

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de La Recherche Scientifique
Université Djilali Bounaama Khemis Miliana



Faculté des Sciences et de la Technologie
Département de la Technologie

Mémoire Présenté

Pour l'Obtention de Diplôme

Master
En AUTOMATIQUE

Spécialité Automatique et Informatique Industrielle

Thème

Commande Avancée d'un Robot Cartésien

présenté par

Mr. SADOUKI Aimen
Mr. ZOUAGHI Abdellah

Jury :

Mr. CHOUYA Ahmed	UDBKM	Examineur
Mr. IKNI Samir	UDBKM	Examineur
Mr. KERRACI Abdelkader	UDBKM.	Encadreur

Année Universitaire 2019/2020

Dédicace

Du fond du cœur, nous dédions ce travail à nos parents qui ont travaillé sans relâche pour faire de nous les hommes que nous sommes aujourd'hui, sans oublier de mentionner tous ceux qui ont participé de toutes les manières possibles à notre croissance aussi petite soit-elle.

C'est une victoire pour nous aujourd'hui, et nous espérons que c'est aussi une victoire pour vous tous!

Remerciements

Premièrement, nous remercions Dieu le tout puissant de nous avoir fourni la volonté et le pouvoir de mettre en place ce travail.

Nous sommes également reconnaissants à nos parents qui ont fait de nous les hommes que nous sommes aujourd'hui, et nous espérons qu'ils sont fiers de nous.

Nous sommes très reconnaissants à tous les professeurs qui nous ont accompagnés depuis notre première année primaire à l'âge de six ans, de jour en jour et surtout à notre promoteur dans ce travail, M. Abdelkader KERRACI, sans oublier nos enseignants, membres du jury qui ont accepté d'examiner ce travail.

Nous tenons également à remercier tous ceux qui ont contribué à ce travail directement ou indirectement et en particulier ceux qui ont établi les routes sur lesquelles nous marchons aujourd'hui.

Résumé

Nos travaux de recherche se sont focalisés sur la modélisation et la commande des robots cartésiens, leurs comportements et usages dans le domaine industriel. Les simulations ont été réalisées sur le modèle dynamique du robot cartésien que nous avons calculé sous l'environnement Matlab/Simulink. Deux lois de commande ont été appliquées au robot, la première est une loi de commande classique de type proportionnelle - intégrale (PD) et la deuxième loi est une loi basée sur le mode glissant qui est une loi de commande robuste. Les résultats de simulation ont été très satisfaisants, ce qui prouve l'exactitude du modèle calculé et la maîtrise des deux lois de commande.

Mots clés : Robot Cartésien, Modèle Dynamique, mode glissant, stabilité.

ملخص

يركز عملنا البحثي على نمذجة الروبوتات الديكارتية والتحكم فيها وسلوكها واستخداماتها في المجال الصناعي قمنا بإجراء عمليات المحاكاة على النموذج الديناميكي للروبوت الديكارتية والذي حسبناه في بيئة المحاكاة Matlab/Simulink. ثم طبقنا قانونين للتحكم على الروبوت، الأول هو قانون تحكم تكاملي تناسبية كلاسيكية (PD) والثاني هو قانون يعتمد على الوضع الانزلاقي وهو قانون تحكم شديد الدقة. وقد كانت نتائج المحاكاة مرضية للغاية، مما يثبت دقة النموذج المحسوب واتقان قانوني التحكم.

الكلمات المفتاحية: الروبوت الديكارتية، النموذج الديناميكي، الوضع المنزلق، الاستقرار.

Abstract

Our research work has focused on the modelling and control of Cartesian robots, their behaviour and uses in the industrial field. We carried out The simulations on the dynamic model of the Cartesian robot that we calculated under the Matlab/Simulink environment. Two control laws were applied to the robot, the first one is a classical proportional-integral (PD) control law and the second one is a law based on the sliding mode which is a robust control law. The simulation results were very satisfactory, proving the accuracy of the calculated model and the mastery of both control laws.

Keywords : Cartesian Robot, Dynamic model, sliding mode, stability.

Table des matières

Résumé	iii
Abstract	iii
Table des matières	iv
Liste des Notations et Abréviations	viii
1 Contexte Général	2
1.1 Introduction	2
1.2 Classification des robots industriels	3
1.3 Qu'est-ce qu'un Robot Cartésien	4
1.4 Applications	6
1.4.1 Solutions de prélèvement et placement	6
1.4.2 Tri	7
1.4.3 Solutions d'emballage et Palettisation	8
1.4.4 Processus d'assemblage	8
1.4.5 Les imprimantes cartésiennes 3D	9
1.5 Conclusion	9
2 Modélisation d'un Robot Cartésien	10
2.1 Introduction	11
2.2 Espace de travail d'un robot cartésien	11
2.3 Précision Statique de la Pose de l'effecteur	12
2.3.1 Résolution	12
2.3.2 Répétabilité	12
2.3.3 Précision	12
2.4 Modèle Géométrique d'un Robot Cartésien	12
2.5 Convention de Denavit Hartenberg (DH)	13

2.5.1	Paramètres de Denavit-Hartenberg	13
2.5.2	Matrice Homogène	14
2.5.3	Matrice Homogène Inverse	15
2.6	Modèle Géométrique :	15
2.6.1	Modèle Géométrique d'un Robot Cartésien	15
2.6.2	Modèle Géométrique Direct	16
2.6.3	Modèle Géométrique Inverse	17
2.7	Modèle Cinématique d'un Robot Cartésien	18
2.7.1	Modèle Cinématique Direct	18
2.7.2	Intérêts de la Matrice Jacobienne	18
2.7.3	Calcul de la Matrice Jacobienne :	19
2.7.4	Les Points Singuliers	19
2.7.5	Modèle Cinématique d'un Robot Cartésien	20
2.7.6	Transposée de la matrice Jacobienne	22
2.7.7	Modèle Cinématique Inverse	22
2.7.8	Matrice Jacobienne Inverse d'un Robot Cartésien	23
2.8	Modèle Dynamique d'un Robot Cartésien	23
2.8.1	Modélisation Dynamique	23
2.8.2	Formalisme de Lagrange	24
2.8.3	Forme Générale des Equations Dynamiques	24
2.8.4	Modèle dynamique avec frottement	25
2.8.5	Modèle dynamique d'un robot cartésien	26
2.9	Conclusion	27
3	Commande du Robot Cartésien	29
3.1	Introduction	29
3.2	Lois de commande classique	30
3.2.1	Loi de commande de type PD	30
3.2.2	Application au robot cartésien	31
3.3	Loi de commande robuste basée sur le mode glissant	37
3.3.1	Surfaces de glissement	38
3.3.2	Conception de la loi de commande	42
3.3.3	Stabilité	43
3.3.4	Simulation	44
3.4	Conclusion	49
	Conclusion	50
	Bibliographie	52

Table des figures

1.1	Principaux types de robots industriels	4
1.2	Robot cartésien(Source :Robotpark/CARTESIAN and GANTRY Robots)	5
1.3	Robot cartésien (Gantry Robot/Robot Portique)	5
1.4	Robot portique (Source :Robotpark/CARTESIAN and GANTRY Robots)	6
1.5	Part pick and place (Source :Yamaha-Motor/Cartesian robots)	7
1.6	Tri (Source :Yamaha-Motor/Cartesian robots)	7
1.7	Emballage et palettisation (Source :Thomson Linear Motion/Palletizing Motion Components)	8
1.8	Part assembly system yamaha-motors	8
1.9	Original Prusa Mini 3D Printer	9
2.1	Espace de travail d'un robot à portique (Macro Dynamic)	11
2.2	Modèle géométrique d'un robot cartésien suivant la convention de DH	14
2.3	Modèle géométrique d'un robot cartésien translation/rotations	15
2.4	Modèle géométrique en 3D	16
2.5	Modèle géométrique simplifier en 2D	16
2.6	Modèle géométrique inverse simplifier en 2D	18
2.7	Robot cartésien à trois degrés de liberté	20
2.8	Diagramme d'un robot cartésien à trois axes	20
3.1	Schéma synoptique de la commande type PD	30
3.2	Schéma utilisé pour la Simulation PD	33
3.3	Position et erreur de position de l'axe X , avec loi de commande PD	34
3.4	Position et erreur de position de l'axe Y , avec loi de commande PD	34
3.5	Position et erreur de position de l'axe Z , avec loi de commande PD	34
3.6	Position et erreur de position de l'axe X , avec des valeurs de gain plus élevées	35
3.7	Position et erreur de position de l'axe Y , avec des valeurs de gain plus élevées	35
3.8	Position et erreur de position de l'axe Z , avec des valeurs de gain plus élevées	36
3.9	Couple et vitesse de l'axe X , avec des valeurs de gain plus élevées	36

3.10	Couple et vitesse de l'axe Y , avec des valeurs de gain plus élevées	36
3.11	Couple et vitesse de l'axe Z , avec des valeurs de gain plus élevées	37
3.12	Condition de glissement	40
3.13	Représentation graphique de les équations (3.20) et (3.28) pour $n = 2$	41
3.14	Vibrations causé par les retards de commutation	41
3.15	Couche de limite	41
3.16	Schéma de la simulation MG	46
3.17	Position et erreur de position de l'axe X , avec des valeurs de gain plus élevées .	47
3.18	Position et erreur de position de l'axe Y , avec des valeurs de gain plus élevées .	47
3.19	Position et erreur de position de l'axe Z , avec des valeurs de gain plus élevées .	47
3.20	Couple et vitesse de l'axe X , avec des valeurs de gain plus élevées	48
3.21	Couple et vitesse de l'axe Y , avec des valeurs de gain plus élevées	48
3.22	Couple et vitesse de l'axe Z , avec des valeurs de gain plus élevées	48

Liste des Notations et Abréviations

Liste des notations

$C(q, \dot{q})\dot{q}$	Vecteur ($n \times 1$) des couples de Coriolis et des forces centrifuges
C_i	Corps i du robot
dq	Variations élémentaires des coordonnées articulaires
dX	Variations élémentaires des coordonnées opérationnelles
e	Vecteur d'erreur
\dot{e}	Erreur de vitesse
E	Énergie cinétique du système
η	Constante de conception
G	Vecteur des couples de gravité.
G_i	Fonction de modélé géométrique inverse du robot
J	Matrice jacobienne
$J(q)$	Matrice jacobienne de dimension ($m \times n$)
J^{-1}	Matrice jacobienne inverse
J^T	Matrice jacobienne transposée
J_{ij}	Élément (i, j) de la matrice jacobienne
K_f	Gain de du filtre
K_p	Gain de commande proportionnel
K_v	Gain de commande dérivé
\mathcal{L}	Lagrangien du robot
$M(q)$	Matrice inertie
\hat{M}	Matrice d'inertie estimée
$\hat{M}(q)$	Erreurs d'estimation de la matrice d'inertie
q	Variation articulaires
\dot{q}	Vitesse
q_d	Position désirée
\tilde{q}	Erreur de position
\dot{q}_r	Vecteur vitesse de référence
$\ddot{\tilde{q}}$	Erreur d'accélération
\ddot{q}	Accélération
$\dot{\tilde{q}}$	Erreur de vitesse

R_0	Origine du repère
R_i	Repère lié au corps C_i
Rot	Rotation
S	Ensemble des points d'équilibre
sat	Fonction saturation
$sign$	Fonction signe
t_f	Temps final
T_i^{i-1}	Matrice de transformation du repère R_i au repère R_{i-1}
T_r	Translation
τ	Vecteur du couple appliqué
U	Énergie potentielle du système
$u(x, t)$	Loi de commande à structure variable
u_{sw}	Un terme d'action de commutation
v_r	Terme de commande additionnel
x_d	État explicite variant dans le temps
x_e	Erreur de suivi de la variable x

Liste des abréviations

<i>CNC</i>	Machine à commande numérique
<i>CSV</i>	Commande à structure variable
<i>2D</i>	2 dimensions
<i>3D</i>	3 dimensions
<i>DH</i>	Devenait-Hartenberg
<i>MC</i>	Mode de convergence
<i>MG</i>	Mode de glissement
<i>MGD</i>	Modèle géométrique direct
<i>MGI</i>	Modèle géométrique inverse
<i>MRP</i>	Mode du régime permanent
<i>PD</i>	Loi de commande Proportionnelle - Dérivée
<i>SSV</i>	Système à structure variable

Introduction

«*Avec suffisamment d'innovation, les applications sont illimitées*», cette expression a été source d'inspiration derrière sur quoi nous avons choisi de travailler dans ce mémoire. C'est assez simple mais aussi vaste que l'horizon. Donc, au début, on a opté pour la simplicité et la fonctionnalité, et quoi de mieux adapté à cela dans le mot de la robotique autre que les robots cartésiens ou portiques. Ce sont de simples robots manipulateurs à trois degrés de liberté avec des applications illimitées, c'est ainsi que nous avons choisi notre sujet.

De notre point de vue, et peut-être même le vôtre, c'est un sujet tellement intrigant. Mais qu'y a-t-il à ajouter à ce domaine qui est largement exploité. Les robots cartésiens sont l'une des premières formes de robotique, et ils étaient en cours de développement depuis longtemps. Nous les avons vus dans divers domaines, et même dans certains endroits auxquels nous ne pouvions pas nous attendre, et en tant qu'étudiants, nous n'avons pas grand-chose à ajouter.

Pour répondre à cela, et comme la robotique est un domaine dans lequel il est très difficile d'entamer, nous avons décidé de passer par une étude assez simple pour toute personne ayant des connaissances de base dans le domaine.

Pour bien traiter notre sujet de projet de fin d'étude, nous avons opter la structure suivante :

Dans le premier chapitre, nous avons évoqué certaines utilisations les plus courantes de ce type de robots.

Dans le deuxième chapitre, nous avons commencé à plonger profondément dans ce que sont ces robots et dans quels domaines ils sont les meilleurs, comme la précision et la répétabilité. Cela nous a conduit vers la modélisation géométrique, cinématique et dynamique des robots à chaîne ouverte simple d'une façon générale.

Le troisième chapitre était un peu différent, et parce qu'au début nous avons opté pour la réalisation et l'étude d'un robot cartésien, mais en raison des circonstances imposées due a l'épidémie du Covid-19, nous avons adapté notre projet de fin d'étude à la simulation du modèle dynamique du robot cartésien que nous avons calculé dans le chapitre 2, sous l'environnement Matlab/Simulink. Deux types de lois de commande on été simulées, une première est de type classique et un deuxième de type avancé basée sur le mode glissant. Les résultats de simulations obtenus ont été très satisfaisants, ce qui prouve l'exactitude du modèle du robot calculé et la maîtrise des algorithmes des lois de commande utilisées.

Au final, nous avons effectué ce travail avec le souci de rendre les premiers pas dans ce domaine beaucoup plus stables et sans nuire à quoi que ce soit, car nous étions convaincus que l'innovation est la seule limite de nos jours.

Contexte Général

Sommaire

1.1 Introduction	2
1.2 Classification des robots industriels	3
1.3 Qu'est-ce qu'un Robot Cartésien	4
1.4 Applications	6
1.4.1 Solutions de prélèvement et placement	6
1.4.2 Tri	7
1.4.3 Solutions d'emballage et Palettisation	8
1.4.4 Processus d'assemblage	8
1.4.5 Les imprimantes cartésiennes 3D	9
1.5 Conclusion	9

1.1 Introduction

Un robot est un agent intelligent mécanique ou virtuel qui peut effectuer des tâches automatiquement ou avec des conseils par télécommande. Un robot a généralement la capacité d'entrée sensorielle (vision, toucher, etc.), de reconnaissance et de mouvement, ce qui signifie qu'un robot devrait au moins avoir des capteurs, des moteurs et des contrôleurs. Il existe plusieurs types de robots, souvent différents en fonction de la fonction, de l'axe, du degré de liberté, de l'espace de travail, etc... Les principaux types de robots incluent aujourd'hui, mais sans s'y limiter, les robots industriels, les robots militaires, les robots médicaux, les robots mobiles, les robots de service et micro et nano robots .

Le développement de machines à commande numérique et la popularité croissante de l'ordinateur ont contribué à la création des premiers robots industriels. Le premier robot industriel connu qui correspond à la définition ISO du terme a été créé par Griffith "Bill" P. Taylor en 1937 et est apparu dans Meccano Magazine l'année suivante [1].

Un robot industriel comme il es defini par **ISO 8373 : 2012** : est : *An automatically controlled, reprogrammable, multi-purpose manipulator programmable in three or more axes, which can be either fixed in place or mobile for use in industrial automation applications*". Ce que veut dire "Un manipulateur polyvalent, reprogrammable et à commande automatique, programmable sur trois axes ou plus, qui peut être soit fixe en place, soit mobile pour être utilisé dans des applications d'automatisation industrielle".

Les termes utilisés dans la définition ci-dessus sont expliqués en détail comme suit :

- **Reprogrammable** : conçu pour que les mouvements programmés ou les fonctions auxiliaires peuvent être modifiées sans altération physique.
- **Multifonction** : capable d'être adapté à une application différente avec une altération physique.
- **Altération physique** : L'altération du système mécanique (le système mécanique n'inclut pas les supports de stockage, les ROM, etc.)
- **Axe** : direction utilisée pour spécifier le mouvement du robot en mode linéaire ou rotatif.

1.2 Classification des robots industriels

À partir de l'an 2004 et en accord avec les fournisseurs de robots, les robots ne devraient être classés que par structure mécanique. [2]

- Robots linéaires (y compris robots cartésiens et portiques)
- Robots SCARA
- Robots articulés
- Robots parallèles (delta)
- Robots cylindriques
- Autres
- Non classés

La figure 1.1 illustre la configuration mécanique de ces types de robots. Ci-dessous, quelques explications et définitions supplémentaires sont données pour les différentes classifications par types de robots.

Le nombre d'axes doit être compris comme la caractéristique de base fournie par le producteur et non comme des axes ajoutés ultérieurement par l'utilisateur.

Les robots ventilés par structure mécanique sont basés sur les définitions suivantes :

Robot articulé : un robot dont le bras a au moins trois articulations rotatives.

Robot cylindrique : un robot dont les axes forment un système de coordonnées cylindriques.

Robot SCARA : un robot, qui a deux joints rotatifs parallèles pour assurer la conformité dans un plan.

Robot cartésien : robot dont le bras a trois articulations prismatiques et dont les axes coïncident avec un système de coordonnées cartésiennes.

Robot parallèle : un robot dont les bras ont des articulations prismatiques ou rotatives simultanées.

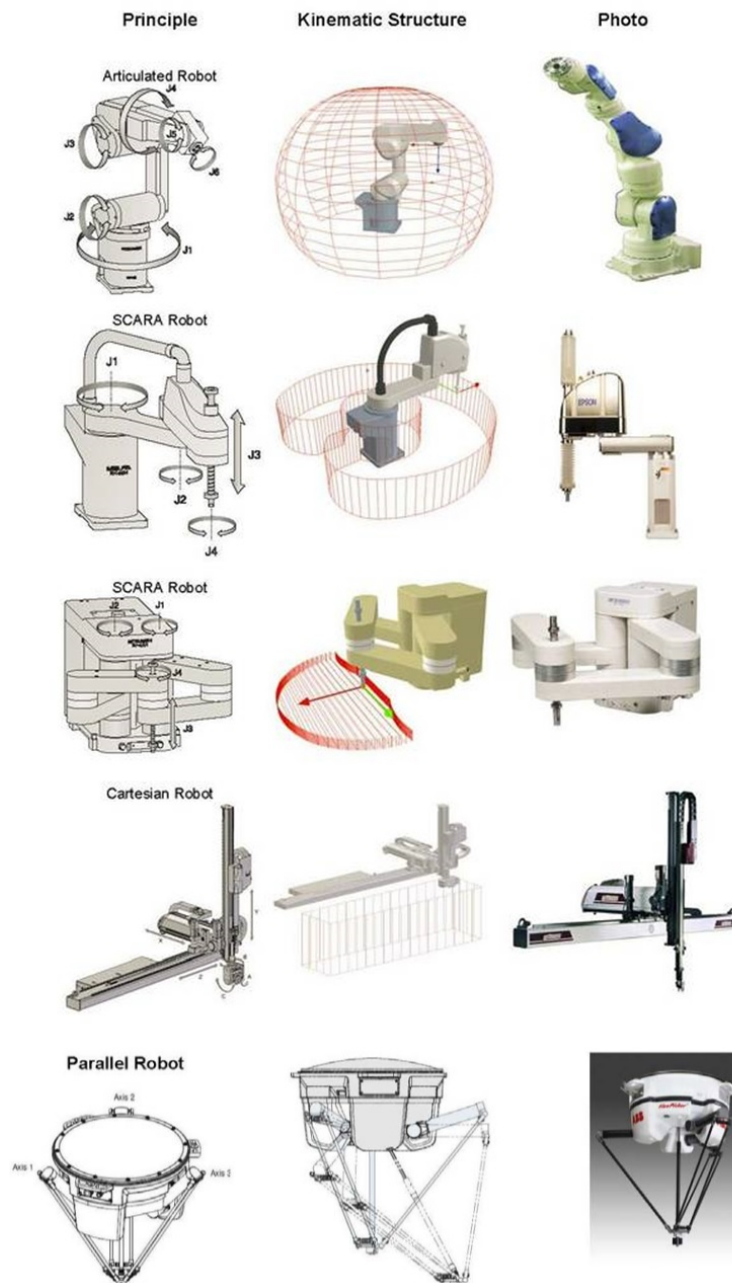


FIGURE 1.1 – Principaux types de robots industriels

1.3 Qu'est-ce qu'un Robot Cartésien

Après avoir présenter la définition des différents types de robots industriels fournie par l'organisation ISO, nous pouvons maintenant approfondir un peu plus notre étude concernant le type cartésien de robots industriels. Alors, qu'est-ce qu'un robot cartésien et à quoi peut-il servir ?!

Un robot à coordonnées cartésiennes (également appelé robot linéaire) est un robot industriel dont les trois principaux axes de contrôle sont linéaires (ils se déplacent en ligne droite plutôt que de tourner (prismatique)) et sont à angle droit les uns par rapport aux autres [3].

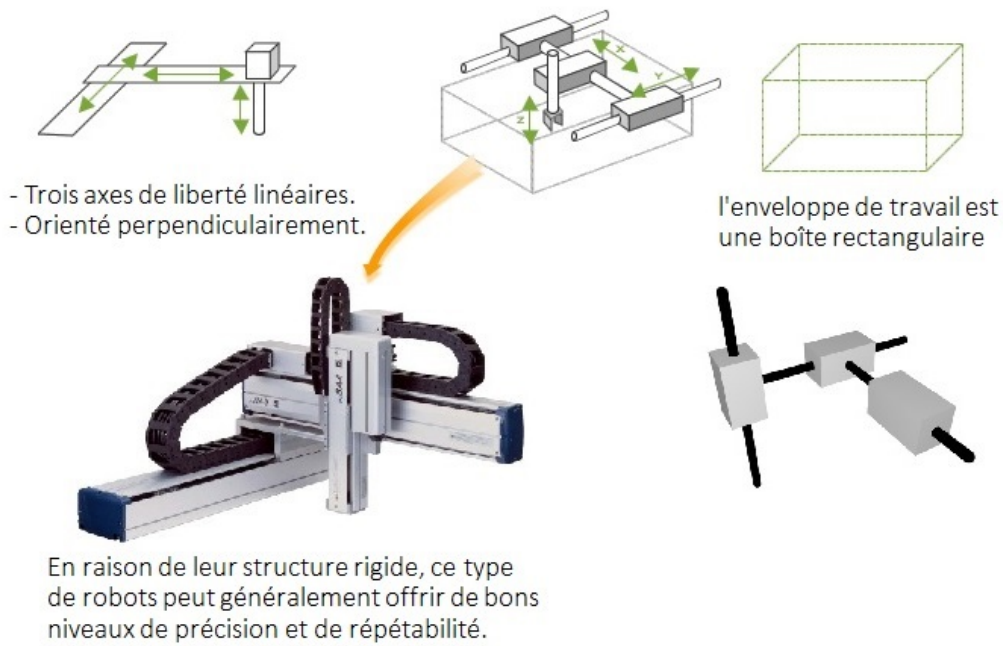


FIGURE 1.2 – Robot cartésien (Source : Robotpark/CARTESIAN and GANTRY Robots)

Les trois articulations coulissantes correspondent au déplacement du poignet de haut en bas, dedans-dehors, d'avant en arrière. Entre autres avantages, cette disposition mécanique simplifie la solution du bras de commande robot. Il a une fiabilité et une précision élevées lorsqu'il fonctionne dans un espace tridimensionnel [4]. En tant que système de coordonnées de robot, il est également efficace pour les déplacements horizontaux et pour l'empilement des bacs [5].

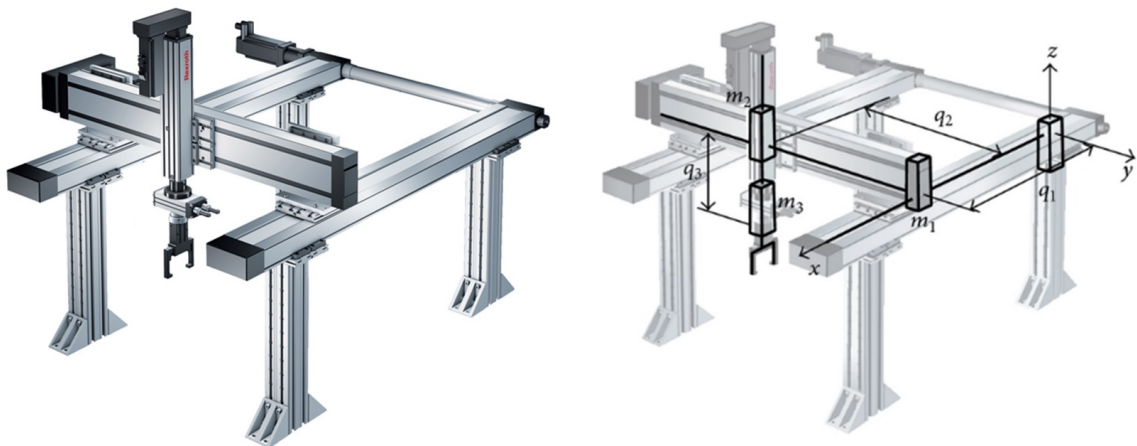


FIGURE 1.3 – Robot cartésien (Gantry Robot/Robot Portique)

Les robots à coordonnées cartésiennes avec l'élément horizontal soutenu aux deux extrémités sont parfois appelés robots à portique ; mécaniquement, ils ressemblent à des portiques, bien que ces derniers ne soient généralement pas des robots. Les robots à portique sont souvent assez grands [6].

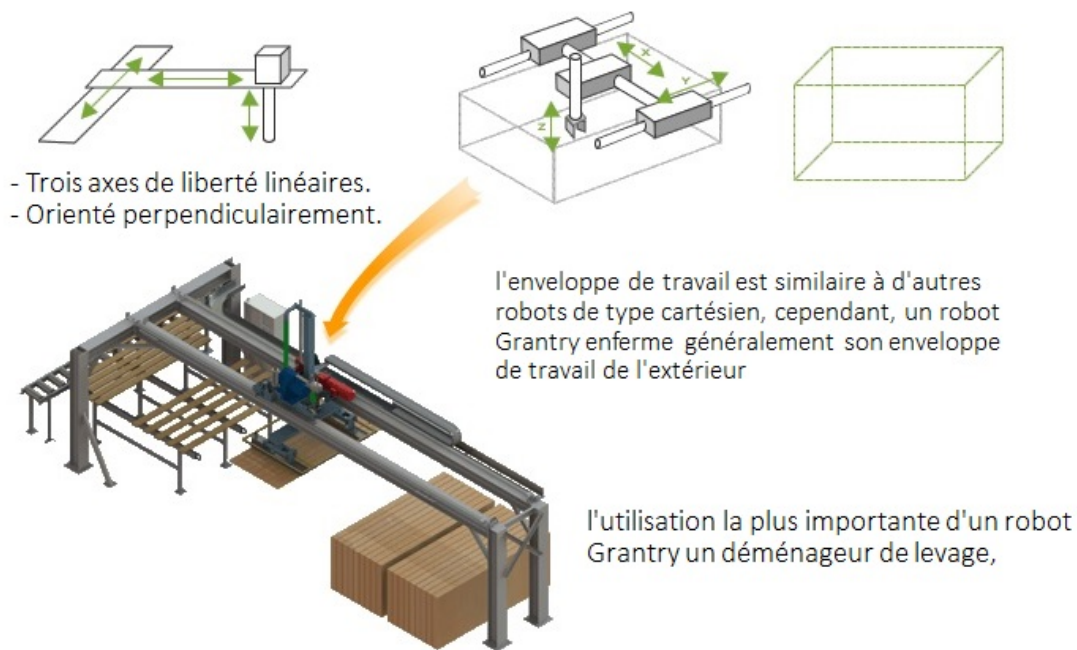


FIGURE 1.4 – Robot portique (Source :Robotpark/CARTESIAN and GANTRY Robots)

1.4 Applications

Avec l'absence d'axes rotatifs, les robots cartésiens ont tendance à avoir un degré de précision plus élevé, ce qui en fait la solution d'automatisation idéale pour ces tâches répétitives banales. Contrairement à d'autres machines d'automatisation, les systèmes de robotique linéaire peuvent être reprogrammés pour s'adapter rapidement aux changements de produits et sont flexibles pour répondre à des exigences uniques. Un robot cartésien peut être plus économique que d'autres types de robots tels que le bras articulé ou SCARA.

De nombreuses industries nécessitent aujourd'hui des processus automatisés pour garantir la répétabilité des processus, réduire la variabilité, répondre aux attentes de qualité et réduire les coûts de fabrication. L'automatisation est vitale pour de nombreuses entreprises pour répondre aux demandes de délais et rester compétitives. Les systèmes robotiques offrent le meilleur investissement en termes de coûts car ils peuvent être reprogrammés ou réutilisés pour répondre à la variété des produits et aux cycles de vie. La robotique linéaire est une solution polyvalente pour ces défis.

Il existe de nombreuses utilisations différentes de la robotique linéaire, mais les plus courantes sont :

1.4.1 Solutions de prélèvement et placement

Des robots de prélèvement et placement à grande vitesse déplacent un produit d'un endroit à un autre avec une précision parfaite. Les erreurs manuelles comme placer le mauvais article au mauvais endroit peuvent être remplacées par la robotique linéaire. La mise en œuvre d'un système "**pick and place**" pour placer les articles à certains endroits et sur les chaînes de montage peut améliorer considérablement l'efficacité et la précision[7] .

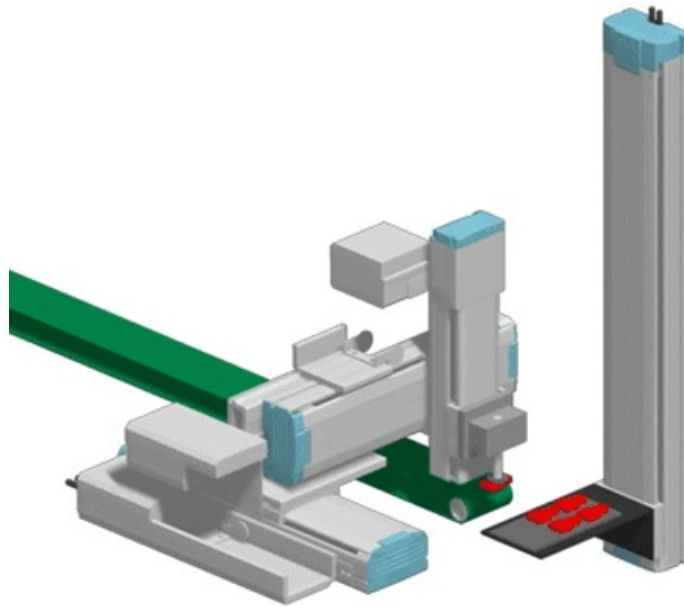


FIGURE 1.5 – Part pick and place (Source :Yamaha-Motor/Cartesian robots)

1.4.2 Tri

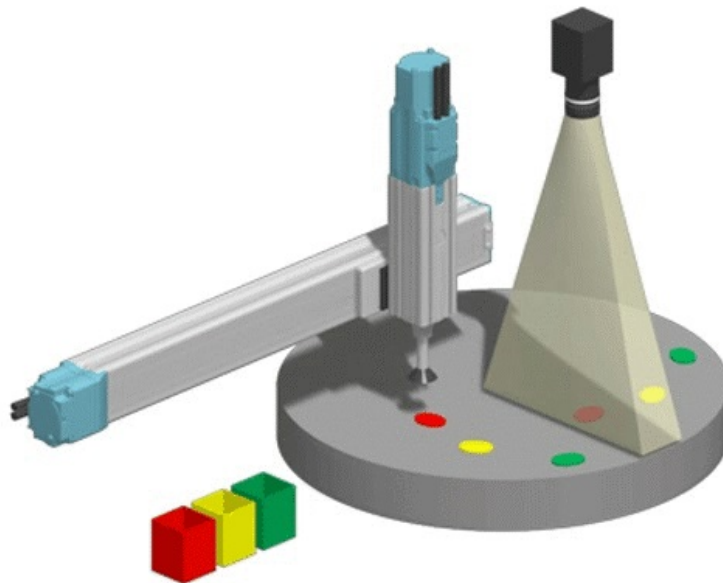


FIGURE 1.6 – Tri (Source :Yamaha-Motor/Cartesian robots)

Un robot linéaire peut rationaliser le processus de tri traditionnellement manuel et monotone, le rendant plus efficace et plus sûr. Lorsqu'il est couplé à un système de vision, des distinctions précises peuvent être faites avec une plus grande cohérence[7].

Les processus d'emballage peuvent fonctionner 24h/24 et 7j/7 lorsque nous utilisons un système de robotique linéaire, car nous pouvons théoriquement fonctionner même avec les lumières éteintes! ce qui va permettre à la production de fonctionner sans arrêt, ce qui est extrêmement difficile aux travailleurs à manipuler manuellement. C'est un excellent moyen d'améliorer le délai d'exécution.

1.4.3 Solutions d'emballage et Palettisation

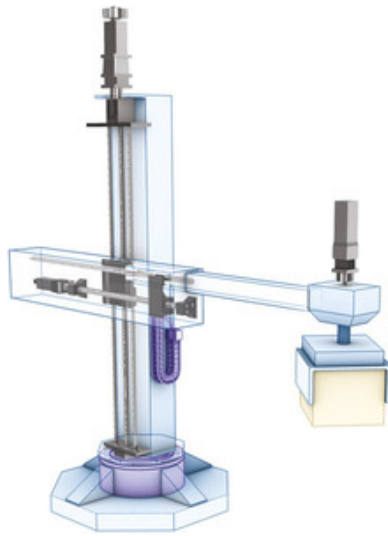


FIGURE 1.7 – Emballage et palettisation (Source :Thomson Linear Motion/Palletizing Motion Components)

Un palettiseur prend les produits et les place selon un modèle prédéterminé pour former des couches, puis les placer sur une palette. Sans automatisation, cela pourrait être un travail manuel dangereux de levage lourd. Les systèmes robotiques augmentent considérablement la productivité et la fiabilité des processus de palettisation. Ils ont un encombrement minimal de l'équipement et sont devenus une solution plus idéale pour un plus large éventail de scénarios d'emballage[8].

1.4.4 Processus d'assemblage

De nombreux processus tels que la distribution, la découpe, la formation, la soudure, etc. sont mieux exécutés par un système robotique linéaire, en particulier lorsque de longs déplacements et une portée étendue sont nécessaires[7].

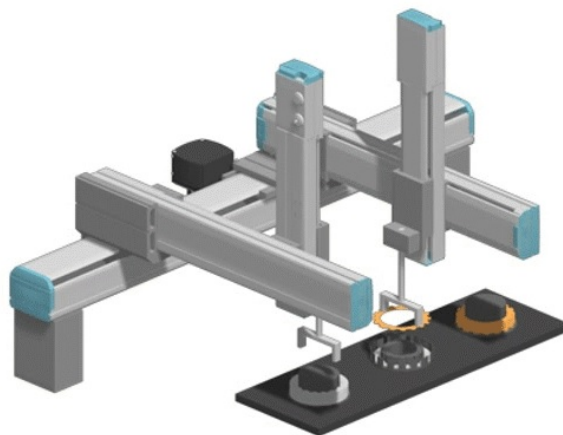


FIGURE 1.8 – Part assembly system yamaha-motors

1.4.5 Les imprimantes cartésiennes 3D

Une application populaire pour ce type de robot est une machine à commande numérique par ordinateur (machine CNC) et l'impression 3D. L'application la plus simple est utilisée dans les machines de fraisage et de dessin où un stylo ou un routeur se déplace sur un plan x-y tandis qu'un outil est soulevé et abaissé sur une surface pour créer une conception précise. Les machines "**Pick and Place**" et les traceurs sont également basés sur le principe du robot à coordonnées cartésiennes [9].



FIGURE 1.9 – Original Prusa Mini 3D Printer

1.5 Conclusion

Dans ce chapitre nous avons les définitions et les classifications concernant les robots industriels en général et des robots cartésiens en particulier.

Comme, nous avons donné des exemples des applications concernant notre cas particulier (d'un robot cartésien). Dans le chapitre suivant, nous allons calculer le modèle géométrique et dynamique des robots cartésiens.

Modélisation d'un Robot Cartésien

Sommaire

2.1	Introduction	11
2.2	Espace de travail d'un robot cartésien	11
2.3	Précision Statique de la Pose de l'effecteur	12
2.3.1	Résolution	12
2.3.2	Répétabilité	12
2.3.3	Précision	12
2.4	Modèle Géométrique d'un Robot Cartésien	12
2.5	Convention de Denavit Hartenberg (DH)	13
2.5.1	Paramètres de Denavit-Hartenberg	13
2.5.2	Matrice Homogène	14
2.5.3	Matrice Homogène Inverse	15
2.6	Modèle Géométrique :	15
2.6.1	Modèle Géométrique d'un Robot Cartésien	15
2.6.2	Modèle Géométrique Direct	16
2.6.3	Modèle Géométrique Inverse	17
2.7	Modèle Cinématique d'un Robot Cartésien	18
2.7.1	Modèle Cinématique Direct	18
2.7.2	Intérêts de la Matrice Jacobienne	18
2.7.3	Calcul de la Matrice Jacobienne :	19
2.7.4	Les Points Singuliers	19
2.7.5	Modèle Cinématique d'un Robot Cartésien	20
2.7.6	Transposée de la matrice Jacobienne	22
2.7.7	Modèle Cinématique Inverse	22
2.7.8	Matrice Jacobienne Inverse d'un Robot Cartésien	23
2.8	Modèle Dynamique d'un Robot Cartésien	23
2.8.1	Modélisation Dynamique	23
2.8.2	Formalisme de Lagrange	24
2.8.3	Forme Générale des Equations Dynamiques	24
2.8.4	Modèle dynamique avec frottement	25
2.8.5	Modèle dynamique d'un robot cartésien	26
2.9	Conclusion	27

2.1 Introduction

Dans l'étude de la robotique, l'une de nos préoccupations majeures est la localisation des objets dans l'espace tridimensionnel. Ces objets sont les liens du manipulateur, les pièces et outils avec lesquels il agit et d'autres objets dans l'environnement des manipulateurs. À un niveau brute mais important au même temps, ces objets sont décrits par seulement deux attributs : leur position et leur orientation.

Naturellement, un sujet d'intérêt immédiat est la manière dont nous représentons ces quantités et les manipulons mathématiquement.

Jusqu'à présent, l'un des principaux obstacles technologiques dans l'industrie de la robotique a été la réduction des erreurs entre le cadre de l'outil et le cadre de l'objectif[10]. La source de cette erreur est facilement identifiable et la modélisation des différences entre le contrôleur et le robot représente généralement la majorité des erreurs entre le cadre de l'outil et le cadre de l'objectif. Cependant, dans certains autres cas, l'inexactitude des pièces jointes et des défauts de fabrication peuvent expliquer les différences entre le positionnement du résultat et l'objectif.

2.2 Espace de travail d'un robot cartésien

L'espace de travail d'un robot cartésien est défini comme l'espace atteignable par l'effecteur final. La taille de l'espace de travail est fonction des facteurs suivants :

- Le nombre de degrés de liberté (généralement de 1 à 6 degrés de liberté (le cas où l'effecteur a trois axes de rotation (x , y et z)) ;
- La longueur des segments glissants constituant le robot ;
- Les limites des articulations (comme les butées mécaniques) ;
- Les interférences mécaniques entre ses éléments.

Cet espace de travail permet de vérifier que l'ensemble de la tâche est accessible. La pose X de l'effecteur final du robot définit la position et l'orientation de l'organe terminal dans son espace de travail. Le nombre de paramètres nécessaires pour définir la pose d'un corps rigide dans l'espace est de six : trois coordonnées définissant la position et trois coordonnées définissant l'orientation. Pour définir la position d'un point particulier de l'effecteur final, les coordonnées cartésiennes sont généralement utilisées. Pour exprimer l'orientation, plusieurs notations existent comme les angles d'Euler, les quaternions, les matrices de rotation ou les cosinus directeurs [11].

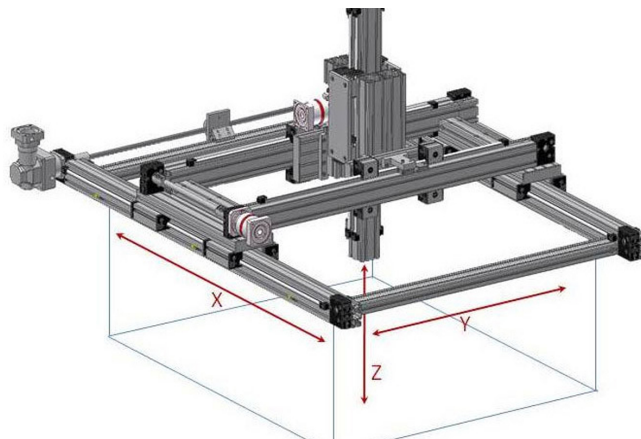


FIGURE 2.1 – Espace de travail d'un robot à portique (Macro Dynamic)

2.3 Précision Statique de la Pose de l'effecteur

Les paramètres identifiés liés à l'étalonnage robotique sont la précision, la répétabilité et la résolution. Chacun des cela dépend des différents composants utilisés (liens (liaisons), moteurs, codeurs, etc.), la procédure de construction, et la capacité du contrôleur.

2.3.1 Résolution

La résolution est définie comme le plus petit mouvement incrémentiel que le robot peut produire physiquement.

2.3.2 Répétabilité

La répétabilité est une mesure de la capacité du robot à revenir à la même position et l'orientation.

2.3.3 Précision

La précision est définie comme la capacité du robot à se déplacer avec précision vers la position souhaitée dans l'espace 3D.

La précision et la répétabilité absolues décrivent la capacité d'un robot pour se déplacer à un endroit souhaité sans aucune déviation. Précision dynamique et répétabilité décrivent la capacité d'un robot à suivre une trajectoire souhaitée avec peu ou pas de variance. De plus, comme dans tous les robots applications zéro dépassement est une nécessité pour éviter les collisions désastreuses avec d'autres parties de la cellule de travail [12].

2.4 Modèle Géométrique d'un Robot Cartésien

Le modèle géométrique est la description mathématique du comportement géométrique du robot. C'est un modèle qui exprime la pose du centre de l'outil par rapport à la base en fonction de la pose des articulations actives [13].

Il existe différents types de modèles géométriques [14] :

- Le modèle géométrique direct (MGD) qui exprime la pose de l'effecteur final x en fonction des variables articulaires q et des paramètres géométriques ε du robot :

$$x = MGD(q, \varepsilon)$$

- Le modèle géométrique inverse (MGI) exprime les variables articulaires q en fonction de la pose de l'effecteur final x et des paramètres géométriques ε du robot :

$$q = MGI(x, \varepsilon)$$

De nombreuses approches existent pour développer le modèle géométrique d'un robot, comme par exemple la convention de *Denavit-Hartenberg* (DH) [15], et la convention de *Khalil-Kleinfinger*, appelée aussi convention de *Denavit-Hartenberg modifiée* [16].

Les modèles géométriques peuvent être développés pour différentes applications et besoins spécifiques. Ainsi, il existe plusieurs méthodes pour décrire la pose de l'effecteur. Ces méthodes diffèrent par les techniques mathématiques appliquées et par la méthode de paramétrage.

Finalement on peut dire que la structure particulière du robot cartésien est liée à des caractéristiques attendues et influe sur la précision de la pose de l'outil. L'amélioration de la précision de pose de l'outil passe par la proposition d'un modèle géométrique et l'identification des paramètres géométriques en fonction de la précision de réalisation de la tâche attendue.

En général, un robot est composé de plusieurs segments ou éléments rigides liés entre eux par des articulations considérées comme idéales (sans jeu mécanique et sans déformation) et dotés d'un effecteur final. Pour garantir la qualité de réalisation de la tâche, la précision de la pose de l'effecteur final doit être maîtrisée. Un modèle géométrique cohérent avec le comportement géométrique de la structure du robot doit être développé.

2.5 Convention de Denavit Hartenberg (DH)

Denavit et Hartenberg ont proposés une méthode pour décrire la transformation homogène entre deux solides adjacents avec une position et orientation particulière des repères.

La convention, ou notation, de Denavit-Hartenberg permet de caractériser la position relative de deux solides avec seulement quatre paramètres (au lieu de six). Ceci au prix de certaines restrictions dans le choix des systèmes de coordonnées (repères).

Pour définir les repères il faut suivre la procédure suivante :

- Nommer les corps du robot de $i = 0$ jusqu'à $i = n$ en commençant par la base du robot avec $i = 0$;
- Nommer les articulations de $i = 1$ à n (1 pour le premier degré de liberté et n pour le dernier) ;
- Pour $i = 0$ jusqu'à $i = n - 1$ fixer l'axe z_i sur l'articulation $i + 1$;
- L'origine du repère R_0 sera un point quelconque de l'axe z_0 , de sorte que, x_0 et y_0 forment un repère orthonormé direct ;
- Pour $i = 1, 2, \dots, n - 1$, l'origine du repère est fixée à l'intersection de l'axe z_i avec la droite perpendiculaire commune à z_{i-1} et z_i . Si les deux axes se coupent, l'origine est le point d'intersection. Si par contre, les axes sont parallèles l'origine est l'origine du repère de l'articulation $i + 1$
- L'axe x_i est la droite perpendiculaire commune à z_{i-1} et z_i , et l'axe y_i est choisi de sorte que le repère soit orthonormé direct

2.5.1 Paramètres de Denavit-Hartenberg

Décomposition de M_{i-1} en 4 transformations élémentaires :

- Rotation autour de z d'un angle θ_i ou θ est $\widehat{(x_{i-1}, x_i)}$ suivant l'axe z_{-1}
- Translation suivant z d'une distance d_i ou d_i est $\overline{(x_{i-1}, x_i)}$ suivant l'axe z_{i-1}
- Translation suivant x d'une distance a_i a_i est $\overline{(z_{i-1}, z_i)}$ suivant l'axe x_i
- Rotation autour de x d'un angle α_i ou α est $\widehat{(z_{i-1}, z_i)}$ suivant l'axe x_i

Remarque :

- Si l'articulation i est de type prismatique, alors d_i est variable.
- Si l'articulation i est de type rotoïdes, alors θ_i est variable.

De ces paramètre on peut dresser le tableau des paramètres de Denavit-Hartenberg

i	α	$a(q)$	d	θ
1	α_1	a_1	d_1	θ_1
2	α_2	a_2	d_2	θ_2
3	α_3	a_3	d_3	θ_3

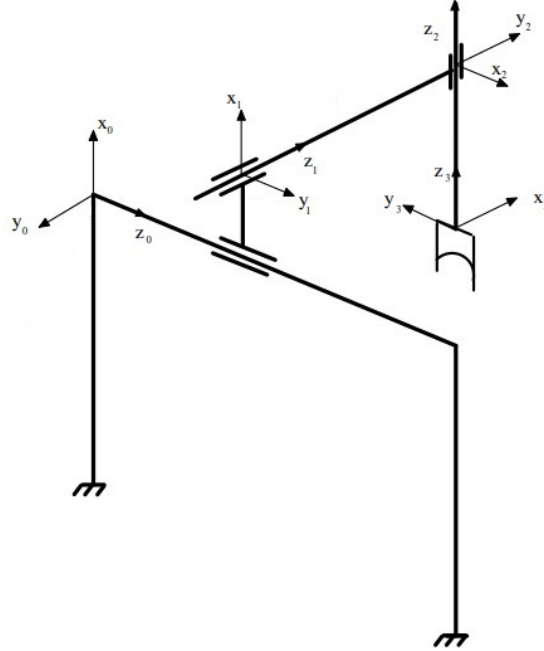


FIGURE 2.2 – Modèle géométrique d'un robot cartésien suivant la convention de DH

2.5.2 Matrice Homogène

Ces transformations se font par rapport au repère courant[17]. D'où :

$$M_{(i-1,i)} = f(q) = R_z(\theta)T_z(d)T_x(a)R_x(\alpha) \quad (2.1)$$

$$M_{(i-1,i)} = R_{(z_{i-1},\theta_i)}T_{(z_{i-1},d_i)}T_{(x_i,a_i)}R_{(x_i,\alpha_i)} \quad (2.2)$$

Avec :

$$R_{(z_{i-1},\theta_i)} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{(z_{i-1},d_i)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{(x_i,a_i)} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{(x_i,\alpha_i)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donc :

$$M_{(i-1,i)} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

$$\Rightarrow M_{(i-1,i)} = \begin{bmatrix} R_{(i-1,i)} & T_{(i-1,i)} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -s\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

2.5.3 Matrice Homogène Inverse

La matrice R est orthogonale donc :

$$R^{-1} = R^T$$

Et :

$$M_{(i,i-1)}^{-1} = M_{(i-1,i)}$$

On aura la matrice $M_{(i,i-1)}$:

$$M_{(i,i-1)} = \begin{bmatrix} R_{(i-1,i)}^T & -R_{(i-1,i)}^T \times T_{(i-1,i)} \\ 0 & 1 \end{bmatrix} \quad (2.5)$$

2.6 Modèle Géométrique :

2.6.1 Modèle Géométrique d'un Robot Cartésien

En appliquant la convention de Denavit-Hartenberg, on aura les deux schéma de la figure ci-dessous :

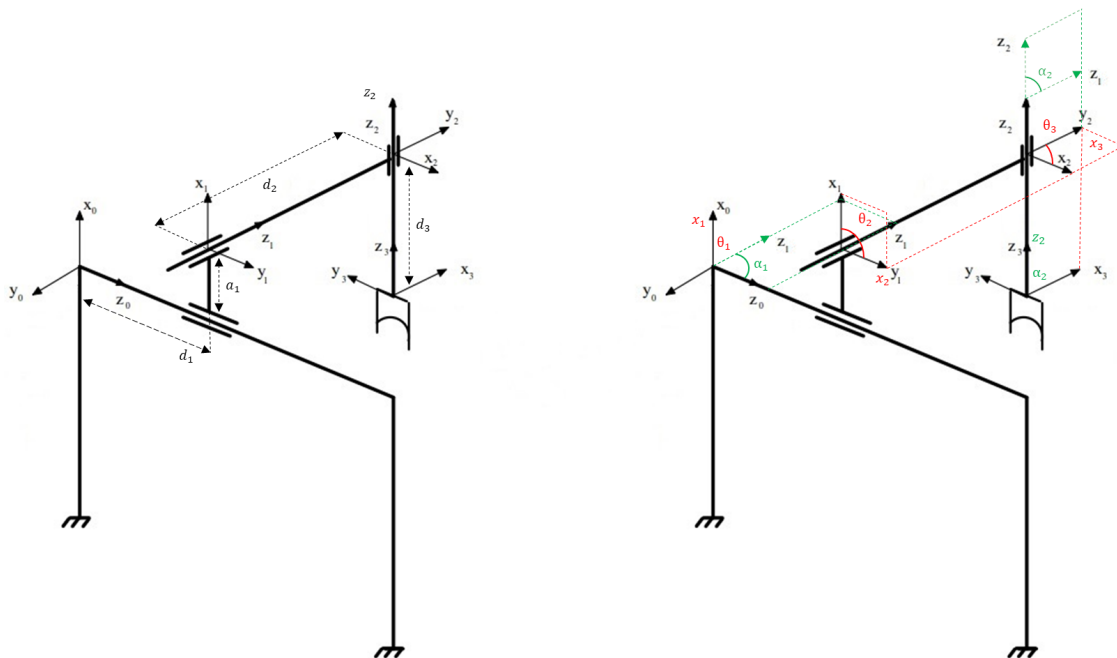


FIGURE 2.3 – Modèle géométrique d'un robot cartésien translation/rotations

A partir du schéma de la figure 2.3, on peut dresser le tableau ci-dessous :

i	α	$a(q)$	d	θ
1	$\frac{\pi}{2}$	a_1	d_1^*	0
2	$\frac{\pi}{2}$	0	d_2^*	$\frac{\pi}{2}$
3	0	0	$-d_3^*$	$\frac{\pi}{2}$

TABLE 2.1 – Tableau des paramètres du modèle géométrique du robot

2.6.2 Modèle Géométrique Direct

2.6.2.1 La Matrice Homogène Directe

Le modèle géométrique direct (MGD) est l'ensemble des relations qui définit l'emplacement de l'effecteur terminal du robot en fonction de ses coordonnées conjoints. Pour une structure en série, elle peut être représentée par la matrice homogène $M_{(0,3)}$.

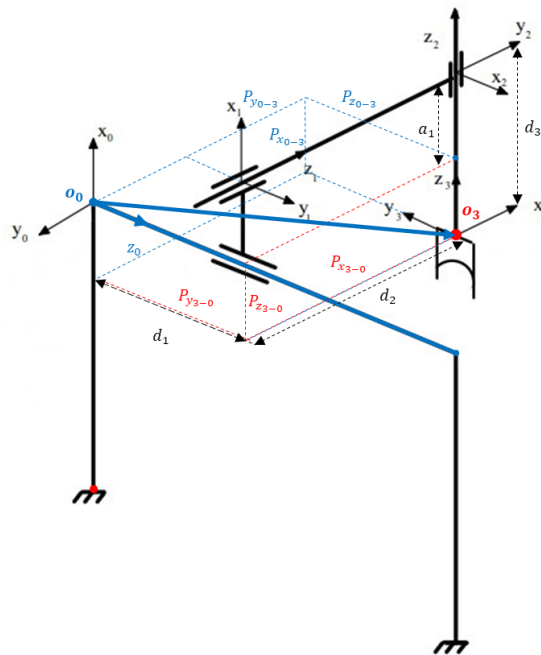


FIGURE 2.4 – Modèle géométrique en 3D

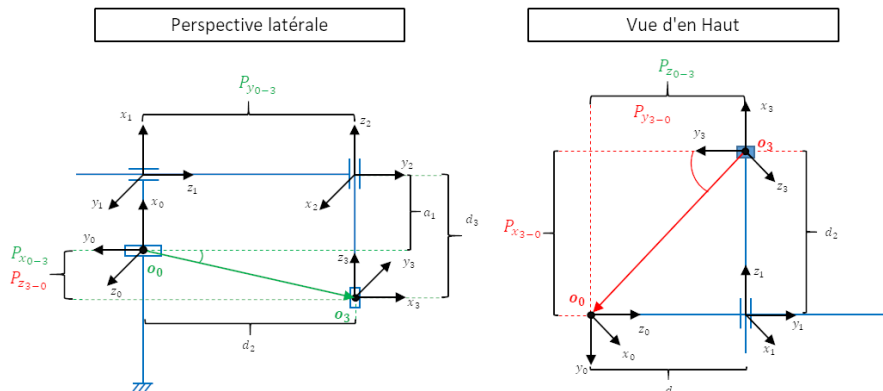


FIGURE 2.5 – Modèle géométrique simplifier en 2D

A partir du tableau (TABLE 2.1) des paramètres de la figure (FIGURE 2.4) nous aurons :

$$M_{(0,1)} = \begin{bmatrix} 1 & 0 & 0 & a_1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & d_1^* \\ 0 & 0 & 0 & 1 \end{bmatrix}, M_{(1,2)} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_2^* \\ 0 & 0 & 0 & 1 \end{bmatrix}, M_{(2,3)} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -d_3^* \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

$$M_{(0,3)} = M_{(0,1)} \times M_{(1,2)} \times M_{(2,3)} = \begin{bmatrix} 0 & 0 & 1 & a_1 - d_3^* \\ -1 & 0 & 0 & -d_2^* \\ 0 & -1 & 0 & d_1^* \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

2.6.2.2 La Matrice Homogène Inverse

on a :

$$R_{(0,3)} = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \implies R_{(0,3)}^T = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$

Nous avons aussi :

$$T_{(0,3)} = \begin{bmatrix} a_1 - d_3^* \\ -d_2^* \\ d_1^* \\ 1 \end{bmatrix}$$

$$M_{(i,i-1)} = \begin{bmatrix} R_{(i-1,i)}^T & -R_{(i-1,i)}^T \times T_{(i-1,i)} \\ 0 & 0 & 0 & 1 \end{bmatrix} \implies M_{(3,0)} = \begin{bmatrix} 0 & -1 & 0 & -d_2^* \\ 0 & 0 & -1 & d_1^* \\ 1 & 0 & 0 & d_3^* - a_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

2.6.3 Modèle Géométrique Inverse

Le problème du calcul des variables conjointes correspondant à un emplacement spécifié de l'effecteur terminal est appelé le problème géométrique inverse. Ce problème est au centre des algorithmes de contrôle informatique pour les robots. Il a en général une solution multiple et sa complexité dépend fortement de la géométrie du robot. Le modèle qui donne toutes les solutions possibles à ce problème est appelé le modèle géométrique inverse (MGI).

Soit G_i une fonction telle que : $q = G_i(\text{Attitude})$

- La fonction G_i est appelée modèle géométrique inverse du robot ;
- La fonction G_i est définie uniquement à l'intérieur de l'espace de travail du robot.

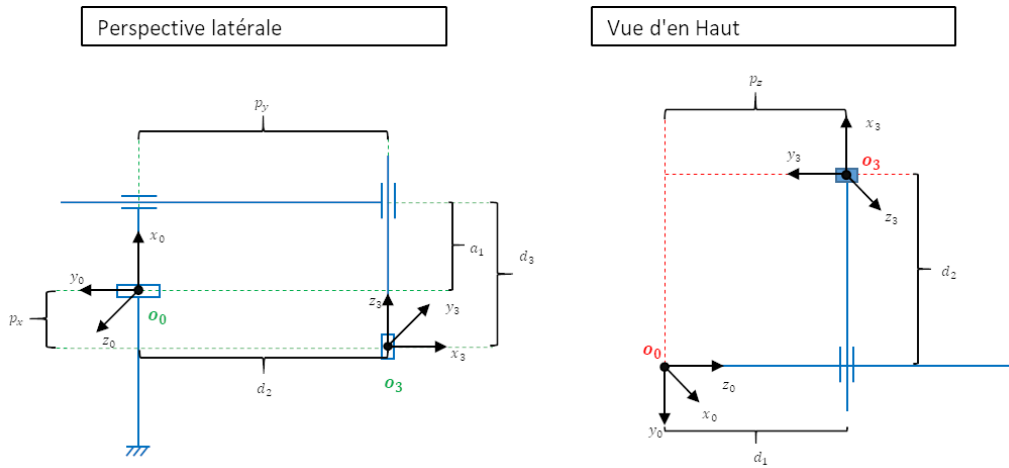


FIGURE 2.6 – Modèle géométrique inverse simplifié en 2D

Dans notre cas, nous avons une seule solution, car nous avons 3 transformations de type prismatique perpendiculaire l'une à l'autre.

Alors soit :

$$p = [p_x \ p_y \ p_z]^T$$

Les coordonnées de O_3 dans R_0 , nous aurons :

$$\begin{cases} p_x = a_1 - d_3 \\ p_y = -d_2 \\ p_z = d_1 \end{cases} \implies \begin{cases} d_3 = a_1 - p_x \\ d_2 = -p_y \\ d_1 = p_z \end{cases} \quad (2.9)$$

2.7 Modèle Cinématique d'un Robot Cartésien

2.7.1 Modèle Cinématique Direct

Le modèle cinématique direct d'un robot manipulateur décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires [16].

Il est donné par :

$$\dot{x} = J(q) \cdot \dot{q} \quad (2.10)$$

Où $J(q)$ désigne la matrice jacobienne de dimension $(m \times n)$ du mécanisme, égale à $\frac{\partial x}{\partial q}$ et la fonction de configuration articulaire q .

La même matrice jacobienne intervient dans le calcul du modèle différentiel direct qui donne les variations élémentaires dX des coordonnées opérationnelles en fonction des variations élémentaires des coordonnées articulaires dq , soit :

$$dX = J(q) \cdot dq \quad (2.11)$$

2.7.2 Intérêts de la Matrice Jacobienne

L'intérêt de la matrice jacobienne est multiple [16] :

- Elle est à la base du modèle différentiel inverse, permettant de calculer une solution locale des variables articulaires q connaissant les coordonnées opérationnelles x ;
- En statique, on utilise le jacobée pour établir la relation liant les efforts exercés par l'organe terminal sur l'environnement aux forces et couples des actionneurs ;
- Elle facilite le calcul des singularités et de la dimension de l'espace opérationnel accessible du robot ;
- Elle facilite le calcul des singularités et de la dimension de l'espace opérationnel accessible du robot.

2.7.3 Calcul de la Matrice Jacobienne :

Le calcul de la matrice jacobienne peut se faire en dérivant le MGD, $X = f(q)$, à partir de la relation suivante :

$$J_{ij} = \frac{\partial f_i}{\partial q_j} \quad (2.12)$$

Avec : $i = 1, \dots, m; j = 1, \dots, n$

où J_{ij} est l'élément (i, j) de la matrice jacobienne J . Cette méthode est facile à mettre en œuvre pour des robots à deux ou trois degrés de liberté.

$$J(q) = \frac{\partial f(q)}{\partial q} = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \dots & \frac{\partial f_1}{\partial q_n} \\ \frac{\partial f_2}{\partial q_1} & \frac{\partial f_2}{\partial q_2} & \dots & \frac{\partial f_2}{\partial q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \frac{\partial f_m}{\partial q_2} & \dots & \frac{\partial f_m}{\partial q_n} \end{bmatrix} \quad (2.13)$$

2.7.4 Les Points Singuliers

Les points singuliers correspondent à certaines configurations des manipulateurs de robot qu'il faut éviter car elles entraînent une perte brutale de rigidité du manipulateur. Au voisinage de ces configurations, le manipulateur peut devenir incontrôlable et les forces articulaires pourraient augmenter considérablement et il pourrait même y avoir un risque d'endommager les mécanismes du manipulateur. Les points singuliers dans un espace de travail peuvent être identifiés mathématiquement lorsque le déterminant dans la matrice jacobienne est nul [18] :

$$\det(J(q)) = 0 \quad (2.14)$$

2.7.5 Modèle Cinématique d'un Robot Cartésien

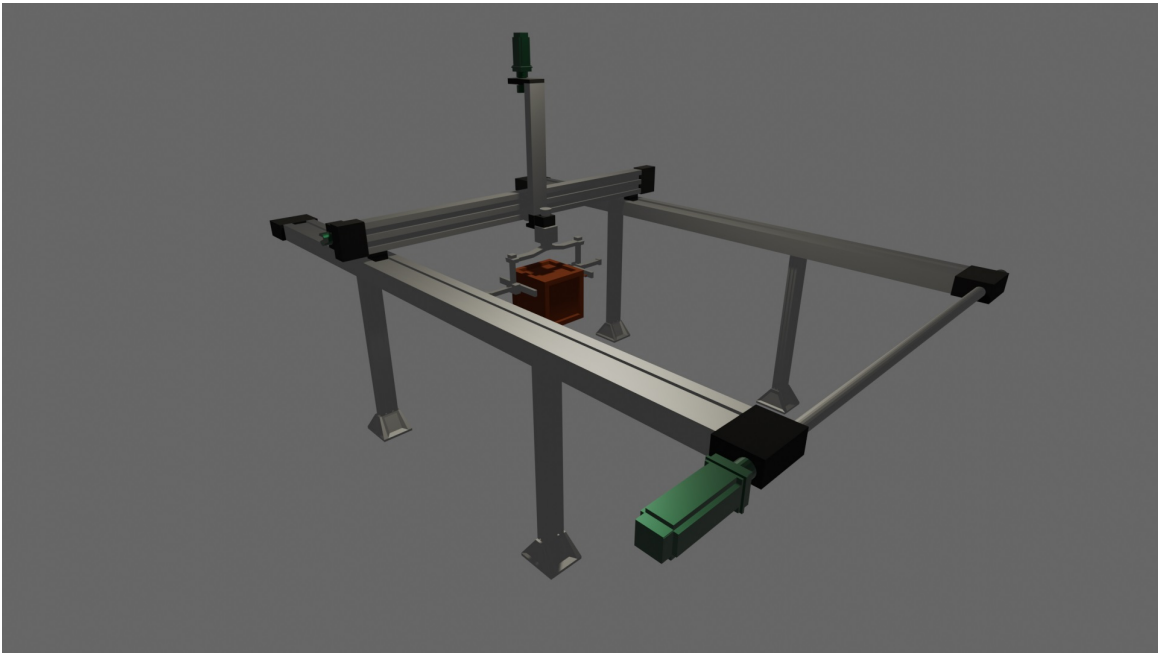


FIGURE 2.7 – Robot cartésien à trois degrés de liberté

Afin d'obtenir la cinématique directe du robot cartésien à trois degrés de liberté, nous devons dessiner un schéma pour le système, (voir Figure 2.8), où q_1, q_2, q_3 sont les positions des articulations ; et m_1, m_2, m_3 représentent les masses des axes. Comme on l'observe, la translation est le mouvement unique que réalise ce genre de robots, alors la cinématique est définie comme :

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} q_1 \\ 0 \\ 0 \end{bmatrix} ; \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ 0 \end{bmatrix} ; \begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (2.15)$$

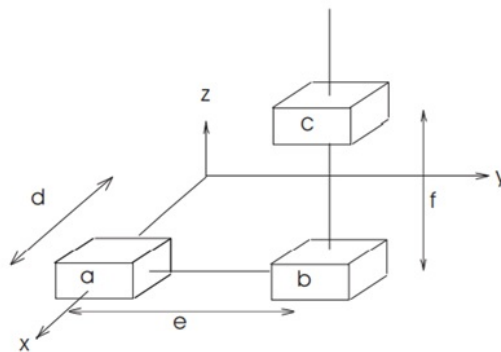


FIGURE 2.8 – Diagramme d'un robot cartésien à trois axes

2.7.5.1 Matrice Jacobienne d'un Robot Cartésien

Afin d'obtenir la matrice jacobienne des trois degrés de liberté du robot cartésien, il est nécessaire pour utiliser la cinématique avancée qui est définie comme :

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (2.16)$$

Si on fait la dérivation partielle de x par rapport à q_1 , q_2 et q_3 nous aurons :

$$\begin{aligned} \frac{\partial x}{\partial q_1} &= \frac{\partial q_1}{\partial q_1} = \dot{q}_1 \\ \frac{\partial x}{\partial q_2} &= \frac{\partial q_1}{\partial q_2} = 0 \\ \frac{\partial x}{\partial q_3} &= \frac{\partial q_1}{\partial q_3} = 0 \end{aligned}$$

La dérivation partielle de y par rapport à q_1 , q_2 et q_3 nous aurons :

$$\begin{aligned} \frac{\partial y}{\partial q_1} &= \frac{\partial q_2}{\partial q_1} = 0 \\ \frac{\partial y}{\partial q_2} &= \frac{\partial q_2}{\partial q_2} = \dot{q}_2 \\ \frac{\partial y}{\partial q_3} &= \frac{\partial q_2}{\partial q_3} = 0 \end{aligned}$$

La dérivation partielle de z par rapport à q_1 , q_2 et q_3 nous aurons :

$$\begin{aligned} \frac{\partial z}{\partial q_1} &= \frac{\partial q_3}{\partial q_1} = 0 \\ \frac{\partial z}{\partial q_2} &= \frac{\partial q_3}{\partial q_2} = 0 \\ \frac{\partial z}{\partial q_3} &= \frac{\partial q_3}{\partial q_3} = \dot{q}_3 \end{aligned}$$

L'équation suivante $\dot{x} = J(q) \cdot \dot{q}$ est décrite comme suit :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} & \frac{\partial x}{\partial q_3} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} & \frac{\partial y}{\partial q_3} \\ \frac{\partial z}{\partial q_1} & \frac{\partial z}{\partial q_2} & \frac{\partial z}{\partial q_3} \end{bmatrix} \cdot \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \quad (2.17)$$

Où les éléments de la matrice jacobienne sont définis comme :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \quad (2.18)$$

2.7.6 Transposée de la matrice Jacobienne

La transposition de la matrice $J(q)(n \times m)$ est une matrice $J^T(q)(m \times n)$ créé formellement. Elle est représentée comme suit :

$$J_{ji}^T(q) = J_{ij}(q); \text{ pour } : 1 \leq i \leq n; 1 \leq j \leq m \quad (2.19)$$

2.7.6.1 Transposée de la Matrice Jacobienne du Robot Cartésien

Dans le cas particulier du robot cartésien, la matrice jacobienne $J(q)$ est égale à la matrice d'identité I , donc sa matrice transposée $J^T(q)$ est la même, donc nous aurons :

$$J^T(q) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.20)$$

2.7.7 Modèle Cinématique Inverse

En mathématiques, et en particulier en algèbre linéaire, une matrice carrée $A (n \times n)$, est réversible, non singulière, non dégénérée ou régulière s'il existe une autre matrice carrée avec $(n \times n)$ appelée matrice inverse A^{-1} , qui vérifiée :

$$A^{-1}A = AA^{-1} = I \quad (2.21)$$

La définition mathématique de la matrice jacobienne inverse est comme suit :

$$J^{-1}(q) = \frac{C^T}{\det(J(q))} \quad (2.22)$$

Où C est la matrice des cofacteurs.

2.7.7.1 La Matrice des Cofacteurs du Robot Cartésien

Afin d'obtenir la matrice des cofacteurs, il est nécessaire d'appliquer la procédure suivante : Considérant une matrice A définie comme suit :

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

Nous obtenons la matrice de cofacteurs suivante :

$$A = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \quad (2.23)$$

$$c_{11} = +(ei - hf) = +(1 - 0) = 1$$

$$c_{12} = -(di - gf) = -(0 - 0) = 0$$

$$c_{13} = +(dh - ge) = +(0 - 0) = 0$$

$$c_{21} = -(bi - hc) = -(0 - 0) = 0$$

$$c_{22} = +(ai - gc) = +(1 - 0) = 1$$

$$\begin{aligned}
c_{23} &= -(ah - gb) = -(0 - 0) = 0 \\
c_{31} &= +(bf - ec) = +(0 - 0) = 0 \\
c_{32} &= -(af - dc) = -(0 - 0) = 0 \\
c_{33} &= +(ae - db) = +(1 - 0) = 1 \\
\implies C &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.24}
\end{aligned}$$

2.7.8 Matrice Jacobienne Inverse d'un Robot Cartésien

Nous avons :

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \implies C^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.25}$$

Nous avons aussi :

$$\det(J(q)) = 1 \implies J^{-1}(q) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.26}$$

Nous pouvons conclure que pour le cas spécifique des trois degrés de liberté, la matrice inverse existe.

2.8 Modèle Dynamique d'un Robot Cartésien

2.8.1 Modélisation Dynamique

Le modèle dynamique des robots joue un rôle important dans leur conception et leur fonctionnement. Pour la conception du robot, le modèle dynamique inverse peut être utilisé pour sélectionner les actionneurs, tandis que le modèle dynamique direct est utilisé pour effectuer des simulations dans le but de tester les performances du robot et d'étudier les mérites relatifs des schémas de commande possibles. En ce qui concerne les opérations du robot, le modèle dynamique inverse est utilisé pour calculer les couples d'actionneurs, qui sont nécessaires pour obtenir un mouvement souhaité. Il est également utilisé pour identifier les paramètres dynamiques nécessaires pour les applications de contrôle et de simulation [14, .p.191,192].

Il existe deux problèmes liés à la dynamique d'un manipulateur que nous souhaitons résoudre. Dans le premier problème, on nous donne un point de trajectoire, et nous souhaitons trouver le vecteur requis des couples d'articulation. Cette formulation de la dynamique est utile pour le problème du contrôle du manipulateur. Le deuxième problème est de calculer comment le mécanisme se déplacera sous l'application d'un ensemble de couples d'articulations.

Comme conclusion, on peut dire que la modélisation dynamique est la représentation mathématique d'un système qui décrit son comportement dans le stimulus interne et externe présenté dans le système.

Plusieurs approches ont été proposées pour modéliser la dynamique des robots. Les plus fréquemment employés en robotique sont [14] :

- La formulation de Lagrange ;
- La formulation de Newton-Euler.

L'approche Newton-Euler est basée sur les formules dynamiques élémentaires et sur une analyse des forces et moments de contrainte agissant entre les articulations. Ce qui signifie que c'est une approche de l'équilibre des forces de la dynamique. La formulation Lagrangienne d'une autre part est une approche énergétique de la dynamique. Bien sûr, pour le même manipulateur, les deux donneront les mêmes équations de mouvement. [11, .p.165]

2.8.2 Formalisme de Lagrange

Dans cette section, nous développons une méthode de Lagrange simple pour présenter la forme générale du modèle dynamique des robots. Tout d'abord, nous considérons un système idéal sans frottement ni élasticité, n'exerçant ni forces ni moments sur l'environnement.

La formulation de Lagrange décrit le comportement d'un système dynamique en termes de travail et d'énergie stockée dans le système. Les équations de Lagrange sont couramment écrites sous la forme :

$$\Gamma_i = \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = \tau \quad (2.27)$$

$$\mathcal{L} = E - U \quad (2.28)$$

Où :

τ : Vecteur des couples appliqués ;

L : Le lagrangien du robot ;

E : Énergie cinétique du système ;

U : Énergie potentielle du système ;

q, \dot{q}, \ddot{q} : sont la position, la vitesse et l'accélération dans l'espace commun respectivement ;

2.8.3 Forme Générale des Equations Dynamiques

2.8.3.1 L'énergie Cinétique :

L'énergie cinétique du système est une fonction quadratique des vitesses articulaires telle que [14, .p.194] [11, .P.182] :

$$E = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (2.29)$$

Où :

$M(q)$: est la matrice d'inertie, elle est symétrique définie positive, ces éléments sont en fonction des positions articulaires.

Les matrices définies positives sont celles qui ont la propriété que leur forme quadratique est toujours un scalaire positif. L'équation 2.29 peut être considérée comme analogue à l'expression familière de l'énergie cinétique d'une masse ponctuelle [11, .P.182] :

$$E = \frac{1}{2} m \dot{q}^2 \quad (2.30)$$

Où :

m : est la masse manipulateur.

2.8.3.2 L'énergie Potentielle :

L'énergie potentielle étant fonction des positions des articulations, la formule d'énergie potentielle s'écrit :

$$U = \sum_{j=1}^n U_j = \sum_{j=1}^n -M_j g^T (\mathcal{L}_{0,j} + S_j) \quad (2.31)$$

Où :

M_j : Le moment des efforts extérieurs exercés sur le corps C_j autour de O_j .

g^T : L'accélération de la pesanteur transposé .

$\mathcal{L}_{0,j}$: Désignant le vecteur d'origine O_0 et d'extrémité O_j

S_j : Vecteur ayant pour origine O_j et pour extrémité le centre de masse du corps C_j .

Par l'exploitation des équations 2.27, 2.29 et 2.31, on obtient le modèle dynamique du robot :

$$\Gamma = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + Q(q) \quad (2.32)$$

Où :

— $C(q, \dot{q})\dot{q}$ est le vecteur ($n \times 1$) des couples de Coriolis et des forces centrifuges, tel que :

$$C\dot{q} = \dot{M}\dot{q} - \frac{\partial E}{\partial q} \quad (2.33)$$

— $Q = [Q_1 \dots Q_n]^T$ est le vecteur des couples de gravité.

Après le calcul des éléments de A, C et Q [14, W, Khalil.P.195.196.197];

En l'absence de frottement et d'autres perturbations, le modèle dynamique de robot rigide peut s'écrire :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (2.34)$$

Où :

$g(q) \in \mathbf{R}^{n \times 1}$ est un vecteur du couple de gravité obtenu comme résultat du gradient sur l'énergie potentielle :

$$g(q) = \frac{\partial U(q)}{\partial q} \quad (2.35)$$

2.8.4 Modèle dynamique avec frottement

La frottement joue un rôle majeur dans la limitation de la qualité des performances du robot. Le frottement non compensé produit une erreur statique, un retard et un comportement de cycle limite.

Donc en considérant la frictions les équations de Lagrange sont couramment écrites sous la forme :

$$\Gamma_i = \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = \tau - f(\tau, \dot{q}_i) \quad (2.36)$$

Pour :

$$i = 1, \dots, n$$

$$\mathcal{L} = E - U \quad (2.37)$$

Où :

$f(\tau, \dot{q}_i) \in \mathbf{R}^{n \times 1}$ est le vecteur des couples de frottement définis comme :

$$f(\tau, \dot{q}_i) = \begin{bmatrix} f_1(\tau_1, \dot{q}_1) \\ f_2(\tau_2, \dot{q}_2) \\ \vdots \\ f_n(\tau_n, \dot{q}_n) \end{bmatrix} \quad (2.38)$$

Donc si on prend en considération les frottements, la dynamique du robot est donnée par [18, .P.174] :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + f(\tau, \dot{q}) = \tau \quad (2.39)$$

2.8.5 Modèle dynamique d'un robot cartésien

Après avoir présenter les notions générale de la modélisation dynamique, dans cette section nous appliquerons la formulation de Lagrange pour obtenir le modèle dynamique de notre robot cartésien à trois degrés de liberté, en se basant sur le modèle cinématique déjà obtenu.

Cependant, comme notre robot n'a que trois articulations prismatiques (pas de rotations), l'équation d'énergie cinétique s'écrira comme :

$$E = \frac{1}{2}mv^2 = \frac{1}{2}\dot{q}^T M(q)\dot{q} \quad (2.40)$$

$$v = \frac{d}{dt} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.41)$$

$$\implies v^2 = \|v\|^2 = v^T v = [x \ y \ z] \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.42)$$

$$\begin{cases} v_1^2 = \dot{q}_1^2 \\ v_2^2 = \dot{q}_1^2 + \dot{q}_2^2 \\ v_3^2 = \dot{q}_1^2 + \dot{q}_2^2 + \dot{q}_3^2 \end{cases} \quad (2.43)$$

En remplaçant les valeurs sur v_1^2 , v_2^2 et v_3^2 dans (2.43) on obtient l'énergie cinétique de chaque lien :

$$\begin{cases} E_1(q, \dot{q}) = \frac{m_1 \dot{q}_1^2}{2} \\ E_2(q, \dot{q}) = \frac{m_2(\dot{q}_1^2 + \dot{q}_2^2)}{2} \\ E_3(q, \dot{q}) = \frac{m_3(\dot{q}_1^2 + \dot{q}_2^2 + \dot{q}_3^2)}{2} \end{cases} \implies E(q, \dot{q}) = \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2(\dot{q}_1^2 + \dot{q}_2^2)}{2} + \frac{m_3(\dot{q}_1^2 + \dot{q}_2^2 + \dot{q}_3^2)}{2} \quad (2.44)$$

Après la distribution, nous pouvons attendre l'équation de la forme suivante :

$$E(q, \dot{q}) = \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2 \dot{q}_1^2}{2} + \frac{m_2 \dot{q}_2^2}{2} + \frac{m_3 \dot{q}_1^2}{2} + \frac{m_3 \dot{q}_2^2}{2} + \frac{m_3 \dot{q}_3^2}{2} \quad (2.45)$$

$$\implies E(q, \dot{q}) = \frac{(m_1 + m_2 + m_3)}{2} \dot{q}_1^2 + \frac{(m_2 + m_3)}{2} \dot{q}_2^2 + \frac{(m_3)}{2} \dot{q}_3^2 \quad (2.46)$$

L'énergie potentielle $U(q)$ est obtenue en considérant dans ce cas $h = q_3$ et $m = m_3$:

$$U(q) = m_3 \cdot g \cdot q_3 \quad (2.47)$$

Maintenant et après avoir calculé l'énergie cinétique et potentielle, nous pouvons calculer le lagrangien \mathcal{L} en utilisant la formule de l'équation 2.28 :

$$\mathcal{L}(q, \dot{q}) = \left[\frac{(m_1 + m_2 + m_3)}{2} \dot{q}_1^2 + \frac{(m_2 + m_3)}{2} \dot{q}_2^2 + \frac{(m_3)}{2} \dot{q}_3^2 \right] - m_3 \cdot g \cdot q_3 \quad (2.48)$$

et pour déduire le modèle dynamique de notre système on suit les étapes ci-dessous :

$$\frac{\partial \mathcal{L}}{\partial \dot{q}} = \begin{bmatrix} (m_1 + m_2 + m_3) & 0 & 0 \\ 0 & (m_2 + m_3) & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \quad (2.49)$$

et on a :

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}} = \begin{bmatrix} (m_1 + m_2 + m_3) & 0 & 0 \\ 0 & (m_2 + m_3) & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} \quad (2.50)$$

et enfin :

$$\frac{\partial \mathcal{L}}{\partial q} = \begin{bmatrix} 0 \\ 0 \\ m_3 \cdot g \end{bmatrix} \quad (2.51)$$

En regroupant les termes on trouve le modèle dynamique du robot cartésien :

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} (m_1 + m_2 + m_3) & 0 & 0 \\ 0 & (m_2 + m_3) & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ m_3 \cdot g \end{bmatrix} \quad (2.52)$$

Où : τ_1, τ_2 et τ_3 : sont les couples appliqués.

Comme nous pouvons l'observer, le modèle dynamique représenté par l'équation 2.52 ne contient pas le terme des forces de frottement.

Si on remplace par les paramètres physiques du robot mentionnés dans le tableau ci-dessous :

Notions	Valeur	Unités
m_1	16.180	kg
m_2	14.562	kg
m_3	12.944	kg
g	9.81	m/s^2

On trouve le modèle dynamique du robot à trois degrés de liberté de la référence [18] :

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} 43.686 & 0 & 0 \\ 0 & 27.506 & 0 \\ 0 & 0 & 12.944 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 126.98064 \end{bmatrix} \quad (2.53)$$

2.9 Conclusion

Dans ce chapitre nous avons donné un aperçu sur quelques notions de base des robot cartésiens, ou nous avons présenter leurs description géométrique en appliquant la convention

de Devenavit-Hartenberg. Après, et en utilisant le formalisme de Lagrange-Euler, nous avons calculé le modèle dynamique du robot cartésien.

A la fin du chapitre, nous avons donné le modèle dynamique du robot basé sur des paramètres réels obtenus de la littérature. Ce modèle dynamique va être vérifié dans le chapitre qui suit, en l'utilisant dans la simulation des lois de commande classique et robuste.

Commande du Robot Cartésien

Sommaire

3.1	Introduction	29
3.2	Lois de commande classique	30
3.2.1	Loi de commande de type PD	30
3.2.2	Application au robot cartésien	31
3.3	Loi de commande robuste basée sur le mode glissant	37
3.3.1	Surfaces de glissement	38
3.3.2	Conception de la loi de commande	42
3.3.3	Stabilité	43
3.3.4	Simulation	44
3.4	Conclusion	49

3.1 Introduction

Nous appelons commande dynamique tout système capable d'assurer la compensation automatique des forces et couples dus aux vitesses et accélérations des segments massifs d'un mécanisme articulé : forces d'inertie, centrifuges et de Coriolis agissant.

La commande des robots cartésiens a fait l'objet de nombreux travaux. Les principales approches utilisées sont :

- La commande classique de type PD.
- La commande par découplage non linéaire (Couple calculé).
- La commande basée sur la passivité.
- La commande fondée sur une fonction de Lyapunov.
- La commande adaptative.
- La commande robuste à structure variable. ...

Dans ce chapitre, nous allons appliquer deux lois de commande au modèle dynamique de notre robot cartésien, la première est de type classique et la deuxième de type avancé, qui est une commande basée sur le mode glissant.

3.2 Lois de commande classique

3.2.1 Loi de commande de type PD

Le modèle dynamique du robot sans le terme de de frottement est donné par :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (3.1)$$

L'objectif de régulation est de maintenir la position de la variable articulaire q du robot constante autour d'une position désirée constante dénotée q_d . La loi de commande proportionnelle dérivée (PD) à la forme de base suivante[19] :

$$\tau = K_p(q_d - q) - K_v\dot{q} \quad (3.2)$$

Soit $\tilde{q} = q - q_d$ l'erreur de position, alors, nous avons $\dot{\tilde{q}} = \dot{q}$ et $\ddot{\tilde{q}} = \ddot{q}$ puisque $\dot{q}_d = 0$. En appliquant cette loi de commande sur la dynamique du robot, nous obtenons la dynamique du système en boucle fermée (voir figure 3.1) :

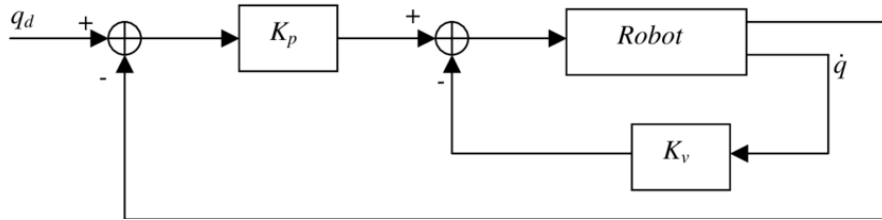


FIGURE 3.1 – Schéma synoptique de la commande type PD

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + K_v\dot{q} + K_p\tilde{q} = 0 \quad (3.3)$$

Les points d'équilibres du système en boucle fermée sont définis par l'ensemble :

$$S = \{(\tilde{q}, \dot{\tilde{q}}) : K_p\tilde{q} + G(\tilde{q} + q_d) = 0, \dot{\tilde{q}} = 0\} \quad (3.4)$$

L'analyse de la stabilité du système en boucle fermée se fait au sens de Lyapunov et pour cela la fonction candidate de Lyapunov est choisie comme :

$$V(\tilde{q}, \dot{\tilde{q}}) = \frac{1}{2}\dot{\tilde{q}}^T M(q)\dot{\tilde{q}} + \frac{1}{2}\tilde{q}^T K_p\tilde{q} + U(q) + U_0 \quad (3.5)$$

où $U(q)$ représente l'énergie potentielle et U_0 est une constante positive choisie pour que la fonction candidate V de Lyapunov soit définie positive. Si nous dérivons V par rapport au temps, nous obtenons :

$$\dot{V}(\tilde{q}, \dot{\tilde{q}}) = \dot{\tilde{q}}^T M(q)\ddot{\tilde{q}} + \frac{1}{2}\dot{\tilde{q}}^T \dot{M}(q)\dot{\tilde{q}} + \dot{\tilde{q}}^T K_p\tilde{q} + \dot{U}(q) \quad (3.6)$$

La dérivée de l'énergie potentielle est calculée comme suit :

$$\dot{U}(q) = \frac{dU(q)}{dt} = \left(\frac{dq}{dt}\right)^T \frac{\partial U(q)}{\partial q} = \dot{q}^T G(q) \quad (3.7)$$

puisque ;

$$\frac{\partial U(q)}{\partial q} = G(q) \quad (3.8)$$

par le remplacement de $M(q)\ddot{q}$ obtenue de l'équation (3.3) dans (3.6), nous obtenons :

$$\dot{V}(\tilde{q}, \dot{\tilde{q}}) = -\dot{\tilde{q}}^T C(q, \dot{q})\dot{\tilde{q}} - \dot{\tilde{q}}^T K_v \dot{\tilde{q}} - \dot{\tilde{q}}^T K_p \tilde{q} - \dot{\tilde{q}}^T G(q) + \frac{1}{2}\dot{\tilde{q}}^T \dot{M}(q)\dot{\tilde{q}} + \dot{\tilde{q}}^T K_p \tilde{q} + \dot{\tilde{q}}^T G(q) \quad (3.9)$$

En exploitant la propriété $\dot{\tilde{q}}^T \left\{ \frac{\dot{M}(q)}{2} - C(q, \dot{q}) \right\} \dot{\tilde{q}} = 0$, nous obtenons directement le résultat :

$$\dot{V}(\tilde{q}, \dot{\tilde{q}}) = \dot{\tilde{q}}^T K_v \dot{\tilde{q}} \quad (3.10)$$

Ce résultat est une fonction semi-définie négative, alors le point d'équilibre $(\tilde{q}, \dot{\tilde{q}}) = (0, 0)$ est stable au sens de Lyapunov. Mais la négativité de (3.10) assure seulement la convergence du deuxième état, ou $\dot{\tilde{q}}$, à zéro, donc l'analyse n'est pas complète. Nous pouvons donc appliquer le théorème de LaSalle pour conclure que les états du système convergeront à l'ensemble S défini par l'équation (3.4), dans un temps fini. Nous avons $\dot{V}(\tilde{q}, \dot{\tilde{q}}) = 0$ seulement pour $\dot{\tilde{q}} = 0$, alors par le remplacement de cette dernière dans la dynamique de la boucle fermée (3.3), nous obtenons :

$$K_p \tilde{q} + G(q) = 0 \quad (3.11)$$

qui est sûrement vrai si :

$$\tilde{q} = -K_p^{-1} G(q) \quad (3.12)$$

alors :

$$\|\tilde{q}\| \leq \|K_p^{-1}\| \|G(q)\| \leq g_b \|K_p^{-1}\| \quad (3.13)$$

Où : g_b est une borne supérieure du vecteur de gravité ce que veut dire :

$$\|G(q)\| \leq g_b(q) \quad (3.14)$$

où : g_b est une fonction scalaire. Pour une articulation rotoïde g_b est une constante indépendante de la variable articulaire q , mais pour une articulation prismatique, g_b peut être dépendante de q .

$\|G(q)\|$: est bornée parce qu'elle ne dépend que des cosinus et sinus des variables articulaires.

De cela nous concluons que le système converge localement vers S défini dans l'équation 3.4 et que la précision du système dépend de la matrice du gain K_p .

En outre, si K_p et K_v sont des matrices définies positives alors toutes les solutions $(\tilde{q}, \dot{\tilde{q}})$ convergent asymptotiquement globalement (si K_p est grand) ou localement (si K_p est petit) à l'ensemble des points d'équilibre S .

L'erreur de l'état stable peut donc, en principe, soit réduit en augmentant K_p arbitrairement, néanmoins le bruit de mesure et autres dynamiques non modélisables limiteront l'usage de hauts gains.

3.2.2 Application au robot cartésien

3.2.2.1 Modèle utilisé

Cette étude de simulation est faite par l'application de loi de commande de type PD au robot cartésien (3 degrés de liberté). Les différentes caractéristiques et le modèle dynamique de ce système sont présentés dans le chapitre 02.

Le modèle dynamique sous sa forme matricielle est :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (3.15)$$

3.2.2.2 Algorithme et simulation de la loi de commande de type PD

La commande proportionnelle dérivée (PD) a la forme de base suivante :

$$\tau = K_p(q_d - q) + K_v\dot{q} \quad (3.16)$$

La dynamique du système en boucle fermée est :

$$M(q)\ddot{\tilde{q}} + C(q, \dot{q})\dot{\tilde{q}} + G(q) + K_v\dot{\tilde{q}} + K_p\tilde{q} = 0 \quad (3.17)$$

Les paramètres de simulation sont les matrices des gains K_p et K_v données par :

$$K_p = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{bmatrix}$$

$$K_v = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{bmatrix}$$

3.2.2.3 Résultats de simulation

Les figures suivantes montrent les positions et les erreurs de position des différents axes (X,Y,Z) de notre robot cartésien lorsque la commande PD est appliquée.

De plus, les consignes utilisées pour la simulation sont ($q_{d1} = 1.2$, $q_{d2} = 2$, $q_{d3} = 1$) :

Le schéma bloqué de la simulation est représenté dans la figure ci-dessous :

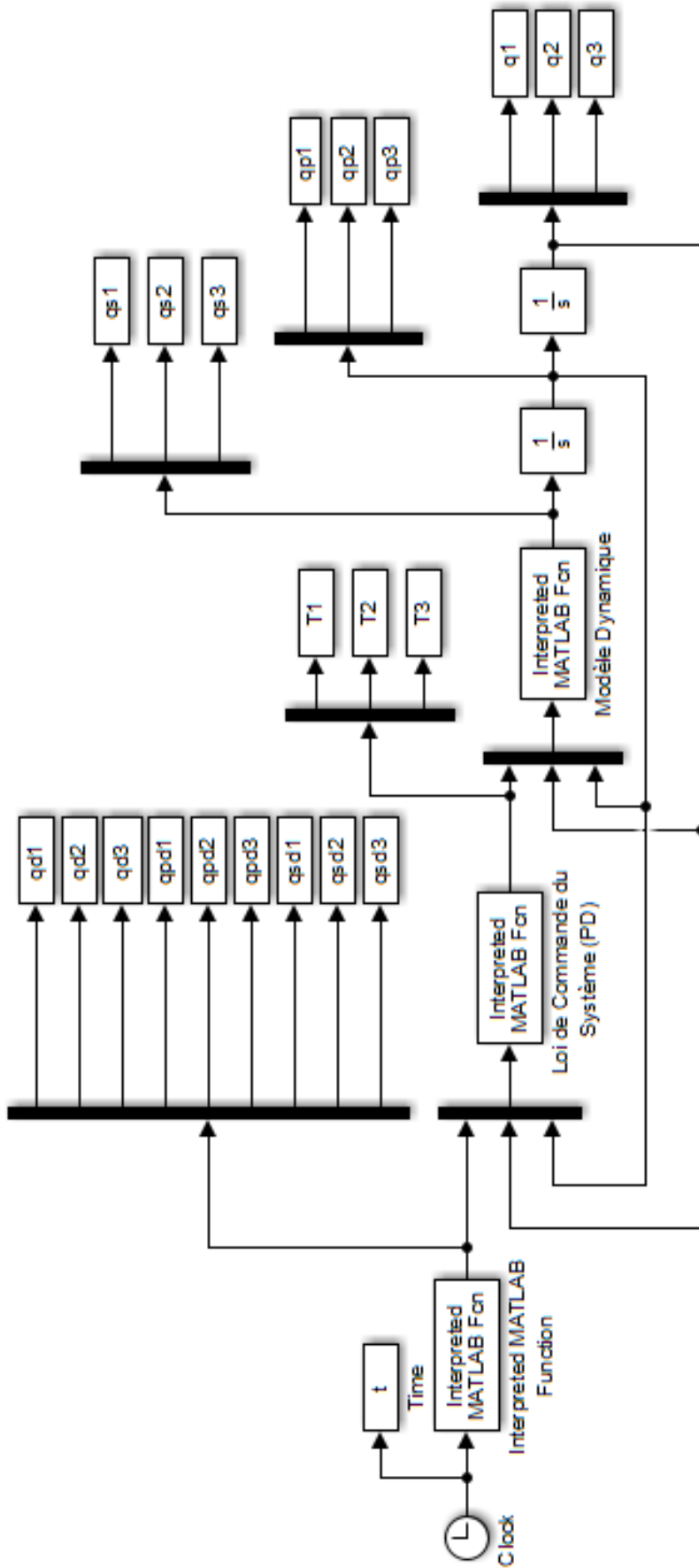
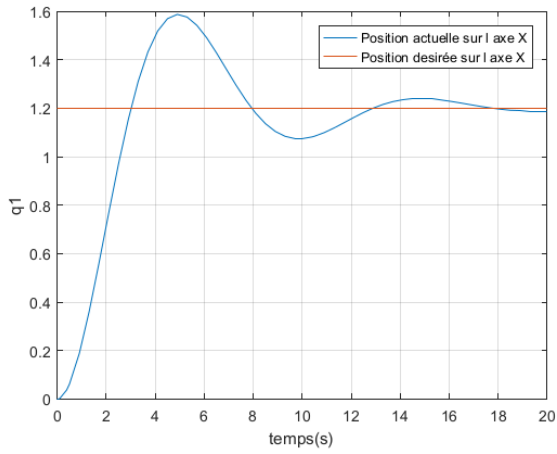
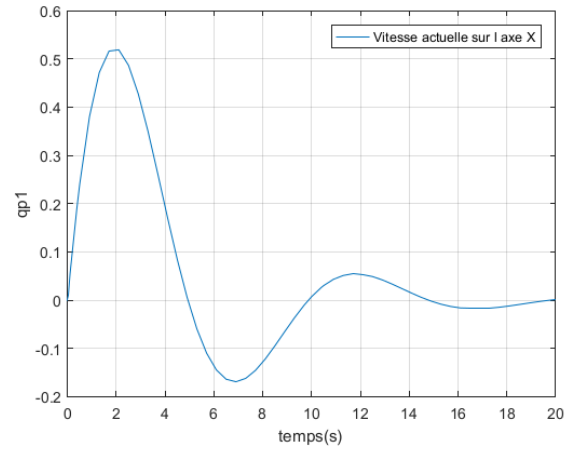


FIGURE 3.2 – Schéma utilisé pour la Simulation PD

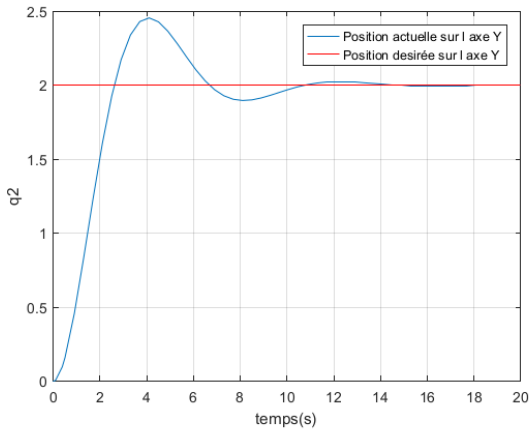


(a) Position

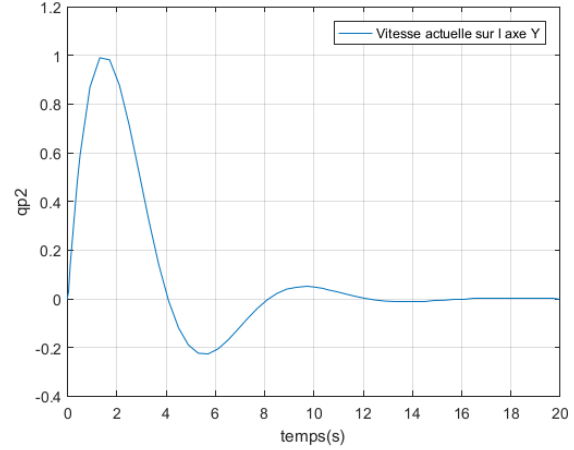


(b) Erreur de position

FIGURE 3.3 – Position et erreur de position de l'axe X, avec loi de commande PD

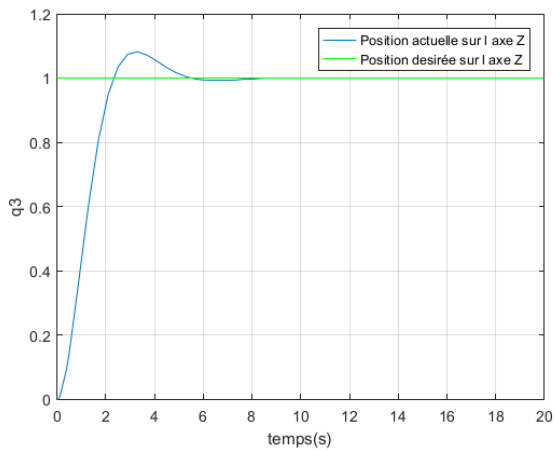


(a) Position

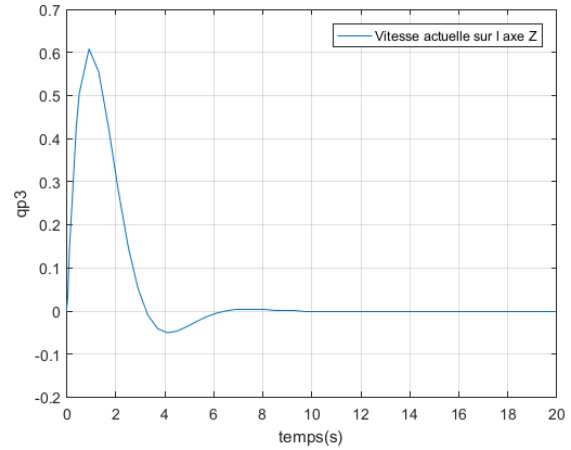


(b) Erreur de position

FIGURE 3.4 – Position et erreur de position de l'axe Y, avec loi de commande PD



(a) Position



(b) Erreur de position

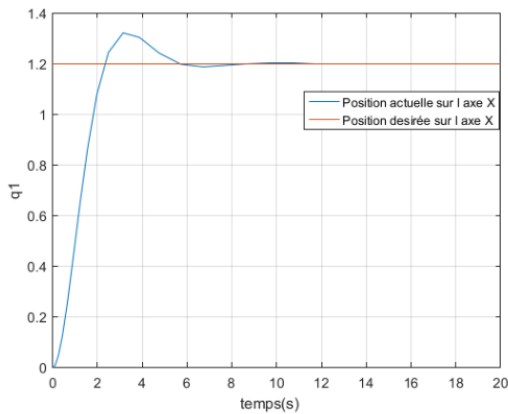
FIGURE 3.5 – Position et erreur de position de l'axe Z, avec loi de commande PD

Avec de petites valeurs de gain comme dans la simulation précédente, le système était un peu lent pour atteindre son état final, et les valeurs d'erreur étaient trop importantes pour être négligées. De toute façon, cela ne peut pas être très préoccupant car le correctif peut se produire simplement en amplifiant les matrices de gain avec des valeurs plus élevées :

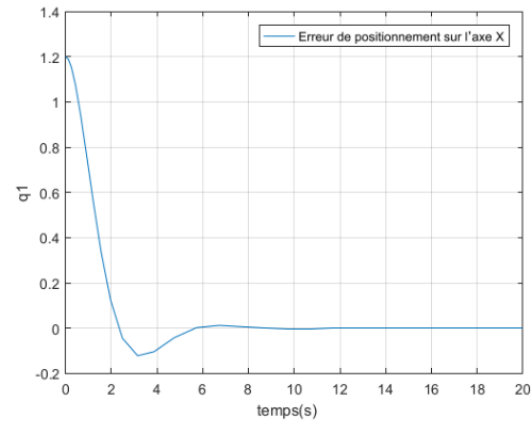
A titre d'exemple, on prend les matrices des gains comme suit :

$$K_p = \begin{bmatrix} 60 & 0 & 0 \\ 0 & 60 & 0 \\ 0 & 0 & 60 \end{bmatrix}$$

$$K_v = \begin{bmatrix} 60 & 0 & 0 \\ 0 & 60 & 0 \\ 0 & 0 & 60 \end{bmatrix}$$

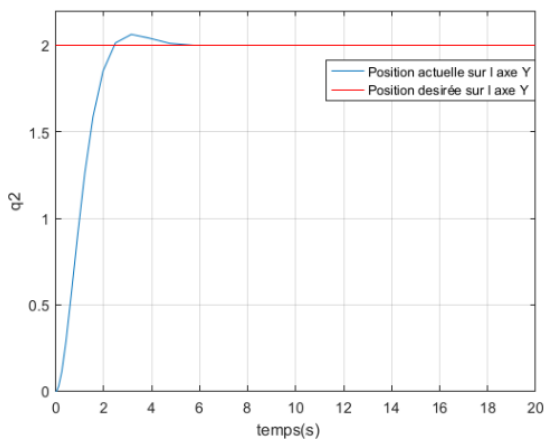


(a) Position

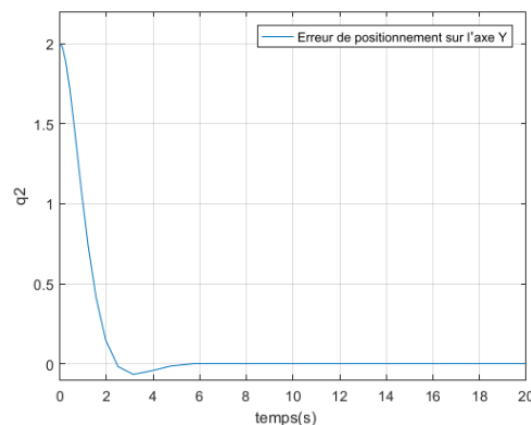


(b) Erreur de position

FIGURE 3.6 – Position et erreur de position de l'axe X, avec des valeurs de gain plus élevées

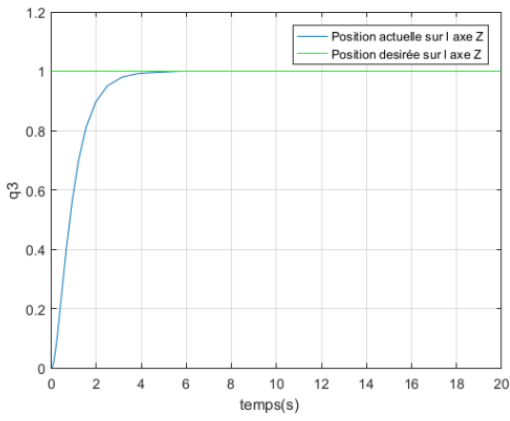


(a) Position

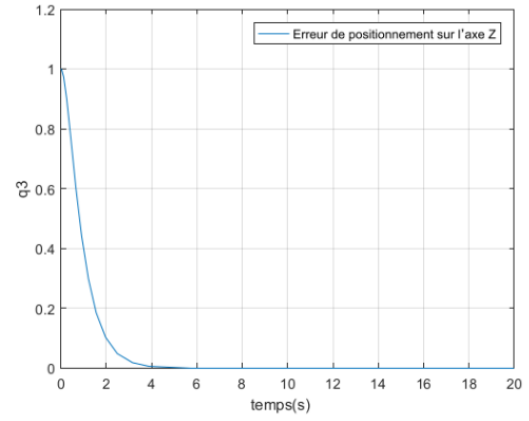


(b) Erreur de position

FIGURE 3.7 – Position et erreur de position de l'axe Y, avec des valeurs de gain plus élevées

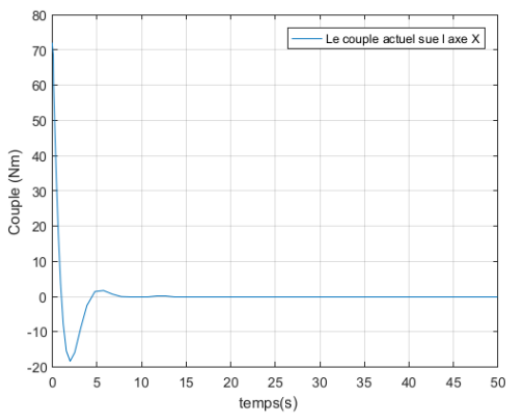


(a) Position

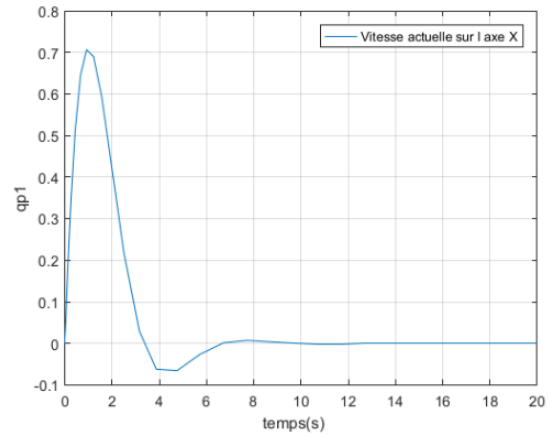


(b) Erreur de position

FIGURE 3.8 – Position et erreur de position de l'axe Z, avec des valeurs de gain plus élevées

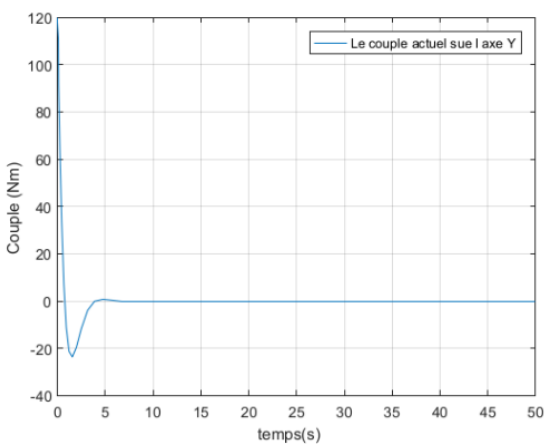


(a) Couple

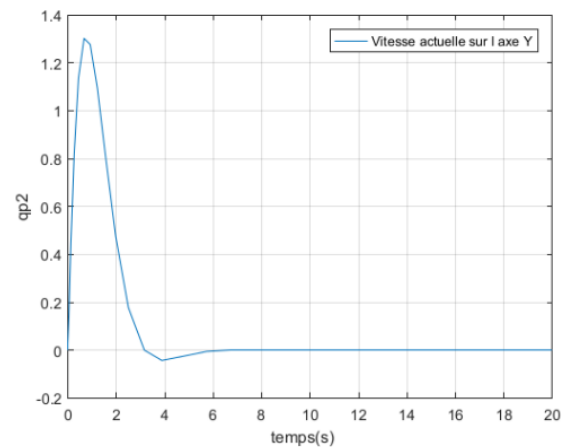


(b) Vitesse

FIGURE 3.9 – Couple et vitesse de l'axe X, avec des valeurs de gain plus élevées

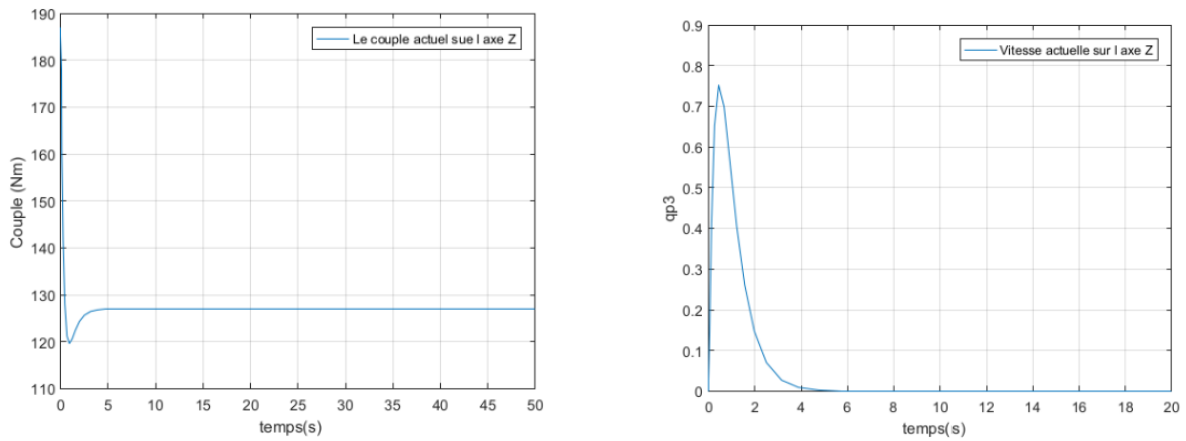


(a) Couple



(b) Vitesse

FIGURE 3.10 – Couple et vitesse de l'axe Y, avec des valeurs de gain plus élevées



(a) Couple

(b) Vitesse

FIGURE 3.11 – Couple et vitesse de l'axe Z , avec des valeurs de gain plus élevées

Grâce à l'observation des courbes, il est facile de dire que la transition du système vers son état permanent est devenue beaucoup plus rapide et que les valeurs d'erreur sont devenues beaucoup plus tolérables.

Comme une dernière remarque pour ce chapitre, nous pouvons dire que la commande PD n'est pas efficace que dans le cas où, les paramètres du système à commander (robot) sont bien définis, c'est-à-dire pas d'incertitudes, pas de frottement et pas de perturbations.

3.3 Loi de commande robuste basée sur le mode glissant

Des méthodes de commande robuste sont conçues pour résister à des facteurs affectant le système tout en donnant de bonnes performances de suivi. Ces méthodes ont principalement recours à un gain de commande élevé pour atténuer l'incertitude du système. La commande par mode glissant (CMG) [20] est une approche de commande robuste qui a été largement utilisée en raison de sa simplicité de conception, de ses propriétés de réduction d'ordre et de ses performances constantes. Cependant, la CMG a vu une application limitée dans le domaine de la robotique en raison du phénomène de vibration à haute fréquence du signal de commande, ce qui le rend hautement inapproprié pour une application pratique [21].

L'objectif est de concevoir une loi de commande stable de sorte que pour une trajectoire souhaitée q_d l'erreur de suivi $q_e = q - q_d$ converge vers zéro.

La loi de commande sera conçu en deux parties principales comme indiqué ci-dessous :

- La première partie impliquera la conception d'un couple de commande basé sur la dynamique du système.
- La deuxième partie sera la loi équivalente qui convertira le signal de couple de commande généré à une commande de position pour être envoyé au moteur.

Les aspects de cette approche seront d'abord présentés pour les systèmes qui ont une seule entrée de commande, ce qui nous va permet de développer une intuition sur les aspects fondamentaux du de cette loi de commande pour les systèmes non-linéaire .

3.3.1 Surfaces de glissement

Considérant le système dynamique à entrée unique ci-dessous :

$$x^{(n)} = f(x) + b(x)u \quad (3.18)$$

où la variable x est la sortie qui nous intéresse, et u est appelé la variable d'entrée de la commande, et $x = [x, \dot{x}, \dots, x^{(n-1)}]$ représente le vecteur d'état. Alors que la fonction $f(x)$ dans 3.18 n'est pas exactement connue, une borne supérieure sur $f(x)$ est fixée par une fonction continue connue de x pour limiter la taille de l'imprécision. De même, le gain de commande $b(x)$ n'est pas exactement connu non plus, seul son signe est connu ; par conséquent, il est également borné par les fonctions continues connues de x .

Le problème de commande est de s'assurer que l'état x suit un état explicite variant dans le temps $x_d = [x_d, \dot{x}_d \dots x_d^{(n-1)}]$ en présence d'incertitude du modèle sur $f(x)$ et $b(x)$. Pour que la tâche de poursuite de trajectoire soit exécutée à l'aide d'une commande finie u , l'état initial désiré $x_d(0)$ doit vérifier la relation suivante :

$$x_d(0) = x(0) \quad (3.19)$$

Par exemple, dans un système de second ordre, la position ou la vitesse ne peut pas changer ; de ce fait, toute trajectoire souhaitée qui peut être exploitable à partir du temps $t = 0$ commence nécessairement à la même position et à la même vitesse que celles du système.

En définissant $x_e = x - x_d$ comme l'erreur de poursuite de la variable x et $x_e = x - x_d = [x_e, \dot{x}_e, \dots, x_e^{(n-1)}]^T$ est le vecteur d'erreur de poursuite. Définissons également une surface $S(t)$ variant dans le temps dans l'espace d'états $R^{(n)}$ avec l'équation scalaire $s(x; t) = 0$ exprimé comme :

$$s(x; t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} x_e \quad (3.20)$$

où λ est une constante strictement positive. Par exemple, si $n = 2$, Eq (3.19) prend la forme suivante :

$$s = \dot{x}_e - \lambda x_e \quad (3.21)$$

C'est-à-dire qu'il consiste simplement en une somme pondérée des erreurs de position et de vitesse ; ainsi, nous pouvons exprimer x_e à partir de l'équation (3.20) comme suit :

$$x_e = \frac{s(x; t)}{\left(\frac{d}{dt} + \lambda \right)^{n-1}} \quad (3.22)$$

Plus spécifiquement, un problème de stabilisation de premier ordre dans s n'est fondamentalement rien d'autre qu'un remplacement du problème de poursuite du vecteur x_d à n dimension (c'est-à-dire le problème original de poursuite du *nième* ordre en x). En effet, s n'a besoin d'être différencié qu'une seule fois pour l'apparition de l'entrée u du fait de la présence du terme $x_e^{(n-1)}$ dans l'expression s dans (3.19).

De plus, les bornes sur s peuvent être évaluées directement comme les bornes du vecteur d'erreur de poursuite x_e , et par conséquent, le scalaire s est considéré comme une véritable mesure des performances de poursuite. Plus précisément, sous l'hypothèse que $x_e(0) = 0$ (en

revanche, nous notons que l'effet des conditions initiales non nulles dans x_e peut être ajouté séparément), nous écrivons :

$$\forall t \geq |s(t)| \leq \Phi \Rightarrow \forall t \geq 0, |x_e^{(i)}(t)| \leq (2\lambda)^i \varepsilon \quad (3.23)$$

$$i = 0, 1, \dots, n - 1$$

$$\text{ou : } \varepsilon = \frac{\Phi}{\lambda^{n-1}}$$

En effet, l'équation (3.20), ou plus précisément l'équation (3.22) dérivé de (3.20) indique clairement que l'erreur de poursuite x_e peut être obtenue à partir de s par une séquence de filtres passe-bas du premier ordre.

En général, une relation d'entrée-sortie de filtre passe-bas du premier ordre est donnée comme suit :

$$\frac{\text{Sortie}_{f_{pb}}}{\text{Entrée}_{f_{pb}}} = K_f \frac{1}{1 + p\tau} \quad (3.24)$$

où K_f est le gain du filtre et τ est la constante de temps du filtre. Soit y_1 la sortie du premier filtre. On peut exprimer la sortie du premier filtre en termes d'intégrale de convolution dans le domaine temporel en prenant en compte le fait que l'entrée est déjà définie comme s :

$$y_1(t) = \int_0^t e^{-\lambda(t-T)} S(T) dT \quad (3.25)$$

En utilisant $|s| \leq \Phi$, nous obtenons ce qui suit :

$$|y_1(t)| \leq \Phi \int_0^t e^{-\lambda(t-T)} dT = \frac{\Phi}{\lambda} e^{-\lambda(t-T)} = \frac{\Phi}{\lambda} (1 - e^{-\lambda t}) \leq \frac{\Phi}{\lambda} \quad (3.26)$$

Nous pouvons appliquer un raisonnement similaire au deuxième filtre, jusqu'à ce que nous arriverons $y_{n-1} = x_e$

$$|x_e| \leq \frac{\Phi}{\lambda^{n-1}} = \varepsilon \quad (3.27)$$

Ainsi, nous avons, en effet, remplacé un problème de poursuite d'ordre n par un problème de stabilisation de premier ordre, et avons quantifié avec l'inégalité (3.23) les transformations auxquelles correspondent les paramètres de performance. Le maintien du scalaire s à zéro, qui est un problème simplifié du premier ordre, peut maintenant être réalisé en choisissant la loi de commande u de (3.18) telle qu'en dehors de $S(t)$.

$$\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s| \quad (3.28)$$

où : η est une constante strictement positive qui assure l'inégalité (??). η est appelée condition condition d'atteignabilité. Une exigence fondamentale est que la dynamique du mode glissant doit être attractive pour l'état du système ou il existe de nombreuses conditions d'atteignabilité définies dans la littérature. Fondamentalement, l'inégalité (3.28) indique que la «distance» au carré de la surface, mesurée par s^2 , diminue tout au long des trajectoires du système. Par conséquent, il limite les trajectoires pour se diriger vers la surface $S(t)$, comme le montre la figure (3.12) [22].

Une autre caractéristique intéressante de l'ensemble invariant $S(t)$ est que lorsque les trajectoires du système sont dans l'ensemble, il est défini par l'équation de l'ensemble lui-même, c'est-à-dire :

$$\left(\frac{d}{dt} + \lambda\right)^{n-1} x_e = 0 \quad (3.29)$$

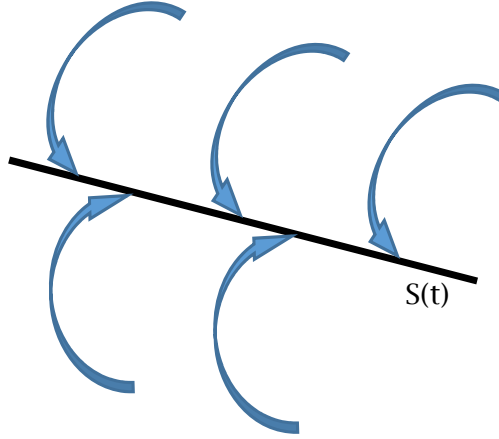


FIGURE 3.12 – Condition de glissement

C'est-à-dire que la surface $S(t)$ représente à la fois un lieu et une dynamique.

Enfin, si la condition (3.19) n'est pas entièrement validée, c'est-à-dire si $x(t) = 0$ est bien éloigné de $x_d(t) = 0$, alors, néanmoins, satisfaire la condition (3.28) donne une garantie d'atteindre la surface $S(t)$ à un temps fini inférieur à $|s(t=0)|/\eta$. En effet, supposons, par exemple, que $s(t=0) > 0$ et définissons " $t_{atteign}$ " comme le temps nécessaire pour atteindre la surface $s = 0$. Intégrons (3.28) entre les points $t = 0$ et $t = t_{atteign}$ (ie, dans l'intervalle $[0, t_{atteign}]$) donne lieu à :

$$0 - s(t=0) = s(t = t_{atteign}) - s(t=0) \leq -\eta(t_{atteign} - 0) \quad (3.30)$$

ce qui signifie que

$$t_{atteign} \leq \frac{s(t=0)}{\eta} \quad (3.31)$$

En conclusion, l'idée derrière (3.20) et (3.28) est d'obtenir une fonction appropriée de l'erreur de poursuite, s , conformément à (3.20), puis de choisir la loi de commande de rétroaction u dans (3.18) telle que s^2 continue à être utilisé comme une fonction de type Liapunov du système en boucle fermée, malgré la présence d'incertitude du modèle et de perturbations. Ensuite, la conception de la loi de commande est une procédure en deux étapes. Premièrement, la sélection d'une loi de commande de rétroaction u est effectuée pour vérifier la condition de glissement (3.28). Cependant, il est nécessaire que la loi de commande soit discontinue tout au long de $S(t)$ pour tenir compte de la présence d'incertitudes de modélisation et de perturbations [22].

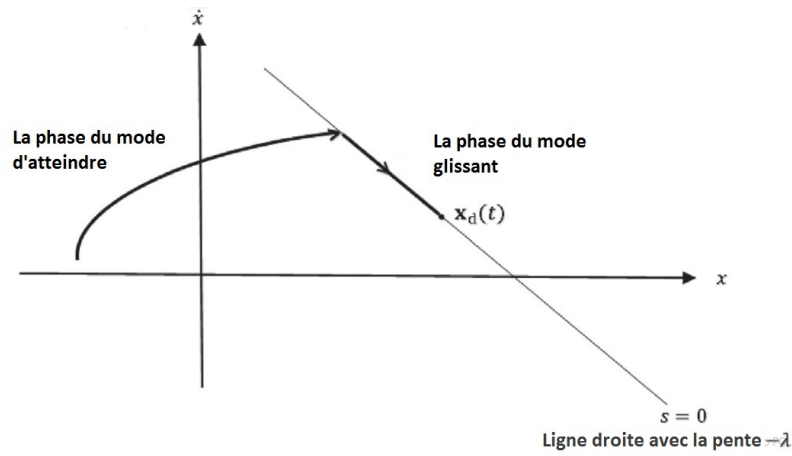


FIGURE 3.13 – Représentation graphique de les équations (3.20) et (3.28) pour $n = 2$

Puisque l'exécution des commutations de contrôle associées n'est pas nécessairement parfaite (par exemple, en pratique, la commutation n'est pas instantanée, et la valeur de s n'est pas connue avec une précision infinie), cela provoque des vibrations (Cf.3.14) [22].

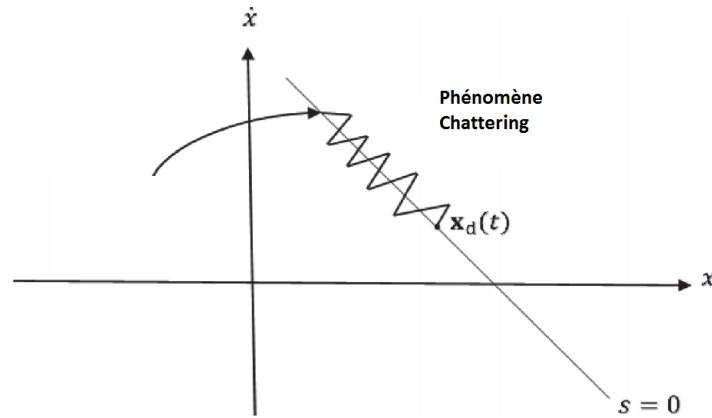


FIGURE 3.14 – Vibrations causé par les retards de commutation

Comme mentionné précédemment, la loi de commande discontinue provoque les vibrations des trajectoires autour de la surface $s = 0$. Ce problème peut être éliminée par lissant les discontinuités au voisinage de la surface de glissement par l'introduction d'une épaisseur de couche limite [22].

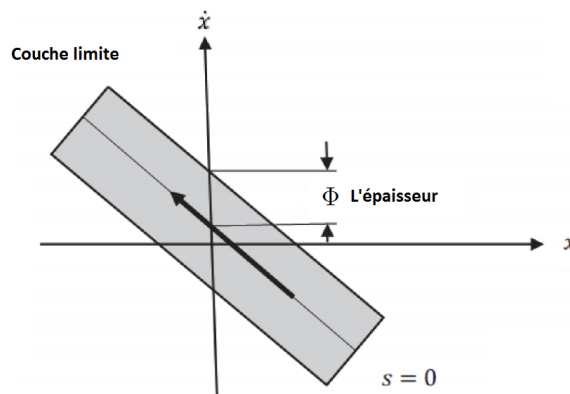


FIGURE 3.15 – Couche de limite

Pour maintenir le fonctionnement du système dans la surface de glissement, un terme d'action de commutation, u_{com} , est ajouté à la loi de commande et est défini par :

$$u_{com} = -Ksat(s) \quad (3.32)$$

et la loi de commande globale peut être exprimée comme suit :

$$u = \hat{u} + u_{com} = \underbrace{(-\hat{f} + \ddot{x}_d - \lambda\dot{x}_e)}_{\text{Commande en base fréquence}} - \underbrace{Ksat(s)}_{\text{Commande en haute fréquence}} \quad (3.33)$$

Avec :

$$sat(s) = \begin{cases} -1 & si : \frac{s}{\Phi} < 0 \\ 0 & si : \frac{s}{\Phi} < 1 \\ 1 & si : \frac{s}{\Phi} < -1 \end{cases} \quad (3.34)$$

où : Φ est l'épaisseur de la couche de limite.

3.3.2 Conception de la loi de commande

Dans le deuxième chapitre, nous avons défini l'équation dynamique du robot manipulateur comme suit :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + f(\tau, \dot{q}) = \tau \quad (3.35)$$

La surface de glissement définie ci-dessous est prise en compte dans la conception de la commande en mode glissant CMG :

$$s = \dot{e} + \lambda e \quad (3.36)$$

où : $e = -q_e = q - q_d$ représente le vecteur d'erreur et λ est supposé être une matrice symétrique définie positive telle que $s = 0$ évoluerait en une surface stable.

Le vecteur vitesse de référence \dot{q}_r est le même que la définition dans [23] :

$$\dot{q}_r = \dot{q}_d - \lambda e \quad (3.37)$$

Par conséquent, on peut définir la surface de glissement comme suit :

$$s = \dot{q} - \dot{q}_r \quad (3.38)$$

Maintenant, le lemme suivant fait référence à la conception de la commande en mode glissant. Considérons le système constitué de (3.35) à (3.37). Si la règle de commande suivante est considérée :

$$\tau = \hat{\tau} - K_{sg}sign(s) \quad (3.39)$$

où :

$$\hat{\tau} = M\ddot{q}_r + \hat{C}\dot{q}_r + G \quad (3.40)$$

et :

$$K_{sg}sign(s) \geq \|\Delta C\dot{q}_r\| + \Gamma_i \quad (3.41)$$

Alors la condition de glissement[23] suivante est satisfaite :

$$\frac{1}{2} \frac{d}{dt} [s^T M s] < -\eta (s^T s)^{\frac{1}{2}}; \quad \eta > 0 \quad (3.42)$$

3.3.3 Stabilité

Dans (3.41), K_{sg} représente l'élément du vecteur de gain glissant K_{sg} et Γ est un vecteur de paramètre de conception qui doit être choisi pour assurer l'inégalité $\Gamma_i \geq F_{up} + \eta_i$ [22].

Soit une fonction de Liapunov candidate donnée comme suit :

$$V = \frac{1}{2}s^T M s \quad (3.43)$$

La matrice M est définie positif, et s est différente de 0 ($s \neq 0$), cela implique que V est toujours supérieur a zéro ($V > 0$); en prenant en compte la propriété symétrique de M et en dérivant (3.43) par rapport au temps, nous donne ce qui suit :

$$\dot{V} = \frac{1}{2}\dot{s}^T M s + \frac{1}{2}s^T (\dot{M} s + M \dot{s}) = \frac{1}{2}\dot{s}^T M s + \frac{1}{2}s^T \dot{M} s + \frac{1}{2}s^T M \dot{s} = s^T M \dot{s} + \frac{1}{2}s^T \dot{M} s \quad (3.44)$$

En utilisant (3.39), on obtient :

$$\dot{V} = s^T M (\ddot{q} - \ddot{q}_r) + \frac{1}{2}s^T \dot{M} s = s^T (M \ddot{q} - M \ddot{q}_r) + \frac{1}{2}s^T \dot{M} s \quad (3.45)$$

On prend \ddot{q} de 3.35 et on la remplace dans (3.46), on obtient :

$$\dot{V} = s^T (M M^{-1} (\tau - C \dot{q} - G - F) - M \ddot{q}_r) + \frac{1}{2}s^T \dot{M} s \quad (3.46)$$

Maintenant on prend \dot{q} de (3.38) et on la remplace dans l'équation précédente; on obtient :

$$\dot{V} = s^T (\tau - C \dot{q} - G - F - M \ddot{q}_r) + \frac{s^T (\dot{M} - 2C) s}{2} \quad (3.47)$$

Dans l'équation précédente le deuxième terme est égale à zéro à cause de la propriété asymétrique; donc nous aurons :

$$\dot{V} = s^T (\tau - C \dot{q} - G - F - M \ddot{q}_r) \quad (3.48)$$

En remplaçant (3.39) et (3.40) successivement pour τ et $\hat{\tau}$ en (3.48) on obtient :

$$\dot{V} = s^T ((\hat{C} - C) \dot{q}_r - K_{sg} \text{sign}(s) - F) = s^T (-\Delta C \dot{q}_r - K_{sg} \text{sign}(s) - F) \quad (3.49)$$

Dans la modélisation des robots manipulateurs, les termes $M(q)$ et $G(q)$ peuvent être déterminer précisément, au même temps ce n'est pas facile de déterminer les termes $C(q, \dot{q})$ et $F(\dot{q})$, à cause de ça, dans cette étude on concéder la matrice C comme suit :

$$C = \hat{C} + \Delta C \quad (3.50)$$

On considère aussi que le vecteur du bruit externe F est définie comme suit :

$$\|F\| \leq F_{up} \quad (3.51)$$

Maintenant, en réorganisant (3.49), nous obtenons ce qui suit :

$$\dot{V} = -s^T (\Delta C \dot{q}_r + F) - s^T K_{sg} \text{sgn}(s) = -s^T (\Delta C \dot{q}_r + F) - \sum_{i=1}^n K_{sg} |s_i| \quad (3.52)$$

Où : $|s_i| = s_i^T \text{sgn}(s_i)$

À ce point, nous pouvons vérifier brièvement que les termes du côté droit de (3.41) sont positifs. Tout d'abord, il est plus facile de dire que le premier terme à droite, $\|\Delta C \dot{q}_r\|$ est positive dans tous les cas, car la norme euclidienne est utilisée. L'autre terme Γ_i est également positif. Parce que, comme nous l'avons déjà dit dans (3.41) $\Gamma_i \geq F_{up} + \eta_i$ où F_{up} est une borne supérieure inconnue définie comme $\|F\| \leq F_{up}$ dans (3.51) ce qui indique également qu'il est absolument positif, et η est une constante de conception strictement positive et garantit l'inégalité. Par conséquent, $\Gamma_i \geq 0$. Maintenant, si l'inégalité K_{sgl} donnée par (3.41) est substituée dans l'équation (3.52) sous l'acceptation de son signe d'égalité, on peut réécrire l'équation (3.52) comme suit :

$$\dot{V} = -s^T(\Delta C \dot{q}_r + F) - \sum_{i=1}^n \|\Delta C \dot{q}_r\| |s_i| - \sum_{i=1}^n F_{up} |s_i| - \sum_{i=1}^n \eta_i |s_i| \quad (3.53)$$

Le premier, deuxième et troisième termes du côté droit de l'équation ci-dessus sont négatifs. Par conséquent, nous pouvons facilement conclure que :

$$\dot{V} \leq - \sum_{i=1}^n \eta_i |s_i| \quad (3.54)$$

Ceci montre que le système en boucle fermée est stable.

3.3.4 Simulation

En mode glissant, une convergence asymptotique vers l'état final sera réalisée. La réponse souhaitée dans le mode atteignant, en général, est d'atteindre la surface de commutation défini comme :

$$s(x) = \Psi^T x = 0 \quad (3.55)$$

dans un temps fini avec un petit dépassement par rapport au surface de commutation.

La loi d'atteinte est une équation différentielle qui détermine la dynamique d'une fonction de commutation $s(x)$. Si $s(x)$ est une équation différentielle asymptotiquement stable, alors, c'est uniquement une condition d'atteinte. En outre, la sélection des paramètres dans l'équation différentielle contrôle la qualité dynamique du système de commande à structure variable CSV dans le mode d'atteinte. La loi atteignante peut être exprimée pratiquement sous une forme générale comme suit [24] :

$$\dot{s} = -Q \text{sign}(s) - Kh(s) \quad (3.56)$$

où : $Q = \text{diag}[q_1, \dots, q_n]$, avec : $q_i > 0$

$K = \text{diag}[k_1, \dots, k_n]$, avec : $k_i > 0$

$h(s) = [h_1(s_1), \dots, h_n(s_n)]$, et $s_i h_i(s_i) > 0$, et $h_i(0) = 0$

Le principe de conception de la loi commande par mode glissant CMG pour les systèmes d'ordre arbitraire est de forcer l'erreur d'une variable et sa dérivée à zéro. Le suivi d'une trajectoire souhaitée $q_d(t)$ est la tâche principale de l'axe du robot. Ici, commençons par définir un vecteur d'erreur de 2 dimension [24] :

$$e = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} q_d - x_1 \\ \dot{q}_d - \dot{x}_2 \end{bmatrix} = \begin{bmatrix} q_d - q \\ \dot{q}_d - \dot{q} \end{bmatrix} \quad (3.57)$$

Puis, un vecteur à n dimensions de la fonction de commutation :

$$s(e) = \Psi e = [\Lambda I] \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \Lambda e_1 + \dot{e}_1 \quad (3.58)$$

où : \dot{e} représente l'erreur de vitesse de poursuite et : $\Lambda = \text{diag}[\lambda_1, \dots, \lambda_n]$, avec : $\lambda_i > 0$

Qui détermine la bande passante du système. Ensuite, la dérivée temporelle de (3.58) est prise comme suit [24] :

$$\dot{s}(e) = \Lambda \dot{e}_1 + \ddot{e}_1 = \Lambda \dot{e}_1 + \ddot{q}_d - \ddot{q} \quad (3.59)$$

Maintenant, le taux constant plus proportionnel atteint la loi représentée par :

$$\dot{s} = -Q \text{sign}(s) - K s \quad (3.60)$$

En remplaçant (3.60) dans (3.59) et en mettant \ddot{q} à part, on obtient :

$$\ddot{q} = Q \text{sign}(s) + K s + \Lambda \dot{e}_1 + \ddot{q}_d \quad (3.61)$$

Enfin, la substitution de (3.61) dans l'équation dynamique du système non linéaire (3.35) conduit également à obtenir la loi de commande finale définie comme suit :

$$M(q)[Q \text{sign}(s) + K s + \Lambda \dot{e}_1 + \ddot{q}_d] + C(q, \dot{q})\dot{q} + g(q) + f(\tau, \dot{q}) = \tau \quad (3.62)$$

En utilisant le logiciel Matlab/Simulink avec les matrices Q , K et Λ sont définies comme suit :

$$Q = \begin{bmatrix} 12 & 0 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 5 \end{bmatrix}, \quad K = \begin{bmatrix} -10 & 0 & 0 \\ 0 & -4.5 & 0 \\ 0 & 0 & -5 \end{bmatrix}, \quad \Lambda = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

Les matrices Q , K et Λ précédentes ont été choisies spécifiquement pour la trajectoire utilisée $(1.2, 2, 1)^T$ car généralement la matrice Q change selon la formule suivante[25] :

$$Q = \varepsilon s^\beta e^{\alpha-t} \quad (3.63)$$

où : ε , β , and α sont des variables choisies en conséquence, pour contrôler le gain en fonction de la trajectoire utilisée.

et :

$$f(\tau, \dot{q}) = 100 \times \dot{q}$$

Nous avons utilisé le même schéma que dans l'étude PD, la seule différence est que nous avons injecté la loi de commande MG à la place du PD :

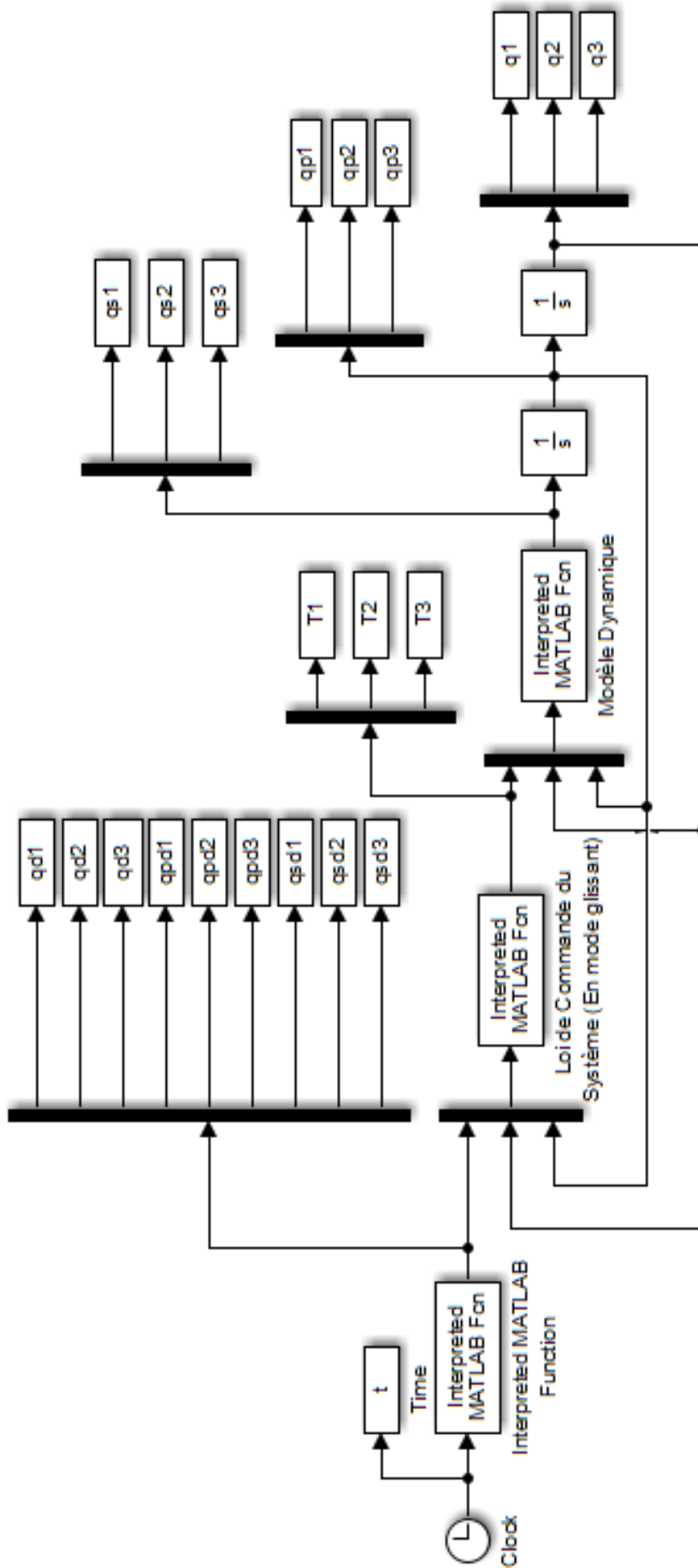
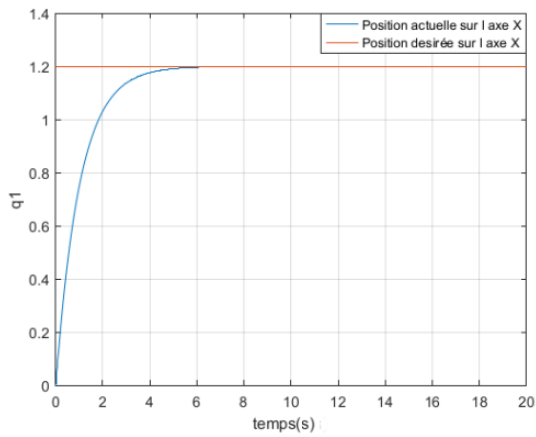
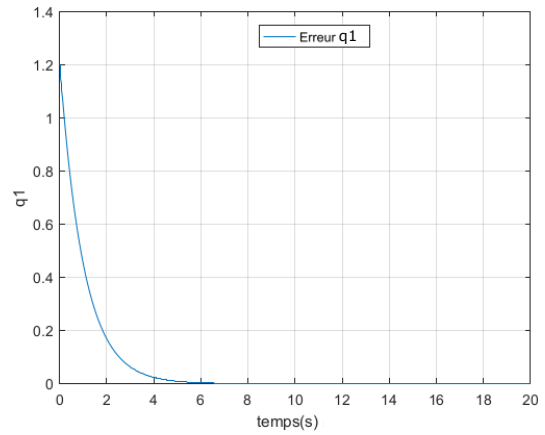


FIGURE 3.16 – Schéma de la simulation MG

Les résultats de la simulation sont les suivants :

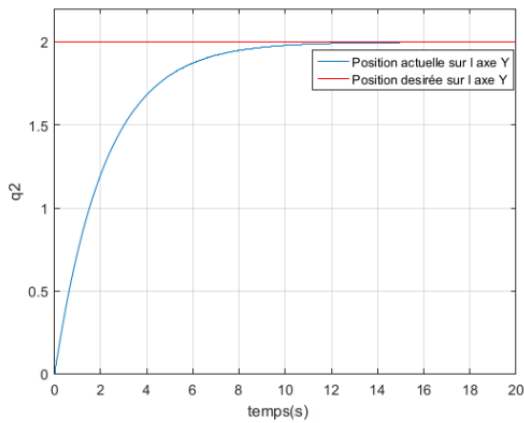


(a) Position

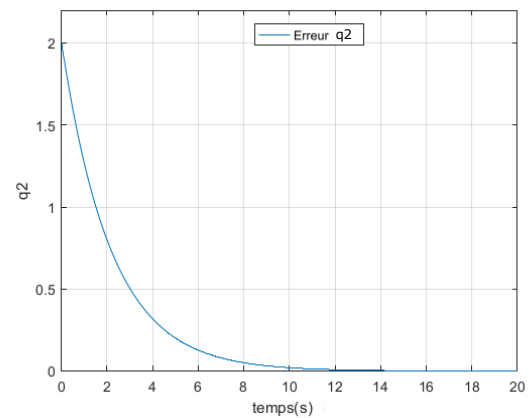


(b) Erreur de position

FIGURE 3.17 – Position et erreur de position de l’axe X , avec des valeurs de gain plus élevées

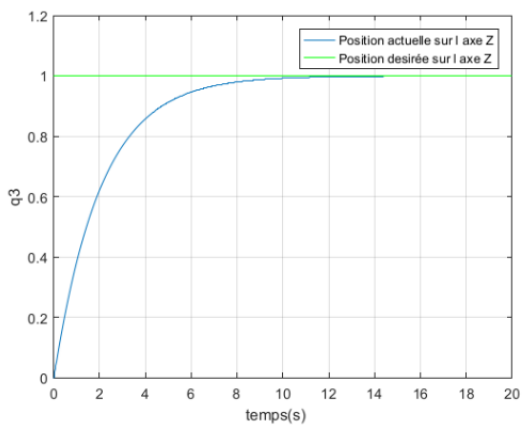


(a) Position

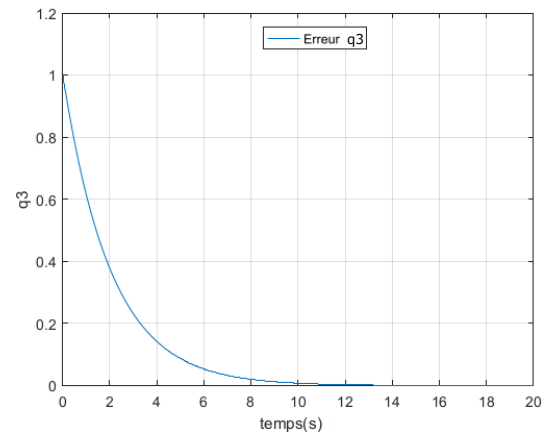


(b) Erreur de position

FIGURE 3.18 – Position et erreur de position de l’axe Y , avec des valeurs de gain plus élevées

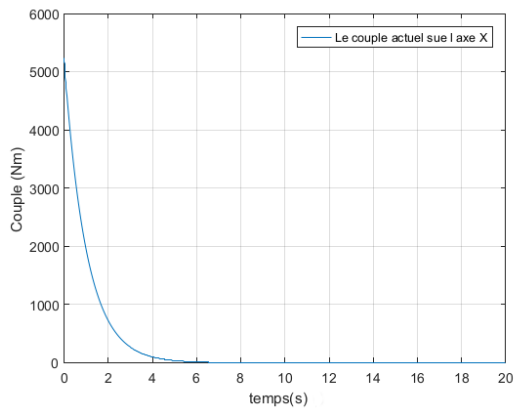


(a) Position

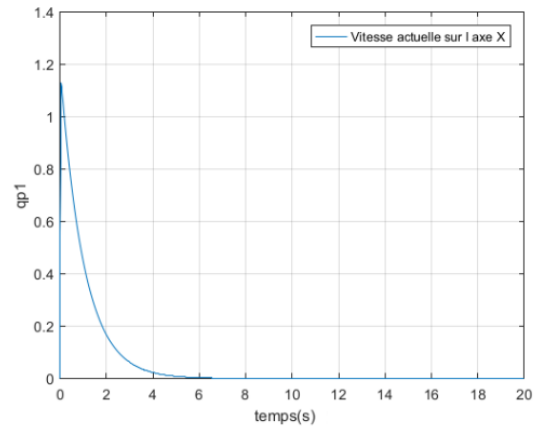


(b) Erreur de position

FIGURE 3.19 – Position et erreur de position de l’axe Z , avec des valeurs de gain plus élevées

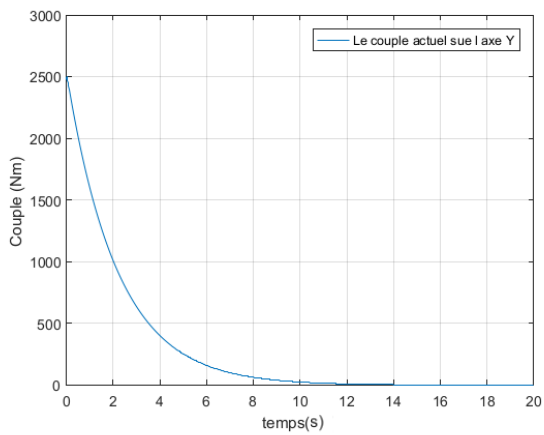


(a) Couple

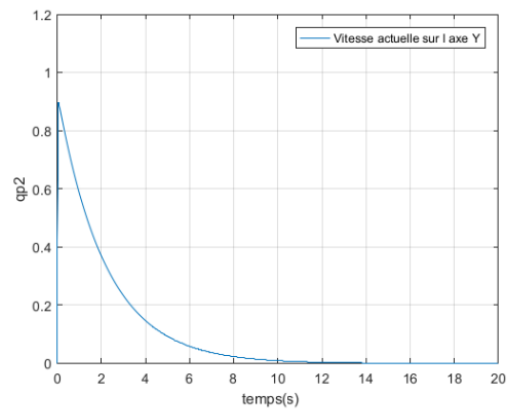


(b) Vitesse

FIGURE 3.20 – Couple et vitesse de l'axe X, avec des valeurs de gain plus élevées

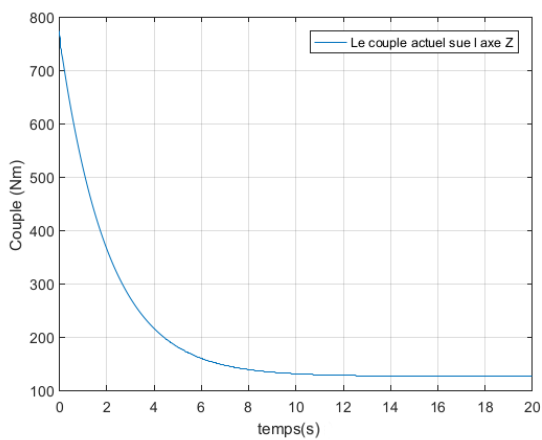


(a) Couple

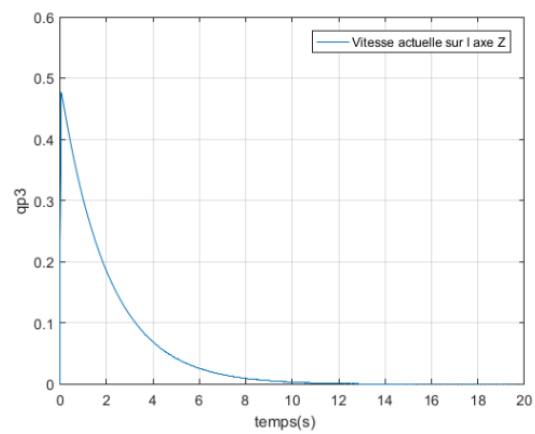


(b) Vitesse

FIGURE 3.21 – Couple et vitesse de l'axe Y, avec des valeurs de gain plus élevées



(a) Couple



(b) Vitesse

FIGURE 3.22 – Couple et vitesse de l'axe Z, avec des valeurs de gain plus élevées

Dans cette partie de la simulation, la stabilité du système est évidente. Comme le montrent les figures ci-dessus, les erreurs de positions sont pratiquement nulles. Les courbes de positions et de vitesses sont sans dépassement. Et les vitesses des axes du robot sont facilement contrôlées par la matrice Q du système tandis que les amplitudes peuvent être contrôlées par la matrice K .

Par comparaison entre les deux lois de commande utilisées, il est facile de constater que les deux lois sont très utilisables sous certaines conditions qui doivent être satisfaites pour obtenir de bons résultats. Pour être plus précis, la loi de commande PD peut être utilisée dans la plupart des systèmes ou l'intervalle de l'erreur de précision est un peu large.

Pour la deuxième commande, il est beaucoup plus précis et peut être encore plus rapide tant que le système peut être contrôlé au maximum.

3.4 Conclusion

Dans ce chapitre, nous avons présenté deux lois de commande, une classique et une autre avancée. Il n'a pas été facile d'obtenir certains résultats par simulation, mais nous avons réussi à obtenir des résultats aussi proches que possible.

Nous avons également vu quelle peut être la différence entre ces deux types de commande, et laquelle peut être la plus avantageuse dans certains cas et sous certaines conditions définies.

En fin de chapitre, nous avons pu présenter les résultats des simulations que nous menons en fonction des études que nous avons effectuées. Et dans ces résultats, nous n'avons pas accordé beaucoup d'attention au temps ou au fait que les résultats étaient suffisamment réalistes pour être exécutés en temps réel. Parce que notre seule attention était de comparer les deux types de commande.

Conclusion

Dans notre projet de fin d'études de master intitulé "Commande Avancée d'un Robot Cartésien" nous nous sommes concentrés sur l'étude des robots cartésiens géométriquement, cinématiquement et dynamiquement et en testant les résultats par simulation.

Nous avons opté pour cette étude avec l'attention en savoir plus sur les robots manipulateurs en particulier et pour obtenir une perspective plus claire sur la façon dont les différents domaines de l'ingénierie peuvent tous être mis en œuvre dans le processus de construction d'une certaine machine.

Tout au long de notre travail, nous avons élaboré un modèle mathématique pour les robots cartésiens à trois degrés de liberté, une synthèse de lois de commande de stabilisation basées d'abord sur l'approche classique PD et ensuite une loi de commande avancée basée la philosophie du mode glissant a été développée. Les résultats théoriques obtenus durant cette étude ont été validés par des simulations l'environnement de simulation Matlab/Simulink.

Par ailleurs, ce projet nous a permis d'appliquer et d'approfondir et de consolider nos connaissances théoriques que nous avons acquis durant notre cursus universitaire.

En fin de compte, ce projet peut être d'un grand intérêt si l'on considère la popularité de ce type de robots dans les environnements industriels. Comme perspectives, on vise à :

- Créez notre propre modèle géométrique de ce type de robots.
- Synthétiser des lois de commande (commande pour contrôler les mouvements, la précision, la répétabilité et la vitesse des axes) basée sur les lois de commande et les approches avancées.

Bibliographie

- [1] Mariane Davids. A brief history of robots in manufacturing. *Robotiq*, Jul 17, 2017 7 :00 AM.
- [2] Keith Dinwiddie. *Basic Robotics.p.116*. 2015.
- [3] D. Zhang and B. Wei. *Mechatronics and Robotics Engineering for Advanced and Intelligent Manufacturing .p.31*. Lecture Notes in Mechanical Engineering. Springer International Publishing, 2016.
- [4] Mingtu Ma ; Yisheng Zhang. *Advanced high strength steel and press hardening : proceedings of the 4th International Conference on Advanced High Strength Steel and Press Hardening (Ichs 2018).p.526*. Singapore : World Scientific Publishing, 2019.
- [5] H.H. Poole. *Fundamentals of Robotics Engineering.p.35*. Springer Netherlands, 2012.
- [6] Cartesian and gantry robots. *Robotpark Academy*.
- [7] Sales Yamaha Robotics Operations, FA Section and Marketing Division. Cartesian robots (application examples). *Yamaha Industrial robots*.
- [8] Thomson linear motion optimized. Pick and place palletizing. *Thomson linear motion*.
- [9] Mélanie R. Fonctionnement d'une imprimante 3d fdm par catégorie. *3D Natives*, 06-06-2017.
- [10] A. Emre Cetin. *Handling Interaction Forces Between an Object and Cartesian Robot Arm in Planar Motion.P.12*. 08-2002.
- [11] John J. Craig. *Introduction to Robotics Mechanics and Control*. Pearson Education, 2005.
- [12] T.C.Yih Kevin L.Conrad, Panayiotis S.Shiakolas. Robotic calibration issues : Accuracy, repeatability and calibration, 2000.
- [13] Luciano Selva Ginani José Maurício S.T. Motta. Theoretical and practical aspects of robot calibration with experimental verification. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 2011.
- [14] W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots*. Kogan Page Science paper edition Modeling, identification & control of robots. Elsevier Science, 2004.
- [15] Hartenberg R.S. Denavit.J. A kinematic notation for lower pair mechanisms based on matrices. transaction of the asme. *journal of Applied Mechanics*, vol. 22, pp. 215-221, 1955.
- [16] Etienne Dombre W.Khalil. *Bases de la modélisation et de la commande des robots-manipulateurs de type série*. 08 mai 2012.
- [17] Mr. Abdelkader Kerraci. Commande de robot manipulateurs. p.08., 2019/2020.
- [18] Fernando Reyes Cortes Pablo Sanchez. *Cartesian Control for Robot Manipulators*. 2010.

- [19] KERRACI Abdelkader. *SYNTHÈSE DES COMMANDES ROBUSTES DES ROBOTS MANIPULATEURS RIGIDES*. MÊmoire, Université d'Oran Es-Sénia, 05 Décembre 2004.
- [20] V Utkin. *Variable structure systems with sliding modes*, pp. 212–222. 1977.
- [21] Chitrlekha Mahanta Nabanita Adhikary. Sliding mode control of position commanded robot manipulators. *the International Federation of Automatic Control* (), 2018.
- [22] İhsan Ömür Bucak. *An In-Depth Analysis of Sliding Mode Control and Its Application to Robotics*. July 16th 2020.
- [23] Weiping Li Jean-Jacques Slotine. *Applied Nonlinear Control*. Pearson, 1991.
- [24] Moldoveanu F. Sliding mode controller design for robot manipulators. *Bulletin of the Transilvania University of Braşov Vol. 7 (56) No. 2*, 2014.
- [25] Baolin Hou Xi Wang. Sliding mode control of a 2-dof manipulator with randombase vibration based on modified exponential reaching law, 29 August 2016.