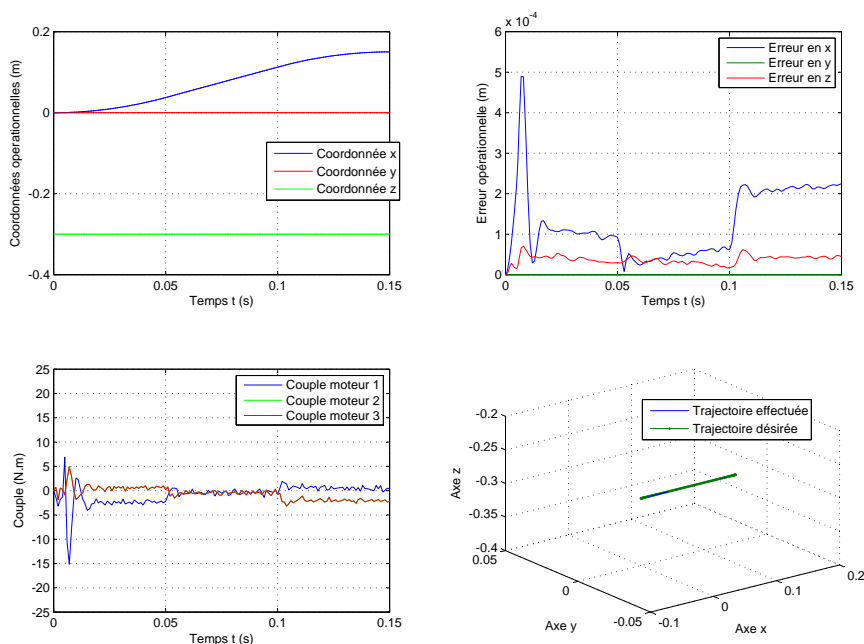


FIGURE V.21: Poursuite de trajectoire opérationnelle - Double boucle

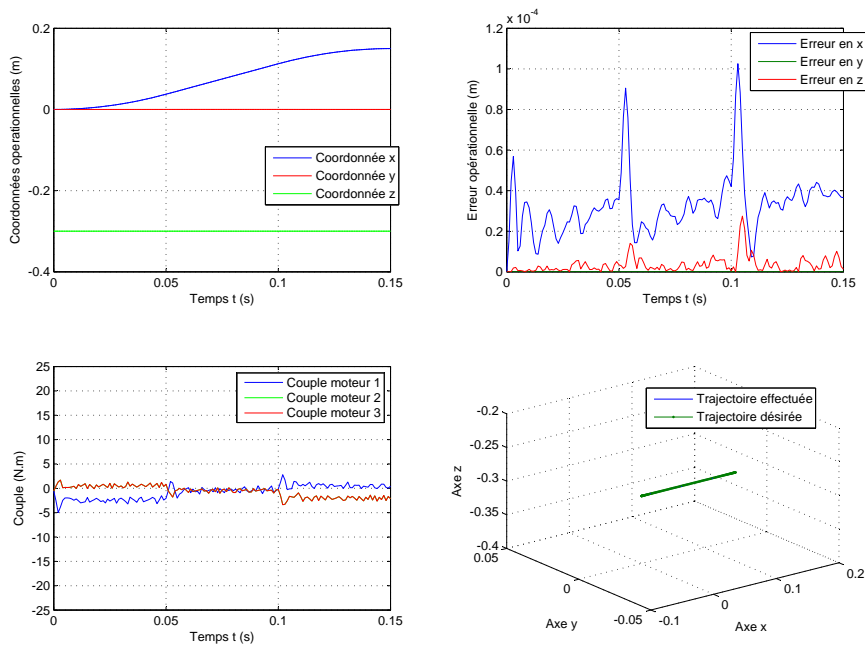


FIGURE V.23: Poursuite de trajectoire opérationnelle - Double boucle + MDI

Dans ce cas là, nous remarquons une amélioration significative et de l'erreur de poursuite et de la dynamique du couple de commande. L'introduction du modèle dynamique dans la commande permet au régulateur de mieux réagir, puisque la plus grande part de la commande est fournie par l'à priori.

6 Tests de performances

La commande sur laquelle nous allons nous basé afin de réaliser non différents tests de performance est la double boucle PID avec à priori de couple en considérant l'effet de l'échantillonnage et de la quantification. Le schéma de simulation est donné sur la figure V.24.

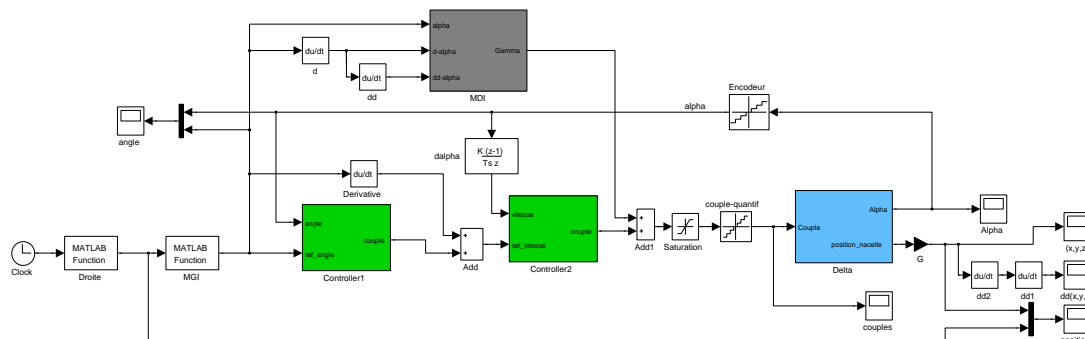


FIGURE V.24: Implantation réelle du régulateur

Les performances peuvent être quantifiées en fonction de plusieurs paramètres :

- Temps de parcours : ou cadence de travail, celle-ci fixe la dynamique du mouvement (accélération et vitesse) ;
- Trajectoire utilisée (forme de la trajectoire) ;
- Charge transportée ;
- L'emplacement de l'organe terminal dans le volume de travail lorsque le robot effectue sa tâche ;
- La rigidité du positionnement final du robot.

Nous allons essayer d'analyser toutes ces considérations.

6.1 Cadence de prise-dépose

L'utilisation première du robot Delta est la palettisation donc la prise-dépose d'objets. Nous étudierons alors plusieurs cas de cadences de prise-dépose et nous analyserons les erreurs de suivi de trajectoire résultantes.

Nous avons pris en premier lieu une trajectoire de prise-dépose simple, composée de trois segments de droites et sans aucune charge transportée.

Le temps effectif de prise et de dépose est de 25 ms et la trajectoire de prise-dépose est donnée sur la figure V.25. Il reste à signaler qu'un cycle de prise-dépose (1 Hz) contient les opérations suivantes :

1. Prise (25 ms) ;
2. Aller ;
3. Dépose (25 ms) ;
4. Retour sur la même trajectoire.

De plus une phase de positionnement a été ajoutée simulant un Delta qui était au repos et qui va se positionner pour commencer sa tâche de prise-dépose (durée $0,6\text{ s}$).

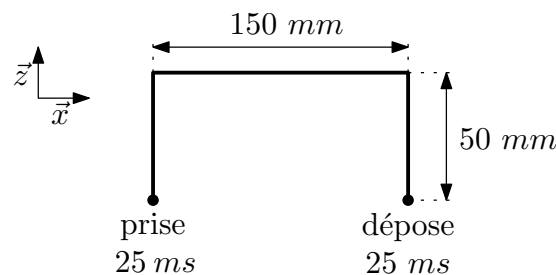


FIGURE V.25: Trajectoire de prise-dépose en segments de droites

- a. Cadence de 1 Hz** Les résultats sont donnés sur la figure V.26 :
- Le premier graphique représente la trajectoire spatiale effectuée ;
 - Le second représente la valeur absolue de l'erreur de poursuite ;
 - Le troisième et le quatrième représentent la dynamique de la vitesse et de l'accélération de la nacelle ;

– Le cinquième couple moteur de commande.

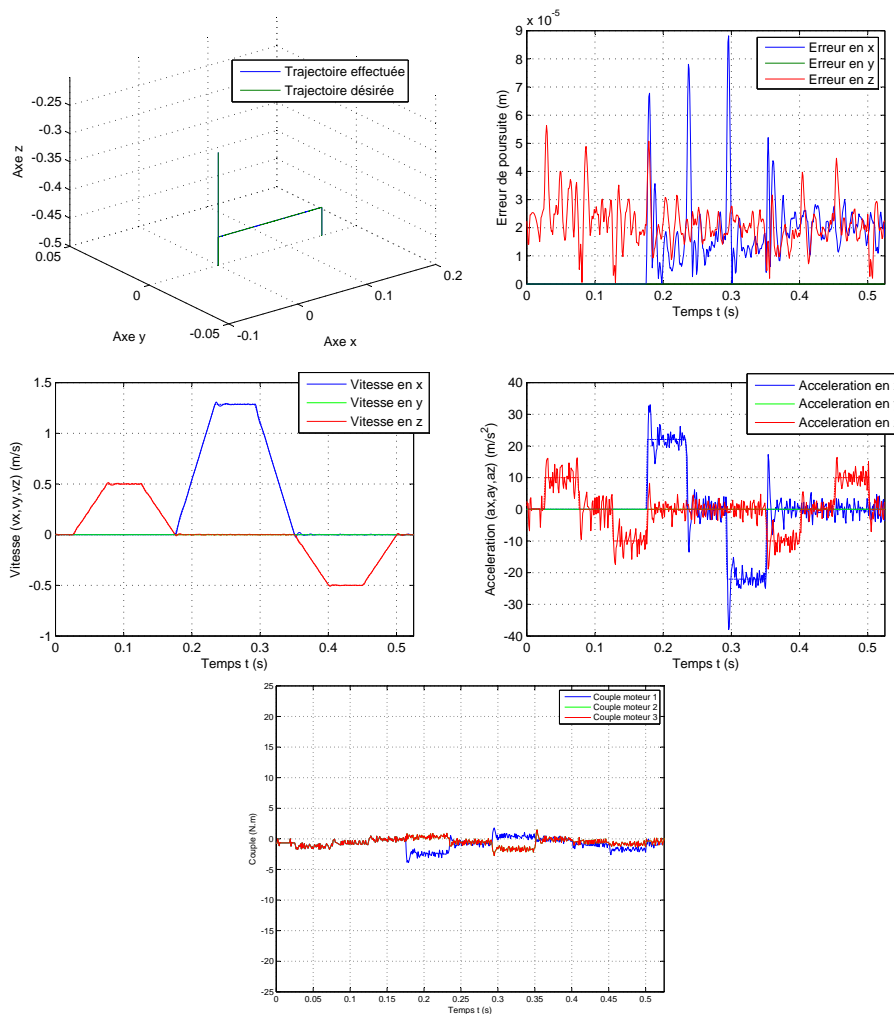


FIGURE V.26: *Prise de pose à 1 Hz avec les segments de droites*

On remarque une erreur maximale de poursuite² de $0,087 \text{ mm}$ et une erreur de positionnement final de $0,022 \text{ mm}$ avec une erreur RMS de $0,020 \text{ mm}$ ³. Quand à la dynamique, elle a une accélération de l'ordre de $4 g$ et une vitesse maximale de l'ordre de $1,3 \text{ m/s}$.

On remarque aussi des pics d'erreur se trouvant sur les transitions de la trajectoire et de la loi de mouvement. Une trajectoire en segments de droite présente des arrêts aux points de jonctions, il y'a donc trois profils en trapèze sur le parcours. Ceci explique le nombre de pics d'erreur.

Le couple moteur est aussi affecté par ces discontinuités dans la trajectoire, donnant des variations brusques des couples moteurs.

2. Les erreurs indiquées sont prise comme étant la valeurs maximale des trois erreurs opérationnelles : en x , en y et en z .

3. Erreur RMS : Root Mean Square ou racine de la moyenne des carrés ce qui fait référence à l'écart type ou à la dispersion de l'erreur (σ).

Il est aussi à noter qu’ayant pris le cas d’un système quantifié, la résolution des capteurs est finie. Ceci conduit à une résolution finie dans l’espace opérationnel. Cette résolution dépend de la position opérationnelle et est donnée par la formule :

$$(\Delta x \ \Delta y \ \Delta z) = MCD(\alpha_1, \alpha_1, \alpha_1, \Delta\alpha, \Delta\alpha, \Delta\alpha)$$

Où : $\Delta\alpha$ est la résolution articulaire due aux capteurs incrémentaux.

En analysant cette fonction tout le long du parcours réalisé, on note que la résolution opérationnelle varie entre $0,016 \text{ mm}$ et $0,024 \text{ mm}$. L’erreur finale et RMS sont donc dans l’ordre de grandeur de la résolution de l’incrément.

A noter que le réglage des paramètres du régulateur s’est fait par ajustement, d’abord de la boucle de vitesse seul, ensuite de la boucle de position, avec l’utilisation d’une saturation de l’action intégrale afin d’éviter le phénomène de la dérive de l’intégrateur.

b. Cadence de 2 Hz Les résultats sont donnés sur la figure V.27.

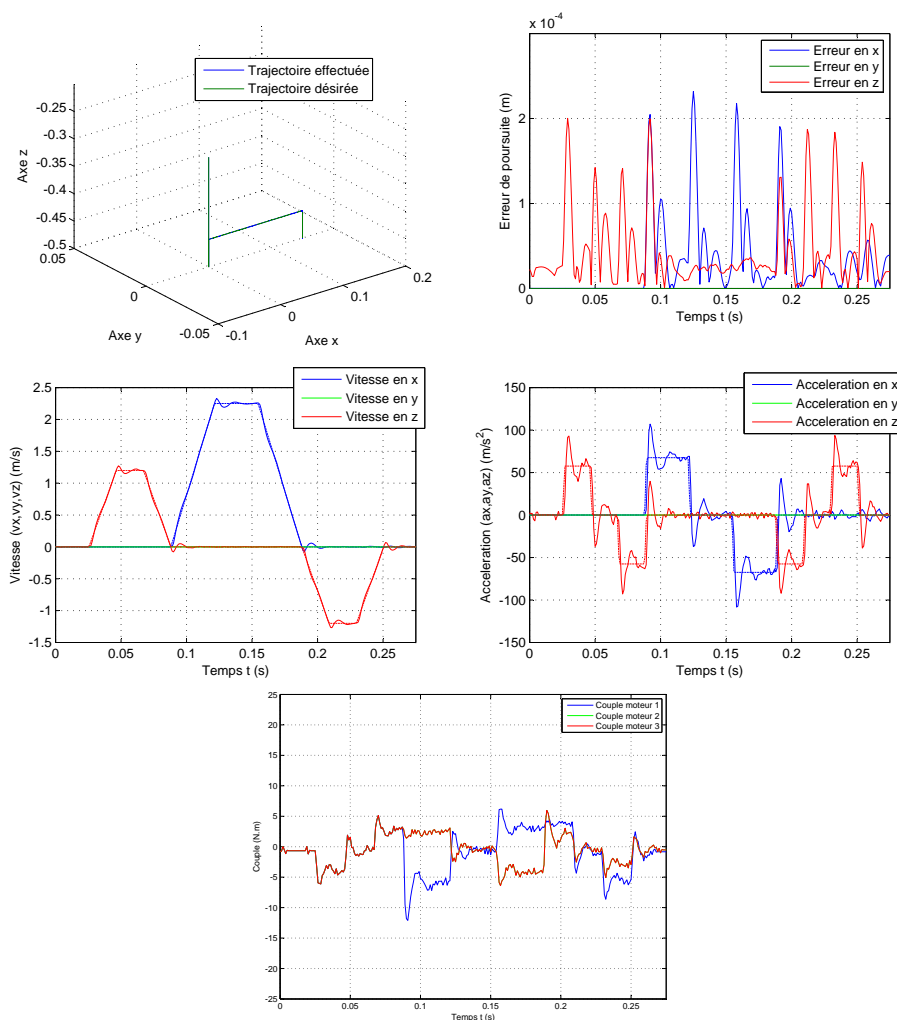


FIGURE V.27: *Prise dépose à 2 Hz avec les segments de droites*

On remarque une erreur maximale de poursuite de $0,23 \text{ mm}$ et une erreur de positionnement final de $0,038 \text{ mm}$ avec une erreur RMS de $0,038 \text{ mm}$. Quand à la dynamique, elle a une accélération de l'ordre de 11 g et une vitesse maximale de l'ordre de $2,3 \text{ m/s}$.

On remarque que l'erreur de poursuite a augmenté de façon significative par rapport à la cadence de 1 Hz . Cependant, l'erreur maximale et RMS restent proches de l'erreur incrémentale.

La même remarque est à faire pour les pics d'erreurs et la discontinuité des couple dans une proportion plus importante.

c. Cadence de 4 Hz Les résultats sont donnés sur la figure V.28.

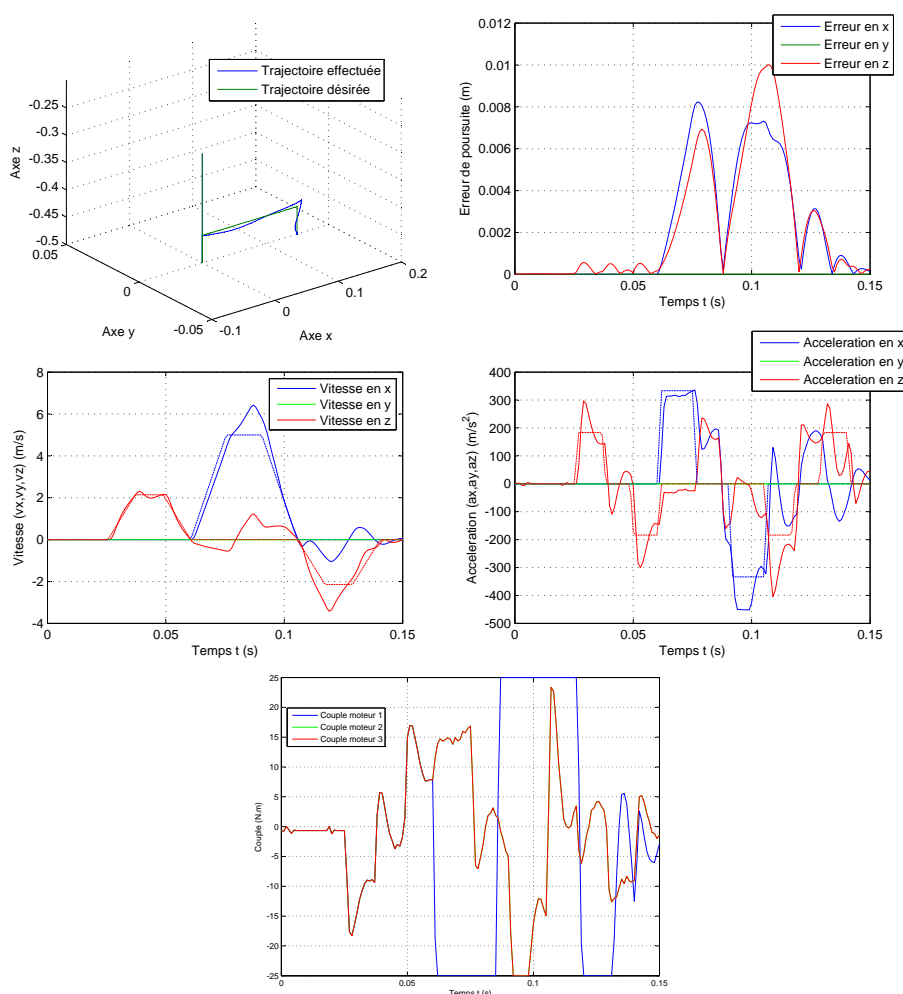


FIGURE V.28: *Prise dépose à 4 Hz avec les segments de droites*

On remarque une erreur maximale de poursuite de 10 mm et une erreur de positionnement final de $0,23 \text{ mm}$ avec une erreur RMS de $1,7 \text{ mm}$. Quand à la dynamique, elle a une accélération de l'ordre de 45 g et une vitesse maximale de l'ordre $6,7 \text{ m/s}$.

En allant à une cadence de 4 Hz on remarque que l'erreur de poursuite devient de l'ordre du cm ce n'est plus acceptable pour le cas du Delta. Cette erreur est due aux fortes discontinuités de la trajectoire et aux couples importants (dépassant la valeur de saturation) que les moteurs doivent fournir afin de vaincre ces discontinuités et atteindre la cadence imposée.

Le robot arrive quand même à se positionner sur le point d'arrivée (présence de l'action intégrale) mais après un décrochage important de la trajectoire de référence (dans les points de discontinuité de la trajectoire).

Afin d'améliorer le suivi de trajectoire nous allons prendre une trajectoire de prise-dépose plus élaborée qui ne contient pas de discontinuités et qui nous permettra d'atteindre l'objectif d'une cadence de 4 Hz .

6.2 Trajectoire de prise dépose

Pour éliminer les discontinuités de la trajectoire en segments de droites, une trajectoire en clothoïdes est adoptée (toujours sans charge transportée).

Cette trajectoire est donnée sur la figure V.29.

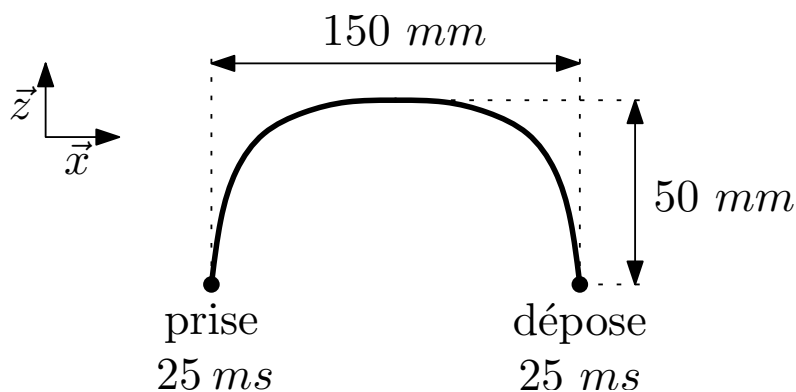


FIGURE V.29: Trajectoire de prise-dépose en clothoïdes

Nous procéderons aux mêmes tests avec cette nouvelle trajectoire.

- a. **Cadence de 1 Hz** Les résultats sont donnés sur la figure V.30.

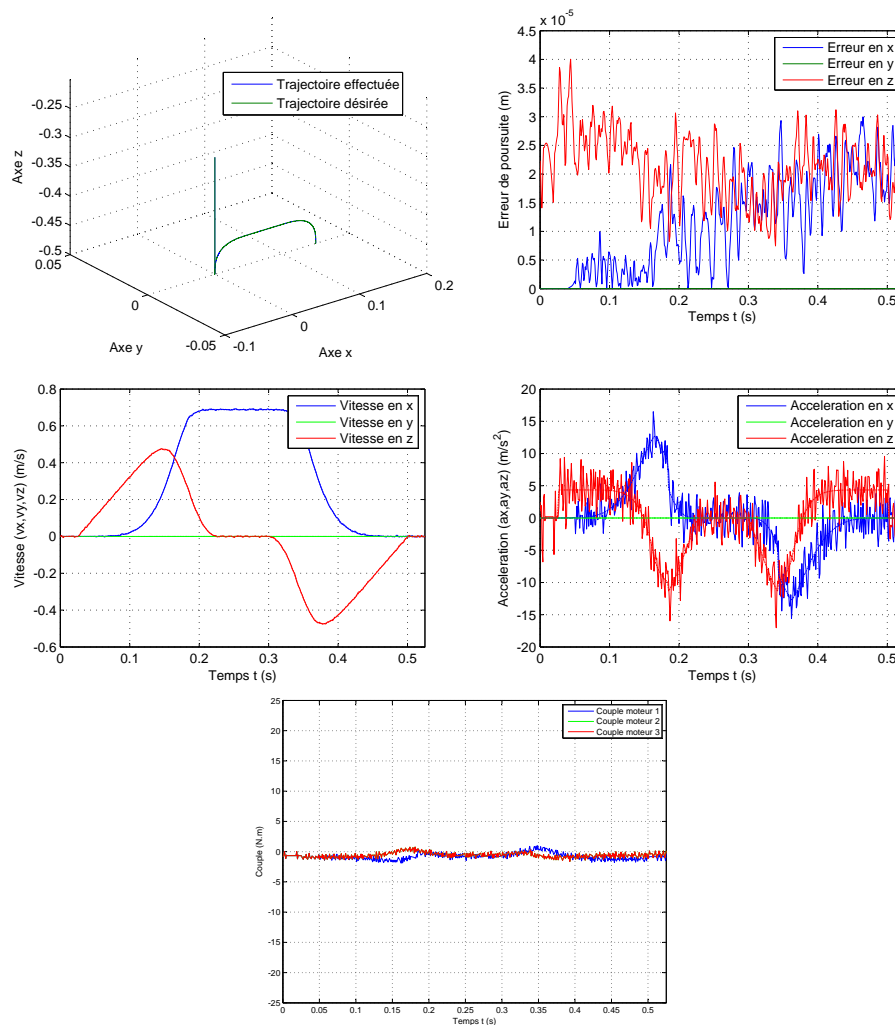


FIGURE V.30: *Prise dépose à 1 Hz avec la clothoïde*

On remarque une erreur maximale de poursuite de $0,040 \text{ mm}$ et une erreur de positionnement final de $0,020 \text{ mm}$ avec une erreur RMS de $0,021 \text{ mm}$. Quand à la dynamique, l'accélération maximale atteinte par le robot est de l'ordre de 2 g avec une vitesse maximale de l'ordre $0,7 \text{ m/s}$. On remarque ainsi que la dynamique est améliorée par rapport à la première trajectoire.

A noter que l'erreur de poursuite reste autour de la résolution incrémentale (entre $0,016 \text{ mm}$ et $0,024 \text{ mm}$), il en est de même pour l'erreur de positionnement et l'erreur RMS.

Pour ce qui est des couples moteurs, leur variation est plus douce et ne présente pas de discontinuité par rapport aux couples mis en jeu pour la 1^{ère} trajectoire à la même cadence.

De plus, en termes de valeur, les couples sont moins importants pour cette trajectoire. On arrive ainsi à diminuer les couples de commande et à diminuer l'erreur en utilisant seulement une trajectoire plus adaptée.

b. Cadence de 2 Hz Les résultats sont donnés sur la figure V.31.

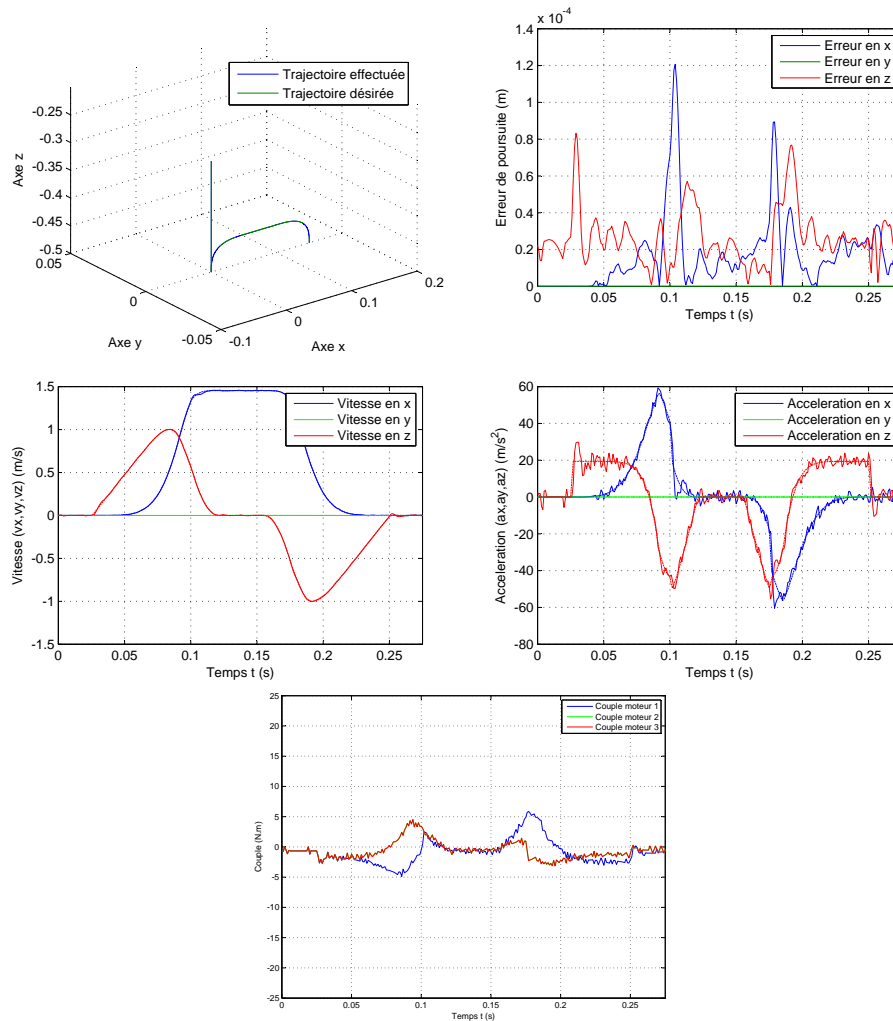


FIGURE V.31: *Prise dépose à 2 Hz avec la clothoïde*

On remarque une erreur maximale de poursuite de 0,12 mm et une erreur de positionnement final de 0,027 mm avec une erreur RMS de 0,023 mm (comparables à la résolution opérationnelle). Quand à la dynamique, elle a une accélération maximale de l'ordre de 6 g, pour une vitesse maximale de l'ordre de 1,5 m/s.

En augmentant la cadence, l'erreur augmente car la dynamique est plus importante (vitesse et accélération) et donc le système devient plus difficile à contrôler. De plus, du fait que la commande soit échantillonnée et quantifiée, l'augmentation de la vitesse d'exécution de la tâche amènera la fréquence d'échantillonnage à devenir de moins en moins négligeable devant la dynamique de la structure, ce qui peut dégrader les performances de la commande.

c. Cadence de 4 Hz Dans le premier cas (trajectoire en segments de droites) cette cadence a donné des résultats très médiocres en termes de suivi de trajectoire. En utilisant

la trajectoire en clothoïdes, la cadence de 4 Hz peut être atteinte avec de meilleures performances de poursuite. Cela est illustré sur la figure V.32.

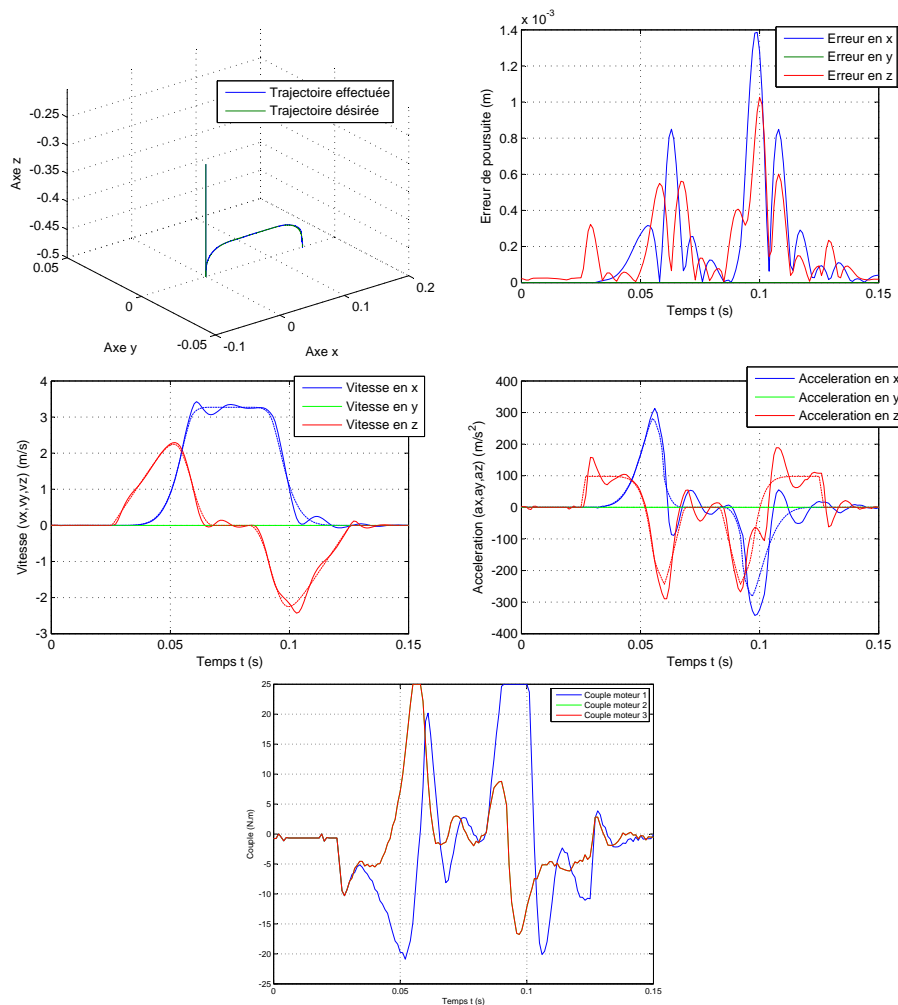


FIGURE V.32: Prise dépose à 4 Hz avec la trajectoire en clothoïdes

On remarque une erreur maximale de poursuite de $1,4\text{ mm}$ et une erreur de positionnement final de $0,041\text{ mm}$ avec une erreur RMS de $0,16\text{ mm}$. Quand à la dynamique, le robot atteint une accélération maximale de l'ordre de 34 g pour une vitesse maximale de l'ordre $3,4\text{ m/s}$.

Dans ce cas là, on remarque que les performances du robot sont moindres, cela est compréhensible vu la cadence et la dynamique imposés. La comparaison des couples moteurs nous permet d'analyser cette situation.

On voit bien que la commande arrive à la saturation dans certains points, ces saturations créent une dégradation du suivi de trajectoire vue que le moteur ne répond pas convenablement à la consigne du régulateur.

En fait les moteurs sont choisis en fonction d'un certain cahier de charge, ce cahier de charge comprend :

- La dynamique : les accélérations et vitesses donc la rapidité du robot ;
- La précision : l'erreur permmissible en concordance avec la dynamique ;
- La position de la trajectoire : la portion de l'espace de travail dans laquelle la trajectoire est réalisée ;
- La charge maximale que devra transporter le robot à la cadence souhaitée.

Tout cela nous amène à dire que le choix des moteurs est imposé par l'adjonction de ces points. Afin de déterminer les moteurs adéquats il sera nécessaire alors d'imposer le compromis souhaité.

Conclusion

La trajectoire en clothoïde présente une meilleure dynamique de l'erreur. De plus, à cadence de travail égale, l'accélération de la nacelle et les couples moteurs maximaux sont moindres. Cela s'explique par le fait que cette trajectoire élimine les discontinuités par rapport à la trajectoire en segments de droites. De plus elle satisfait une variation de courbure progressive ce qui rend l'accélération radiale continument variable.

L'utilisation de la trajectoire en clothoïdes assure une variation plus progressive des couples moteurs. Ceci est un avantage indéniable pour diminuer l'usure des moteurs et des hacheurs. Cette trajectoire sera dorénavant utilisée comme la trajectoire de référence en ce qui concerne la tâche de prise-dépose.

6.3 Robustesse par rapport à la charge transportée

Afin de vérifier la validité de notre système de commande et la qualité de sa robustesse par rapport à un changement paramétrique (ou perturbation), nous allons introduire une incertitude sur la masse de la nacelle. Cela correspondra à la variation de la charge transportée par le robot. Les tests se feront selon la cadence de travail du robot. Deux tests seront ainsi effectués : cadence de $2 Hz$ et de $4 Hz$.

Dans notre cas, nous prendrons plusieurs charges additionnelles⁴ de masses allant de $200 g$ à $2 kg$ pour la cadence de prise-dépose de $2 Hz$. Les résultats des tests sont donnés sur la figure VI.1 dans l'annexe B.

Le suivi de trajectoire commence à se dégrader à partir d'une masse de charge de $1,5 kg$, on atteint alors une erreur maximale de l'ordre de $3 mm$ ce qui est à la limite de l'acceptable, cette erreur ne conduit cependant pas au décrochage du robot de sa trajectoire de référence comme c'est le cas pour la masse de $2 kg$.

On peut alors considérer qu'une telle masse est autour la masse maximale pouvant être transportée par le robot à une cadence de $2 Hz$.

En ce qui concerne la cadence à $4Hz$, nous prendrons aussi plusieurs charges additionnelles de masse allant de $50 g$ à $500 g$.

4. Il faut noter que la masse additionnelle n'est pas prise en compte dans le MDI.

A noter que les paramètres du régulateur ont été réajusté car dans le premier cas, l'objectif de minimisation de l'erreur finale pouvait être atteint sans dégrader le suivi de performance. Pour la cadence de $4Hz$, les mouvements sont très rapides, l'action drivée (régulateur de position) a été augmenté afin de permettre un suivi meilleur de la consigne, mais cela a eu pour conséquence de dégrader l'erreur finale (à l'arrivée).

La figure VI.2 de l'annexe B représente les résultats de ces différents tests.

Le suivi de trajectoire commence à se dégrader à partir d'une masse de charge de $100 g$, on atteint alors une erreur maximale de l'ordre de $2,1 mm$ ce qui est à la limite de l'acceptable (par rapport au cas $150 g$), de plus cette erreur ne conduit pas au décrochage du robot de sa trajectoire de référence comme c'est le cas pour la masse de $500 g$.

Notons dans ces tests l'effet de la saturation des couples moteurs qui donne des pics d'erreur de poursuite. Ces pics d'erreurs augmentent avec la masse et la saturation du couple jusqu'au décrochage de la trajectoire de référence (grande erreur en fin de parcours).

On considère que la masse maximale pouvant être transportée par le robot à une cadence de $4 Hz$ est autour de $100 g$.

L'étude faite précédemment sur la charge transportable par le robot Delta a été réalisée directement sur le robot commandé. Cette étude a eu pour but de montré les limites d'un système déjà existant (avec les couples maximaux des moteurs imposés), ces limites se manifestent en terme de masse transportée à une cadence donnée. Cela nous permet de quantifier réellement le comportement de notre système vis-à-vis de l'incertitude sur la masse transportée. Ce comportement prendra donc les considérations réelle comme : l'effet de la commande, des échantillonnages, de la quantification des consignes de couples,...

Une autre approche peut être adoptée ayant pour but le dimensionnement des moteurs (pour le dimensionnement et la construction du robot). Cette approche ne sera pas abordés dans ce travail (le robot étant réalisé on cherche seulement à caractériser ses performances), nous donnons cependant le schéma de principe présenté sur la figure V.33, Schéma qui se bas sur l'utilisation du MDI afin de prévoir les couples désirés.

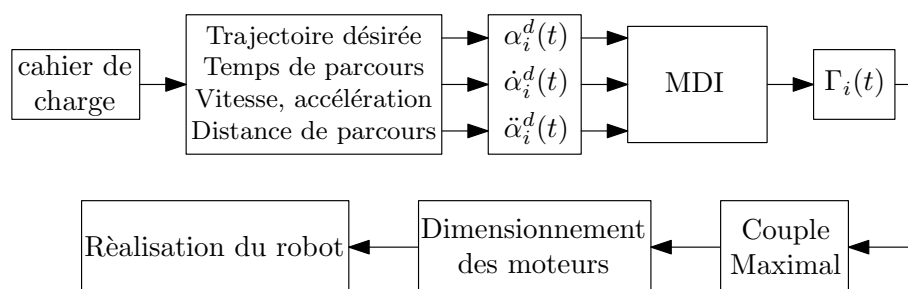


FIGURE V.33: Dimensionnement des moteurs selon les couples souhaités

Dans ce cas là, on dimensionne les moteurs selon les couples maximaux nécessaires à la réalisation de la dynamique de trajectoire souhaités ($\alpha^d(t)$, $\dot{\alpha}^d(t)$, $\ddot{\alpha}^d(t)$). Dans la plupart des cas on prend en plus une marge de sécurité sur la valeur trouvée. Il sera alors nécessaire de refaire le test que nous avons fait précédemment afin de vérifier les résultats sur le système bouclé.

6.4 Robustesse par rapport aux perturbations sur le positionnement

Afin de tester la rigidité du positionnement final de notre robot, ajoutons une force de perturbation au niveau de la nacelle. Cette force est introduite par le biais d'un bloc SimeMechanics jouant le rôle d'un actionneur virtuel placé sur la nacelle.

Cet actionneur simule une personne ou un élément extérieur qui appliqueraient une perturbation sur le positionnement de la nacelle. En appliquant une perturbation de 10 N sur le positionnement final de la trajectoire de prise-dépose, on obtient les résultats présentés sur la figure V.34.

A noter que la perturbation se présente sous la forme d'un créneau de force appliqué d'une durée de $t = 1\text{ s}$ et le robot est déjà positionné.

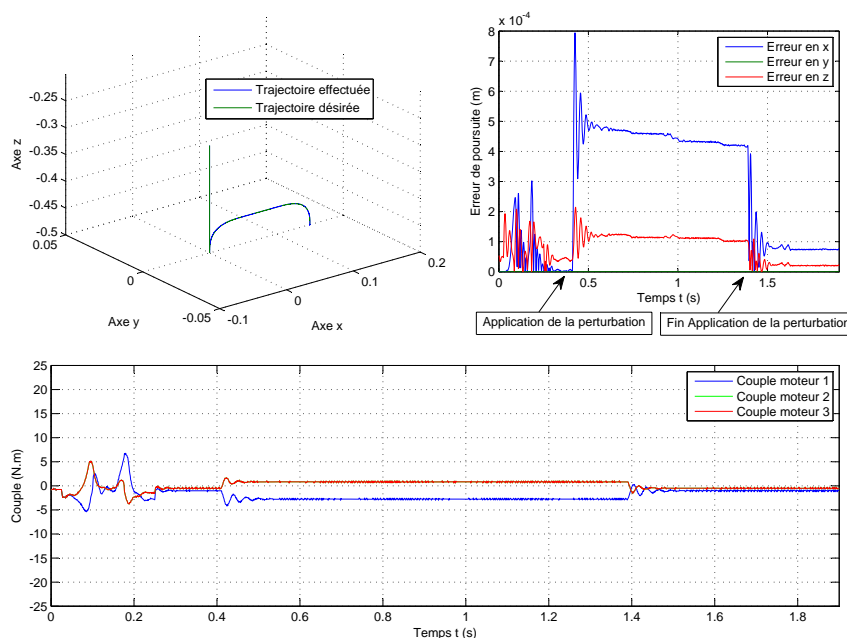


FIGURE V.34: Application d'une perturbation sur le positionnement de la nacelle

On voit que le positionnement est très rigide. La force de perturbation appliquées n'a pratiquement aucun effet sur le positionnement (erreur maximale de $0,8\text{ mm}$). Cela est obtenu grâce l'action intégrale du régulateur de position qui a réglé suffisamment grande.

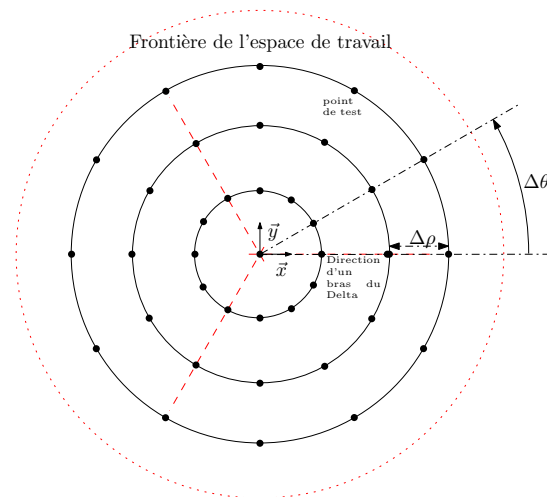
L'augmentation de l'action intégrale peut causer le phénomène d'Integral wind-up, cela a été réglé en saturant la commande due à l'action intégrale (gel de l'action intégrale).

6.5 Robustesse des performances dans le volume de travail

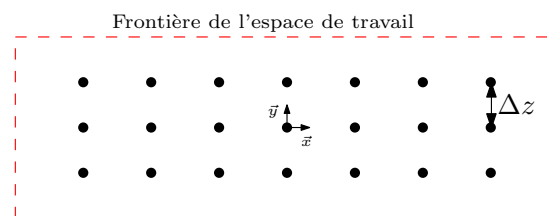
Nous allons essayer de caractériser et d'analyser les variations de l'erreur de poursuite du robot pour une trajectoire de test donnée (segment de droite verticale d'une longueur

de 50mm), la commande du robot est réalisée par un régulateur dont les paramètres ont été optimisés pour le cas $x = 0$ et $y = 0$ (axe $z = 0$).

Le test se fera sur un certain nombre de points de l'espace de travail du robot (111 points au total), suivant le schéma de la figure V.35. Ces points ont été choisis, dans un premier temps, selon une distribution horizontale (radiale avec variation angulaire) et verticale (trois plans horizontaux) régulières.



(a) vue de dessus des points de test sur l'espace de travail



(b) vue de côté des points de test sur l'espace de travail

FIGURE V.35: Points de test pour le réglage des paramètres du régulateur

Les points de test seront répartis d'une manière uniforme sur l'espace de travail du robot, les dimensions de cet espace ayant été déjà été établi précédemment (figure IV.11 avec $D = 668\text{ mm}$, $H = 232\text{ mm}$ et $Z_h = -320\text{ mm}$). Les distances entre les points adjacents sera la suivante :

- Distance entre deux points consécutifs sur le même rayon : $\Delta\rho = 83.5\text{ mm}$;
- Angles de séparation entre deux rayons porteurs de points : $\Delta\theta = 30^\circ$;
- Distance entre les trois plans horizontaux de test : $\Delta z = 58\text{ mm}$.

Pour éviter une éventuelle confusion entre l'erreur de poursuite et la résolution opérationnelle (qui est variable dans l'espace de travail), la comparaison des erreurs se fera soit

au niveau angulaire (la résolution angulaire est constante), soit en effectuant les mesures opérationnelles à travers les mesures articulaires, en passant par le MGD.

Il est à noter que la symétrie ternaire du robot Delta se répercute sur le comportement du robot, c'est à dire que celui-ci aura la même symétrie ternaire. Les erreurs obtenues en un point donné seront donc les mêmes que pour un autre point situé à un angle de $\frac{2\pi}{3}$ et possédant la même distance par rapport à l'origine du repère (c'est à dire le même rayon).

La symétrie ternaire du comportement du Delta est très évidente, sa présence nous permettras de diviser le nombre de points de test par 3 (on n'aura plus que 39 points à tester). En effectuant une analyse un peu plus fine, nous avons constaté qu'en plus de cette symétrie ternaire, le robot Delta possédait une autre particularité qui permettait de réduire encore plus le nombre de points de test.

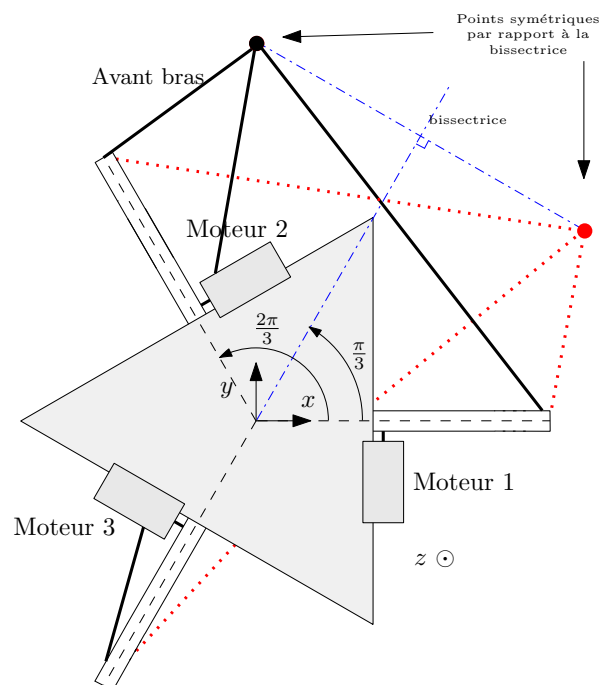


FIGURE V.36: Symétrie hexanaire dans le comportement du robot Delta

Sur le schéma descriptif précédent, nous remarquons qu'entre les deux positions de la nacelle (point noir et point rouge), qui sont symétriques par rapport à la bissectrice de l'angle formé par le bras 1 et le bras 2, l'effet sur le moteur 1 est permuté avec celui que subissait le moteur 2 au niveau de l'autre point. De plus, le moteur 3 subit exactement les mêmes effets (dynamiques) pour les deux positions de la nacelle (point noir et point rouge) vu que les effets des chaînes cinématique 1 et 2 sont seulement permutés.

Malgré les symétries constatées sur le robot Delta, nous décidons quand même de faire les tests sur tout l'espace de travail, ceci afin de confirmer les symétries que possède le comportement dynamique du Delta d'une manière expérimentale.

Résultats des tests d'analyse

Les conditions de test imposées sont :

- Longueur de parcours selon l'axe z : 50mm
- Temps de parcours : 35ms
- La dynamique est de 35g pour l'accélération maximale et de $2,3\text{m/s}$ pour la vitesse maximale.

Après avoir effectué les tests sur l'ensemble de l'espace de travail du robot, nous constatons que les performances du Delta (erreurs) possèdent effectivement une symétrie hexanaire, ceci confirme les suppositions que nous avons émises au préalable. Il nous suffit donc d'écrire les résultats sur $1/6^{\text{ime}}$ de l'espace de travail pour caractériser le comportement du robot sur l'ensemble de l'espace. Les résultats sont portées sur le tableau VI.3 en annexe A. Pour plus de clarté, nous avons aussi tracé les variations de l'erreur moyenne de poursuite du Delta (moyenne des valeurs absolues) en fonction de la position sur un plan horizontal (x, y) . Nous aurons plusieurs courbes chacune correspondant à des points de tests situés sur le même plan horizontal ($z = \text{cste}$).

Ces courbes nous permettent en effet de faire une analyse globale plus aisée et une interprétation des résultats plus intuitive.

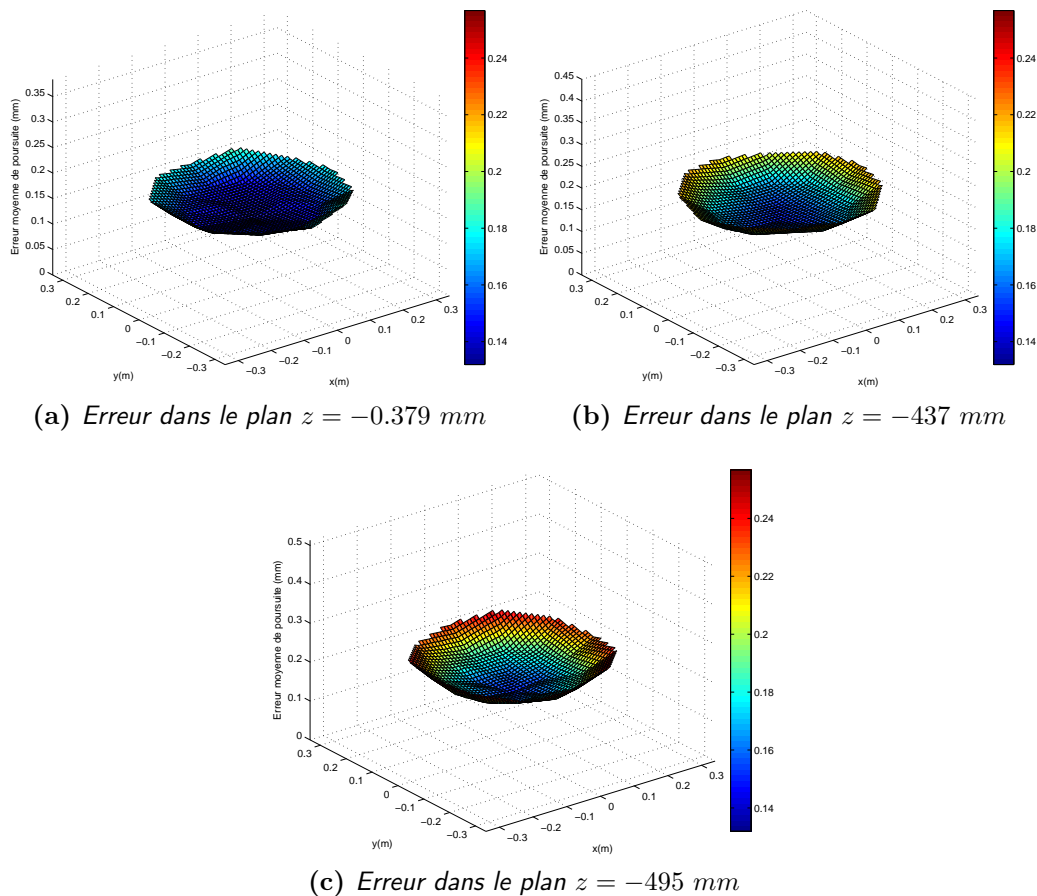


FIGURE V.37: Évolution de l'erreur moyenne dans l'espace de travail

En visualisant les courbes précédentes, nous faisons les remarques suivantes :

- La régulateur a été réellement optimisé pour la trajectoire selon l'axe z dans le 1^{er} plan (c'est l'erreur moyenne de poursuite la plus faible).
- L'éloignement du point de test par rapport au point de réglage modifie les performances en terme d'erreur de poursuite (éloignement dans le même plan (x, y) et l'éloignement selon les trois plan à z différents). Leurs valeurs ne varient pas dans une grande proportion. La variation maximale va du simple au double seulement.
- Ces erreurs restent cependant faibles connaissant la dynamique assez rapide qui a été imposée. Bien que nous ayant balayé tout l'espace de travail, notre régulateur a été capable de garder de très bonnes performances, malgré la modification totale du point de fonctionnement. Cela est obtenue avec l'ajout de l'à priori de couple. Cet ajout permet de faire travailler réellement la double boucle vitesse-position autour d'un point de fonctionnement.

En conclusion, nous pouvons dire que les performances de notre régulateur sont variables dans tout le volume de travail.

7 Commande par apprentissage itératif (ILC)

Dans cette commande nous allons essayer de simplifier le régulateur en adoptant un régulateur PD et profiter du fait que le robot peut avoir comme application de réaliser des trajectoires répétitives pour diminuer ainsi l'erreur.

En effet, dans certaines applications industrielles bien particulières, la tâche du robot se résume en la prise et la pose d'éléments à partir de points de prise (et de dépose) connus avec **précision**, selon une trajectoire de transport fixe et inchangée et avec des points de départ (point de prise) **identiques** entre les opérations de prise-dépose (itérations) consécutives.

7.1 Présentation de la commande

L'aspect répétitif d'un certain nombre de tâches industrielles a fait naître, dans le milieu de l'automatique (au cours des années soixante), l'idée d'essayer d'exploiter la répétabilité du processus afin d'améliorer la commande du système. En effet, étant donné que la trajectoire désirée reste toujours la même, pour toutes les itérations qu'effectue le système, la méthode itérative se base sur la génération d'un signal de commande $U_i(t)$ (l'indice i est relation avec l'itération) en fonction du signal de commande de l'itération précédente $U_{i-1}(t)$ et de l'erreur de poursuite de l'itération précédente, notée $e_{i-1}(t)$. La figure suivante illustre le concept de base de cette méthode.

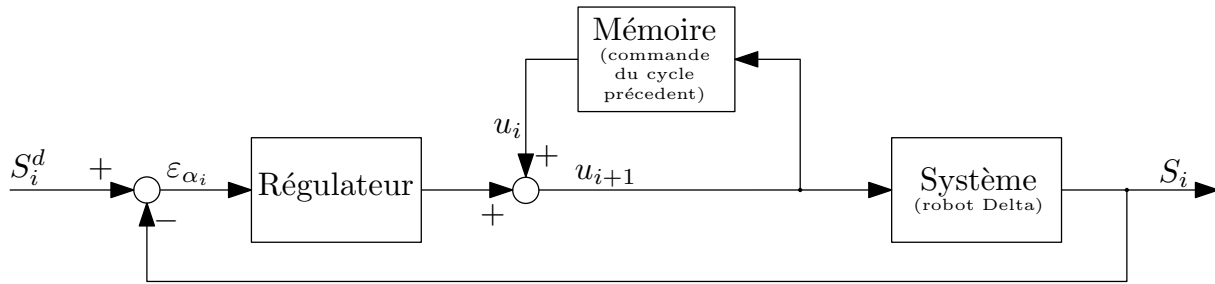


FIGURE V.38: Principe de fonctionnement de la commande itérative

La formulation mathématique de cette loi de commande est la suivante :

$$U_{i+1} = U_i + \Phi(\varepsilon_i) \quad (\text{V.2})$$

ou $\Phi(\varepsilon)$ représente, dans le cas général, une fonction de l'erreur sur la trajectoire ε_i . Cette fonction peut être soit linéaire (dans la plupart des cas), soit prenant la forme d'une fonction mathématique particulière (voir [XT03]).

7.1.1 Topologies de la commande par apprentissage itératif

Différentes variantes de la commande ILC (Itérative Learning Control ou commande par apprentissage itératif) ont été développées, selon que l'on utilise l'erreur actuelle seulement (Current Cycle Learning), méthode illustrée dans la Figure V.38, ou que l'on utilise l'erreur courante et précédente (Current and Previous Cycle Learning). Il est aussi possible d'utiliser une implémentation en cascade par rapport à un système de commande déjà présent, auquel on ne peut apporter de changement interne (régulateur commercial). Les architectures externes de la commande itérative sont présentées d'une manière plus explicite dans [XKH09].

7.1.2 Conditions de fonctionnement de la commande ILC

Afin que la commande par apprentissage itératif soit efficace (convergence de la sortie S_i vers la sortie désirée S_d pour $i \rightarrow \infty$), un certain nombre de conditions doivent être vérifiées pour garantir la convergence de la sortie S , ces conditions sont les suivantes :

a. Continuité LIPSCHITZIÈNE globale (GLC) Cette condition est directement issue de la théorie des suites numérique, sa vérification est **nécessaire** pour que la convergence de la commande soit assurée. L'expression mathématique de cette condition est la suivante :

soit un système (non linéaire dans le cas réel) dont la dynamique est décrite par les équations d'état suivantes :

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), t) & \text{avec } x(0) &= x_0 \\ y(t) &= g(x(t), u(t), t) \end{aligned} \quad (\text{V.3})$$

Avec $t \in [0, T]$ qui est la période d'itération du processus, $x(t) \in \mathfrak{R}^n$, $u(t) \in \mathfrak{R}^m$ et $y(t) \in \mathfrak{R}^p$ ($m = p = 1$ pour le cas d'un système monovariable).

La condition de continuité LIPSCHITZIÈNE globale stipule que les fonctions $f(x, u, t)$ et $g(x, u, t)$ sont continues au sens de LIPSCHITZ par rapport à leur variables respectives (x, u, t) , c'est à dire qu'il existe pour les deux fonctions f et g , deux constantes L_f et L_g tel que :

$$\begin{aligned} \|f(x_1, u_1) - f(x_2, u_2)\| &\leq L_f(\|x_1 - x_2\| + \|u_1 - u_2\|) \\ \|g(x_1, u_1) - g(x_2, u_2)\| &\leq L_g(\|x_1 - x_2\| + \|u_1 - u_2\|) \end{aligned} \quad (\text{V.4})$$

En pratique, il peut être difficile de vérifier la condition de continuité LIPSCHITZIÈNE pour des systèmes non linéaires, on utilise alors un petit artifice mathématique qui consiste à dériver les deux fonctions f et g par rapport à leur variables respectives et à appliquer le théorème des accroissements finis, on obtient alors :

$$\|f(x_1, u_1) - f(x_2, u_2)\| \leq \left\| \frac{\partial f}{\partial x} \right\| \|x_1 - x_2\| + \left\| \frac{\partial f}{\partial u} \right\| \|u_1 - u_2\| \quad (\text{V.5})$$

avec $x \in [x_1, x_2]$ et $u \in [u_1, u_2]$.

Dans le cas ou les deux dérivées sont bornées (système physique de surcroît stable par la boucle de retour), on peut toujours choisir une constante L_f tel que

$$L_f \geq \max\left\{ \left\| \frac{\partial f}{\partial x} \right\|, \left\| \frac{\partial f}{\partial u} \right\| \right\}$$

Ce qui vérifie, bien évidemment, la condition de continuité LIPSCHITZIÈNE globale citée dans (V.4). La même démarche est appliquée pour le cas de la fonction g afin d'aboutir à la vérification de la condition de continuité LIPSCHITZIÈNE et de caractériser la constante L_g .

b. Identité des conditions initiales (IIC) Les conditions initiales jouent un rôle fondamental dans l'utilisation d'une commande ILC.

Cette conditions est aussi **nécessaire** pour garantir le bon fonctionnement de la commande itérative, c'est à dire sa convergence vers la consigne désirée S_d .

Cette condition stipule que les conditions initiales des différentes itérations x_i^0 doivent être les même, le cas contraire conduirait à la divergence de la sortie S_i lorsque $i \rightarrow \infty$ selon (voir [XT03]).

7.2 Application de la commande sur le Delta

L'application de la commande itérative sur le robot Delta se fera selon le schéma Simulink suivant :

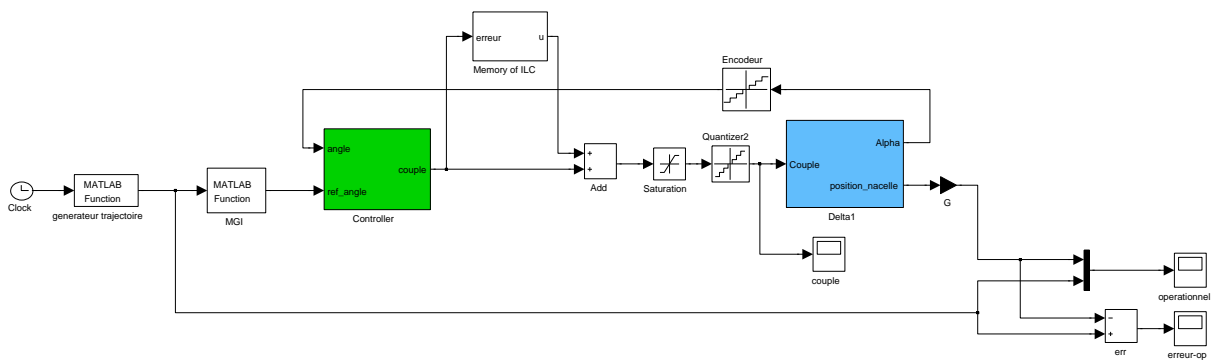


FIGURE V.39: Schéma de l'implémentation de la commande ILC sur le Delta

Nous remarquons que le schéma utilisé pour implémenter la commande ILC sur le robot Delta est légèrement différent du schéma de commande de base présentée précédemment, en effet, après avoir effectuer plusieurs tests sur le Delta en utilisant la commande ILC de base, les résultats obtenus ont été assez peu convainquant avec un faible taux de diminution de l'erreur, la variation rapide des paramètres dynamiques du Delta n'a vraisemblablement pas concordé avec les conditions de fonctionnement de cette méthode.

Il nous fallait donc adopter une autre démarche, afin que la méthode itérative soit utilisable et que les résultats des commandes classiques soient vraiment améliorés.

La contre performance de ces méthodes nous a amené à comprendre que la variation de la dynamique du Delta avait un effet très important sur cette commande et c'est la non prise en compte de cette dynamique qui engendre le dysfonctionnement de cette méthode. Il fallait donc prendre en compte cette variation de dynamique (en faisant varier le gain k de la commande en fonction du point de fonctionnement du robot), cette démarche a été testée en premier lieu mais la aussi, notre essai a été infructueux à cause de la méconnaissance et de la complexité des paramètres dynamiques du robot (modèle dynamique approximé), approximation qui peut avoir un effet inverse (divergence).

Après plusieurs recherches sur l'application des algorithmes itératifs sur les robots manipulateurs, nous avons vu que la loi d'apprentissage utilisée pour ces derniers avait une architecture particulière (voir [KNS91]), nous nous sommes inspiré de ces travaux pour appliquer l'algorithme de commande itératif sur le robot Delta, le schéma de commande correspond à ce qui a été présenté sur la figure V.39.

La forme de cette commande est donnée par la formule suivante (voir aussi figure V.1) :

$$\begin{cases} u_{i+1} = u_i + K \times \phi(e) \\ \phi(e) = K_p e + K_d \dot{e} \end{cases} \quad (\text{V.6})$$

Avec :

- K : facteur caractéristique de la commande itérative ;
- K_p, K_d : gain du régulateur PD temporel.

Nous avons donc deux régulateur PD, l'un par rapport au temps et l'autre par rapport aux itérations.

Comme test préliminaire nous appliquerons cette méthode sur une trajectoire selon l'axe x d'une longueur de 10 cm parcouru en 0,1 s. Les résultats de ce test sont présentés sur la figure V.40 (valeur de l'erreur opérationnelle).

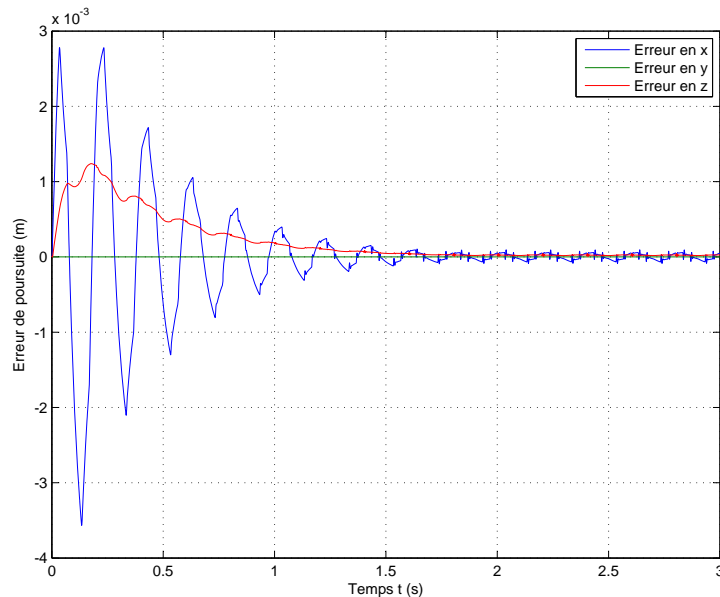
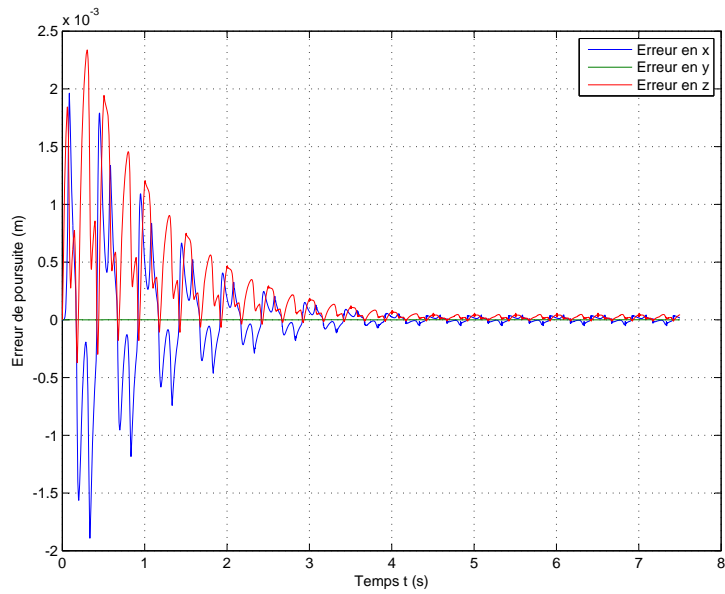


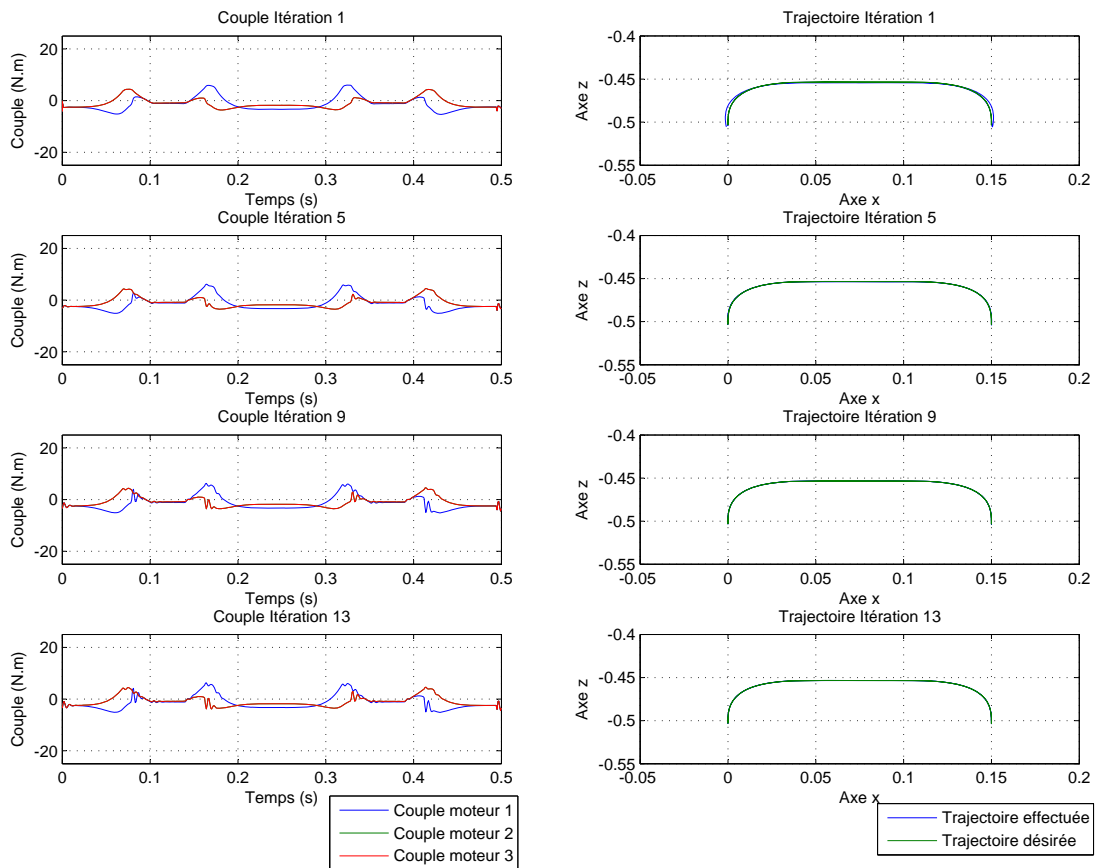
FIGURE V.40: Erreur en fonction du temps (15 itérations) - trajectoire selon x

Le test de cette méthode sur la trajectoire de prise-dépose en clothoïde (figure V.29) avec une cadence de 2Hz et de 4Hz est donnée sur la figure V.41 et V.42 qui représentent :

- les valeurs des erreurs opérationnelles pendant le processus d'apprentissage ;
- Les couples mis en jeux lors des itérations, afin de voir le processus d'apprentissage qui est effectué par la commande ILC.

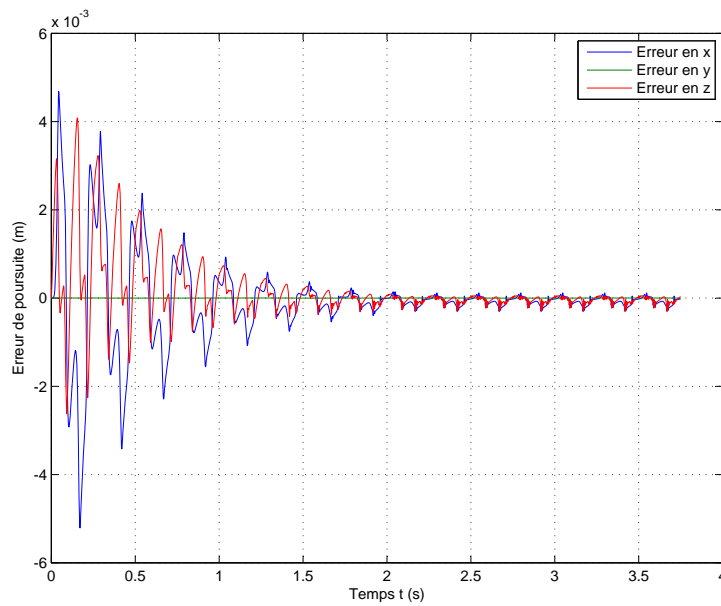


(a) Erreurs de poursuite

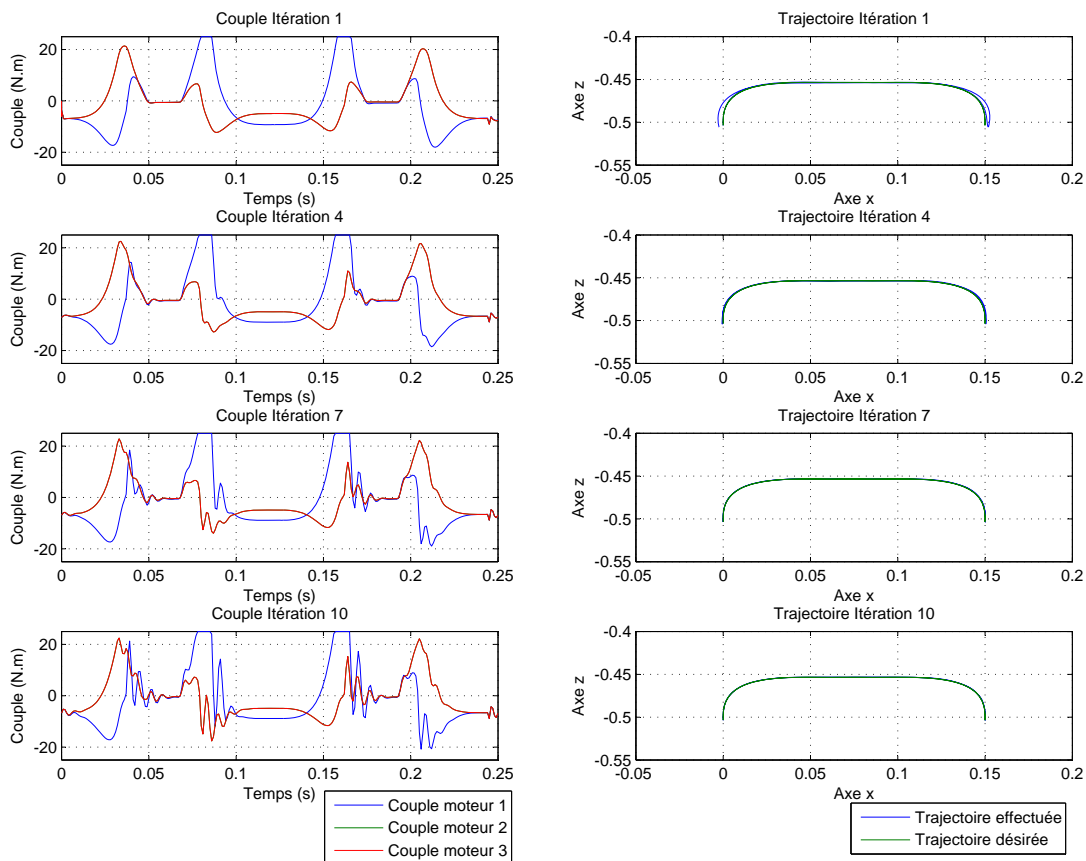


(b) Couples de commande et trajectoires

FIGURE V.41: Test de la commande ILC (15 itérations) - trajectoire prise-dépose à 2Hz



(a) Erreurs de poursuite



(b) Couples de commande et trajectoires

FIGURE V.42: Test de la commande ILC (15 itérations) - trajectoire prise-dépose à 4Hz

Les tableaux VI.1 et VI.2 en annexe A donnent l'évolution des erreurs maximales,

moyennes et les erreurs RMS sur parcours suivant les itérations ⁵ (cas $2Hz$ et $4Hz$).

Pour le cas $2 Hz$, Nous remarquons qu'après treize itérations, l'erreur atteint moins de 1% de l'erreur initiale (erreur du régulateur Proportionnel Dérivé seul) et que cette dernière est de l'ordre de la résolution incrémentale. La résolution finie des capteurs engendre une erreur résiduelle entre deux incréments successifs, l'effet intégrateur de la commande itérative pourrait alors faire diverger la sortie (phénomène en V).

De plus, la saturation du couple peut aussi avoir cet effet là. En effet, cela empêche le contrôleur itératif d'annuler l'erreur ce qui le fait diverger par la suite. On vérifie ça dans le cas de la cadence à $4 Hz$. L'erreur minimale est atteinte alors en 10 itérations, après cela le couple de commande est saturé.

Il suffit alors d'arrêter l'algorithme dès-que la résolution incrémentale est atteinte (après 13 itérations) pour la cadence de $2 Hz$. De même que pour la cadence de $4 Hz$ (après 10 itérations) à cause de la saturations du couple (voir [V.42b](#)).

Cette méthode a permis tout en utilisant un correcteur très simple (PD) de ramener l'erreur au niveau de l'erreur obtenue avec la double boucle avec à priori de couple. L'inconvénient c'est que cette méthode reste très contraignante de part ses suppositions (même trajectoire, même conditions).

7.3 Commentaires et conclusion

L'application d'une commande par apprentissage itératif amène un certain nombre de remarques, notamment par rapport aux conditions que doit vérifier le système pour assurer le bon fonctionnement de la méthode.

Les processus industriels ayant recours aux robots (et notamment ceux utilisant le Delta) intègrent dans la plupart des cas, un contrôle visuel permettant de rendre le robot encore plus efficace et plus flexible, notamment pour situer les objets qui doivent être manipulés, ceci veut dire que le point de départ (d'une itération) varie d'une manipulation à une autre, ceci est aussi vrai pour le point d'arrivée, qui peut éventuellement varier d'une itération à l'autre impliquant la variation de la trajectoire (puisque la distance entre le point de prise et de dépose varie).

Ces deux constatations nous amène à dire, et ce malgré la convergence de l'erreur de poursuite vers une valeur minimale, que cette méthode reste restrictive de part ses suppositions, les recherches à venir visant à améliorer cette méthode doivent donc s'orienter impérativement vers la résolution de ces problèmes de manque de flexibilité.

5. Une itération : un aller-retour sur la trajectoire de test du robot.

Conclusion

Notre travail s'est focalisé sur deux aspects du robot manipulateur parallèle Delta : la modélisation et la synthèse des commandes.

Pour ce qui est de modélisation, la caractérisation du robot a englobée plusieurs niveaux de modélisation, nous nous sommes principalement intéressés pour notre part au modèle géométrique, cinématique et dynamique de la structure.

Cette modélisation nous a permis de mettre en évidence les particularités d'un robot parallèle comme le Delta et ceci sur plusieurs points :

- La modélisation géométrique ne fait pas appel à des méthodes systématiques comme c'est le cas pour les robots séries mais à l'utilisation des formes géométriques caractérisant les liaisons passives et actionnées du robot (cercle pour un pivot et sphère pour une rotule). L'analyse des contraintes de la boucle du robot parallèle amène à lier ces équations pour en tirer le modèle géométrique. Le développement de ces méthodes sera donc particulier à chaque robot parallèle.
- Vu la complexité du modèle géométrique, la mise en évidence du modèle cinématique n'est pas aisée pour le cas du robot Delta, l'approximation numérique de celui-ci a donné cependant les résultats escomptés.
- Pour ce qui est du modèle dynamique, la difficulté essentielle pour le cas du robot Delta réside dans le mouvement de ses barres parallèles. Ces mouvements étant très complexes, cela amènerait l'utilisation d'un torseur (mouvement spatial) afin de les définir. Nous avons donc choisis d'utiliser une approximation qui consiste à reporter les masses des barres parallèles sur la nacelle et le bras, facilitant ainsi le développement du modèle. La validité du modèle a été ensuite confirmée.

Afin de compléter ce travail, on se devait d'avoir un autre modèle du robot que celui découlant des équations analytiques. Notre choix s'est alors porté sur l'utilisation du logiciel SimeMechanics qui nous a permis de construire de façon indépendante des équations dynamiques, un modèle de simulation et de visualisation 3D du robot.

Avec cette double modélisation (mécanique ou structurelle sous SimeMechanics et analytique) nous avons pu valider la totalité des équations géométriques et dynamiques du robot et avoir ainsi une base solide en vue du développement des lois de commande.

En adoptant cette démarche, notre souci était de s'approcher au plus près du cas réel. Nous avons trouvé inapproprié de commander un modèle de simulation analytique par ses mêmes équations analytiques. En adoptant l'utilisation du modèle SimeMechanics, le système de commande est plus réaliste.

Dans un second temps, nous nous sommes fixés des objectifs de réalisation de tâches bien précises (prise-dépose pour le robot Delta) et nous avons développé les commandes et la génération de trajectoire nécessaire à cela.

Pour ce qui est du régulateur, après une rapide analyse sur les régulateur PID, double boucle vitesse-position et double boucle vitesse-position avec à priori de couple notre choix s'est porté sur cette dernière variante. Afin de s'approcher du cas réel, nous avons ajouté au régulateur une quantification et un échantillonnage des commandes et des mesures.

Une partie importante de notre travail a été ensuite consacrée à la quantification des performances de notre commande. La commande double boucle avec à priori de couple à été testée et éprouvée sous plusieurs aspects de performances, les résultats obtenus sont très intéressants.

Ces tests nous amènent à dire que cette commande est adaptée au cas du robot Delta (robot rapide et précis), c'est pour cela qu'elle y est très largement utilisée.

Nous avons ensuite apporté notre contribution en testant un autre type de commande : la commande par apprentissage itératif. Cette commande adapté au seul cas des tâches répétitives, implémente un régulateur temporel simple et tire profit de la répétition de la tâche afin de s'améliorer au fur à mesure des itérations.

Les différents tests réalisés nous amènent à conclure que le robot Delta présente de forts couplages dynamique et qu'une commande simple (type PID) spécifiquement pour les cadences de travail élevés est incapable de donner de bonnes performances de suivi trajectoire. Bien que la commande itérative ait donné de bons résultats, la double boucle avec à priori de couple est plus flexible et permet des opérations de prise-dépose à des positions différents (utilisation avec la vision) et cela sans passer par une phase d'apprentissage.

Annexes

ANNEXE A : Résultats des différents tests (tableaux)

Itérations	Erreur Max (<i>mm</i>)	Erreur Moy (<i>mm</i>)	Erreur RMS (<i>mm</i>)
1	2,337	1,007	1,221
2	1,943	0,827	1,003
3	1,205	0,515	0,622
4	0,759	0,321	0,386
5	0,467	0,200	0,240
6	0,294	0,124	0,149
7	0,183	0,077	0,092
8	0,128	0,048	0,057
9	0,081	0,030	0,035
10	0,057	0,019	0,022
11	0,042	0,011	0,014
12	0,036	0,007	0,009
13	0,035	0,006	0,007

TABLE VI.1: Évolution de l'erreur Max, Moy et RMS suivant les itérations (ILC) - cadence de 2 Hz

Itérations	Erreur Max (<i>mm</i>)	Erreur Moy (<i>mm</i>)	Erreur RMS (<i>mm</i>)
1	5.212	2.361	2.735
2	3.777	1.827	1.995
3	2.378	1.148	1.256
4	1.557	0.726	0.802
5	1.075	0.463	0.519
6	0.754	0.299	0.345
7	0.540	0.196	0.236
8	0.397	0.136	0.168
9	0.302	0.102	0.126
10	0.263	0.080	0.099

TABLE VI.2: Évolution de l'erreur Max, Moy et RMS suivant les itérations (ILC) - cadence de 4 Hz

$\theta/\Delta\theta$	$\rho/\Delta\rho$	$z/\Delta z$	Erreur Max (mm)	Erreur Moy (mm)	Erreur RMS (mm)
0	0	0	0.3313	0.1319	0.1496
0	0	1	0.3652	0.1488	0.1658
0	0	2	0.3793	0.1497	0.1702
0	1	0	0.3263	0.1457	0.1630
0	1	1	0.3692	0.1588	0.1758
0	1	2	0.3611	0.1580	0.1738
0	2	0	0.3377	0.1521	0.1693
0	2	1	0.3825	0.1841	0.2031
0	2	2	0.3795	0.1952	0.2110
0	3	0	0.4105	0.1836	0.2065
0	3	1	0.4353	0.2250	0.2505
0	3	2	0.3944	0.2334	0.2546
1	1	0	0.3469	0.1403	0.1576
1	1	1	0.3671	0.1575	0.1740
1	1	2	0.3801	0.1665	0.1817
1	2	0	0.3327	0.1518	0.1676
1	2	1	0.3475	0.1754	0.1905
1	2	2	0.3675	0.1895	0.2016
1	3	0	0.3306	0.1781	0.1938
1	3	1	0.4045	0.2220	0.2410
1	3	2	0.3966	0.2462	0.2643
2	1	0	0.3439	0.1382	0.1556
2	1	1	0.3609	0.1532	0.1691
2	1	2	0.3760	0.1601	0.1756
2	2	0	0.3080	0.1464	0.1607
2	2	1	0.3449	0.1688	0.1819
2	2	2	0.3547	0.1876	0.1977
2	3	0	0.3584	0.1919	0.2094
2	3	1	0.3341	0.2118	0.2285
2	3	2	0.4045	0.2568	0.2736

TABLE VI.3: Évolution de l'erreur Max, moyenne et RMS suivant le point de test dans le volume de travail

ANNEXE B : Résultats des tests de variation de charge

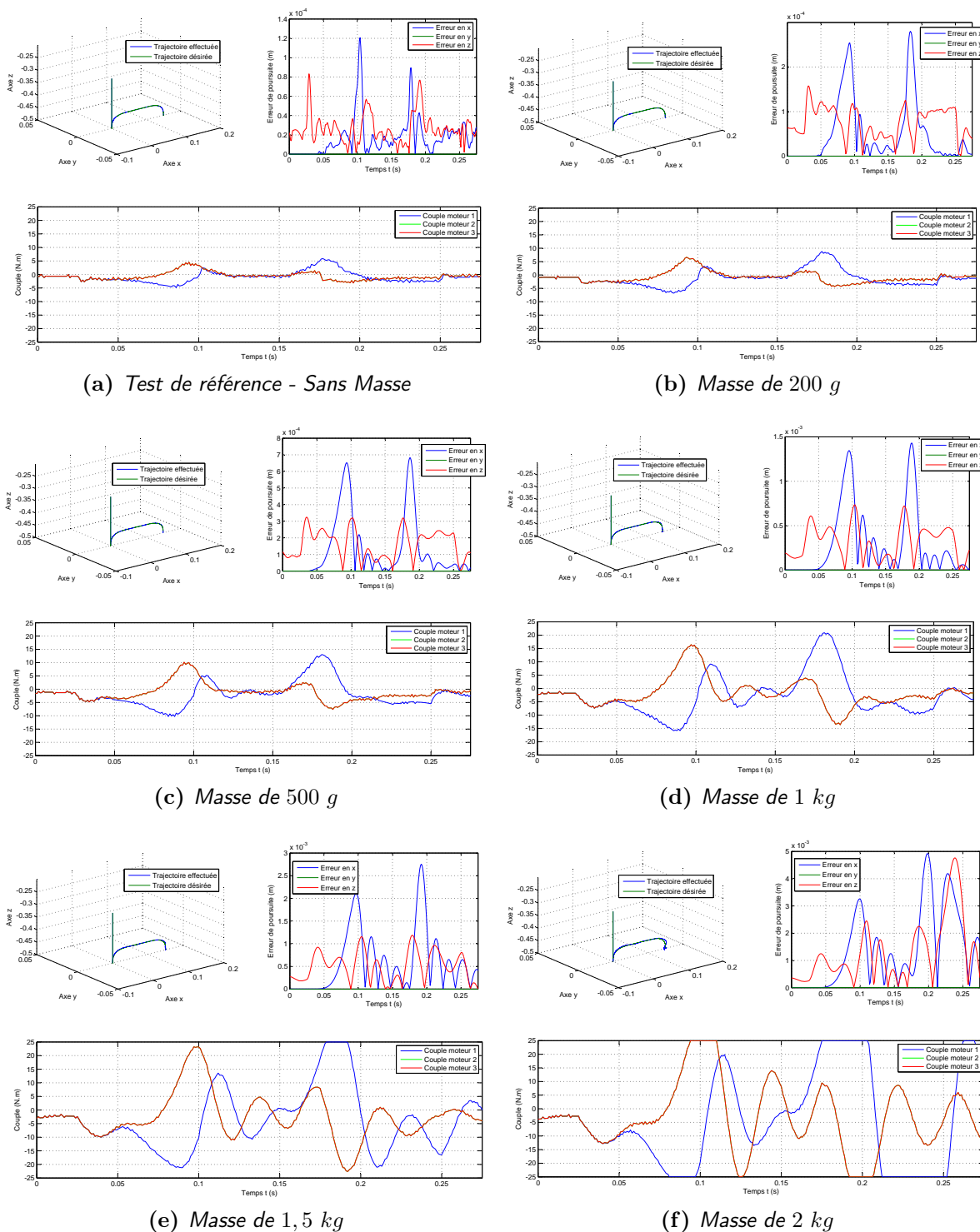
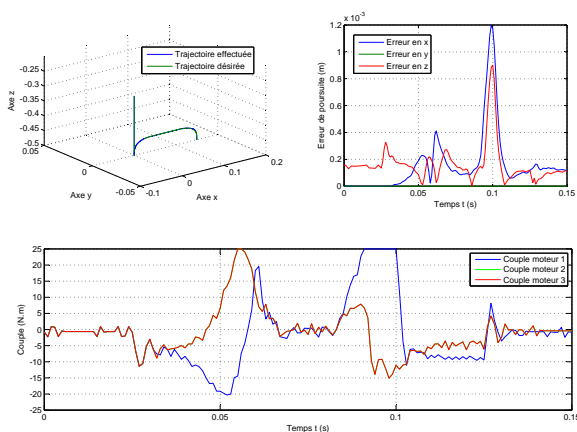
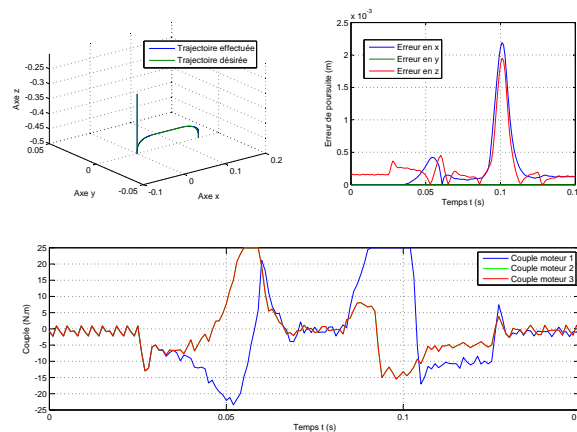


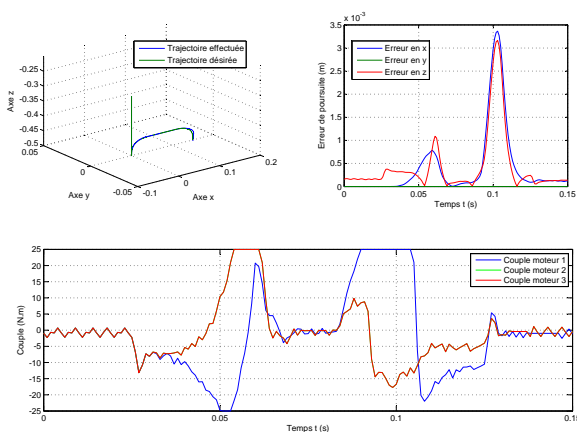
FIGURE VI.1: Tests pour différentes charges à une cadence de 2 Hz



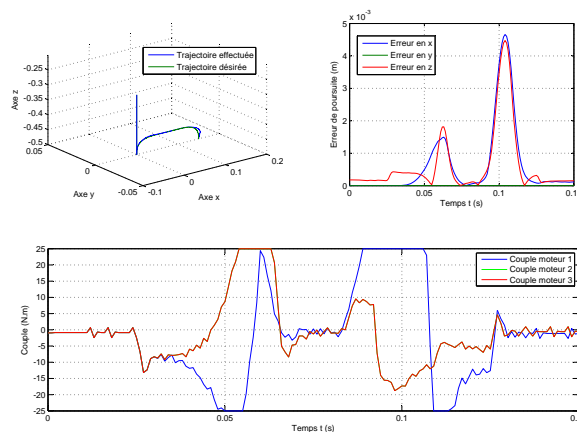
(a) Test de référence - Sans Masse



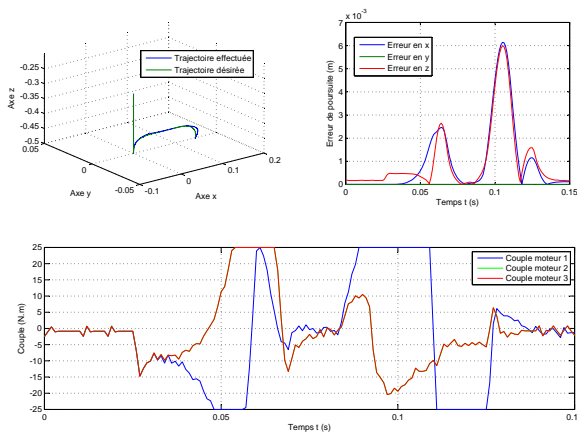
(b) Masse de 50 g



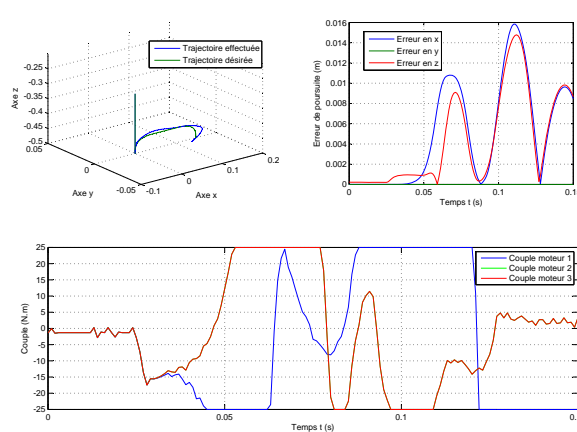
(c) Masse de 100 g



(d) Masse de 150 g



(e) Masse de 200 g



(f) Masse de 500 g

FIGURE VI.2: Tests pour différentes charges à une cadence de 4 Hz

Bibliographie

- [Bou08] *Bouakrif, Farah* : **Commande par Apprentissage Itératif des Robots Manipulateurs**. Thèse de doctorat, Ecole Nationale Polytechnique d'Alger, 2008.
- [Cla91] *Clavel, Raymond* : **Conception d'un robot parallèle rapide à 4 degrés de liberté**. Thèse de doctorat, Ecole Polytechnique Federale de Lausanne - EPFL, 1991.
- [Cod91] *Codourey, Alain* : **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.
- [EW07] *Etienne, Dombre and Wisama, Khalil* (editors) : **Robot Manipulators : Modeling, Performance Analysis and Control**. Control Systems, Robotics & Manufacturing Series. Wiley-ISTE, illustrated edition, January 2007.
- [KD99] *Khalil, Wisama et Dombre, Etienne* : **Modélisation, identification et commande des robots**. Hermes Sciences Publications, Paris, 2^e édition, 1999.
- [KNS91] *Kuc, Tae-yong, Nam, Kwanghee, and S. Lee, Jin* : **An iterative learning control of robot manipulators**. IEEE, 1991.
- [Mer97] *Merlet, Jean Pierre* : **Les robots parallèles**. Hermes Sciences Publications, Paris, 2^e édition, 1997.
- [MSM05] *Mark W., Spong, Seth, Hutchinson, and M., Vidyasagar* : **Robot Modeling and Control**. Wiley, November 2005.
- [SK08] *Siciliano, Bruno and Khatib, Oussama* (editors) : **Springer Handbook of Robotics**. Springer, 1st edition, June 2008.
- [XKH09] *Xu, Jian-Xin, K. Panda, Sanjib, and H. Lee, Tong* (editors) : **Real-time Iterative Learning Control**. Springer, 1st edition, 2009.
- [XT03] *Xu, Jian-Xin and Tan, Ying* (editors) : **Linear and Nonlinear Iterative Learning Control**. Springer, 1st edition, 2003.

Modélisation et Commande d'un robot parallèle de type Delta

Résumé

Ce mémoire d'ingénieur s'articule sur l'étude d'un robot industriel d'un genre particulier, ce robot appartient en effet à la classe des robots parallèles. Ce type de robots est souvent méconnue du grand public, ceci est dû au fait que le nombre d'applications utilisant ces robots reste assez restreint. Le robot Delta qui a fait l'objet de notre étude déroge quelque peu à cette règle, son comportement précis et en même temps rapide a fait de lui un robot très apprécié par certaines industries (microélectronique, horlogerie, chirurgie médicale, agroalimentaire . . .), il demeure certainement le robot parallèle le plus utilisé et le plus connu.

L'étude qui a été menée a débutée par une introduction générale sur les robots parallèles, par la suite, le robot Delta a été présenté en mettant un accent sur ses caractéristiques propres. Nous avons ensuite élaboré un modèle de simulation du Delta en utilisant l'outil de construction SimMechanics du logiciel Matlab-Simulink. Notre étude a été poursuivie par la modélisation du robot Delta. Le noyau de notre travail a été la partie consacrée à l'élaboration de lois de commande adaptées au cas du Delta (rapidité et précision).

Mots-clés : Robots manipulateurs, Robots parallèles, Delta, Commande des robots

Abstract

This engineering memory hinges on the study of a particular sort of industrial robot, in fact, this robot belongs to the family of parallel robots. This kind of robots is almost unknown from the general public, this is due to the restricted number of applications using this robots. The Delta robot, which was the subject of our study, is a little bit apart from this, the speed and the accuracy of his behavior had bring them very appreciate for some industrial applications (microelectronics, clock making, medical surgery, food processing. . .), he is certainly the most used and the well-known parallel robot.

The study had begun by a general introduction on parallel robots, then, the Delta robot was described with underlining his own characteristics. After that, we have developed à Delta simulation model by using SimMechanics toolbox of Matlab-Simulink. Next, we have pursued by modeling the Delta robot. The core of our work was the part devoted to the development of an appropriate control laws for the Delta robot (quickness and accuracy).

Keywords : Manipulator Robots, Parallel robots, Delta, Robot's Control

ملخص

تتطرق هذه المذكرة إلى دراسة آلية روبوتية صناعية من نوع خاص، فهي تنتمي إلى صنف الآليات المتوازية. هذا النوع من الآليات غير معروف في كثير من الأحيان لدى عامة الناس، وهذا بسبب العدد المحدود من التطبيقات التي تستخدم فيه هذه الروبوتات. الروبوت من نوع دلتا Delta الذي كان موضوع دراستنا قد لا يتماشى مع هذه القاعدة، سلوكه الدقيق والسريع جعله محل اهتمام الكثير من الصناعات (الإلكترونيات الدقيقة والساعات، الجراحة الطبية، الصناعة الغذائية) ويبقى من المؤكد أن الدلتا هو الآلية المتوازية الأكثر شهرة واستخداماً.

بدأت الدراسة التي قمنا بها بمقدمة عامة في الآليات المتوازية، ثم قدمنا الروبوت دلتا، مع التركيز بشكل أكبر على خصائصه. بعد ذلك قمنا بتطوير نموذج محاكاة للدلتا باستخدام أداة بناء ميكانيكية SimMechanics. وتابعتنا دراستنا بنمذجة الروبوت دلتا. أخيراً، تطرقنا للجزء الأساسي لعملنا المتمثل في تطوير قوانين تحكم تتماشى مع خصائص الدلتا (السرعة والدقة).

كلمات مفتاحية : الأذرع الآلية، الآليات المتوازية، الدلتا، التحكم في الآليات.
