



DEPARTEMENT DE PHYSIQUE
DEPARTMENT OF PHYSICS

ATTESTATION DE CORRECTION DE LA THESE DE
DOCTORAT/Ph.D

Nous, Professeurs, **BODO Bertrand**, **MBINACK Clément**, **CHEUKEM André**, et Professeur **ESSIMBI ZOBO Bernard**, respectivement Examineurs et Président du jury de la thèse de Doctorat/Ph.D de Monsieur **MEDZO ABA Charles**, Matricule **12W1781**, préparée sous la codirection du Professeur **MELINGUI Achille** et Docteur **MVOGO AHANDA Joseph**, et intitulée : « **Stratégies de commandes stables des robots manipulateurs industriels à architecture de commande interne inconnue et inaccessible, évoluant dans un environnement déterministe ou aléatoire** », soutenue le Mardi , **30 Juillet 2024**, en vue de l'obtention du grade de Docteur/Ph.D en Physique, Spécialité **Energie et Systèmes Electriques et Electroniques**, attestons que toutes les corrections demandées par le Jury de soutenance ont été effectuées.

En foi de quoi, la présente attestation lui est délivrée pour servir et valoir ce que de droit.

Fait à Yaoundé le *04/05/25*

Examineur

BODO Bertrand, Pr, UY1

Le Président du Jury

ESSIMBI ZOBO Bernard, Pr,



Le Chef de Département de Physique

Jayaka Jean-Marie Brionvenou
Professeur

RÉPUBLIQUE DU CAMEROUN

Paix-Travail-Patrie

UNIVERSITÉ DE YAOUNDE I

Centre de Recherche et de formation
Doctotale en Sciences, Technologies
et Geosciences
Unité de Recherche et de Formation
Doctorale en Physique et Applications
B.P 812 Yaoundé
Email : crfd_stg@uy1.uninet.cm



REPUBLIC OF CAMEROON

Peace-Work-Fatherland

THE UNIVERSITY OF YAOUNDE I

Postgraduate Shool of Science,
Technology and
Geosciences
Research and Postgraduate Training
Unit for Physics and Applications
P.O. Box 812 Yaoundé
Email : crfd_stg@uy1.uninet.cm

THÈSE

En vue de l'obtention du grade de
Docteur/PhD de l'Université de Yaoundé I,
(Spécialité Automatique)

Laboratoire d'Électronique Électrotechnique et Automatique



Présentée et soutenue le 30 Juillet 2024 par :

MEDZO ABA Charles

Matricule : 12W1781

(Master en Physique)

**Stratégies de commandes stables des robots manipulateurs industriels à
architecture de commande interne inconnue et inaccessible, évoluant
dans un environnement déterministe ou aléatoire**

Devant un jury composé de :

ESSIMBI ZOBO BERNARD, PR	Université de Yaoundé I, Cameroun	Président
MELINGUI ACHILLE, MC	Université de Yaoundé I, Cameroun	Rapporteur
MVOGO AHANDA J.J.B, CC	Université d'Ebolowa, Cameroun	Rapporteur
BODO BERTRAND, PR	Université de Yaoundé I, Cameroun	Examineur
CHEUKEM ANDRÉ, MC	Université de Dschang, Cameroun	Examineur
MBINACK CLÉMENT, MC	Université de Yaoundé I, Cameroun	Examineur

Année Académique 2024-2025

Remerciements

L'achèvement de ce travail mené sur plusieurs années nous procure une grande satisfaction. C'est l'occasion pour nous de remercier toutes les personnes qui nous ont soutenu durant toutes ces années.

Je tiens à exprimer ma profonde gratitude à M. Achille MELINGUI, Maître de Conférences à l'Ecole Nationale Supérieure Polytechnique de l'Université de Yaoundé 1 comme superviseur et à M. Joseph Jean-Baptiste MVOGO AHANDA, Chargé de Cours à l'Université d'Ebolowa comme Directeur, pour la confiance qu'ils m'ont témoignée en acceptant la direction scientifique de mes travaux. Je leur suis reconnaissant de m'avoir fait bénéficier tout au long de ce travail de leur expérience, leur rigueur intellectuelle, leur dynamisme et de leur efficacité que je n'oublierai jamais. Soyez rassuré de mon attachement et de ma profonde gratitude.

Je souhaite remercier l'ensemble des membres du jury qui ont examiné cette thèse pour leurs remarques constructives, leurs observations enrichissantes. Je tiens à remercier chaleureusement le Professeur ESSIMBI ZOBO Bernard pour avoir accepté de présider mon jury. Je remercie vivement le Professeur BODO Bertrand ; le Maître de Conférences CHEUKEM Clément et le Maître de Conférences MBINACK Clément pour avoir accepté d'examiner cette thèse.

Certaines expériences de cette thèse ont été réalisées au Centre de Recherche en Informatique, Signal et Automatique de Lille (CRISal) de l'Université de Lille. Nous profitons de cette occasion pour remercier le Professeur Rochdi MERZOUKI, pour son accompagnement.

Je remercie tous les membres de l'équipe de recherche du laboratoire de Robotique et Vision de l'ENSP-YDE 1 pour leurs nombreux conseils, leurs disponibilités et leurs soutiens pendant ces années de thèse.

Je réserve une pensée spéciale à tous les enseignants du Département de Physique de la Faculté Science de l'Université de Yaoundé 1. Je remercie le Professeur NDJAKA Jean Marie, Chef de Département, ainsi que son personnel pour m'avoir offert un cadre et une atmosphère de travail agréable. Mes remerciements vont également aux enseignants du laboratoire d'électronique pour les enseignements reçus, les conseils et l'encadrement reçu

tout au long de mon cursus académique.

Je tiens également à exprimer ma profonde gratitude au Dr OLLE OLLE Daniel, Directeur Adjoint de L'Ecole Normale Supérieure de l'Enseignement Technique de l'Université d'Ebolowa et Chef de Département du Génie Informatique de la même institution ; ainsi que le personnel dudit département pour leurs soutiens, leurs conseils et leurs assistances.

Je remercie la famille ABA NTSENE pour leur soutien moral, spirituel et matériel.

Je remercie ma chère épouse NGOH MOTTO Francine Linda pour son soutien quotidien indéfectible et son enthousiasme contagieux. Notre couple a grandi en même temps que ce projet scientifique, le premier servant de socle solide à l'épanouissement du second.

Le regretté professeur Jean Bosco MBEDE m'a transmis cette passion pour la recherche, la robotique et a profondément enrichi mes connaissances scientifiques. Je te dédie cette thèse.

Résumé

Les robots industriels réalisent de nos jours des tâches plus complexes telles que la collaboration homme-robot et la préhension souple des objets dans des environnements non structurés où ils subissent des perturbations aléatoires. Une commande efficace dans de tels environnements nécessite une interaction du robot avec son environnement et par conséquent le développement des stratégies de commandes en couple. Cependant, la commande en couple n'est pas toujours possible, car pour des raisons de sécurité et de protection de la propriété intellectuelle, plusieurs constructeurs fabriquent des robots industriels avec une architecture de commande interne inconnue et inaccessible. Dans ce contexte, la commande dans l'espace de tâche est la plus efficace. Cette stratégie de commande est régulièrement confrontée au problème de stabilité globale. Ce problème de stabilité s'accroît lorsque l'architecture du contrôleur interne est inconnue et inaccessible et davantage lorsque le robot évolue en environnement aléatoire. Dans cette thèse, nous proposons premièrement un contrôleur externe adaptatif basé sur un réseau neuronal à fonction radiale, qui approxime les dynamiques du contrôleur interne inconnu et inaccessible afin d'imposer les dynamiques désirées par l'utilisateur en éliminant les effets du contrôleur interne dans la boucle de commande. Par la suite, nous proposons une approche de commande adaptative hybride qui combine une méthode adaptative indirecte pour le rejet des perturbations déterministes et directe pour le rejet des perturbations aléatoires. L'utilisation de la théorie de Lyapunov nous permet de démontrer que les lois de commandes proposées assurent une stabilité semi-globale en boucle fermée. Les simulations réalisées donnent des performances en suivi de trajectoire de l'ordre $1 \times 10^{-5} m$ en environnement déterministe et de l'ordre de $1.5 \times 10^{-4} m$ en environnement aléatoire. Les résultats des expériences réalisées sur les robots SCORBOT-ER de Intelek et UR5 de Cobot, ont donné les performances en suivi de trajectoire de l'ordre de $1 \times 10^{-4} m$ en environnement déterministe et de $1.8 \times 10^{-3} m$ en environnement aléatoire.

Mots clés : Robot Manipulateur Industriel, Architecture fermée, Commande dans l'espace des tâches, Commande adaptative, Environnements aléatoires, Réseaux de neurones.

Abstract

Nowadays, industrial robots realize more complex tasks, such as human-robot collaboration and the flexible prehension of objects in unstructured environments where they are subject to random disturbances. In such environments, effective control requires the robot to interact with its environment, and therefore the development of torque control strategies. However, torque control is not always possible, because for reasons of safety and protection of intellectual property, many manufacturers produce industrial robots with an unknown and inaccessible internal control architecture. In this context, task space control is the most efficient. This control strategy is regularly confronted with the problem of global stability. This problem of stability is accentuated when the architecture of the internal controller is unknown and inaccessible, and even more so when the robot is evolving in a random environment. In this thesis, we propose firstly an adaptive external controller based on a radial function neural network, which approximates the dynamics of the unknown and inaccessible internal controller in order to impose the dynamics desired by the user by eliminating the effects of the internal controller in the control loop. We then propose a hybrid adaptive control approach that combines an indirect adaptive method for rejecting deterministic disturbances and a direct method for rejecting random disturbances. The use of Lyapunov theory enables us to demonstrate that the proposed control laws ensure semi-global closed-loop stability. The simulations carried out give trajectory-following performances of the order of $1 \times 10^{-5} m$ in a deterministic environment and of the order of $1.5 \times 10^{-4} m$ in a random environment. The results of experiments carried out on Inteltek's SCORBOT-ER and Cobot's UR5 robots gave trajectory-following performances of the order of $1 \times 10^{-4} m$ in a deterministic environment and $1.8 \times 10^{-3} m$ in a random environment.

Key words : Industrial Robot Manipulator, Closed Architecture, Task Space Control, Adaptive Control, Random Environments, Neural Networks.

Table des matières

Liste des figures	viii
Liste des tableaux	x
Introduction Générale	1
0.1 Contexte et motivation	1
0.2 Problématique	2
0.3 Objectifs et contributions	3
0.3.1 Objectifs	3
0.3.2 Contributions	3
0.4 Organisation du manuscrit	4
1 État de l’art	7
1.1 Introduction	7
1.2 Historique de l’évolution des supports de commande des robots manipulateurs industriels	8
1.3 Commande des robots manipulateurs industriels	10
1.3.1 Généralités sur la commande des robots manipulateurs industriels	10
1.3.2 Stratégies de commande pour la stabilité globale des RMIs	13
1.3.3 Commandes des RMIs en environnement aléatoire	16
1.4 Préliminaires	18
1.4.1 Modélisation des robots manipulateurs industriels	18
1.4.2 Les réseaux de neurones à fonctions de base radiale	24
1.4.3 La théorie de Lyapunov	27
1.4.4 L’inégalité de Young	31
1.4.5 Processus stochastiques	32
1.4.5.1 Équations différentielles stochastiques de type Itô	35
1.4.5.2 Équations différentielles aléatoires	37

1.5	Conclusion	38
2	Méthodes et Outils	39
2.1	Introduction	39
2.2	Commande dans l'espace de tâches des RMIs à architecture de commande interne inconnue et inaccessible évoluant en environnement déterministe . .	40
2.2.1	Présentation de l'approche de commande	40
2.2.2	Présentation de l'algorithme de commande	43
2.2.3	Analyse de la stabilité du système bouclé	48
2.3	Commande Neuro-Adaptative des RMIACIIIs évoluant dans un environnement aléatoire	57
2.3.1	Présentation de l'approche de commande	58
2.3.2	Modélisation d'un robot manipulateur industriel en environnement aléatoire	59
2.3.3	Linéarisation du couple stochastique généralisée τ_c	64
2.3.4	Présentation de l'algorithme de commande	66
2.3.5	Analyse de la stabilité	70
2.4	Description des plates-formes expérimentales	76
2.4.1	Description du Scrobot ER-4U	76
2.4.2	Description du robot UR5	77
2.5	Conclusion	79
3	Résultats et discussions	81
3.1	Introduction	81
3.2	Commande dans l'espace des tâches d'un robot manipulateur industriel à architecture de commande interne inconnue et inaccessible évoluant en environnement déterministe	81
3.2.1	Résultats des simulations effectuées sur le modèle numérique du Scrobot ER-4U	82
3.2.2	Résultats des expériences effectuées sur le robot Scrobot ER-4U . .	84
3.2.3	Résultats des expériences effectuées sur le robot UR5	92
3.3	Commande neuro adaptative dans l'espace de tâche des RMIACIIIs évoluant dans un environnement aléatoire	95
3.3.1	Résultats des simulations effectuées sur le Scrobot ER-4U	96

3.3.2 Résultats des simulations effectuées sur le robot UR5	104
3.4 Conclusion	106
Bibliographie	111

Liste des figures

1	Architecture générale de commande dans l'espace des tâches des RMIs.	2
1.1	Les consoles de programmations	9
1.2	Principe de la commande dans l'espace articulaire	11
1.3	Architecture générale de commande dans l'espace des tâches des RMIs.	12
1.4	Cinématique générale d'un RMI planaire à n DDL	19
1.5	Axes de coordonnées de la convention de Denavit-Hartenberg [58].	20
1.6	Structure d'un RBF NN.	26
1.7	Illustration de la définition intuitive de la stabilité.	28
1.8	Types de stabilité.	29
2.1	Architecture générale de commande dans l'espace des tâches des RMIs.	41
2.2	Organigramme de la commande dans l'espace des tâches d'un RMI à architecture de commande interne inconnue et inaccessible	47
2.3	Modèle d'un RMI n-DDL en environnement aléatoire.	59
2.4	Propagation des accélérations aléatoires.	60
2.5	Vue d'ensemble du schéma de contrôle proposé.	69
2.6	Scorbot ER-4u.	76
2.7	Vue d'ensemble du dispositif expérimental avec l'annotation des principaux composants.	77
2.8	Robot UR5.	78
3.1	Résultats des simulations : (a) Erreurs de suivi sur l'axe X, (b) Erreurs de suivi sur l'axe Y et (c) Erreurs de suivi sur l'axe Z.	83
3.2	Erreurs de suivi sur les axes X,Y,Z pour $\varpi_i = \{10^{-3}, 0.5, 1\}$, $i = 1, 2, 3$	85
3.3	Signaux de commande des vitesses \dot{q}_{cj} , $j = \{1, 2, 3, 4\}$, $\varpi_i = \{10^{-3}, 0.5, 1\}$, $i = 1, 2, 3$	86
3.4	Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $k = \{1, 2, 3\}$	87

3.5	Résultats expérimentaux pour $k = \{1, 2, 3\}$, commandes de vitesse \dot{q}_{cj} , $j = \{1, 2, 3, 4\}$	87
3.6	Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $\alpha_{ii} = \{1, 2, 4\}$	89
3.7	Commandes de vitesse \dot{q}_{cj} , $j = \{1, 2, 3, 4\}$ pour $\alpha_{ii} = \{1, 2, 4\}$	89
3.8	Résultat de l'expérience : Suivi de trajectoire de l'effecteur.	91
3.9	Résultat de l'expérience, Erreurs de suivi sur chaque axe.	91
3.10	Résultat de l'expérience, Signaux de commande.	91
3.11	Résultat de l'expérience effectué sur l'UR5 : Suivi de trajectoire	93
3.12	Résultat de l'expérience effectuée sur l'UR5 : Vitesse de commande.	95
3.13	Résultat de l'expérience effectué sur l'UR5 : Erreur de suivi	95
3.14	Perturbations aléatoires ξ_1 , ξ_2 , et ξ_3	96
3.15	Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $\varrho_2 = \{0.01, 0.5, 1\}$	99
3.16	Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$ pour $\varrho_2 = \{0.01, 0.5, 1\}$, Com- mandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$	99
3.17	Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3$	100
3.18	Résultats de simulation pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3$; Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$	101
3.19	Erreurs de suivi sur chaque axe.	102
3.20	Suivi de Trajectoire.	103
3.21	Résultats de l'étude comparative, Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$ pour les deux contrôleurs.	103
3.22	Suivi de Trajectoire	105
3.23	Erreurs de suivi.	106
3.24	Vitesses de commande	106

Liste des tableaux

2.1	Description des mouvements du Scorbot ER-4U	77
2.2	Paramètres du SCORBOT ER-4U	77
2.3	Paramètre de Denavit Hartenberg de l'UR5	78
2.4	Paramètres d'inertie de l'UR5	78
3.1	Valeurs Max,Min de l'erreur de suivi sur les axes X,Y,Z pour $\varpi_i = \{10^{-3}, 0.5, 1\}$, $i = 1, 2, 3$	85
3.2	Commandes de vitesse \dot{q}_{cj}	86
3.3	Valeurs Max,Min de l'erreur de suivi sur les axes X,Y,Z pour $k = 1, 2, 3$	88
3.4	Commandes de vitesse \dot{q}_{cj} pour $k = 1, 2, 3$	88
3.5	Valeurs Max/Min de l'erreur de suivi sur les axes X,Y,Z pour $\alpha_{ii} = \{1, 2, 4\}$	88
3.6	Valeurs Max/Min des signaux de Commandes de vitesse \dot{q}_{cj} pour $\alpha_{ii} =$ $\{1, 2, 4\}$	90
3.7	Valeurs Max/Min de l'erreur de suivi sur les axes X,Y,Z et temps de réponse des différents contrôleurs.	92
3.8	Valeurs Max/Min des commandes de vitesse	92
3.9	Valeurs Max/Min des erreurs de suivi sur les axes	94
3.10	Valeurs Max/Min des angles et commandes de vitesse	94
3.11	Valeurs de l'erreur de suivi sur les axes pour $\varrho_2 = \{0.01, 0.5, 1\}$	98
3.12	Commande de vitesse \dot{q}_{cj} pour $\varrho_2 = \{0.01, 0.5, 1\}$	98
3.13	Valeurs de l'erreur de suivi sur les axes X,Y,Z pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3$	100
3.14	Commandes de vitesse \dot{q}_{cj} pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3, 4$	101
3.15	Valeurs Max/Min de l'erreur de suivi sur les axes X,Y,Z et temps de réponse des différents contrôleurs	102
3.16	Signaux de commandes.	103
3.17	Valeurs Max/Min des erreurs de suivi sur les axes.	105
3.18	Valeurs Max/Min des signaux de commande	105

Acronymes et abréviations

AMF	American Machine and Foundry
BLF	Barrier Lyapunov functions
CC	Courant Continu
DDL	Dégré De Liberté
DSC	Dynamic Surface Control
MDD	Modèle Dynamique Direct
MDI	Modèle Dynamique Inverse
MCD	Modèle Cinématique Direct
MCI	Modèle Cinématique Inverse
MIMO	Multi-Input Multi-Output
MGD	Modèle Géométrique Direct
MGI	Modèle Géométrique Inverse
PD	Proportional Derivative
PID	Proportional Integral Derivative
PMDC	Permanent Magnet Direct Current
PMC	Perceptron MultiCouche
PUMA	Programmable Universal Machine for Assembly
RBF NN	Radial Basis Function Neural Network
RDE	Random Differential Equation
RIA	Robotic Industries Association
RMI	Robot Manipulateur Industriel
RMIACIII	Robot Manipulateur Industriel à Architecture de Commande Interne Inconnue et Inaccessible
SDE	Stochastic Differential Equation
SCARA	Selective Compliance Assembly Robot Arm
T3	The Tomorrow Tool
UR	Universal Robots

Introduction générale

0.1 Contexte et motivation

L'industrie a subi de grands cycles d'innovation au fil des années qui ont profondément transformé la production industrielle, son organisation, ainsi que la dynamique économique dans son ensemble. Pour augmenter la production industrielle, il est devenu nécessaire de remplacer l'intervention humaine pour certaines tâches répétitives, dangereuses ou pénibles dans les unités de production. Dans cette mouvance, Jacques Vaucanson inventa le premier automate en 1737 [1]. Les automates ont fortement impacté la production industrielle grâce à l'automatisation des tâches, mais ils présentent certaines limites dans le processus d'automatisation des tâches complexes telles que la prise de décisions intelligentes, la préhension des objets, le temps d'exécution des tâches, etc. Par conséquent, s'inspirant des téléopérateurs utilisés dans l'industrie nucléaire dans les années 1950 pour la manipulation d'éléments radioactifs, George Devol inventa en 1954 le premier robot industriel, Unimate. Ce bras articulé fut exploité par General Motors en 1961 pour extraire les pièces d'une machine de moulage sous pression. La rapidité et la précision des robots manipulateurs dans l'exécution des tâches répétitives ont favorisé leur développement pour des tâches plus avancées telles que le soudage, l'assemblage, la manutention, le polissage et le poinçonnage. Cette génération de robot industriel était appelée à évoluer dans des environnements à la fois structurés et non structurés.

L'avènement de l'industrie de quatrième génération en 2012 va redéfinir et réorganiser les moyens de production. Les unités de production doivent désormais intégrer des systèmes de supervision en temps réel, être intelligentes et permettre la connexion horizontale et verticale entre les personnes, les machines, les objets, les systèmes d'informations et de communications afin de gérer dynamiquement les systèmes complexes [3–5]. Les robots industriels sont désormais destinés à réaliser des tâches et des applications plus complexes et futuristes, et doivent pouvoir interagir de manière intelligente avec leur environnement. Ainsi, le champ d'application des robots industriels s'est élargi à d'autres secteurs d'activité, tels que l'aéronautique, la marine, la défense et l'agriculture, dans lesquels ils sont exposés à des phénomènes stochastiques, comme des vibrations ou des perturbations aléatoires. **L'objectif principal de cette thèse est d'élaborer des stratégies de**

commande stable pour des robots manipulateurs industriels évoluant dans des environnements à la fois déterministes et aléatoires.

0.2 Problématique

L'utilisation des robots industriels dans des environnements aléatoires requiert une interaction constante avec l'environnement. Cette interaction permet au robot d'ajuster sa posture en fonction de la dynamique de l'environnement. De plus, les perturbations aléatoires affectent principalement le couple du robot. Il est donc nécessaire de développer une boucle de rejet de ces perturbations. Dans un tel contexte, la stratégie de commande adéquate est la commande dans l'espace des tâches qui intègre une boucle de contrôle interne et une boucle de contrôle externe (Fig.1).

La boucle de contrôle externe génère les consignes de vitesses/positions à partir des informations issues de l'environnement d'opération. Tandis que la boucle de contrôle interne génère quant à elle les couples nécessaires aux actionneurs afin de satisfaire les consignes de vitesses/positions. Cependant, pour des raisons de sécurité et de protection de la propriété intellectuelle, la boucle interne de la plupart des robots industriels commercialisés n'est pas toujours connue et accessible à l'utilisateur. Dans ce contexte, l'utilisateur ne peut qu'élaborer les lois de commande dédiées à la boucle externe, car la loi de commande de la boucle interne est déjà conçue et intégrée par le concepteur. Cela soulève le problème de stabilité globale du système composé, car il est démontré que deux sous-systèmes indépendamment stables ne garantissent pas la stabilité globale du système composé [6], [7]. Par conséquent, la question de recherche traitée dans cette thèse est la suivante :

« **Quelle stratégie de commande serait appropriée pour les robots manipulateurs industriels à structure interne inconnue et inaccessible opérant dans des environnements déterministes ou aléatoires ?** »

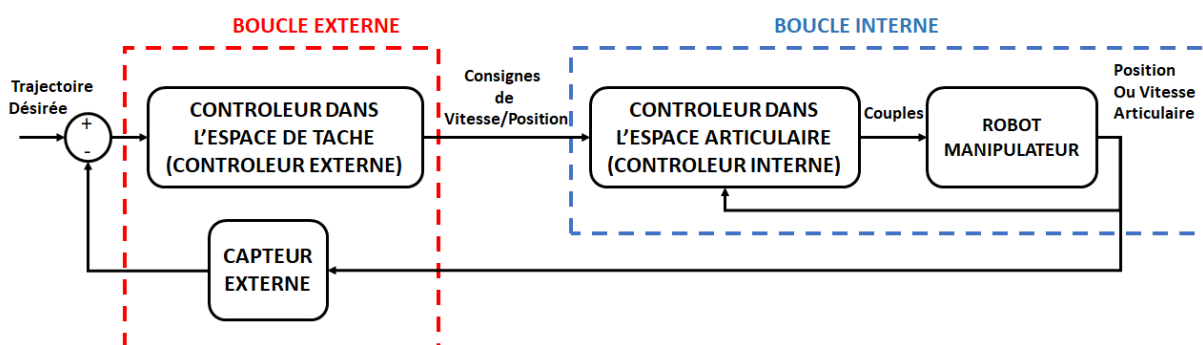


FIGURE 1 – Architecture générale de commande dans l'espace des tâches des RMIs.

0.3 Objectifs et contributions

0.3.1 Objectifs

L'élaboration des lois de commande des RMIs à contrôleur interne inaccessible et inconnu opérants dans des environnements déterministes ou aléatoires est l'objectif principal de cette thèse. Les objectifs spécifiques visés sont les suivants :

1. Développer une stratégie de commande efficace permettant de contourner la structure fermée des RMIs ,
2. Développer une stratégie de commande efficace pour les RMIs à contrôleur interne fermé permettant le rejet des perturbations déterministes et aléatoires.

0.3.2 Contributions

Plusieurs travaux ont été menés sur la commande dans l'espace de tâche et le rejet des perturbations aléatoires. Cependant, ces deux problématiques ont généralement été traitées de manière indépendante. Ce qui est problématique dans le contexte actuel de la robotique industrielle, où les performances en termes de rapidité, de stabilité et de robustesse sont essentielles.

Pour résoudre le problème de stabilité globale des stratégies de commande dans l'espace de tâche, les travaux effectués jusqu'à présent sur cette thématique s'appuient sur l'une des trois hypothèses simplificatrices suivantes :

- *H1 : La structure du contrôleur interne est connue et modifiable [8–10].*
- *H2 : La structure du contrôleur interne est connue mais non modifiable [11].*
- *H3 : La structure du contrôleur interne est inconnue, mais peut garantir la stabilité et la limitation des erreurs de suivi des vitesses articulaires [6, 7]*

Nous pouvons constater que les deux premières hypothèses ne peuvent pas d'assurer la stabilité globale du système composé (Voir Figure 1)) lorsque le contrôleur interne est inconnu et non modifiable. Par conséquent, toute approche de commande basée sur l'une de ces deux hypothèses ne peut pas être mise en œuvre dans le cadre de cette thèse. La dernière hypothèse suggère qu'il suffit que le contrôleur externe soit stable pour garantir la stabilité globale du système. Cependant, comme nous l'avons mentionné précédemment, la stabilité globale d'un système composé n'est pas toujours garantie lorsque les sous-systèmes sont stables indépendamment.

La première contribution de cette thèse vise à lever le verrou théorique relatif aux trois hypothèses su-citées. Ainsi, nous proposons un contrôleur externe qui contient la structure d'un contrôleur interne susceptible de garantir la stabilité du système bouclé tout en éliminant les effets du contrôleur interne conçu par le fabricant. Pour cela, nous exploitons le fait que pour des mesures de stabilité, la plupart des contrôleurs internes des RMIs contiennent un terme proportionnel en vitesse $K_d(\dot{q} - \dot{q}_c)$ avec K_d la matrice de gain inconnue, \dot{q} le vecteur des vitesses articulaires, et \dot{q}_c le vecteur des commandes des vitesses articulaires (sorties du contrôleur externe). A partir de cette observation, nous considérons que le contrôleur interne à la forme $u = -K_d(\dot{q} - \dot{q}_c) + \psi\left(q, \dot{q}, q_c, \int_0^t q(\varsigma)d\varsigma, \int_0^t q_c(\varsigma)d\varsigma\right)$, avec $\psi(*)$ une fonction inconnue et K_d une matrice de constante inconnue. L'idée ici est de concevoir \dot{q}_c qui est fonction de l'approximation de $\psi(*)$ et qui contraint le contrôleur interne à atteindre la stabilité.

En ce qui concerne le rejet des perturbations aléatoires, la littérature actuelle compte deux grandes approches. Pour mieux les appréhender, il est bon de savoir que les dynamiques excitatrices des perturbations aléatoires sont représentées par la matrice $\Gamma(q) \in \mathbb{R}^{n \times r}$, avec q le vecteur de n positions angulaires des articulations du robots et r la dimension de l'espace de travail [13]. La première approche de la littérature [14]- [15] consiste à effectuer le rejet des perturbations aléatoires en utilisant la norme de Frobenius $\|\Gamma(q)\|_F^2$ de la matrice $\Gamma(q)$. La seconde approche quant à elle utilise la norme cartésienne $\Gamma(q)\Gamma^T(q)$. Cependant, $\Gamma(q)$ dépend des paramètres dynamiques et cinématiques du robot, par conséquent ces approches sont sensibles aux erreurs de modélisation. **La seconde contribution** de cette thèse propose une méthode adaptative directe de rejet des perturbations aléatoires et une méthode adaptative indirecte pour la compensation des incertitudes de modélisation et des paramètres incertains. Pour ce faire, nous démontrons que pour tout vecteur $y \in \mathbb{R}^n$ le terme $\Gamma(q)\Gamma^T(q)y$ peut se mettre sous une forme paramétrique linéaire tel que $\Gamma(q)\Gamma^T(q)y = Q(q, y)\Theta$, avec $Q(q, y) \in \mathbb{R}^{n \times \bar{m}}$ le regressor, et $\Theta \in \mathbb{R}^{\bar{m}}$ le vecteur des paramètres du robot. Le vecteur Θ est mis à jour en ligne pour permettre le rejet optimal des perturbations aléatoires .

0.4 Organisation du manuscrit

Cette thèse est organisée en trois chapitres en plus de l'introduction et de la conclusion générale.

— Chapitre 1 : Nous présentons dans un premiers temps l'historique des supports de

commande des robots manipulateurs industriels. Ensuite, nous exposons un état de l'art sur les stratégies de commande existantes de ces robots. Enfin, nous présentons les outils mathématiques nécessaires à l'élaboration des lois de commande.

- Chapitre 2 : Ce chapitre présente nos deux principales contributions à savoir, la synthèse de la loi de commande des robots industriels à architecture de commande interne fermée et inconnue et la loi de commandes des robots à architecture de commande interne fermée et inconnue soumis aux perturbations aléatoires.
- Chapitre 3 : Il présente les résultats des simulations numériques et des expériences des algorithmes de commande développés au chapitre (2). Nous faisons une analyse détaillée de ces résultats et nous les comparons à ceux de la littérature.

Le manuscrit se termine par une conclusion qui résume les contributions principales et présente une discussion sur les travaux futurs à mener afin d'améliorer les résultats obtenus.

État de l'art

1.1 Introduction

De nos jours, les robots manipulateurs industriels (RMIs) sont appelés à réaliser des tâches de plus en plus complexes dans divers domaines, et à évoluer dans des environnements plus ou moins structurés où ils sont soumis à des perturbations de nature déterministe ou aléatoire. Dans le transport maritime, par exemple, les RMIs utilisés pour la maintenance des coques des navires subissent les mouvements aléatoires générés par les vagues [18, 19]. Dans le transport aérien, les RMIs utilisés pour le pilotage des avions sont soumis aux mouvements aléatoires induits par le vent. Pour les interventions chirurgicales robotisées, les RMIs doivent s'adapter aux battements du cœur. Comme nous l'avons mentionné précédemment, seules les stratégies de commande en couple peuvent garantir des performances acceptables dans ce contexte. Cependant, la forte concurrence sur le marché des RMIs contraint plusieurs constructeurs à concevoir des robots manipulateurs industriels à architecture de commande interne fermée et inaccessible.

Dans ce chapitre, nous présentons tout d'abord l'historique de l'évolution des supports de commande des robots manipulateurs industriels. Nous aborderons ensuite les généralités sur la commande des RMIs ainsi que les contributions significatives existantes sur la commande des RMIs en environnements déterministes et aléatoires. Enfin, nous présenterons les outils mathématiques nécessaires à la conception des lois de commande dans l'espace des tâches et au rejet des perturbations aléatoires.

1.2 Historique de l'évolution des supports de commande des robots manipulateurs industriels

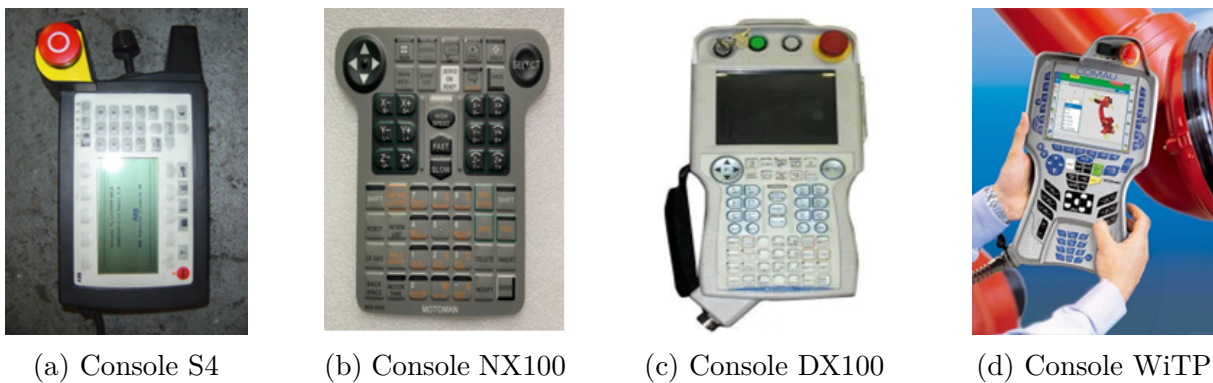
Au fil des années, les supports de commande des robots manipulateurs industriels ont connu plusieurs mutations en raison des avancées technologiques. Ils peuvent être classés en quatre générations distinctes.

La première génération de robots industriels s'étend de 1950 à 1967 [20]. Les robots de cette génération étaient des systèmes purement mécaniques actionnés par des actionneurs pneumatiques. Ces actionneurs étaient commandés par des systèmes logiques combinatoires constitués de portes logiques. Ces dernières étaient essentiellement constituées de commutateurs à cames ou de relais mécaniques [2, 20]. Les facteurs tels que l'encombrement, le coût de production, la consommation énergétique et les nuisances sonores n'ont pas favorisé l'expansion des supports de commande de cette génération.

La deuxième génération de robots manipulateurs industriels, qui se situe entre 1970 et 1980, connaît son essor grâce au développement des microprocesseurs et des actionneurs électriques [2, 20, 21]. L'apparition du tout premier automate programmable industriel (API) de la marque Allen-Bradley en 1970 et du tout premier micro-ordinateur, le Mircral, en 1973, a révolutionné les supports de commande des robots de cette génération. Contrairement à ceux de la première génération, ces robots sont capables de suivre des trajectoires de référence et d'exécuter des mouvements point à point. Ils sont contrôlés par des microprocesseurs, des API et des micro-ordinateurs. Les limites majeures de ces supports de commande sont la polyvalence des tâches à effectuer et le diagnostic. En effet, chaque support de commande est programmé dès la conception pour exécuter une tâche bien précise et ne peut pas être utilisé pour d'autres tâches. Les diagnostics/supervisions s'effectuaient au moyen de voyants lumineux, sans qu'aucune indication sur la cause réelle de la défaillance ne soit fournie.

L'introduction de la vision industrielle dans les unités de production a favorisé l'apparition des robots industriels de troisième génération, qui s'étend de 1980 à 2010 [2]. Ces robots se caractérisent par leur grande capacité d'interaction avec l'opérateur et l'environnement grâce à des interfaces de commande. Contrairement aux robots de la deuxième génération, les robots de la troisième sont programmés en ligne ou hors ligne à l'aide d'unités de commande conçues à cet effet. Cela leur confère un important potentiel opérationnel. De plus, le diagnostic et la supervision s'effectuent à l'aide d'interfaces de contrôle. Les systèmes de commande des robots ont considérablement évolué durant

cette période. En 1982, la société IBM a mis au point un langage de programmation à la fois puissant et facile d'utilisation : l'AML. Les ingénieurs pouvaient créer rapidement et facilement des programmes d'application en utilisant un ordinateur personnel IBM. En 1992, la compagnie suédoise ABB lance la première console de contrôle des robots : Le S4 (Figure 1.1a). En 1994, la société japonaise Yaskawa Motoman conçoit la première console de commande capable de contrôler deux robots de manière synchronisée : le MRC. En 2004, elle améliore le système de commande et propose : Le NX100 (Figure 1.1b). Le NX100 peut commander quatre robots de manière synchrone (jusqu'à 38 axes). Par la suite, elle développe la version Dx100 (Figure 1.1c) qui permet de contrôler huit robots sur 72 axes. Deux ans plus tard, la société italienne Comau lance la première console sans fil : le WiTP (Figure 1.1d). Les robots de la troisième génération ne permettaient pas d'effectuer des tâches complexes telles que le tri des objets, l'inspection de la qualité ou la détection des défauts.



(a) Console S4

(b) Console NX100

(c) Console DX100

(d) Console WiTP

FIGURE 1.1 – Les consoles de programmations

Grâce à l'avènement des circuits logiques programmables (FPGA, CPLD), la dernière génération des RMIs intègre de nouvelles technologies, notamment l'Internet des objets (IoT), le cloud computing et l'intelligence artificielle. Ces robots sont destinés à effectuer des tâches plus complexes, telles que le tri, le rangement, l'analyse des procédés, l'impression 3D, ainsi que la surveillance et le diagnostic des procédés. La puissance de calcul des supports de commande embarqués permet l'implémentation de lois de commande intelligentes. Contrairement aux trois premières générations de supports de commande, où seule la commande dans l'espace articulaire était possible, ceux de la quatrième génération permettent non seulement la commande dans l'espace articulaire, mais aussi la commande dans l'espace de tâches. Par conséquent, dans cette thèse, nous nous focalisons essentiellement sur les RMIs dotés de supports de commande de la quatrième génération.

1.3 Commande des robots manipulateurs industriels

Commander un RMI consiste à générer un ensemble d'actions permettant à son organe terminal, couramment appelé effecteur, d'occuper une séquence de positions et d'orientations afin d'exécuter la tâche assignée. Pour des performances optimales, les algorithmes de commande des RMIs doivent intégrer des termes assurant le rejet des perturbations causées par les incertitudes de modélisation et les phénomènes déterministes ou aléatoires. Dans cette section, nous présentons d'abord les stratégies de commande des RMIs, à savoir la commande dans l'espace articulaire et la commande dans l'espace de tâche. Nous exposerons ensuite quelques travaux réalisés sur les stratégies de commande dans l'espace de tâche visant à assurer la stabilité du système global. Enfin, nous présenterons les travaux effectués sur la commande des RMIs dans un environnement aléatoire.

1.3.1 Généralités sur la commande des robots manipulateurs industriels

Les tâches effectuées par les robots manipulateurs industriels se subdivisent en deux grandes catégories. La première catégorie englobe les tâches telles que la manutention, le polissage, etc. qui requièrent une interaction du robot avec l'environnement. La deuxième catégorie regroupe les tâches ne nécessitant pas d'interaction avec l'environnement, telles que la pulvérisation, la soudure à l'arc ou la peinture. Généralement, les tâches qui nécessitent une interaction du robot avec l'environnement sont réalisées à l'aide des stratégies de commande dans l'espace des tâches, tandis que celles qui ne requièrent aucune interaction sont réalisées à l'aide des stratégies de commande dans l'espace articulaire du robot.

Commande dans l'espace articulaire

La Figure 1.2 illustre la structure générale d'une commande dans l'espace articulaire. Elle comprend un modèle géométrique inverse et un contrôleur interne. La trajectoire cartésienne souhaitée x_d est convertie par ce modèle géométrique inverse en une trajectoire articulaire de référence q_r . Le couple τ est généré par le contrôleur interne à l'aide le signal d'erreur $\Delta e = q_r - q$. Dans la littérature, plusieurs travaux à l'instar de ceux de Vaccaro et al. [22], Visak et al. [23], Sung-moon et al. [24] ont implémenté cette stratégie de commande. Vaccaro et al. [22] proposent une commande dans l'espace articulaire qui permet d'obtenir les positions, les vitesses et les accélérations de référence des va-

riables articulaires. Visak et al. [23] proposent une méthode d'apprentissage basée sur le Renforcement Learning pour convertir directement les coordonnées de l'espace cartésien en positions articulaires de référence. Sung-moon et al. [24] conçoivent un contrôleur de couple dans l'espace articulaire basé sur la commande par retard de temps avec détection de collision. Pour répondre à la même problématique, Hu et al. [25] proposent une commande articulaire décentralisée capable d'estimer et de compenser les non-linéarités ainsi que les perturbations déterministes. De façon analogue, Seraji et al. [26] proposent une commande articulaire décentralisée robuste dans l'espace articulaire. Cette dernière combine la commande PID et la commande par anticipation position-vitesse-accélération pour chaque articulation. Comme nous l'avons mentionné plus haut, la commande dans l'espace articulaire est utilisée lorsque le robot n'interagit pas avec son environnement [23, 27, 28]. Lorsque la tâche nécessite un retour d'information de l'environnement, la stratégie de commande adéquate est la commande dans l'espace des tâches.

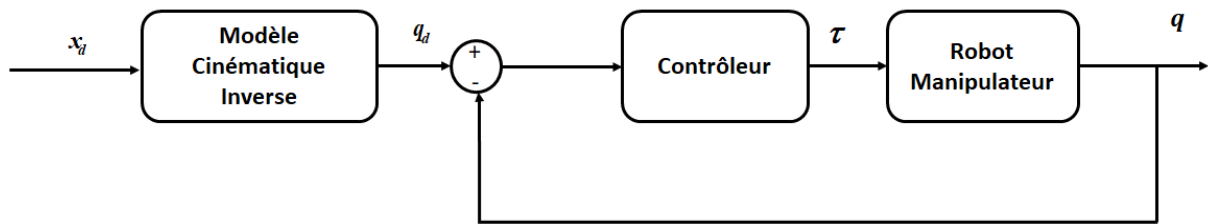


FIGURE 1.2 – Principe de la commande dans l'espace articulaire

Commande dans l'espace des tâches

La structure générale de la commande dans l'espace des tâches est présentée dans la Figure 1.3. Elle comprend principalement deux boucles de contrôle, une boucle externe et une boucle interne. La boucle externe assure l'asservissement des positions cartésiennes du robot à partir du signal d'erreur $\Delta x = x_d - x$ et des informations provenant de l'environnement d'opération (effort de l'effecteur, bruits externes, etc.). Elle comprend un contrôleur cartésien qui génère les consignes de vitesse/position articulaires de référence q_r/\dot{q}_r nécessaires pour garantir l'asservissement des positions ou de vitesses cartésiennes. La boucle interne utilise les informations proprioceptives du robot (positions, vitesses ou accélérations articulaires) pour garantir l'asservissement des vitesses/positions articulaires de référence. Elle comprend un contrôleur articulaire qui génère les couples u .

L'implémentation de la commande dans l'espace des tâches rencontre un certain nombre de difficultés liées à la structure mécanique du robot, au nombre et aux types de capteurs

utilisés. Ces difficultés soulèvent trois problèmes majeurs qui empêchent cette stratégie de commande d'assurer la stabilité du système global. Il s'agit des incertitudes et erreurs de modélisation, de l'absence de capteur de vitesse et des configurations particulières. Plusieurs solutions ont été proposées dans la littérature pour solutionner ces problèmes. Pour le problème de singularités, Shuzhi et al. [29] ont proposé une approximation neuronale de la matrice jacobienne inverse. Xian et al. [30] éliminent le problème de singularité en classifiant l'espace des tâches du robot manipulateur en plusieurs sous régions sans configurations singulières. La stabilité globale est garantie en proposant un contrôleur pour chaque sous région. Pour faire face aux incertitudes de modélisation, García et al. [31] ont proposé une commande neuro adaptative dans l'espace des tâches par mode glissant pour les RMIs à jacobien incertain. Shuzhi et al. [29] introduisent un paramètre robuste qui supprime les incertitudes de modélisation du robot et celles introduites par le réseau de neurones. Afin de résoudre le problème d'absence de capteur de vitesse cartésienne/articulaire, Xinwu et al. [32] ont proposé un observateur par mode glissant qui estime les signaux de vitesse.

Les contributions suscitées s'intéressent particulièrement aux problématiques classiques rencontrées dans la littérature, à savoir la gestion des singularités, les incertitudes de modélisation et l'absence de capteurs de vitesse. De nos jours, la forte concurrence sur le marché des RMIs oblige les constructeurs à concevoir des RMIs dotés d'une architecture de commande interne fermée et inaccessible. En outre, les RMIs sont appelés à évoluer dans des environnements complexes où ils peuvent être soumis à des perturbations de natures diverses(aléatoires et déterministes). Cela accentue le problème de la stabilité globale des approches de commande dans l'espace des tâches (voir la 1.3) et pose la problématique du rejet des perturbations aléatoires.

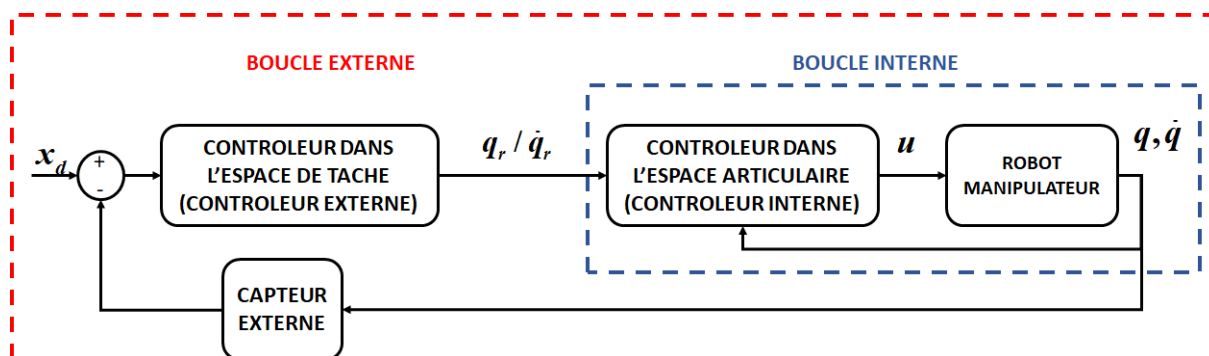


FIGURE 1.3 – Architecture générale de commande dans l'espace des tâches des RMIs.

1.3.2 Stratégies de commande pour la stabilité globale des RMIs

Comme l'indique la Figure 1.3, la commande des RMIs dans l'espace des tâches nécessite le développement d'un contrôleur externe et d'un contrôleur interne. Afin de garantir la stabilité globale des RMIs commandés par les lois de commande dans l'espace de tâche, la littérature actuelle fait appel à trois hypothèses simplificatrices. La première suppose que le contrôleur interne est connu et accessible (H1). La deuxième suppose que le contrôleur interne est connu, mais inaccessible (H2). La troisième émet l'hypothèse selon laquelle le contrôleur interne est inconnu et inaccessible, mais pour garantir la stabilité globale, il faut que les erreurs de vitesses articulaires soient bornées et que la norme de ces dernières soit intégrable (H3). Les sous-sections suivantes présentent les contributions pertinentes qui ont été faites sur la base de ces trois hypothèses.

Contrôleur interne connu et accessible : Hypothèse 1

L'Hypothèse (H1) est implicitement faite lorsque le contrôleur interne est connu et accessible [33, 34]. Ainsi, pour implémenter la commande, l'utilisateur peut modifier la loi de commande du contrôleur interne embarqué dans le RMI [8–10]. Cette hypothèse s'applique généralement aux RMIs vendus avec une interface de contrôle interne accessible. En exploitant cette hypothèse, plusieurs chercheurs [11, 35–37] proposent des lois de commande dans l'espace des tâches qui assurent la stabilité asymptotique du système globale (Voir Figure 1). Izadbakhsh et al. [35] proposent une commande robuste et adaptative indépendante du modèle. Ils ont utilisés la théorie d'approximation universelle et le théorème de Stone-Weierstrass pour compenser les incertitudes structurées et non structurées associées aux perturbations externes et aux paramètres dynamiques du RMI. Le contrôleur interne de leur loi de commande est de type proportionnelle et le contrôleur externe est construit autour de la matrice jacobienne inverse. Les travaux de Dogan et al. [37] utilisent la technique d'apprentissage répétitive pour modéliser la pseudo-inverse de la matrice jacobienne. Le contrôle PID fait office de contrôleur interne dans leurs travaux [37]. La commande adaptative et robuste qui permet à un opérateur humain d'interagir simultanément avec un groupe de robots essaims dans un environnement réduit fut élaboré par Liu et al. [36]. Dans leurs travaux [36], le contrôle interne est un PID adaptatif et le contrôleur externe est obtenu grâce à la pseudo inverse du Jacobien.

D'autres approches de commande ont été proposés en considérant la même l'Hypothèse (H1). M. Galicki et al. [38] conçoivent une commande adaptative robuste à temps fini

dans l'espace des tâches. Cette commande assure la stabilité globale à temps fini des RMIs commandés dans l'espace des tâches. Elle est construite autour d'une commande à mode glissant terminal non singulier (contrôleur interne) et utilise l'inverse exact de la matrice jacobienne (contrôleur externe). Munoz et al. [39] proposent également une commande qui garantit la stabilité globale à temps fini des RMIs commandés dans l'espace des tâches. Cette commande permet à un RMI de suivre un contour de trajectoire à temps prédéfini dans l'espace des tâches. Le contrôleur interne de leur commande est un contrôleur à temps prédéfini et le contrôleur externe est obtenu grâce à la pseudo inverse de la matrice jacobienne. Reza et al. [40] proposent une commande adaptative dans l'espace de tâche dédiée aux RMIs à l'aide de l'expansion de la série de Fourier sans mesure de la vitesse. Cette commande [40] garantit la stabilité asymptotique globale des RMIs commandés dans l'espace des tâches.

Lorsque le contrôleur interne est inaccessible, les travaux sus-mentionnés ne peuvent plus garantir la stabilité globale des systèmes décrits dans la Figure 1.3.

Contrôleur interne connu et inaccessible : Hypothèse 2

Plusieurs travaux scientifiques ont été menés en considérant l'Hypothèse (H2). Elle stipule que le contrôleur interne est connu mais inaccessible. Dans cette hypothèse, on suppose que le fabricant donne des informations détaillées sur l'architecture du contrôleur interne embarquée mais ne permet pas à l'utilisateur du RMI d'effectuer des modifications. Afin de garantir la stabilité globale du système, l'utilisateur doit concevoir la loi de commande du contrôleur externe en tenant compte de l'architecture du contrôleur interne.

En considérant cette hypothèse, Wang et al. [11] conçoivent un contrôleur adaptatif. Ils considèrent que le contrôleur interne inaccessible est soit un PI ou un PID. Afin de garantir la stabilité globale du système et compte tenu de l'indépendance entre les contrôleurs (contrôleur dans l'espace des tâches et le contrôleur dans l'espace articulaire), ils utilisent la technique de modulation dynamique pour synchroniser les deux contrôleurs. Bing Xiao et al. [41] conçoivent une commande de suivi exponentielle de trajectoire des RMIs à paramètres dynamiques et cinématiques incertains basée sur un observateur. Cette commande garantit la stabilité globale exponentielle des RMIs à architecture de commande interne connue, mais inaccessible. Afin d'obtenir cette stabilité, ils exploitent la théorie qui stipule que si le système observateur-contrôleur est exponentiellement stable, alors les erreurs d'estimation et de suivi de trajectoire peuvent globalement converger de manière exponentielle vers leurs points d'équilibre respectifs. Le contrôleur externe qu'ils proposent

est adaptatif et approxime en ligne la pseudo inverse de la matrice Jacobienne [41]. H. Wang et al. [42] proposent une commande adaptative avec compensation des incertitude de modélisation cinématique et dynamique, qui garantit une stabilité globale et uniforme. Dans leur travaux ils considèrent que la structure du contrôleur interne est PI.

Contrôleur interne inconnu et inaccessible mais garantissant des erreurs en vitesse bornées et intégrables : Hypothèse 3

L'Hypothèse (H3) est utilisée lorsque l'architecture du contrôleur interne est inconnue et inaccessible mais pour garantir la stabilité, les erreurs en vitesse doivent être bornées et intégrables. En plus, les deux contrôleurs doivent être capable de se synchroniser [6, 43]. Dans ce contexte, pour garantir la stabilité globale, l'utilisateur conçoit la loi de commande du contrôleur externe. Cette commande fournit des signaux de référence (position/vitesse) qui permettent au contrôleur interne de suivre avec précision la trajectoire de référence.

Très peu de travaux ont été menés en considérant cette hypothèse. Néanmoins, Zhi et Guoqiang [43] ont développé une commande adaptative distribuée dans l'espace de tâche capable d'assurer le suivi coopératif de trajectoire d'un ensemble de RMIs. Pour y parvenir, ils estiment la tâche globale via un observateur et par la suite, des algorithmes coopératifs distribués sont conçus. En utilisant le théorème de stabilité de Lyapunov, ils démontrent que la commande proposée assure la stabilité asymptotique du système globale. Le problème de suivi visuel de trajectoire¹ des RMIs sans mesure de la vitesse dans l'espace-image, en tenant compte simultanément des incertitudes du modèle de la caméra et du modèle du RMI a intéressé Wang et al. [6]. Ils ont conçu un observateur de l'espace-image qui exploite les informations de vitesses de l'espace-image issues du modèle cinématique inconnu du RMI pour proposer une commande adaptative. Ils assurent eux aussi une stabilité asymptotique du système globale.

Dans cette section, nous avons effectué la revue de littérature des stratégies de commande dans l'espace des tâches qui résolvent le problème de stabilité globale des RMIs. Ces commandes s'appliquent aux RMIs évoluant en environnement déterministe. Nous allons présenter dans la section suivante les travaux pertinents sur la commande des RMIs en environnement aléatoire.

1. Obtenue grâce à la technique de commande qui consiste à commander le déplacement d'un système dynamique grâce à des informations visuelles issues d'un ou plusieurs capteurs visuels

1.3.3 Commandes des RMIs en environnement aléatoire

L'environnement aléatoire est un environnement non structuré dans lequel on ne peut pas prédire un événement, anticiper les actions du RMI, etc. Dans ce type d'environnement, seules les stratégies de commande dans l'espace des tâches peuvent être utilisées. L'utilisation des RMIs en environnement aléatoire a contraint les chercheurs à élaborer des lois de commandes capables d'effectuer le rejet des dites perturbations. Les stratégies de commande qui ont été développées pour cette problématique se sont appuyées sur les approches de modélisation des systèmes stochastiques. Dans la littérature, on distingue deux approches de modélisation. La première approche [14, 15, 44–47] les modélise par des équations différentielles stochastiques (SDE) de type Itô en considérant les perturbations aléatoires comme des bruits blancs. La deuxième approche [16, 48–51] les modélise comme des équations différentielles aléatoires (RDEs) en considérant les perturbations aléatoires comme des bruits colorés. La revue de littérature sur la commande des RMIs en environnement aléatoire sera centrée sur ces deux approches.

Première approche de modélisation des vibrations aléatoires

Plusieurs techniques de commande basées sur un modèle stochastique de type Itô ont été proposées dans la littérature. En se basant sur le principe d'équivalence de la mécanique et le mouvement relatif, Cui et al. [14] réalisent une commande de suivi de trajectoire d'un robot manipulateur rigide à articulation rotoïde de n DDL soumis aux perturbations aléatoires. Ils démontrent que les effets des vibrations aléatoires sur le robot se manifestent par l'apparition des accélérations aléatoires qui se propagent à travers tous les segments du robot. Un modèle dynamique Hamiltonien stochastique pour décrire les mouvements du manipulateur en présence des vibrations aléatoires est construit. À partir de ce modèle, ils proposent une commande Backstepping vectorielle. Su et al. [52] ont proposé une commande adaptative neuronale de suivi asymptotique de trajectoire. Pour cela, ils utilisent les propriétés de fonctions Numbness pour effectuer la mise à jour en ligne des lois de commandes virtuelles auxiliaires de la technique de Backstepping. Liu et al. [44] modélisent un RMI à n DDL à articulations flexibles soumis à des vibrations aléatoires. Ils proposent une commande adaptative basée sur la technique de commande des surfaces dynamiques.

La commande adaptative par retour d'état d'un robot non-linéaire stochastique a été proposée par Hui et al. [15]. Pour cela, la technique de la commande des surfaces dyna-

miques (DSC) est utilisée pour surmonter le problème de "l'explosion de la complexité" générée dans la procédure de conception Backstepping. Le RBF NN est utilisé pour compenser les termes non-linéaires qui excitent les perturbations stochastiques du robot qui se répètent dans la loi de commande. Wu et al. [45] ont conçu une commande Backstepping pour un système composé d'une sphère roulante et d'une voiture évoluant dans un environnement vibratoire. En utilisant la loi de Lagrange, le modèle dynamique sans perturbation est d'abord construit. Par la suite, selon le principe du mouvement relatif, les vibrations aléatoires de l'environnement sont transformées en bruits aléatoires et introduites dans l'équation de Lagrange construite. Sun et al. [46] ont proposé une fonction de Lyapunov de type tangente (BLF) pour borner les états du système en présence de perturbations aléatoires. Ils ont proposé par la suite une technique Backstepping filtrée à temps fini pour concevoir une commande adaptative floue de suivi de trajectoire [47].

Les approches de commande utilisant la modélisation de type Itô pour le rejet des perturbations aléatoires bien qu'ayant produit des résultats satisfaisants en simulation ont très peu été évalués dans des systèmes réels. En effet, les systèmes stochastiques de type Itô ont la particularité d'avoir des puissances infinies et par conséquent ne correspondent à aucun système réel. Afin de se rapprocher des systèmes réels, des travaux ont proposé des modèles sous forme d'équation différentielle aléatoire (RDE).

Deuxième approche de modélisation des vibrations aléatoires

Dans la littérature, on retrouve plusieurs lois de commandes attrayantes qui utilisent le modèle stochastique de type RDE. En exploitant certaines théories comme celle de la relativité du mouvement, Cui et al. [14] ont résolu le problème de modélisation et de commande des robots manipulateurs à articulations flexibles actionnés par des moteurs à courant continu soumis à des perturbations aléatoires. Ils démontrent que les vibrations aléatoires de l'environnement vont impacter les dynamiques mécaniques du robot par un couple stochastique et les dynamiques électriques par un bruit thermique. En utilisant la technique de séparation, ils parviennent à isoler les dynamiques du robot qui excitent les perturbations aléatoires. Ils appliquent la commande Backstepping vectorielle pour permettre au robot de réaliser le suivi de trajectoire. Mingyue et al. [49] ont proposé une commande adaptative par retour d'états pour robot manipulateur évoluant en environnement aléatoire, et qui ne nécessite pas un capteur de vitesse. Pour cela, un observateur à gain élevé est conçu pour surmonter la difficulté liée aux vitesses non mesurables en présence des bruits aléatoires. Guodong et al. [50] ont utilisé les équations de Kane pour

construire le modèle dynamique d'un robot industriel à six DLL soumis à des vibrations aléatoires. Temurtas et al. [51] ont proposé une commande prédictive neuronale généralisée pour les robots manipulateurs soumis à des perturbations aléatoires.

Il ressort de la brève revue de littérature présentée précédemment que les approches existantes de commande dans l'espace des tâches des RMIs présentent deux principales limites :

- elles peuvent poser des problèmes de stabilité globale lorsque le contrôleur interne n'est pas accessible ;
- elles nécessitent une approche adaptative spécifique pour le rejet des perturbations aussi bien aléatoires que déterministes, spécifique aux commandes en temps réel.

Comme solution à ces deux problèmes, cette thèse propose un contrôleur externe qui contient la structure du contrôleur interne susceptible de garantir la stabilité du système bouclé tout en éliminant les effets du contrôleur interne conçu par le fabricant. En outre, la thèse propose une approche adaptative de rejet des perturbations aléatoire, construite à partir d'un modèle RDE des phénomènes stochastiques, et d'un régresseur des dynamiques qui élimine les perturbations aléatoires. Pour développer les deux approches de commandes, cette thèse utilise des outils mathématiques, et il est par conséquent nécessaire de rappeler les principes de base.

1.4 Préliminaires

Plusieurs outils mathématiques sont utilisés pour élaborer une loi de commande. Dans cette section, nous présentons les outils utilisés pour la modélisation du robot et les perturbations aléatoires, ainsi que pour l'analyse de la stabilité et l'approximation des dynamiques incertaines et inconnues.

1.4.1 Modélisation des robots manipulateurs industriels

L'élaboration des lois de commande exige la description mathématique du comportement des systèmes robotiques dans l'environnement où ils évoluent : la modélisation. Plusieurs travaux se sont particulièrement intéressés à la modélisation des robots manipulateurs industriels en environnement déterministe notamment ceux de Spong [53], Siciliano [54], Khan [55], etc. Pour cela, considérons le RMI à n -DDL représenté par la Figure 1.4.

a. Modèle géométrique

Le modèle géométrique direct (MGD) exprime la pose de l'organe terminal $X \in \mathbb{R}^n$ du robot en fonction des variables articulaires $q \in \mathbb{R}^n$ dans l'espace cartésien [56]. Il est défini par (1.4)

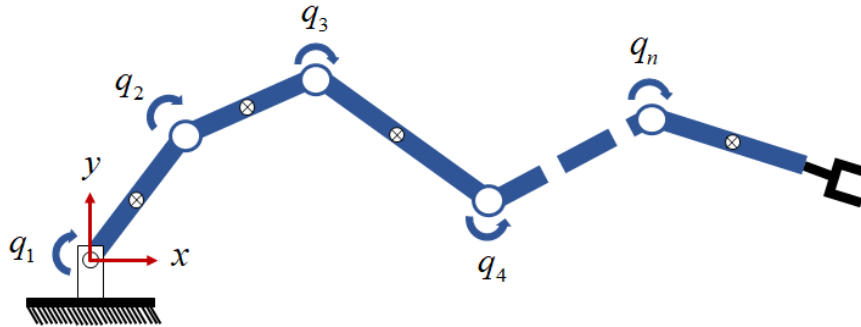


FIGURE 1.4 – Cinématique générale d'un RMI planaire à n DDL .

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

avec $f(q)$, la matrice de transformation homogène. Il est obtenu grâce aux formalismes de Denavit-Hartenberg qui décrivent la position relative de deux solides dans l'espace. Dans la littérature, on retrouve deux approches :

- La convention de Denavit-Hartenberg [57], introduite en 1955, par Jacques Denavit et Richard S. Hartenberg ;
- La convention de Denavit-Hartenberg modifiée, appelée aussi convention de Khalil-Kleinfinger [58], qui est utilisée depuis 1986.

Les axes de coordonnées de la convention de Denavit-Hartenberg modifiée (voir Figure 1.5) sont définis par les règles suivantes [59] :

- L'axe z_i du solide i est porté par l'axe de l'articulation i ;
- L'axe x_i du solide i est porté par la perpendiculaire commune aux axes z_i et z_{i-1} ;
- L'axe y_i du solide i est formé par la règle de la main droite pour compléter les coordonnées du système (x_i, y_i, z_i) .

La matrice de transformation homogène définissant l'interaction entre les solides $i-1$ et i liés respectivement aux repères R_{i-1} et R_i s'exprime en fonction de 4 paramètres :

- α_i , l'angle de rotation entre l'axe z_{i-1} et l'axe z_i autour de l'axe x_{i-1} ;

- d_i , la distance entre l'axe z_{i-1} et l'axe z_i le long de l'axe x_{i-1} ;
- θ_i , l'angle de rotation entre les axes x_{i-1} et x_i autour de l'axe z_i ;
- r_i : la distance entre les axes x_{i-1} et x_i le long de l'axe z_i .

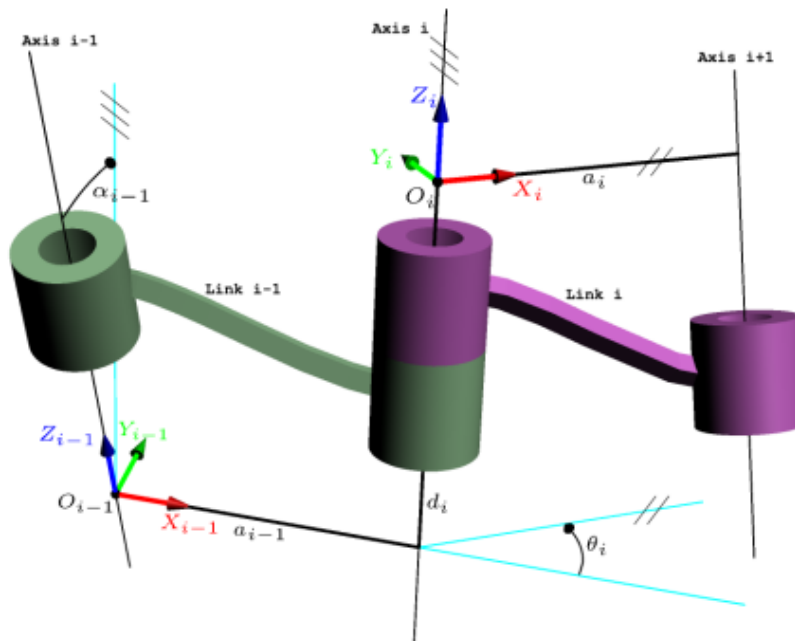


FIGURE 1.5 – Axes de coordonnées de la convention de Denavit-Hartenberg [58].

La variable q_i de l'articulation i , définissant l'orientation θ_i et la position relative r_i entre les articulations $i - 1$ et i sont définis par la relation (1.2)

$$q_i = \bar{\sigma}_i \theta_i + \sigma_i r_i, \quad (1.2)$$

avec :

- $\sigma_i = 0$: si l'articulation i est rotoïde ;
- $\sigma_i = 1$: si l'articulation i est prismatique ;
- θ_i : L'angle de rotation entre les axes x_{i-1} et x_i autour de z_i ;
- r_i : la distance entre l'axe z_i et l'axe z_{i-1} suivant l'axe x_i ;
- $\bar{\sigma}_i = 1 - \sigma_i$.

La Matrice de transformation homogène du repère R_i vers le repère R_{i-1} est donnée par

$${}^{i-1}M_i = Rot(x, \alpha_i) \times Trans(x, d_i) \times Rot(z, \theta_i) \times Trans(z, r_i), \quad (1.3)$$

avec :

- *Rot* (*axe*, *angle*), la rotation autour d'un axe i d'un angle θ ;
- *Trans* (*axe*, *distance*), la translation le long d'un axe i d'une distance d .

En multipliant les matrices (4×4) de rotation et de translation élémentaires (autour ou le long d'un seul axe), on obtient la matrice suivante

$${}^{i-1}M_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & d_i \\ \cos \alpha_i \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i & r_i \sin \alpha_i \\ \sin \alpha_i \sin \theta_i & \sin \alpha_i \cos \theta_i & \cos \alpha_i & r_i \cos \alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (1.4)$$

Le modèle géométrique inverse (MGI) [21, 60] permet de calculer les coordonnées articulaires $q \in \mathbb{R}^n$ en fonction de la pose de l'organe terminale (effecteur) $X \in \mathbb{R}^n$ du robot :

$$q = g(X). \quad (1.5)$$

Il n'existe pas une méthode analytique générale pour trouver le MGI, mais certaines méthodes sont plus ou moins adaptées à certaines chaînes cinématiques de RMI :

- la méthode de Pieper [61] permet de déterminer le MGI d'un robot à six DDLs possédant trois articulations rotoïdes d'axes concourants ou trois articulations prismatiques ;
- la méthode de Paul [62] permet de déterminer analytiquement les variables articulaires de manière successive en allant de q_1 vers q_n ; Cette méthode est très adaptée pour les RMIs
- les méthodes géométriques [63] utilisent les techniques de la géométrie pour trouver directement chaque variable articulaire.

Lorsque le MGI n'a pas de forme explicite, on utilise des procédures numériques telles que :

- la méthode de Newton RAPHSON [61] ;
- les méthodes basées sur le modèle différentiel inverse [64] ;
- les méthodes basées sur la transposée de la matrice jacobienne [65].

b. Modèle cinématique

Le modèle cinématique direct (1.6) d'un RMI décrit la relation entre les vitesses opérationnelles et les vitesses articulaires [66]

$$\dot{x} = J(q) \dot{q}, \quad (1.6)$$

avec :

- $\dot{x} \in \mathbb{R}^{\bar{r}}$, le vecteur des vitesses opérationnelles.
- $\dot{q} \in \mathbb{R}^n$, le vecteur des vitesses articulaires.
- $J(q) \in \mathbb{R}^{\bar{r} \times n}$, la matrice jacobienne.

En dérivant la matrice de transformation homogène du MGD, on obtient la matrice Jacobienne analytique définie par l'expression (1.7)

$$J_{ij} = \frac{\partial f_i(q)}{\partial q_j} \quad i = 1, \dots, m; j = 1, \dots, n, \quad (1.7)$$

avec J_{ij} , l'élément (i, j) de la matrice Jacobienne J .

Le modèle cinématique inverse (1.8) établit la relation entre les vitesses articulaires \dot{q} et les vitesses opérationnelles. Il est défini par l'expression

$$\dot{q} = J^{-1} \dot{X}. \quad (1.8)$$

Remarque : Si le déterminant de la matrice Jacobienne (J) est nulle et si elle n'est pas une matrice carrée, on ne peut calculer son inverse. Pour éviter ces situations, l'on utilise la pseudo-inverse de la matrice jacobienne (1.9) pour déterminer le MCI.

$$J^\dagger(q) = J(q)^T [J(q)J(q)^T]^{-1}. \quad (1.9)$$

L'utilisation de $J^\dagger(q)$ nous permet d'éviter les configurations dites singulières. Elles définissent un ensemble de coordonnées cartésiennes (poses) que le RMI ne peut atteindre, où les efforts subis par les actionneurs peuvent tendre vers l'infini ce qui provoquent leurs détériorations.

c. Modèle dynamique

Le modèle dynamique décrit la relation entre les couples et/ou forces appliqués aux actionneurs du RMI et les positions, vitesses, accélérations. Le modèle dynamique [55]

d'un système est décrit par la relation (1.10)

$$\tau = f(q, \dot{q}, \ddot{q}, f_e), \quad (1.10)$$

avec :

- τ , le vecteurs des couples/forces des actionnaires, selon que l'articulation est rotoïde ou prismatique ;
- q , le vecteur des positions articulaires ;
- \dot{q} , le vecteur des vitesses articulaires ;
- \ddot{q} , le vecteur des accélérations articulaires ;
- f_e , les perturbations extérieures.

Plusieurs formalismes sont utilisés pour obtenir le modèle dynamique des robots. Les plus exploités en robotique sont le formalisme de Lagrange et celui de Newton-Euler. Nous utilisons le formalisme de Lagrange dans notre étude. Ce formalisme a été appliqué dans plusieurs travaux de modélisation des RMIs notamment ceux de [53–56]. Le Lagrangien décrit le comportement d'un système dynamique en terme d'énergie [53]. Les équations de Lagrange sont généralement écrites sous la forme suivante

$$\Gamma = \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q}, \quad (1.11)$$

avec :

- $L = K - P$, le Lagrangien du système dynamique ;
- K , l'énergie cinétique ;
- P , l'énergie potentielle.

Afin d'établir le modèle dynamique des RMIs rigides, les Hypothèses 1.4.1 sont généralement posées [53, 54, 58] :

Hypothèse 1.4.1.

- *le système est idéal sans frottement ou élasticité ;*
- *les inerties des actionneurs sont négligées ;*
- *les efforts exercés par l'organe terminal sur l'environnement sont négligeables.*

Les travaux de Spong [53], de Siciliano [54], de Khalil [55, 56], etc. ont établi la forme générale du modèle dynamique (1.12) d'un RMI,

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau, \quad (1.12)$$

avec :

- $q \in \mathbb{R}^n$, le vecteur position angulaire des bras du robot ;
- $M(q) \in \mathbb{R}^{n \times n}$, la matrice d'inertie de la chaîne de corps rigides du robot ;
- $G(q) \in \mathbb{R}^n$, les couples de gravité ;
- $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$, les couples de Coriolis et centrifuges ;
- $\tau \in \mathbb{R}^n$, les couples des moteurs.

Afin de garantir la stabilité des lois de commandes, certaines propriétés doivent être appliquées au modèle dynamique des RMIs. Parmi lesquelles la Propriété 1.4.1 qui définit le bornage de la matrice d'inertie. Cette propriété, permet d'éviter les configurations singulières [53, 54].

Propriété 1.4.1. *La matrice d'inertie est définie positive, symétrique, et satisfait l'inégalité suivante :*

$$\lambda_{\min}(M(q))\|y\|^2 \leq y^T M(q)y \leq \lambda_{\max}(M(q))\|y\|^2, \quad (1.13)$$

avec $y \in \mathbb{R}^n$. $\lambda_{\min}(\cdot)$ et $\lambda_{\max}(\cdot)$ représentent respectivement la valeur propre minimale et maximale de la matrice (\cdot) [67].

Propriété 1.4.2. *La matrice d'inertie et la matrice de Coriolis satisferont la propriété suivante, connue sous le nom de symétrie oblique [67] :*

$$y^T (\dot{M}(q) - 2C(q, \dot{q})) y = 0, \forall y \in \mathbb{R}^n, \quad (1.14)$$

où $\dot{M}(q)$ est la dérivée de la matrice d'inertie.

Propriété 1.4.3. *La matrice jacobienne est linéairement paramétrable dans le sens où :*

$$J(q) \dot{\xi} = Y_J(q, \xi) \theta_J, \quad (1.15)$$

avec :

- $\theta_J \in \mathbb{R}^p$, le vecteur des paramètres cinématiques ;
- $\xi \in \mathbb{R}^n$, une entrée quelconque ;
- $Y_J(q, \xi) \in \mathbb{R}^{r \times p}$, la matrice de régression cinématique.

1.4.2 Les réseaux de neurones à fonctions de base radiale

Un RBF NN est une classe particulière des réseaux de neurones Feedforward de type PMC qui utilise des fonctions de bases radiales comme fonctions d'activation. Ils sont

largement utilisés pour approximer les fonctions, prédire des séries temporelles, classifier et contrôler les systèmes. Ils ont été formulés pour la première fois en 1988 par Broomhead et Lowe [69, 70]. Les RBFNNs sont formés de trois couches donc une couche d'entrée, une couche cachée avec une fonction d'activation dite à base radiale (RBF) non linéaire centrée autour d'un point et munie d'un rayon et d'une couche de sortie linéaire. Les valeurs des poids synaptiques de la couche d'entrée valent 1. Par conséquent, l'information inscrite sur la couche d'entrée est transmise sans distorsion vers les neurones de la couche cachée. L'apprentissage se déroule en deux phases. La première phase d'apprentissage est non supervisé et la deuxième phase est supervisé.

Une fonction de base radiale est une fonction d'activation centrée en un point déterminé par le vecteur des poids synaptiques et dont la position ainsi que la largeur sont ajustées par apprentissage. Soit ϕ une fonction symétrique autour d'un centre μ_j définie par [71] :

$$\phi_j(x) = \phi(\|x - \mu_j\|), \quad (1.16)$$

avec : $\|\cdot\|$, la norme de la fonction (*).

Un modèle ou réseau RBF est une combinaison linéaire de fonctions radiales de centres μ_j

$$y(x) = \sum_{j=1}^N w_j \phi(\|x - \mu_j\|, \sigma_j), \quad (1.17)$$

avec :

- x , le vecteur d'entrées de RBF NN ;
- $y(x)$, le vecteur de sortie du RBF NN ;
- μ_j , les centres des fonctions à base radiale ;
- σ_j , les largeurs ou rayons des fonctions ;
- w_j , les poids du réseau.

Les réseaux RBFs sont généralement entraînés à partir d'une paire de valeurs d'entrées $x(t)$ et de sorties $y(t)$ avec $t \in \mathbb{R}^n$. L'apprentissage d'un RBF est généralement dite hybride. Car elle se déroule en deux phases :

a. La première phase : Apprentissage Non supervisée.

La première phase consiste à construire le réseau en choisissant les paramètres de la couche cachée. Il s'agit donc de trouver le nombre de μ_j appropriés au problème posé. Une fois le nombre de centre déterminé, il faut déterminer leurs largeurs σ_j . Pour cela, on utilise la méthode de classification k-means.

b. La deuxième phase : Apprentissage supervisée.

Cette phase consiste à déterminer les poids de chaque neurone de sortie grâce à l'équation matricielle suivante [72] :

$$\begin{bmatrix} \phi(\|x_1 - \mu_1\|) & \cdots & \phi(\|x_1 - \mu_M\|) \\ \phi(\|x_2 - \mu_1\|) & \cdots & \phi(\|x_2 - \mu_M\|) \\ \vdots & \vdots & \vdots \\ \phi(\|x_N - \mu_1\|) & \cdots & \phi(\|x_N - \mu_M\|) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}. \quad (1.18)$$

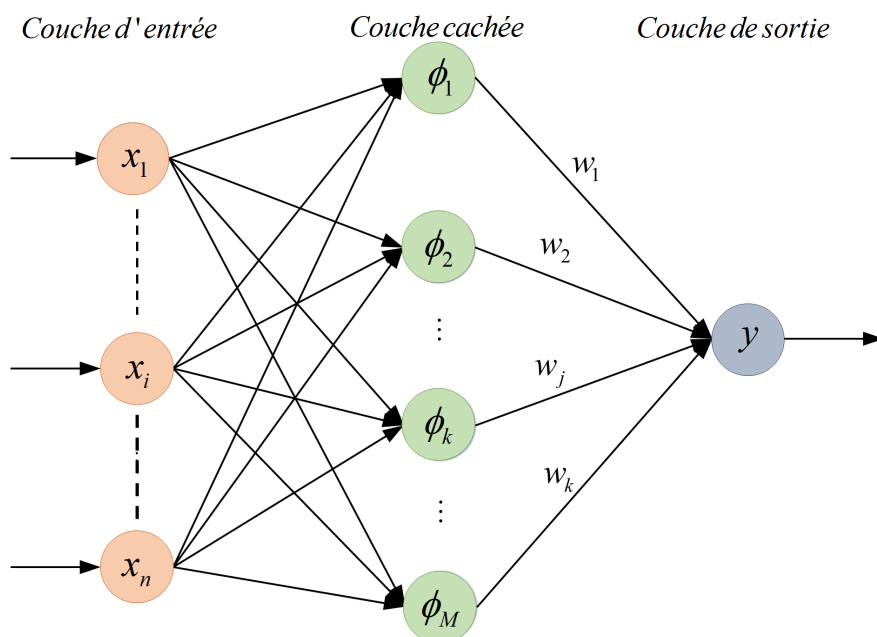


FIGURE 1.6 – Structure d'un RBF NN.

L'équation (1.18) peut alors se mettre sous la forme

$$\Phi \times W = Y, \quad (1.19)$$

Ainsi, pour tout $\phi(y)$ une fonction continue définie par un ensemble compact $\Xi \subset \mathbb{R}^n$, il existe un RBF NN tel que pour tout $\delta_\phi > 0$, avec un nombre suffisant de nœuds $\iota > 1$, tel que [73]

$$\phi(y) = W_\phi^T S_\phi(y) + \delta_\phi(y), \quad (1.20)$$

où :

- $|\delta_\phi(y)| < \epsilon_\phi$ est l'erreur d'approximation ;
- $W_\phi = [w_1, w_2, \dots, w_\iota]^T \in \mathbb{R}^\iota$ est le vecteur de poids ;

- $y \in \Xi \subset \mathbb{R}^l$ est le vecteur d'entrée ;
- $S_\phi(y) = [S_1(y), S_2(y), \dots, S_\iota(y)]^T$ est le vecteur des fonctions de base radiale ;
- $S_i(y)$ est une fonction gaussienne définie par :

$$S_i(y) = \exp \left[-\frac{(y - \mu_i)^T (y - \mu_i)}{\eta^2} \right], i = 1, 2, \dots, \iota, \quad (1.21)$$

où

- $\mu_i = [\mu_{i1}, \mu_{i2}, \dots, \mu_{il}]^T$ est le vecteur des centres gaussiens ;
- η est la largeur des fonctions gaussiennes.

Les approches d'approximation des dynamiques du controleur interne et de rejet des perturbations déterministes sont construites autour d'un RBF NN. Les Lemmes 1.4.1, 1.4.2 ci-dessous sont exploites pour cette phase d'élaboration de l'algorithme de commande.

Lemme 1.4.1. *Soit $S(\bar{a}_l) = [S_1(\bar{a}_l), S_2(\bar{a}_l), \dots, S_\iota(\bar{a}_l)]^T$ un vecteur de fonction de base d'un RBF NN avec $\bar{a}_l = [a_1, a_2, \dots, a_l]^T$. Pour tous les k et l choisis de telle sorte que $0 < k \leq l$, l'inégalité suivante tient [73] :*

$$\|S(\bar{a}_l)\|^2 \leq \|S(\bar{a}_k)\|^2. \quad (1.22)$$

Preuve : Soit :

$$\begin{aligned} \|S(\bar{a}_l)\|^2 &= \sum_{i=1}^j \left(\prod_{t=1}^l \exp \left[-\frac{1}{\eta^2} (a_t - \mu_{it})^2 \right] \right)^2, \\ \|S(\bar{a}_l)\| &\leq \sum_{i=1}^j \left(\prod_{t=1}^k \exp \left[-\frac{1}{\eta^2} (a_t - \mu_{it})^2 \right] \right)^2, \\ \|S(\bar{a}_l)\| &\leq \|S(\bar{a}_k)\|^2. \end{aligned} \quad (1.23)$$

Lemme 1.4.2. *Pour tout $y \in \mathbb{R}$ et $\varepsilon > 0$, l'inégalité suivante est valable [74]*

$$0 \leq |y| - y \times \tanh \left(\frac{y}{\varepsilon} \right) \leq 0.2785 \times \varepsilon, \quad (1.24)$$

ou encore

$$0 \leq |y| \leq y \times \tanh \left(\frac{y}{\varepsilon} \right) + 0.2785 \times \varepsilon. \quad (1.25)$$

1.4.3 La théorie de Lyapunov

La stabilité est le critère de performance principale des systèmes dynamiques. Un système est stable si, lorsqu'il est écarté de sa position d'équilibre par un signal externe, il

revient à cette position d'équilibre une fois que ce signal disparaît [58]. Parmi les outils d'analyse de la stabilité des systèmes tels que la méthode de Lagrange-Dirichlet, la méthode de Poisson ou de Poincaré, nous avons choisi la méthode de Lyapunov. En effet, la méthode de Lyapunov utilise l'énergie totale du système pour analyser la stabilité et n'a besoin ni des pôles du système, ni des solutions de l'équation d'état. Ceci s'écrit mieux à notre étude car notre système contient des dynamiques inconnues et incertaines. La notion de stabilité des systèmes est étroitement liée à celle du point d'équilibre. Plusieurs définitions encadrent ces notions. Pour cela, considérons les systèmes définis par :

$$\begin{cases} \dot{x} = f(x), \\ x(t_0) = x_0. \end{cases} \quad (1.26)$$

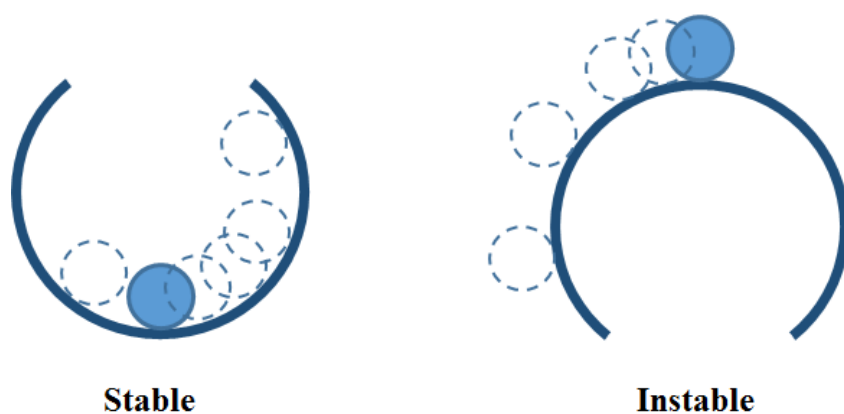


FIGURE 1.7 – Illustration de la définition intuitive de la stabilité.

Definition 1.4.1. (*Point d'équilibre*)

L'état x_e est dit état ou point d'équilibre pour le système (1.26), si $x(t_1) = x_e$ implique $x(t) = x_e$ pour tout $t \geq t_1$. Ou si l'état x_e vérifie l'équation $f(x_e) = 0$ [75] (Voir Figure 1.7).

Definition 1.4.2. (*Stabilité*)

Lorsqu'un système dynamique est légèrement perturbé de son point d'équilibre, s'il reste proche de ce point d'équilibre, on dira alors que ce point d'équilibre est stable. S'il s'éloigne et ne revient pas vers le point d'équilibre on dira qu'il est instable [76] (Voir Figure 1.7).

Alexander Lyapunov s'est inspiré de la stabilité intuitive (1.4.2) pour démontrer qu'il existe un rayon ρ d'une boule centrée sur le point d'équilibre tel qu'il est possible de trouver une sous-boule de rayon $\rho(\varepsilon)$ qui permet à ce que toute trajectoire issue de

n'importe quelle condition initiale dans cette sous-boule de rayon ρ ne quittera jamais la boule de rayon ε . D'où la définition de la stabilité au sens de Lyapunov suivante :

Definition 1.4.3. (*Stabilité au sens de Lyapunov*)

Le point d'équilibre x_e est dit stable si $\forall \varepsilon > 0$, il existe $\rho(\varepsilon) > 0$ tel que [75] (Voir Figure 1.8)

$$\|x_0 - x_e\| \leq \rho \Rightarrow \|x(t) - x_e\| \leq \varepsilon, \forall t \geq t_0. \quad (1.27)$$

Definition 1.4.4. (*Point d'équilibre attractif*)

Le point d'équilibre x_e est dit attractif [75] s'il existe $\rho > 0$ tel que

$$\|x_0 - x_e\| \leq \rho \Rightarrow \lim_{t \rightarrow \infty} \|x(t) - x_e\| = 0. \quad (1.28)$$

Le point d'équilibre x_e est dit attractif si $x(t)$, à un certain voisinage et pendant un certain temps, converge vers le point d'équilibre x_e (Voir Figure 1.8).

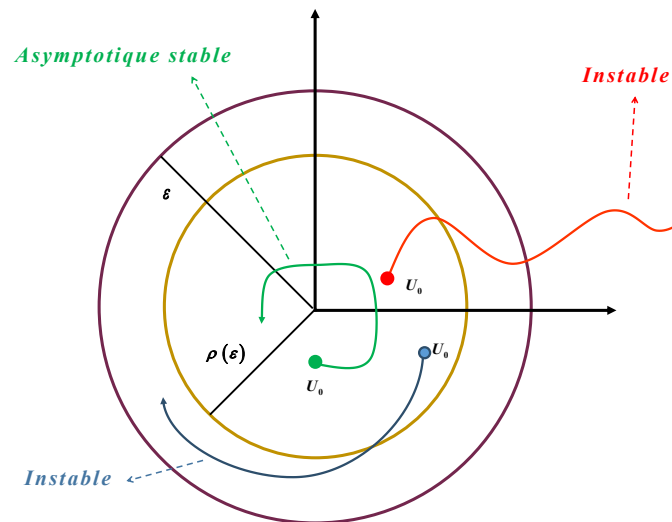


FIGURE 1.8 – Types de stabilité.

Definition 1.4.5. (*Stabilité asymptotique*)

Le point d'équilibre x_e est asymptotiquement stable s'il est stable et attractif [75].

Definition 1.4.6. (*La stabilité exponentielle*)

Le point d'équilibre x_e est exponentiellement stable s'il existe $\alpha > 0$ tel que, quelque soit $\varepsilon > 0$, il existe $\rho(\varepsilon)$ qui vérifie [75] :

$$\forall t \geq t_0, \|x_0 - x_e\| \leq \rho \Rightarrow \|x(t) - x_e\| \leq \varepsilon \|x_e - x_0\| e^{-\alpha(t-t_0)},$$

avec α le taux de stabilité.

La stabilité exponentielle indique que l'état $x(t)$ converge vers le point d'équilibre d'une manière exponentielle. La variable α caractérise la vitesse de cette convergence.

Definition 1.4.7. (*Stabilité globale*)

Le point d'équilibre est globalement (asymptotiquement ou exponentiellement) stable, si toutes les conditions de stabilité précédentes sont vérifiées pour n'importe quelle condition initiale dans \mathbb{R}^n [75].

La différence entre les définitions (1.4.5), (1.4.6) et la définition (1.4.7) réside sur le fait que les conditions initiales sont prises autour d'un point d'équilibre fixe x_e dans les définitions (1.4.5), (1.4.6) ce qui n'est pas le cas dans la définition (1.4.7) ($x_e \in \mathbb{R}^n$). La stabilité au sens de Lyapunov repose sur le principe physique qui stipule que si l'énergie totale d'un système est dissipée de manière continue, alors ce système doit rejoindre un point d'équilibre. Le théorème de Lyapunov s'entoure d'un vocabulaire particulier donc il est primordiale de présenter.

Definition 1.4.8. [76]

- Une fonction $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite définie positive si $V(x) > 0, \forall x \neq 0$ et $V(x) = 0$ lorsque $x = 0$;
- Une fonction $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite définie semi-positive si $V(x) > 0, \forall x \neq 0$ et $V(x) \geq 0$ lorsque $x = 0$;
- Une fonction $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite définie négative si $V(x) < 0, \forall x \neq 0$ et $V(x) \geq 0$ lorsque $x = 0$;
- La fonction quadratique $V(x) = x^T Q x$, ou Q une matrice $n \times n$ réelle symétrique, est dite définie positive si toutes les valeurs propres de la matrice sont strictement positives.

Nous pouvons alors définir respectivement le candidat de Lyapunov et la fonction de Lyapunov.

Definition 1.4.9. (*Candidat de Lyapunov*)

Toute fonction $V(x)$ définie positive et continue est appelée candidat de Lyapunov [76].

Definition 1.4.10. (*Fonction de Lyapunov*)

Une fonction de Lyapunov est un candidat de Lyapunov de classe \mathbb{C}^1 ayant la propriété suivante [76]

$$\dot{V}(x) \leq 0, \forall x \neq 0; \quad \dot{V}(x) = 0 \text{ si } x = 0. \quad (1.29)$$

Soit le système (1.26) de vecteur d'état $x(t)$, possédant un point d'équilibre x_e et $\delta(x_e)$ une boule de rayon $\rho(\varepsilon)$ centrée en x_e . S'il existe une fonction scalaire $V(x) : \delta(x_e) \rightarrow \mathbb{R}$ alors :

Théorème 1.4.1. (*Stabilité au sens de Lyapunov*)

x_e est un point d'équilibre localement stable dans $\delta(x_e)$ si et seulement si [75] :

- $V(x_e) = 0$;
- $V(x) > 0$ définie positive, $\forall x \in \delta(x_e)$, $x \neq x_e$;
- $\dot{V}(x_e) \leq 0$ semi-définie négative $\forall x \in \delta(x_e)$.

Théorème 1.4.2. (*Stabilité asymptotique*)

x_e est asymptotiquement stable si et seulement si [75] :

- $V(x_e) = 0$;
- $V(x) > 0$ définie positive, $\forall x \in \delta(x_e)$, $x \neq x_e$;
- $\dot{V}(x_e) < 0$ est définie négative $\forall x \in \delta(x_e)$.

L'état d'équilibre $x_e(t) = 0$ est globalement asymptotiquement stable.

Théorème 1.4.3. (*Stabilité asymptotique globale*)

$x_e = 0$ est globalement asymptotiquement stable si et seulement si [75] :

- $V(x_e) = 0$;
- $V(x) > 0$ définie positive, $\forall x \in \delta(x_e)$, $x \neq x_e$;
- $\dot{V}(x_e) < 0$ est définie négative $\forall x \in \delta(x_e)$;
- $\dot{V}(x) \rightarrow -\infty$ quand $\|x\| \rightarrow \infty$.

1.4.4 L'inégalité de Young

L'inégalité de Young est très exploitée pour exprimer un produit de deux termes par leur somme. Elle va nous permettre de séparer les termes non-linéaires incertains et les dynamiques qui excitent les vibrations aléatoires, des états du système.

Lemme 1.4.3. *Pour tous réels a, b positifs et tous réels p, q strictement positifs tels que $\frac{1}{p} + \frac{1}{q} = 1$ nous avons l'inégalité suivante :*

$$ab \leq \frac{a^p}{p} + \frac{b^q}{q}. \quad (1.30)$$

Le cas classique de cette inégalité est lorsque deux $p = 2$ et $q = 2$

$$ab \leq \frac{a^2}{2} + \frac{b^2}{2}. \quad (1.31)$$

Et nous avons de cette dernière, pour tout $\varepsilon > 0$

$$ab \leq \frac{a^2}{2\varepsilon} + \frac{\varepsilon b^2}{2}. \quad (1.32)$$

1.4.5 Processus stochastiques

Encore appelé fonction aléatoire, processus aléatoire, un processus stochastique est une variable, vecteur ou fonction aléatoire qui décrit les résultats des expériences aléatoires ou d'un phénomène aléatoire.

Les processus aléatoires ou stochastiques sont représentés par des outils mathématiques destinés à modéliser leurs évolutions dans le temps. Ils sont décrits par des familles discrètes ou continues de variables ou de vecteurs aléatoires $(X_t)_{t \in I}$. Où I est l'ensemble des temps d'observation des états du processus. Plusieurs définitions encadrent ce concept.

Definition 1.4.11. (*Espace de probabilité*)

Le triplet $(\Omega, \mathcal{F}, \mathbb{P})$ est appelé espace de probabilité. Il est dit complet si pour tout $D \in \Omega$ négligeable, D est contenu dans \mathcal{F}

avec :

- Ω : un ensemble quelconque.
- \mathcal{F} : une famille des parties de Ω ($\mathbb{P}(\Omega)$).
- \mathbb{P} : l'ensemble des variables probabilistes.
- D : un sous ensemble.

Definition 1.4.12. (*Un processus stochastique*) Un processus stochastique $X = (X_t)_{t \in I}$ est une famille de variables aléatoires, indexée par I et définie dans l'espace de probabilité $(\Omega, \mathcal{F}, \mathbb{P})$ à valeurs dans un espace mesurable (E, B) , qu'on appelle espace d'états [77].

Remarque 1.4.1.

- (a.) $\omega \in \Omega \rightarrow X_t(\omega)$ est une variable aléatoire si t est une variable fixe.
- (b.) $\omega \in \Omega \rightarrow X_t(\omega)$ est une fonction réelle appelée trajectoire du processus si ω est une variable fixe.

Remarque 1.4.2.

- (a.) Si $T \subseteq \mathbb{N}$, le processus est à temps discret.
- (b.) Si $T = [0, a]$ tel que $a > 0$, le processus est à temps continu.

Definition 1.4.13. (*Mesurabilité*)

Un processus stochastique X est mesurable si pour tout $A \in B(\mathbb{R}^d)$, l'ensemble $\{(t, \omega) ; X_t(\omega) \in A\}$ appartenant au produit $B([0, +\infty[) \otimes \mathcal{F}$; l'application (2.114) est mesurable [77, 78]

$$([0, +\infty[\times \Omega, B([0, +\infty[\times F)) \rightarrow (\mathbb{R}^d, B(\mathbb{R}^d)), \quad (1.33)$$

$$(s, \omega) \rightarrow X_s(\omega).$$

Dans les processus aléatoires, la notion de variable, vecteur et fonction sont très importantes. Mais elles dépendent toutes de la notion d'espace métrique.

Definition 1.4.14. (*Espace métrique*)

Un espace métrique (X, d) est un ensemble X muni d'une application $d : X \times X \rightarrow \mathbb{R}$ appelée distance ou métrique, possédant les trois propriétés suivantes [79] :

1. $\forall x, y \in X, d(x, y) \geq 0$ et $d(x, y) = 0 \Leftrightarrow x = y$;
2. $\forall x, y \in X, d(x, y) = d(y, x)$ (la symétrie) ;
3. $\forall x, y, z \in X, d(x, z) \leq d(x, y) + d(y, z)$ (l'inégalité triangulaire).

La notion d'espace métrique d'éléments aléatoires est très importante dans les processus aléatoires. Pour cela, soient $W := W([0, T], \mathbb{R}^m)$ toutes fonctions continues de $[0, T]$ dans \mathbb{R}^m , dotées d'une métrique uniforme définie par

$$d(x, y) := \sup_{0 \leq t \leq T} \|*\|, \quad (1.34)$$

avec $\|*\|$ la norme euclidienne sur \mathbb{R}^m .

Definition 1.4.15. (*Variable aléatoire* [79])

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilités, on appelle variable aléatoire sur $(\Omega, \mathcal{F}, \mathbb{P})$ une fonction mesurable X de Ω dans \mathbb{R} vérifiant :

$$\forall B \in \mathbb{B}(\mathbb{R}_n), X^{-1}(B) = \{\omega \in \Omega / X(\omega) \in B\} \in \mathcal{F}. \quad (1.35)$$

Definition 1.4.16. (*Vecteur aléatoire* [79])

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilités :

1. On appelle vecteur aléatoire sur $(\Omega, \mathcal{F}, \mathbb{P})$, une fonction mesurable X de Ω dans \mathbb{R}^n .
2. On appelle élément aléatoire sur $(\Omega, \mathcal{F}, \mathbb{P})$, une fonction mesurable X de Ω dans (X, d) .

avec :

- (X, d) , l'espace métrique ;
- ω , l'espace topologique ;
- \mathbb{B} , la Tribue Borélienne sur ω ;
- B , les boréliens (les éléments de la tribu Borélienne).

Un processus X est dit adapté par rapport $\{\mathcal{F}\}_{t \geq 0}$ si pour tout t , la variable aléatoire X_t est \mathcal{F}_t -mesurable [78].

Definition 1.4.17. (*Variation bornée, Variation finie [77]*)

Soit $X = (X_t, t \geq 0)$ un processus aléatoire :

1. Il est dit à variation bornée sur $[0, T]$ si et seulement si

$$\sup \sum_{i=1}^{n-1} |X_{t_{i+1}} - X_{t_i}| < K; \quad (1.36)$$

2. Il est dit à variation finie sur $[0, T]$ si et seulement si

$$\sup \sum_{i=1}^{n-1} |X_{t_{i+1}} - X_{t_i}| < \infty. \quad (1.37)$$

La meilleure approximation d'un processus aléatoire $X = (X_t, t \geq 0)$ d'un espace probabiliste $(\Omega, \mathcal{F}, \mathbb{P})$ est une variable aléatoire \mathcal{B} -mesurable appelée espérance conditionnelle.

Definition 1.4.18. (*Espérance conditionnelle [78]*)

Soit X une variable aléatoire réelle sur $(\Omega, \mathcal{F}, \mathbb{P})$ telle que $\mathbb{E}(|X|) < \infty$. On appelle espérance conditionnelle de X sachant \mathcal{B} , noté $\mathbb{E}(X|\mathcal{B})$, toute variable aléatoire Y satisfaisant les deux conditions :

1. $Y \subseteq \mathcal{B}$, c'est-à-dire Y est \mathcal{B} -mesurable ;

2. pour tout $A \in \mathcal{B}$, on a

$$\int_A X \, d\mathbb{R} = \int_A Y \, d\mathbb{R}. \quad (1.38)$$

En fait, toute variable aléatoire Y satisfaisant la définition 1.4.18 est appelée un vecteur de \mathcal{B} . Le théorème 1.4.4 suivant fixe l'existence et l'unicité de l'espérance conditionnelle.

Théorème 1.4.4. (*Existence de l'espérance conditionnelle [78]*)

1. L'espérance conditionnelle $\mathbb{E}(X|\mathcal{B})$ existe ;
2. L'espérance conditionnelle est unique dans le sens que si Y et Y' sont deux versions $\mathbb{E}(X|\mathcal{B})$ alors $Y = Y'$.

La martingale est une notion probabiliste qui permet de généraliser l'étude des processus aléatoires. Elle compte deux variantes la sous-martingale et la sur-martingale. En référence aux jeux d'hasard, cette notion permet d'établir à un processus aléatoire $\{X_n\}_n$ une espérance conditionnelle identique à X_n pour tout $m \geq n$.

Definition 1.4.19. (*Martingale [77]*)

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilités et \mathcal{F}_t une filtration de cet espace. Un processus aléatoire $(X_t)_{t \geq 0}$ intégrable, (c'est-à-dire vérifiant $\mathbb{E}(|X_t|) < \infty$) pour tout t est :

1. Une martingale si pour $s \leq t$, $\mathbb{E}(X_t | \mathcal{F}_s) = X_s$;
2. Une sur-martingale si pour $s \leq t$, $\mathbb{E}[X_t | \mathcal{F}_s] \geq X_s$;
3. Une sous-martingale si pour $s \leq t$, $\mathbb{E}[X_t | \mathcal{F}_s] \leq X_s$.

Les processus stochastiques sont de plus en plus présents dans les systèmes physiques et d'ingénierie en raison de la pluralité des environnements d'utilisation ou de travail de ces systèmes. La modélisation des systèmes physiques faisant intervenir des processus stochastiques est l'un des principaux problèmes de l'ingénierie actuelle. La littérature actuelle propose deux approches de modélisation :

- Les dynamiques du système sont modélisées par des équations différentielles stochastiques (EDS) de type Itô, dans lesquelles les perturbations aléatoires sont représentées par des bruits blancs ;
- Les dynamiques du système sont modélisées par des équations différentielles aléatoires (EDAs), dans lesquelles les perturbations aléatoires sont représentées par des bruits colorés et stationnaires.

1.4.5.1 Équations différentielles stochastiques de type Itô

La première approche de modélisation des systèmes soumis aux perturbations aléatoires [14, 15, 44–46] les décrivent par des équations différentielles de type Itô avec des perturbations aléatoires représentées par des bruits blancs. Un bruit blanc est un processus aléatoire de second ordre ayant une densité spectrale de puissance identique sur toutes les fréquences de sa bande passante. Cela est équivalent à une fonction d'auto-corrélation nulle en tout point sauf à l'origine. Ceci signifie qu'un bruit blanc est un processus aléatoire dé-corrélé. Cela se traduit par une puissance moyenne infinie donc une énergie infinie. Bien qu'inexistant dans la réalité, cette notion de bruit blanc est intéressante dans certains problèmes pratiques. C'est un excellent outil d'approximation d'une excitation quelconque. Il permet de calculer la réponse d'un système plus ou moins amorti. On distingue principalement deux types de bruits blancs. Les bruits blancs forts et les bruits blancs faibles.

- Un bruit blanc est dit fort si les variables aléatoires du processus stochastique sont centrées, indépendantes et identiquement distribuées.
- Un bruit blanc est dit faible si les variables aléatoires du processus stochastique sont centrées et de variances finies.

Les équations différentielles stochastiques d'un processus stochastique X sont définies par [78] :

$$dX_t = b(t, X_t) dt + \sigma(t, X_t) dB_t. \quad (1.39)$$

Ce qui, en terme intégrale, s'écrit

$$X_t^i = X_0^i + \int_0^t b_i(s, X_s) ds + \sum_{j=1}^m \int_0^t \sigma_{i,j}(s, X_s) dB_s^j, \quad 1 \leq i \leq d, \quad (1.40)$$

avec :

- $b(t, x) = (b_i(t, x))_{1 \leq i \leq d}$, la dérivée de l'EDS, b est un vecteur mesurable de \mathbb{R}^d ;
- $\sigma(t, x) = \sigma_{i,j}(t, x)_{\substack{0 \leq i \leq d \\ 0 \leq j \leq m}}$, est une matrice $d \times m$ mesurable appelé coefficient de diffusion de l'EDS;
- $B = (B^1 \cdots B^m)$, est un mouvement brownien standard en dimension m .

La solution d'une EDS est une fonction aléatoire. Il s'agit donc d'un processus qu'on note

$$X = (X_t)_{t \geq 0}. \quad (1.41)$$

La solution $E(b, \sigma)$ d'une EDS est :

- un espace de probabilité filtré $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t \geq 0}, \mathbb{P})$ satisfaisant les conditions habituelles;
- un $\{F_t\}_{t \geq 0}$ -un mouvement brownien $B = (B^1 \cdots B^m)$ dans \mathbb{R}^m définit sur cet espace de probabilité;
- un processus $\{F_t\}_{t \geq 0}$ -adapté continu $X = (X^1 \cdots X^d)$ à valeurs dans \mathbb{R}^d tel que l'équation (2.45) soit vérifiée, c'est à dire, coordonnée par coordonnée, pour tout $0 \leq i \leq d$.

Théorème 1.4.5. (Deuxième formule d'Itô [77])

f une fonction définie sur $\mathbb{R}_+ \times \mathbb{R}$ de classe C^1 par rapport à t , de classe C^2 par rapport à x , à dérivées bornées, on a :

$$f(t, X_t) = f(0, X_0) + \int_0^t f'_x(s, X_s) dX_s + \int_0^t f'_s(s, X_s) ds + \frac{1}{2} \int_0^t f''_{xx}(s, X_s) \sigma_s^2 ds. \quad (1.42)$$

ce qui l'on note :

$$\begin{aligned} df(t, X_t) &= \left[f'_t(t, X_t) + \frac{1}{2} f''_{xx}(t, X_t) \sigma_t^2 \right] + f'_x(t, X_t) dX_t \\ &= f'_t(t, X_t) dt + f'_x(t, X_t) dX_t + \frac{1}{2} f''_{xx}(t, X_t) dX_t. \end{aligned} \quad (1.43)$$

Théorème 1.4.6. (formule d'Itô [77])

La formule d'Itô montre que :

$$d[X_1 X_2](t) = X_1(t) dX_2(t) + X_2(t) dX_1(t) + \sigma_1(t) \sigma_2(t) dt \quad (1.44)$$

La relation (1.44) est nommée le lemme d'Itô. $\sigma_1(t) \sigma_2(t)$ correspond au crochet de X_1, X_2 noté $\langle X_1, X_2 \rangle$ est défini par :

$$\langle X_1, X_2 \rangle = \int_0^t \sigma_1(t) \sigma_2(t) dt \quad (1.45)$$

1.4.5.2 Équations différentielles aléatoires

Les bruits colorés sont les outils utilisés par la deuxième approche de modélisation des perturbations aléatoires [16, 48, 49]. Contrairement aux bruits blancs, les bruits colorés ont une densité spectrale de puissance qui varie sur l'ensemble de fréquence de la bande passante suivant la relation f^γ , où f représente la fréquence et γ un réel. Par conséquent, ils décrivent des processus aléatoires colorés caractérisés par une puissance moyenne et une énergie finie. Considérons un processus stochastique $X : I \times \Omega \rightarrow \mathbb{R}^d$ de trajectoire continue $f : I \times \mathbb{R}^d \times \Omega \rightarrow \mathbb{R}^d$.

Definition 1.4.20. (Équations différentielles ordinaires aléatoires [79])

Une équation différentielle ordinaire aléatoire (1.46) sur \mathbb{R}^d est une équation différentielle ordinaire non autonome (EDO) (1.47) pour presque tout $\omega \in \Omega$ définie par :

$$\frac{dX_t}{dt} = f(t, X_t(\omega), \omega), \quad X_t(\omega) \in \mathbb{R}^d, \quad (1.46)$$

$$x' = \frac{dx}{dt} = F_\omega(t, x), \quad x = X_t(\omega) \in \mathbb{R}^d. \quad (1.47)$$

Definition 1.4.21. Un processus stochastique X_t défini dans l'intervalle I est appelé solution de l'équation différentielle aléatoire (1.46) si pour tout $\omega \in \Omega$, $X_t(\omega)$ est une solution de l'équation différentielle ordinaire déterministe non autonome (1.47).

Généralement, les EDAs sont souvent confrontées au problème de condition initiale

$$\frac{dx_t}{dt} = F_\omega(t, x), \quad x = X_t(\omega) \in \mathbb{R}^d, \quad x(t_0) = x_0, \quad x \in \mathbb{R}^d. \quad (1.48)$$

avec $F_\omega(t, x) := f(t, X_t(\omega), \omega)$ où $f : [0, T] \times \mathbb{R}^d \times \Omega \rightarrow \mathbb{R}^d$ est une fonction continue.

La solution de (1.48) est une fonction continue et différentiable $x : [t_0, T] \rightarrow \mathbb{R}^d$ avec

$x(t_0) = x_0$ tel que pour tout $t \in [t_0, T]$ on a

$$\frac{d}{dt}x(t) = F_\omega(t, x)$$

L'intégration de l'équation différentielle aléatoire (1.48) donne l'équation suivante :

$$x(t) = x_0 + \int_0^t F_\omega(s, x) ds, \quad t \in [t_0, T]. \quad (1.49)$$

Definition 1.4.22. *La solution de (1.48) est dite unique sur I si pour deux processus arbitraires X_t et Y_t solutions par chemin dit pathwise du problème (2.8) alors $X_t = Y_t$, pour tout $(t, \omega) \in I \times \Omega^*$, avec $\Omega^* \subset \Omega$, et $P(\Omega^*) = 1$.*

La solution d'une EDA est un processus stochastique défini sur un intervalle I et est indépendant de ω .

1.5 Conclusion

Dans ce chapitre, nous avons présenté les travaux existants les plus pertinents sur la commande dans l'espace articulaire et des tâches. Ensuite nous avons présenté les travaux sur la commande des RMIs soumis à des perturbations déterministe et aléatoires. Enfin nous avons donné quelques les préliminaires nécessaires à nos travaux. Mais au préalable, une revue de littérature sur l'évolution des supports de commande a été effectuée afin de mieux définir le cadre de nos travaux. En ce qui concerne les stratégies de commande dans l'espace de tâche des RMIs, les lois de commande proposées dans la littérature utilisent toutes des hypothèses simplificatrices. Par conséquent, ces lois ne peuvent pas garantir la stabilité globale des systèmes de commande dans l'espace des tâches si l'on fait abstraction de ces hypothèses. Nous constatons également que les travaux de recherche menés sur la commande des RMIs en environnement aléatoire présentent de mauvaises performances lorsque les dynamiques du robot sont inconnues ou incertaines. En effet, ils utilisent les techniques de séparation pour isoler les dynamiques du robot des perturbations aléatoires. Cependant, on remarque que dans toutes les approches existantes, les dynamiques aléatoires dépendent des paramètres du robot. Dans ce contexte, ces lois de commande ne peuvent pas être adaptatives. Dans le chapitre suivant, nous proposons des stratégies de commande stables pour les robots manipulateurs industriels à architecture de commande interne inconnue et inaccessible, évoluant dans un environnement déterministe ou aléatoire.

Méthodes et Outils

2.1 Introduction

La commande des RMIs à architecture de commande interne inconnue et inaccessible en présence des vibrations aléatoires soulève principalement le problème du rejet des perturbations aléatoires résultant de ces vibrations. Pour y faire face, il faut résoudre au préalable le problème lié à l'inaccessibilité du contrôleur interne afin de garantir la stabilité globale du système bouclé. Dans la littérature, ces deux problématiques sont traitées séparément. Par conséquent, il est nécessaire de proposer une approche de commande qui traite simultanément ces deux problématiques. Les solutions existantes pour le rejet des perturbations aléatoires sont par ailleurs non adaptatives donc sensibles aux incertitudes de modélisation.

Dans la première partie de ce chapitre, nous considérons que l'environnement du robot manipulateur est déterministe et nous proposons une commande capable de modifier les dynamiques du contrôleur interne afin de garantir la stabilité globale du système bouclé. Une fois cette problématique traitée, nous abordons celle liée au rejet des perturbations aléatoires dans la deuxième partie du chapitre. Nous considérons alors que l'environnement est aléatoire et concevons une commande adaptative capable d'éliminer les dynamiques d'excitations aléatoires qui représentent l'impact des vibrations aléatoires issues de l'environnement sur le robot. Dans la troisième partie du chapitre, nous présenterons les plates-formes utilisées.

2.2 Commande dans l'espace de tâches des RMIs à architecture de commande interne inconnue et inaccessible évoluant en environnement déterministe

Dans cette section, nous concevons une commande adaptative indirecte robuste qui tient compte des incertitudes des paramètres dynamiques et des erreurs de modélisation d'un RMI dont l'architecture de commande interne est inconnue et inaccessible. À cette fin, nous utilisons un RBFNN pour approximer les paramètres incertains du robot et les dynamiques inconnues du contrôleur interne. À partir de là, nous élaborons une loi de commande externe capable d'éliminer les effets des dynamiques inconnues du contrôleur interne et de rejeter les perturbations de nature déterministes. Un terme de contrainte est introduit pour permettre à la commande de converger rapidement vers les performances prescrites par l'utilisateur.

2.2.1 Présentation de l'approche de commande

La stratégie de commande dans l'espace de tâche telle que décrite par la Figure 2.1 est confrontée à un problème de stabilité globale. En effet, plusieurs facteurs tels que les incertitudes paramétriques du robot, le bruit des capteurs, les retards de communication entre les deux boucles perturbent la stabilité. De plus, les contrôleurs internes et externes sont conçus indépendamment donc leur association ne garantit pas la stabilité du système bouclé ainsi formé (Figure 2.1) même si chacun d'eux est stable indépendamment. Pour résoudre ce problème, nous exploitons le fait que la plupart des contrôleurs internes possèdent un terme dérivé ($\dot{q} - \dot{q}_c$) permettant de garantir la stabilité et la précision, avec \dot{q}_c les commandes de vitesses articulaires conçues par l'utilisateur (sortie du contrôleur externe) et \dot{q} le vecteur des vitesses articulaires du robot. De plus, bien que le contrôleur interne soit inconnu et fermé, l'on sait que la plupart des contrôleurs internes des RMIs sont des contrôleurs PI ou PID avec ou sans compensation de gravité et de friction [6,9,11].

Dans le cas où c'est un contrôleur PI avec compensation de gravité, son expression est

$$u = g(q) - K_p(q - q_c) - K_d(\dot{q} - \dot{q}_c), \quad (2.1)$$

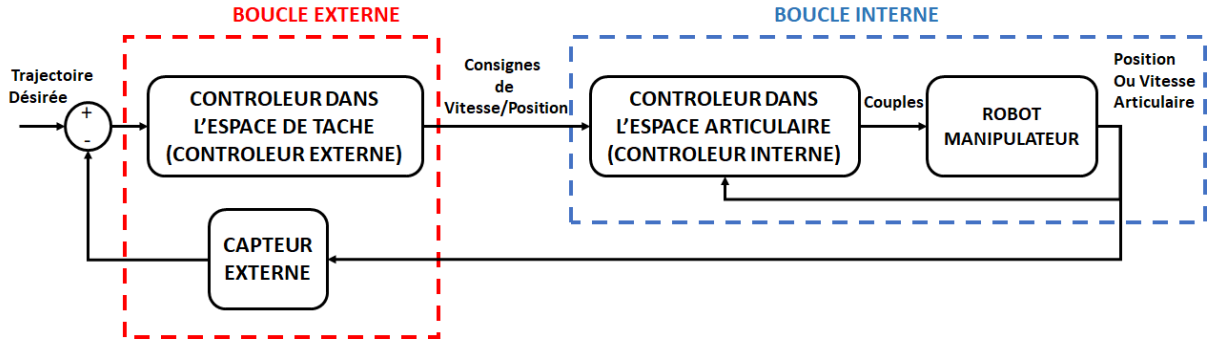


FIGURE 2.1 – Architecture générale de commande dans l'espace des tâches des RMIs.

Dans le cas où c'est un contrôleur PID avec compensation de gravité son expression est

$$u = g(q) - K_p(q - q_c) - K_d(\dot{q} - \dot{q}_c) - K_i \int_0^t (q(\varsigma) - q_c(\varsigma)) d\varsigma, \quad (2.2)$$

avec :

- K_p , le paramètre proportionnel ;
- K_d , le paramètre dérivé ;
- K_i , le paramètre intégral ;
- $g(t)$, le compensateur de gravité.

A partir de (2.1) et (2.2), nous déduisons que le contrôleur interne est une combinaison du terme dérivé et d'une fonction inconnue qui dépend des signaux q , \dot{q} , q_c , $\int_0^t q(\varsigma)d\varsigma$, et $\int_0^t q_c(\varsigma)d\varsigma$. Par conséquent on peut admettre que la forme générale d'un contrôleur interne est :

$$u = -K_d(\dot{q} - \dot{q}_c) + \psi \left(q, \dot{q}, q_c, \int_0^t q(\varsigma)d\varsigma, \int_0^t q_c(\varsigma)d\varsigma \right), \quad (2.3)$$

avec :

- $u \in \mathbb{R}^n$, le vecteur des couples de commande des actionneurs ;
- $\psi \left(q, \dot{q}, q_c, \int_0^t q(\varsigma)d\varsigma, \int_0^t q_c(\varsigma)d\varsigma \right) \in \mathbb{R}^n$, le vecteur inconnu représentant les informations inconnues du contrôleur interne et les incertitudes de modélisation ;
- $K_d \in \mathbb{R}^{n \times n}$, la matrice de gain inconnue du contrôleur interne.

Il est bien connu dans la théorie de la commande que les gains du contrôleur doivent toujours être positifs pour assurer la stabilité du système bouclé [53]. Ainsi, le signe de K_d est la seule information connue de K_d . Par conséquent, contrairement à l'approche de commande développée par Wang [11] qui suppose que K_d est une matrice diagonale, nous

utilisons un RBFNN pour estimer K_d^{-1} et un autre RBFNN est utilisé pour approximer la fonction $\psi(*)$.

L'idée principale de notre loi de commande est de concevoir une commande externe \dot{q}_c qui approxime les dynamiques inconnus du contrôleur interne $\psi(*)$ et K_d^{-1} , annule leurs effets et impose un contrôleur interne évolué qui assure le suivi de trajectoire et garantit la stabilité globale du système [Capteur externe - Contrôleur externe - Contrôleur interne - Robot manipulateur]. En outre, l'une des principales contraintes lors de la conception d'un algorithme de commande des robots manipulateurs industriels est la réduction du temps de calcul pour des applications à temps réels. Pour satisfaire cette exigence, nous exploitons l'une des propriétés inhérentes du RBFNN qui permet de réduire le nombre des lois de mise à jour des paramètres adaptatifs. Cette approche introduit des erreurs d'approximation, ce qui entraîne une perte de précision pendant la phase transitoire. Par conséquent, nous introduisons un terme qui contraint le système à converger rapidement vers les performances spécifiées par l'utilisateur.

L'approche de commande se résume en deux étapes principales :

- Concevoir le vecteur des vitesses articulaires de référence

$$\dot{q}_r = h_1(q, x, x_{des}), \quad (2.4)$$

capable de compenser les erreurs de suivi de trajectoire dans l'espace des tâches. Par la suite, définir une surface glissante $s = \dot{q} - \dot{q}_r$ et concevoir un modèle d'état du système de commande du RMI prenant en compte les incertitudes de modélisation dynamique, cinématique et la structure inconnue du contrôleur interne.

- A partir du modèle d'état du système de commande obtenu, concevoir un contrôleur externe ayant la forme

$$\dot{q}_c = \dot{q}_r - \hat{W}h_2(q, \dot{q}, q_r, \dots), \quad (2.5)$$

où \hat{W} est l'approximation de K_d^{-1} et h_2 est une fonction non-linéaire qui contient les dynamiques inconnues du contrôleur interne.

Il est aisé de constater que si l'on substitue (2.5) dans (2.3) la structure du contrôleur interne est modifiée comme suite

$$u = -K_d(\dot{q} - \dot{q}_r) - K_d\hat{W}h_2(q, \dot{q}, q_r, \dots) + \psi(q, \dot{q}, \dots). \quad (2.6)$$

La principale difficulté de notre approche est la conception de \hat{W} , $h_1(q, x, x_{des})$, et $h_2(q, \dot{q}, q_r, \dots)$.

2.2.2 Présentation de l'algorithme de commande

Afin de faire face aux incertitudes de modélisation du RMI et de résoudre le problème d'inaccessibilité du contrôleur interne, nous proposons une stratégie de commande adaptative indirecte. La partie adaptative de la loi de commande permet d'annuler les effets des dynamiques inconnues du contrôleur interne et d'assurer le rejet des incertitudes cinématiques et dynamiques du système. Cependant, le contrôleur adaptatif introduit des erreurs d'approximation, ce qui entraîne des erreurs de suivis élevées pendant la phase de transition. Pour résoudre ce problème, nous exploitons les propriétés de la fonction exponentielle afin de concevoir un vecteur de contrainte qui oblige le RMIACIII à respecter les performances définies par l'utilisateur.

Considérons l'erreur de suivi dans l'espace des tâches définie par

$$\tilde{x}(t) = x(t) - x_{des}(t). \quad (2.7)$$

Le vecteur de contraintes qui représente les performances prescrites par l'utilisateur est $\varpi \in \mathbb{R}^{\bar{r}}$. Définissons $\zeta(x) \in \mathbb{R}^{\bar{r}}$, le vecteur de contrainte de performances qui garantit le respect des contraintes de l'utilisateur.

$$\zeta(x) = k \times \text{diag}([\exp(\tilde{x}_1^2 - \varpi_1^2), \exp(\tilde{x}_2^2 - \varpi_2^2), \dots, \exp(\tilde{x}_{\bar{r}}^2 - \varpi_{\bar{r}}^2)]) \tilde{x}, \quad (2.8)$$

avec :

- \tilde{x}_i , le $i^{\text{ème}}$ élément de l'erreur ;
- ϖ_i , le $i^{\text{ème}}$ élément du vecteur de contrainte $\varpi \in \mathbb{R}^{\bar{r}}$;
- k , une constante positive ;
- $\exp(*)$, la fonction exponentielle ;
- $\text{diag}(*)$, la matrice diagonale $\bar{r} \times \bar{r}$.

Lorsque l'erreur de suivi dans l'espace des tâches \tilde{x}_i (2.7) dépasse les performances prescrites désirées comprises dans la plage $[-\varpi_i, \varpi_i]$, $\zeta(x)$ augmente de manière exponentielle. Cela va réduire la vitesse de référence de l'effecteur (2.10) et par conséquent la vitesse de référence articulaire (2.9). Cela provoque l'augmentation de la surface glissante (2.11). Cette augmentation permet le rejet des perturbations déterministes et rend la

commande robuste pendant la phase transitoire. Cependant, l'erreur $\tilde{x}_i \geq 1$ m est inimaginable en ingénierie. Il est donc préférable de choisir le critère de performance $\varpi_i \in]0, 1]$.

La vitesse articulaire de référence (2.9) est obtenue grâce au MCI (1.8) du RMI décrit dans la Sous-Section 1.4.1.2. Pour tenir compte des incertitudes cinématiques, nous approximations en ligne la matrice Jacobienne. Afin d'éviter les configurations singulières, nous utilisons l'estimé de la pseudo-inverse de la matrice Jacobienne.

$$\dot{q}_r = \hat{J}^\dagger(q)\dot{x}_r, \quad (2.9)$$

avec

- $\hat{J}^\dagger(q) = \hat{J}(q)^T [\hat{J}(q)\hat{J}(q)^T]^{-1}$; l'estimé de la pseudo-inverse de la matrice jacobienne ;
- $\hat{J}(q)$, l'estimé de la matrice jacobienne $J(q)$;
- \dot{x}_r , la vitesse de référence de l'effecteur définie comme suit

$$\dot{x}_r = \dot{x}_{des} - \alpha\zeta(x), \quad (2.10)$$

où $\alpha \in \mathbb{R}^{\bar{r} \times \bar{r}}$ est une matrice diagonale définie positive.

Afin de garantir la robustesse, la précision, la stabilité et la rapidité de notre loi de commande, nous définissons la surface glissante suivante

$$s = \dot{q} - \dot{q}_r. \quad (2.11)$$

La dérivé de l'erreur de suivie est définie par

$$\dot{\tilde{x}} = \dot{x} - \dot{x}_{des}. \quad (2.12)$$

En utilisant l'expression du MCD (1.6) et la Propriété (1.4.3) qui stipule que $J(q)\dot{q} = Y_J(q, \dot{q})\theta_J$, $\dot{\tilde{x}}$ devient

$$\begin{aligned} \dot{\tilde{x}} &= J(q)\dot{q} - \dot{x}_{des}, \\ &= Y_J(q, \dot{q})\theta_J - \dot{x}_{des}. \end{aligned} \quad (2.13)$$

Du fait des différentes incertitudes, il est difficile d'obtenir la valeur exacte de θ_J . Par cela, nous allons l'exprimer avec $\hat{\theta}_J$ et $\tilde{\theta}_J$. Ainsi l'on a

$$\theta_J = \tilde{\theta}_J + \hat{\theta}_J. \quad (2.14)$$

L'introduction de (2.14) dans (2.13) nous donne

$$\begin{aligned}
 \dot{\tilde{x}} &= Y_J(q, \dot{q}) (\tilde{\theta}_J + \hat{\theta}_J) - \dot{x}_{des}, \\
 &= Y_J(q, \dot{q}) \tilde{\theta}_J + Y_J(q, \dot{q}) \hat{\theta}_J - \dot{x}_{des}, \\
 &= Y_J(q, \dot{q}) \tilde{\theta}_J + \hat{J}(q) \dot{q} - \dot{x}_{des}.
 \end{aligned} \tag{2.15}$$

En exploitant le (2.11), le (2.15) devient :

$$\begin{aligned}
 \dot{\tilde{x}} &= Y_J(q, \dot{q}) \tilde{\theta}_J + \hat{J}(q) (s + \dot{q}_r) - \dot{x}_{des}, \\
 &= \hat{J}(q) s + \hat{J}(q) \dot{q}_r + Y_J(q, \dot{q}) \tilde{\theta}_J - \dot{x}_{des}.
 \end{aligned} \tag{2.16}$$

A partir du (2.11) et (2.10), l'équation (2.16) devient

$$\begin{aligned}
 \dot{\tilde{x}} &= Y_J(q, \dot{q}) \tilde{\theta}_J + \hat{J}(q) s + \dot{x}_r - \dot{x}_{des}, \\
 &= Y_J(q, \dot{q}) \tilde{\theta}_J + \hat{J}(q) s - \alpha \zeta(x).
 \end{aligned} \tag{2.17}$$

Le résultat final de la dérivé de l'erreur \tilde{x} est :

$$\dot{\tilde{x}} = Y_J(q, \dot{q}) \tilde{\theta}_J + \hat{J}(q) s - \alpha \zeta(x), \tag{2.18}$$

Exprimons la dérivé de la surface glissante (2.19)

$$\begin{aligned}
 s &= \dot{q} - \dot{q}_r, \\
 \dot{s} &= \ddot{q} - \ddot{q}_r \\
 M(q) \dot{s} &= M(q) \ddot{q} - M(q) \ddot{q}_r.
 \end{aligned} \tag{2.19}$$

D'après l'expression (1.12) décrivant le modèle dynamique du RMIACIII, nous avons $M(q) \ddot{q} = u - C(q, \dot{q}) \dot{q} - g(q)$. Ainsi l'expression (2.19) devient

$$M(q) \dot{s} = u - C(q, \dot{q}) \dot{q} - g(q) - M(q) \ddot{q}_r. \tag{2.20}$$

En considérant la forme générale du contrôleur interne des RMIACIII définie dans (2.3) et l'expression de la surface glissante (2.11), nous obtenons la représentation d'état

du système commande suivante :

$$\begin{aligned}
 M(q) \dot{s} &= u - C(q, \dot{q}) (\dot{q}_r + s) - g(q) - M(q) \ddot{q}_r, \\
 &= u - C(q, \dot{q}) \dot{q}_r - C(q, \dot{q}) s - g(q) - M(q) \ddot{q}_r, \\
 &= -K_d(\dot{q} - \dot{q}_c) - C(q, \dot{q}) s + \psi(*) - C(q, \dot{q}) \dot{q}_r - g(q) - M(q) \ddot{q}_r, \\
 &= -K_d(\dot{q} - \dot{q}_c) - C(q, \dot{q}) s + \chi(Z),
 \end{aligned} \tag{2.21}$$

avec :

- $\chi(Z) = \psi(*) - C(q, \dot{q}) \dot{q}_r - g(q) - M(q) \ddot{q}_r$, une fonction inconnue ;
- $Z = [q, \dot{q}, q_c, \dot{q}_r, \ddot{q}_r, \int_0^t q(\varsigma) d\varsigma, \int_0^t q_c(\varsigma) d\varsigma]^T$, le vecteur des signaux non-mesurables.

Remarque :

$\chi(Z)$ contient les dynamiques inconnus du contrôleur interne ψ et les paramètres des dynamiques incertaines tels que la matrice d'inertie, la matrice de Coriolis et le couple de gravité.

La représentation d'état du système de commande des RMIs (2.21) contient la fonction inconnue $\chi(Z)$, qui prend en entrée les signaux non-mesurables contenus dans le vecteur Z . Pour garantir la stabilité globale du système, nous proposons le contrôleur externe suivant :

$$\dot{q}_c = \dot{q}_r - \hat{W}v, \tag{2.22}$$

où :

$$v = \hat{\theta}_d \bar{U} + \hat{J}^T(q) \tilde{x}, \tag{2.23}$$

avec :

- $\hat{W} \in \mathbb{R}^{n \times n}$, l'estimé de K_d^{-1} ;
- $\hat{\theta}_d$, l'estimé du paramètre adaptatif θ_d utilisé pour approximer $\chi(Z)$;
- \bar{U} est une fonction robuste, définie comme par

$$\bar{U} = \left[U_1 \tanh\left(\frac{U_1 s_1}{\bar{\varepsilon}_1}\right), \dots, U_n \tanh\left(\frac{U_n s_n}{\bar{\varepsilon}_n}\right) \right]^T \in \mathbb{R}^n, \tag{2.24}$$

où :

- $\tanh(*)$, une fonction tangente hyperbolique ;
- $\bar{\varepsilon}_i > 0$, un paramètre de conception ;

— s_i , le $i^{\text{ième}}$ élément de s , $i = \{1, 2, \dots, n\}$.

La fonction U_i est définie comme suit

$$U_i = \|S_{\chi_i}(\bar{Z})\| + 1. \quad (2.25)$$

avec $\bar{Z} = [q, \dot{q}_r]^T$.

Les lois de mise à jour des paramètres adaptatifs sont :

$$\dot{\hat{\theta}}_d = \Gamma_d \times S^T(\bar{Z})\bar{U} - \gamma_d \hat{\theta}_d, \quad (2.26)$$

$$\dot{\hat{W}}_i = \Gamma_{W_i} v_i s - \gamma_{W_i} \hat{W}_i, \quad (2.27)$$

$$\dot{\hat{\theta}}_J = \Gamma_J Y_J^T(q, \dot{q}) \tilde{x} - \gamma_J \hat{\theta}_J. \quad (2.28)$$

où :

- \hat{W}_i représente la $i^{\text{ième}}$ colonne de la matrice \hat{W} ;
- v_i le $i^{\text{ième}}$ élément de v .

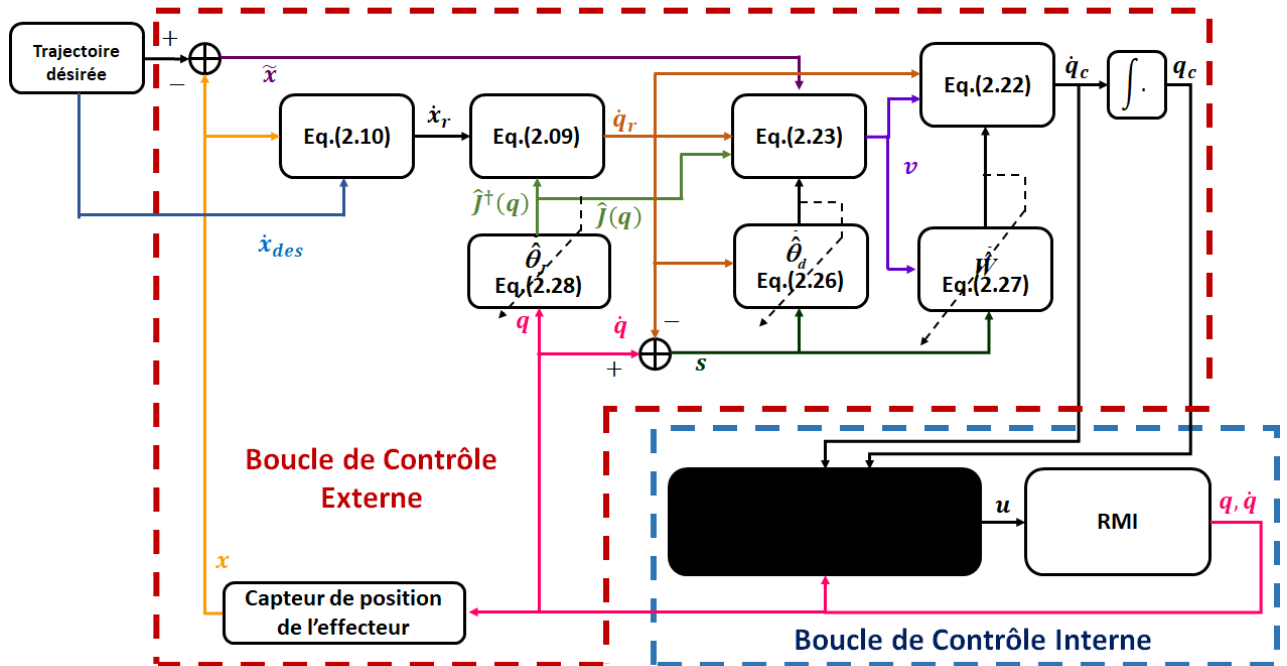


FIGURE 2.2 – Organigramme de la commande dans l'espace des tâches d'un RMI à architecture de commande interne inconnue et inaccessible

Remarque

La définition de la loi de commande (2.22) et des lois d'adaptations (2.26-2.28) sera justifiée pendant l'analyse de la Stabilité. La fonction optimale h_2 présentée dans la Sous-section 2.2.1 est v et h_1 est $\hat{J}^\dagger(q) \dot{x}_r$. Les paramètres de conception suivants sont des constantes positives : $\Gamma_d, \gamma_d, \Gamma_J, \gamma_J, \Gamma_{W_i}$, et γ_{W_i} , $i = \{1, 2, \dots, n\}$.

L'organigramme représenté dans la Figure 2.2 résume l'approche commande proposée. Dans la sous-section suivante, nous allons démontrer que l'approche de commande proposée définie par les lois de commandes (2.22), (2.23) et les lois d'adaptation (2.26); (2.27) et (2.28) garantissent la stabilité globale du système bouclé (Figure 2.2).

2.2.3 Analyse de la stabilité du système bouclé

L'utilisation des RBF NNs pour approximer les dynamiques inconnues du contrôleur interne ainsi que les paramètres incertains n'est valable que dans un certain ensemble compact nommé « domaine d'approximation ». Il est difficile d'identifier ce domaine d'approximation avec précision. Une fois définies, les entrées du RBF NN doivent varier dans ce domaine afin de garantir la stabilité du système (voir Figure 2.2). Une telle stabilité est appelée semi-globale uniformément bornée (SGUUB). Ainsi, le système bouclé décrit par la Figure 2.2 est stable si et seulement si tous les signaux mis en jeu sont uniformément bornés autour du domaine d'approximation, indépendamment de l'influence et de la nature des signaux externes. Pour que cela soit effectif, il faut correctement choisir les paramètres de conception. Le théorème 2.2.1 décrit les conditions que doivent remplir ces paramètres afin que la loi de commande (2.22), ainsi que les lois d'adaptation (2.26), (2.27) et (2.28), puissent garantir la SGUUB du système global (Figure 2.2). L'analyse de stabilité consiste donc à démontrer ce théorème.

Théorème 2.2.1. *Les paramètres de conception $\bar{\varepsilon}_i, \gamma_d, \gamma_J$ et γ_{W_i} doivent être suffisamment petits et $k, \alpha_i, \Gamma_d, \Gamma_J$, et Γ_{W_i} suffisamment grands pour que la commande de vitesse articulaire définie par l'Équation (2.22), et les lois d'adaptations (2.26), (2.27) et (2.28), assurent le bornage de tous les signaux du système bouclé et contraignent l'erreur de suivi de trajectoire $\tilde{x}(t)$ à converger vers zéro. C'est-à-dire, $\tilde{x}(t) \rightarrow 0$ lorsque $t \rightarrow \infty$.*

Preuve :

La loi de commande (2.22) et les lois d'adaptation (2.26), (2.27) et (2.28) font face à cinq erreurs. Trois d'entre elles sont liées aux paramètres adaptatifs, une est due à une

erreur de suivi et une autre à une erreur de glissement. C'est pour cette raison, que nous considérons le candidat de Lyapunov $V(t) \in \mathbb{R}$ suivant

$$V(t) = \frac{1}{2}s^T M(q)s + \frac{1}{2}\tilde{x}^T \tilde{x} + \frac{1}{2\Gamma_d}\tilde{\theta}_d^2 + \frac{1}{2\Gamma_J}\tilde{\theta}_J^T \tilde{\theta}_J + \frac{1}{2} \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \tilde{W}_i, \quad (2.29)$$

où :

— \tilde{W}_i est la $i^{\text{ème}}$ colonne de la matrice $\tilde{W} = K_d^{-1} - \hat{W}$;

— $\tilde{\theta}_d = \theta_d - \hat{\theta}_d$;

— $\tilde{\theta}_J = \theta_J - \hat{\theta}_J$.

La dérivée du candidat de Lyapunov par rapport au temps donne

$$\dot{V}(t) = \frac{1}{2}s^T \dot{M}(q)s + s^T M(q)\dot{s} + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d}\tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J}\tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i. \quad (2.30)$$

Remplaçons $M(q)\dot{s}$ défini dans (2.21) dans (2.30)

$$\begin{aligned} \dot{V}(t) &= \frac{1}{2}s^T \dot{M}(q)s + s^T [-K_d(\dot{q} - \dot{q}_c) - C(q, \dot{q})s + \chi(Z)] \\ &\quad + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d}\tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J}\tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i, \end{aligned} \quad (2.31)$$

$$\begin{aligned} &= \frac{1}{2}s^T [\dot{M}(q) - 2C(q, \dot{q})]s + s^T [-K_d(\dot{q} - \dot{q}_c) + \chi(Z)] \\ &\quad + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d}\tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J}\tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i. \end{aligned}$$

Or d'après la Propriété 1.4.2, nous obtenons

$$\dot{V}(t) = s^T [-K_d(\dot{q} - \dot{q}_c) + \chi(Z)] + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d}\tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J}\tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i. \quad (2.32)$$

De l'équation $s = \dot{q} - \dot{q}_r$ exprimons, \dot{q}_r

$$\dot{q} = \dot{q}_r + s. \quad (2.33)$$

L'équation (2.32) devient :

$$\begin{aligned}
 \dot{V}(t) &= s^T [-K_d(\dot{q}_r + s - \dot{q}_c) + \chi(Z)] + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i, \\
 &= s^T [-K_d(\dot{q}_r - \dot{q}_c) + \chi(Z)] + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i - s^T K_d s, \\
 &= -s^T K_d (\dot{q}_r - \dot{q}_c) + s^T \chi(Z) + \tilde{x}^T \dot{\tilde{x}} + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i - s^T K_d s.
 \end{aligned} \tag{2.34}$$

Cependant, $\chi(Z)$ est un vecteur inconnu qui comporte les dynamiques inconnues du contrôleur interne et les états du système non-mesurables. Nous allons par conséquent approximer chacune de ses composantes $\chi_i(Z)$, avec $i = \{1, 2, \dots, n\}$, par un modèle RBF NN. Ainsi nous avons

$$\chi_i(Z) = W_{\chi_i}^T S_{\chi_i}(Z) + \delta_{\chi_i}(Z). \tag{2.35}$$

Le produit vectoriel $s^T \chi(Z)$ peut être réécrit sous la forme

$$s^T \chi(Z) = \sum_{i=1}^n s_i (W_{\chi_i}^T S_{\chi_i}(Z) + \delta_{\chi_i}(Z)) \tag{2.36}$$

En appliquant le théorème des inégalités, nous avons

$$\begin{aligned}
 s^T \chi(Z) &\leq \sum_{i=1}^n |s_i| (\|W_{\chi_i}\| \|S_{\chi_i}(Z)\| + |\delta_{\chi_i}(Z)|), \\
 &\leq \sum_{i=1}^n |s_i| (\|W_{\chi_i}\| \|S_{\chi_i}(Z)\| + \epsilon_{\chi_i}).
 \end{aligned} \tag{2.37}$$

Définissons la constante θ_d comme suit :

$$\theta_d = \max\{\epsilon_{\chi_1}, \epsilon_{\chi_2}, \dots, \epsilon_{\chi_n}, \|W_{\chi_1}\|, \|W_{\chi_2}\|, \dots, \|W_{\chi_n}\|\}. \tag{2.38}$$

Le vecteur Z contient les signaux non mesurables ou difficilement mesurables $q, \dot{q}, \dot{q}_c, x, \dot{x}, \int_0^t q(\phi) d\phi$.

Pour éviter d'utiliser ses signaux, nous exploitons le Lemme 1.4.1. Ainsi, l'inégalité suivante est valable :

$$\|S_{\chi_i}(Z)\| \leq \|S_{\chi_i}(\bar{Z})\|, \tag{2.39}$$

avec $\bar{Z} = [q, \dot{q}_r]$.

En substituant (2.38) et (2.39) dans (2.37), et en utilisant (2.25), il résulte que

$$\begin{aligned}
 s^T \chi(Z) &\leq \sum_{i=1}^n |s_i| (\theta_d \|S_{\chi_i}(Z)\| + \theta_d), \\
 &\leq \theta_d \sum_{i=1}^n |s_i| (\|S_{\chi_i}(\bar{Z})\| + 1), \\
 &\leq \theta_d \sum_{i=1}^n |s_i U_i|.
 \end{aligned} \tag{2.40}$$

L'équation (2.40) permet de borner $s^T \chi(Z)$ par une fonction n'utilisant pas des signaux difficilement mesurables. Cependant cette fonction dépend de $|s_i U_i|$. Pour enlever cette valeur absolue, nous exploitons le Lemme 1.4.2. On obtient :

$$\begin{aligned}
 s^T \chi(Z) &\leq \theta_d \sum_{i=1}^n s_i U_i \tanh\left(\frac{s_i U_i}{\bar{\varepsilon}_i}\right) + \beta_0, \\
 &\leq \theta_d s^T \bar{U} + \beta_0,
 \end{aligned} \tag{2.41}$$

avec $\beta_0 = 0.2785 \times \theta_d \sum_{i=1}^n \bar{\varepsilon}_i$.

En substituant les Équations (2.41) et (2.18) dans (2.34) on obtient

$$\begin{aligned}
 \dot{V}(t) &= -s^T K_d (\dot{q}_r - \dot{q}_c) + \theta_d s^T \bar{U} + \beta_0 + \tilde{x}^T \left(Y_j(q, \dot{q}) \tilde{\theta}_J + \hat{J}(q) s - \alpha \zeta(x) \right) + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d \\
 &\quad + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{W}_i - s^T K_d s
 \end{aligned}$$

$$\begin{aligned}
 \dot{V}(t) &= -s^T K_d (\dot{q}_r - \dot{q}_c) + \theta_d s^T \bar{U} + \tilde{x}^T \hat{J}(q) s + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J \\
 &\quad + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{W}_i - s^T K_d s - \tilde{x}^T \alpha \zeta(x) + \tilde{x}^T Y_j(q, \dot{q}) \tilde{\theta}_J + \beta_0
 \end{aligned}$$

$$\begin{aligned}
 \dot{V}(t) &= -s^T \left[K_d (\dot{q}_r - \dot{q}_c) + \hat{\theta}_d \bar{U} + \tilde{x}^T \hat{J}(q) \right] + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{W}_i \\
 &\quad - s^T K_d s - \tilde{x}^T \alpha \zeta(x) + \tilde{x}^T Y_j(q, \dot{q}) \tilde{\theta}_J + \beta_0 - s^T \tilde{\theta}_d \bar{U}
 \end{aligned} \tag{2.42}$$

Pour assurer la stabilité du système bouclé (Figure 2.1), la vitesse de commande articulaire est définie par :

$$\dot{q}_c = \dot{q}_r - \hat{W} v, \tag{2.43}$$

avec :

$$v = \hat{\theta}_d \bar{U} + \hat{J}^T(q) \tilde{x}. \quad (2.44)$$

L'équation (2.42) devient

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) - s^T K_d \hat{W} [\hat{\theta}_d \bar{U} + \hat{J}^T(q) \tilde{x}] + [\hat{\theta}_d + \tilde{\theta}_d] s^T \bar{U} + \beta_0 \\ & + s^T \hat{J}(q) \tilde{x} + \tilde{x}^T Y_J(q, \dot{q}) \tilde{\theta}_J + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i \end{aligned} \quad (2.45)$$

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) - s^T K_d \hat{W} [\hat{\theta}_d \bar{U} + \hat{J}^T(q) \tilde{x}] + s^T [\hat{\theta}_d \bar{U} + \hat{J}(q) \tilde{x}] \\ & + s^T \bar{U} \tilde{\theta}_d + \beta_0 + \tilde{x}^T Y_J(q, \dot{q}) \tilde{\theta}_J + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i \end{aligned}$$

d'où

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) + s^T [1 - K_d \hat{W}] [\hat{\theta}_d \bar{U} + \hat{J}^T(q) \tilde{x}] + s^T \bar{U} \tilde{\theta}_d + \beta_0 \\ & + \tilde{x}^T Y_J(q, \dot{q}) \tilde{\theta}_J + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i. \end{aligned} \quad (2.46)$$

En substituant (2.44) dans (2.46) on obtient :

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) + s^T [1 - K_d \hat{W}] v + \beta_0 + s^T \bar{U} \tilde{\theta}_d + \tilde{x}^T Y_J(q, \dot{q}) \tilde{\theta}_J \\ & + \frac{1}{\Gamma_d} \tilde{\theta}_d \dot{\tilde{\theta}}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \dot{\tilde{\theta}}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i. \end{aligned} \quad (2.47)$$

On sait que $K_d \tilde{W} = K_d (K_d^{-1} - \hat{W}) = 1 - K_d \hat{W}$, alors on a $K_d \hat{W} = 1 - K_d \tilde{W}$. De plus, en se basant sur le fait que θ_d et θ_J sont des constantes, donc leurs dérivées sont nulles, on a $\dot{\tilde{\theta}}_d = -\dot{\hat{\theta}}_d$ et $\dot{\tilde{\theta}}_J = -\dot{\hat{\theta}}_J$.

L'équation (2.47) devient

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) + s^T K_d \tilde{W} v + \beta_0 + \frac{1}{\Gamma_d} \tilde{\theta}_d \left[s^T \bar{U} \Gamma_d - \dot{\tilde{\theta}}_d \right] \\ & + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \left[\Gamma_J \tilde{x}^T Y_J(q, \dot{q}) - \dot{\tilde{\theta}}_J \right] + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i. \end{aligned} \quad (2.48)$$

De (2.48), nous obtenons les lois d'adaptation suivantes :

$$\dot{\hat{\theta}}_d = \Gamma_d \times S^T(\bar{Z}) \bar{U} - \gamma_d \hat{\theta}_d, \quad (2.49)$$

$$\dot{\hat{\theta}}_J = \Gamma_J Y_J^T(q, \dot{q}) \tilde{x} - \gamma_J \hat{\theta}_J, \quad (2.50)$$

où Γ_d et Γ_J sont des matrices diagonales des gains d'adaptation tandis que γ_d et γ_J

sont des constantes appelées communément en anglais σ -modification coefficient. Leur rôle est d'assurer la convergence des lois de mise à jour.

Substituons (2.49) et (2.50) dans (2.47), on a

$$\begin{aligned} \dot{V}(t) &\leq -s^T K_d s - \tilde{x}^T \alpha \zeta(x) + s^T K_d \tilde{W} v + \beta_0 + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i \\ &\quad + \frac{1}{\Gamma_d} \tilde{\theta}_d \left[s^T \bar{U} \Gamma_d - s^T \bar{U} \Gamma_d + \gamma_d \hat{\theta}_d \right] + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \left[\Gamma_J \tilde{x}^T Y_J(q, \dot{q}) - \Gamma_J \tilde{x}^T Y_J(q, \dot{q}) + \gamma_J \hat{\theta}_J \right] \\ \dot{V}(t) &\leq -s^T K_d s - \tilde{x}^T \alpha \zeta(x) + s^T K_d \tilde{W} v + \beta_0 + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\tilde{W}}_i + \frac{1}{\Gamma_d} \tilde{\theta}_d \gamma_d \hat{\theta}_d + \frac{1}{\Gamma_J} \tilde{\theta}_J^T \gamma_J \hat{\theta}_J \end{aligned} \quad (2.51)$$

On sait que :

$$\theta_d = \hat{\theta}_d + \tilde{\theta}_d \Rightarrow \hat{\theta}_d = \theta_d - \tilde{\theta}_d, \text{ de même } \theta_J = \hat{\theta}_J + \tilde{\theta}_J \Rightarrow \hat{\theta}_J = \theta_J - \tilde{\theta}_J$$

Nous avons :

$$\frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d \hat{\theta}_d = \frac{\gamma_d}{\Gamma_d} \left(-\tilde{\theta}_d^2 + \tilde{\theta}_d \theta_d \right) \leq -\frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 + \frac{\gamma_d}{\Gamma_d} \theta_d^2. \quad (2.52)$$

On déduit que :

$$\frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \hat{\theta}_J \leq -\frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J + \frac{\gamma_J}{\Gamma_J} \theta_J^T \theta_J. \quad (2.53)$$

Puisque \tilde{W} est une matrice et v un vecteur, le terme $s^T K_d \tilde{W} v$ peut être réécrit de la manière suivante :

$$s^T K_d \tilde{W} v = \sum_{i=1}^n \tilde{W}_i^T v_i K_d s, \quad (2.54)$$

À partir de la définition de \tilde{W} et comme K_d est une matrice de constante, nous obtenons l'équation suivante

$$\dot{\tilde{W}}_i = \frac{d}{dt} K_{d_i}^{-1} - \dot{W}_i = -\dot{W}_i, \quad (2.55)$$

où $K_{d_i}^{-1}$ est la $i^{\text{ème}}$ colonne de K_d^{-1} .

Substituons (2.52), (2.53), (2.54), (2.55), dans (2.51),

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) + \sum_{i=1}^n s K_d \tilde{W}_i^T v_i + \beta_0 - \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d \dot{\hat{W}}_i \\ & + \frac{\gamma_d}{\Gamma_d} [-\tilde{\theta}_d^2 + \theta_d^2] + \frac{\gamma_J}{\Gamma_J} [-\tilde{\theta}_J^T \tilde{\theta}_J + \theta_J^T \theta_J] \end{aligned} \quad (2.56)$$

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) - \frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 - \frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d [\Gamma_{W_i} s v - \dot{\hat{W}}_i] \\ & \beta_0 + \frac{\gamma_d}{\Gamma_d} \theta_d^2 + \frac{\gamma_J}{\Gamma_J} \theta_J^T \theta_J \end{aligned}$$

Posons $\beta_1 = \beta_0 + \frac{\gamma_d}{\Gamma_d} \theta_d^2 + \frac{\gamma_J}{\Gamma_J} \theta_J^T \theta_J$.

Nous obtenons alors la troisième loi d'adaptation

$$\dot{\hat{W}}_i = \Gamma_{W_i} v_i s - \gamma_{W_i} \hat{W}_i, \quad (2.57)$$

avec Γ_{W_i} la matrice diagonale des gains d'adaptation et γ_{W_i} le σ -modification coefficient de la loi d'adaptation \hat{W} . L'équation (2.56) devient

$$\dot{V}(t) \leq -s^T K_d s - \tilde{x}^T \alpha \zeta(x) - \frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 - \frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J + \beta_1 + \sum_{i=1}^n \frac{1}{\Gamma_{W_i}} \tilde{W}_i^T K_d [\Gamma_{W_i} s v - \Gamma_{W_i} s v + \gamma_{W_i} \hat{W}_i],$$

$$\dot{V}(t) \leq -s^T K_d s - \tilde{x}^T \alpha \zeta(x) - \frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 - \frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J + \sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T K_d \hat{W}_i + \beta_1 \quad (2.58)$$

Cependant on sait que : $\tilde{W}_i^T \hat{W}_i \leq -\tilde{W}_i^T \tilde{W}_i + K_d^{-1}$ d'où (2.58) devient

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - \tilde{x}^T \alpha \zeta(x) - \frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 - \frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J + \beta_1 \\ & - \sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T K_d \tilde{W}_i + \sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T K_d K_{d_i}^{-1}. \end{aligned} \quad (2.59)$$

Notons que $K_d K_{d_i}^{-1}$ est un vecteur dont le $i^{\text{ème}}$ élément est égal à un, alors que tous les autres éléments sont nuls. Ainsi, nous obtenons l'égalité suivante

$$\left(K_d K_{d_i}^{-1} \right)^T \left(K_d K_{d_i}^{-1} \right) = 1, \forall i = \{1, 2, \dots, n\}. \quad (2.60)$$

En utilisant (2.60) et l'inégalité de Young, on a

$$\sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T K_d K_{d_i}^{-1} \leq \sum_{i=1}^n \frac{\gamma_{W_i}}{2\Gamma_{W_i}} \tilde{W}_i^T \tilde{W}_i + \sum_{i=1}^n \frac{\gamma_{W_i}}{2\Gamma_{W_i}}. \quad (2.61)$$

En se basant sur l'évolution de la fonction exponentielle, il s'en-suit que $\exp(\tilde{x}_i^2 - \varpi_i^2) \geq \exp(-1) = 0.3679$, pour tout $\tilde{x}_i \in \mathbb{R}$ et $\varpi_i \in]0, 1]$. Par conséquent, le terme $-\tilde{x}^T \alpha \zeta(x)$ est borné par :

$$-\tilde{x}^T \alpha \zeta(x) \leq -0.3679 \times k \times \tilde{x}^T \alpha \tilde{x}. \quad (2.62)$$

En substituant (2.61) et (2.62) dans (2.59), on obtient

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - 0.3679 \times k \times \tilde{x}^T \alpha \tilde{x} - \frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 - \frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J \\ & - \sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \tilde{W}_i + \beta_2, \end{aligned} \quad (2.63)$$

où $\mathbf{I}_{n \times n}$ est une matrice identité $n \times n$ et $\beta_2 = \sum_{i=1}^n \frac{\gamma_{W_i}}{2\Gamma_{W_i}} + \beta_1$.

Étant donné que K_d est inconnu, pour conclure sur le signe de $\dot{V}(t)$ et par conséquent sur la stabilité du système globale (Figure 2.2), nous allons évaluer le signe de $K_d - \frac{1}{2} \mathbf{I}_{n \times n}$. Pour cela, nous avons deux cas, à savoir, $\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \geq 0$ et $\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) < 0$.

Remarque 2.2.1. Notez que le cas $\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) < 0$ est peu probable en pratique. Habituellement, les gains de rétroaction $K_{d_{ii}}$, $i = \{1, 2, \dots, n\}$ sont suffisamment importants pour réduire le temps de réponse et l'erreur de suivie durant le régime permanent. Par conséquent, en pratique, $K_{d_{ii}} \gg \frac{1}{2}$ de sorte que la matrice $K_d - \frac{1}{2} \mathbf{I}_{n \times n}$ soit toujours définie positive [67, 80, 81]. Néanmoins, nous étudions les deux cas pour montrer que la stabilité du système globale (Figure 2.2) est garantie pour tout K_d positivement défini.

Cas 1 : Considérons que $\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) < 0$.

Dans ce cas, la matrice $K_d - \frac{1}{2} \mathbf{I}_{n \times n}$ est indéfinie ou définie négative, ce qui signifie que le terme $\tilde{W}_i^T \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \tilde{W}_i$ peut être positif ou négatif. Par conséquent, le signe du terme $-\sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \tilde{W}_i + \beta_2$ est indéfini. Si l'on choisit les paramètres de conception $\{\gamma_d, \gamma_J, \gamma_{W_i}, \bar{\varepsilon}_i\}$ aussi petits que possible et $\{\Gamma_d, \Gamma_J, \Gamma_{W_i}\}$ aussi significatifs que possible, ce terme peut devenir petit. Par conséquent, si les paramètres de conception $\{k, \alpha_{ii}\}$ sont choisis suffisamment grands. Alors, d'après (2.63) on aura $\dot{V}(t) \leq 0$. D'après la théorie de Lyapunov, les signaux du système bouclé sont bornés et $\tilde{x}(t) \rightarrow 0$ lorsque $t \rightarrow \infty$.

Cas 2 : Nous considérons que $\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \geq 0$.

Cette dernière signifie que la matrice $K_d - \frac{1}{2} \mathbf{I}_{n \times n}$ est définie semi-positive. Par conséquent, pour tout $y \in \mathbb{R}^n$, les inégalités suivantes sont valides

$$\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) y^T y \leq y^T \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) y \leq \lambda_{\max} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) y^T y. \quad (2.64)$$

En utilisant (2.64) dans (2.63), on obtient

$$\begin{aligned} \dot{V}(t) \leq & -s^T K_d s - 0.3679 \times k \times \tilde{x}^T \alpha \tilde{x} - \frac{\gamma_d}{\Gamma_d} \tilde{\theta}_d^2 - \frac{\gamma_J}{\Gamma_J} \tilde{\theta}_J^T \tilde{\theta}_J \\ & - \sum_{i=1}^n \frac{\gamma_{W_i}}{\Gamma_{W_i}} \tilde{W}_i^T \lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \tilde{W}_i + \beta_2. \end{aligned} \quad (2.65)$$

Étant donné que la matrice de gain de rétroaction K_d est positive avec de grandes valeurs, il existe donc une constante positive β_3 qui satisfait les deux contraintes suivantes :

$$\beta_3 \leq \min \left\{ 0.7358 \times k \times \alpha_i, 2 \times \gamma_d, 2 \times \gamma_J, 2 \times \gamma_{W_i} \times \frac{\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right)}{\lambda_{\max} (K_d)} \right\}, \quad (2.66)$$

et

$$\beta_3 \leq \frac{2 \times \lambda_{\min} (K_d)}{\lambda_{\max} (M(q))}. \quad (2.67)$$

En substituant (2.66) et (2.67) dans (2.65), il en ressort que :

$$\dot{V}(t) \leq -\beta_3 V(t) + \beta_2. \quad (2.68)$$

Multiplions (2.68) par $e^{\beta_3 t}$ on obtient :

$$\begin{aligned} \dot{V}(t) e^{\beta_3 t} & \leq -\beta_3 V(t) e^{\beta_3 t} + \beta_2 e^{\beta_3 t}, \\ \dot{V}(t) e^{\beta_3 t} + \beta_3 V(t) e^{\beta_3 t} & \leq \beta_2 e^{\beta_3 t}, \\ \frac{d}{dt} (V(t) e^{\beta_3 t}) & \leq \beta_2 e^{\beta_3 t}. \end{aligned} \quad (2.69)$$

En intégrant l'inégalité (2.69) dans l'intervalle $[0, t]$ est

$$\begin{aligned} \int_0^t \frac{d}{dt} (V(t) e^{\beta_3 t}) dt & \leq \int_0^t \beta_2 e^{\beta_3 t} dt, \\ [V(t) e^{\beta_3 t}]_0^t & \leq \left[\frac{\beta_2}{\beta_3} e^{\beta_3 t} \right]_0^t, \\ V(t) e^{\beta_3 t} - V(0) e^{0 \times t} & \leq \frac{\beta_2}{\beta_3} e^{\beta_3 t} - \frac{\beta_2}{\beta_3} e^{\beta_3 \times 0}, \end{aligned} \quad (2.70)$$

Nous pouvons alors établir que

$$V(t) \leq \left(V(0) - \frac{\beta_2}{\beta_3} \right) e^{-\beta_3 t} + \frac{\beta_2}{\beta_3}. \quad (2.71)$$

L'équation (2.71) prouve que tous les signaux du système bouclé (Voir Figure 2.2) s , \tilde{x} , $\tilde{\theta}_d$, $\tilde{\theta}_J$, et \tilde{W}_i , $i = \{1, 2, \dots, n\}$ sont SGUUB [73]. De plus, $\|\tilde{x}\| \leq \beta_4 = \sqrt{\frac{2\beta_2}{\beta_3}}$ quand $t \rightarrow \infty$. Cela signifie que la limite de l'erreur de suivi en régime permanent β_4 peut devenir

petite avec un bon choix de paramètres de conception de sorte que l'erreur de suivi $\tilde{x}(t)$ converge vers le voisinage de l'origine lorsque $t \rightarrow \infty$. Ce dernier point complète la preuve du Théorème 2.2.1.

Remarque 2.2.2. *L'analyse de la relation entre les paramètres de conception et le temps de réponse est nécessaire pour les applications en temps réel.*

Notez que pour le cas $\lambda_{\min} \left(K_d - \frac{1}{2} \mathbf{I}_{n \times n} \right) \geq 0$, nous avons :

$$\|\tilde{x}\| \leq \sqrt{2 \left[V(0) - \frac{\beta_2}{\beta_3} \right] e^{-\beta_3 t} + 2 \frac{\beta_2}{\beta_3}}. \quad (2.72)$$

Si $V(0) = \frac{\beta_2}{\beta_3}$, alors $\|\tilde{x}\| \leq \beta_4$. Si $V(0) \neq \frac{\beta_2}{\beta_3}$, pour tout $\beta_5 > \beta_4$, il existe $T > 0$, tel que pour tout $t > T$, on a $\|\tilde{x}\| \leq \beta_5$. Par conséquent, il existe β_5 tel que,

$$\beta_5 = \sqrt{2 \left[V(0) - \frac{\beta_2}{\beta_3} \right] e^{-\beta_3 T} + 2 \frac{\beta_2}{\beta_3}}. \quad (2.73)$$

Alors ,

$$T = -\frac{1}{\beta_3} \ln \left(\frac{\beta_5^2 - 2 \frac{\beta_2}{\beta_3}}{2 \left[V(0) - \frac{\beta_2}{\beta_3} \right]} \right). \quad (2.74)$$

On observe que le temps de réponse T peut devenir petit si β_3 est assez grand et β_2 assez petit. Ces exigences contrastent avec celles de (2.66) et (2.67). Par conséquent, par la méthode d'essai et d'erreur, un compromis doit être trouvé de sorte que les paramètres de conception $\bar{\varepsilon}_i$, γ_d , γ_J , et γ_{W_i} soient choisis aussi petits que possible pour respecter les contraintes (2.66) et (2.67), tandis que k , α_i , Γ_d , Γ_J , et Γ_{W_i} sont choisis aussi grands que possible pour améliorer les performances de suivi et le temps de réponse.

Dans cette section, nous avons considéré que le RMIACII évolue dans un environnement déterministe. Dans celle qui suit, nous allons considérer qu'il évolue dans un environnement aléatoire. Nous allons élaborer une loi de commande qui permet à un RMIACIII de réaliser des tâches en respectant les contraintes de précision et de stabilité, malgré le caractère aléatoire de l'environnement dans lequel il évolue.

2.3 Commande Neuro-Adaptative des RMIACIIIs évoluant dans un environnement aléatoire

La commande développée dans cette section est une extension de celle de la section 2.2. Contrairement aux approches que l'on retrouve dans la littérature [14, 16, 44, 45, 49] qui sont sensibles aux erreurs de modélisation et aux variations des paramètres cinématiques

et dynamiques du RMI, donc non adaptative, l'approche développée dans cette thèse est adaptative, plus précise, rapide et robuste.

2.3.1 Présentation de l'approche de commande

Nous proposons une commande adaptative hybride qui combine la commande adaptative directe pour le rejet des perturbations aléatoires et la commande adaptative indirecte pour le rejet des incertitudes de modélisation et des paramètres incertains du robot. En effet, les récents travaux [14, 16, 44, 45, 49] sur la commande des RMIACIIIs soumis aux vibrations aléatoires proposent des stratégies de commande non-adaptatives. Ce qui est un réel problème lorsque les paramètres du RMIACIIIs varient ou sont incertains. En plus, ils considèrent que le contrôleur interne du RMI est accessible et connu.

Les premières approches de rejet des perturbations aléatoires [14, 44, 45, 49] ont utilisé la technique de séparation pour isoler les perturbations aléatoires représentées par la matrice d'excitation aléatoire $\Gamma(q) \in \mathbb{R}^{n \times r}$ du modèle dynamique du robot, où r est la dimension de l'espace des tâches. Ils obtiennent ainsi $\|\Gamma(q)\|_F^2$ qui représente la norme de Frobenius de $\Gamma(q)$. Pour rejeter les perturbations, ils considèrent que $\|\Gamma(q)\|_F^2$ a une borne supérieure. La deuxième approche fait recours à l'inégalité de Young pour séparer les dynamiques des perturbations aléatoires à celles du robot et obtient $\Gamma(q)\Gamma^T(q)y$. Dans ces deux approches, nous constatons que la matrice d'excitation aléatoire $\Gamma(q)$ dépend des paramètres du robot. Par conséquent, ces approches sont sensibles aux variations des paramètres cinématiques ou aux incertitudes de modélisation, du robot par conclusion non adaptatives.

Pour développer notre approche de commande, nous démontrons que $\Gamma(q)\Gamma^T(q)y$ peut se mettre sous une forme paramétrique linéaire telle que

$$\Gamma(q)\Gamma^T(q)y = Q(q, y)\Theta, \quad (2.75)$$

avec :

- $Q(q, y) \in \mathbb{R}^{n \times \bar{m}}$, la matrice qui regroupe les dynamiques du robot. Elle est appelée matrice de régression ou regressueur.
- $\Theta \in \mathbb{R}^{\bar{m}}$, le vecteur qui regroupe les paramètres cinématiques du robot.

Notre approche adaptative consiste à réajuster en ligne les éléments de Θ .

2.3.2 Modélisation d'un robot manipulateur industriel en environnement aléatoire

Nous présentons dans cette section, le modèle d'un RMI soumis à des vibrations aléatoires telle que élucidée dans les travaux de Cui et al. [13]. Les modèles géométriques et cinématiques des RMIs ne sont pas affectés par les vibrations aléatoires. Par conséquent, les modèles géométrique et cinématique de la Section 2.2.1 restent valables dans ce contexte. Considérons le RMI à $n - DDL$ représenté dans la Figure 2.3.

Principe de propagation des vibrations aléatoires

Les vibrations aléatoires sur un RMI connecté à un point fixe O_0 du sol vont produire sur ce point O_0 , des accélérations aléatoires $\xi \in \mathbb{R}^r$ qui sont des processus stationnaires indépendants. Selon le principe des mouvements relatifs des particules, ces accélérations vont se décomposer en deux composantes donc une composante tangentielle a_{ci2} qui va se propager au segment suivant $i + 1$ et une composante normale a_{ci1} qui va générer sur le segment i , d'après la deuxième lois de Newton, une force d'inertie de nature stochastique défini par

$$F_{ci} = -m_i \times a_{ci1}, \quad (2.76)$$

avec m_i la masse du $i^{\text{ième}}$ segment et a_{ci1} l'accélération perpendiculaire au segment i . La composante tangentielle a_{ci2} qui s'est propagée au segment suivant $i + 1$ va également se décomposer en deux autres composantes (normale et tangentielle). Le phénomène se reproduit jusqu'au dernier segment n .

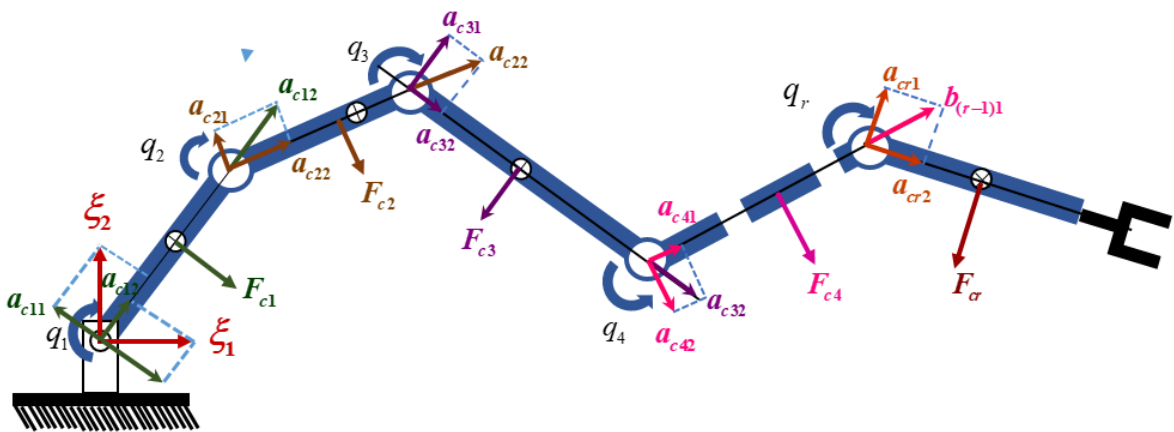


FIGURE 2.3 – Modèle d'un RMI n-DDL en environnement aléatoire.

Modélisation du RMIACIII en environnement aléatoire

La Figure 2.4 décrit le principe de propagation des accélérations aléatoires dans un RMI soumis aux vibrations aléatoires .

A. Première étape : Décomposition de $\vec{\xi}$ (Figure 2.2.a)

La projection de ξ_i dans la direction Y (si le segment évolue dans le plan vertical) ou dans la direction Z (si le segment évolue dans le plan horizontal) par rapport à la base du robot (Voir Figure 2.4) va se décomposer en deux accélérations. Une accélération perpendiculaire a_{c11} au segment 1 et une autre a_{c12} dans la direction du segment :

$$a_{c11} = -\xi_1 \sin(q_1) + \xi_2 \cos(q_1), \quad (2.77)$$

$$a_{c12} = \xi_1 \cos(q_1) + \xi_2 \sin(q_1).$$

D'après la deuxième loi de Newton, a_{c11} va créer une force stochastique F_1 et a_{c12} va se propager le long du segment 2. La force stochastique F_1 est définie par :

$$F_1 = -m_1 a_{c11} = m_1 (\xi_1 \sin(q_1) - \xi_2 \cos(q_1)) \quad (2.78)$$

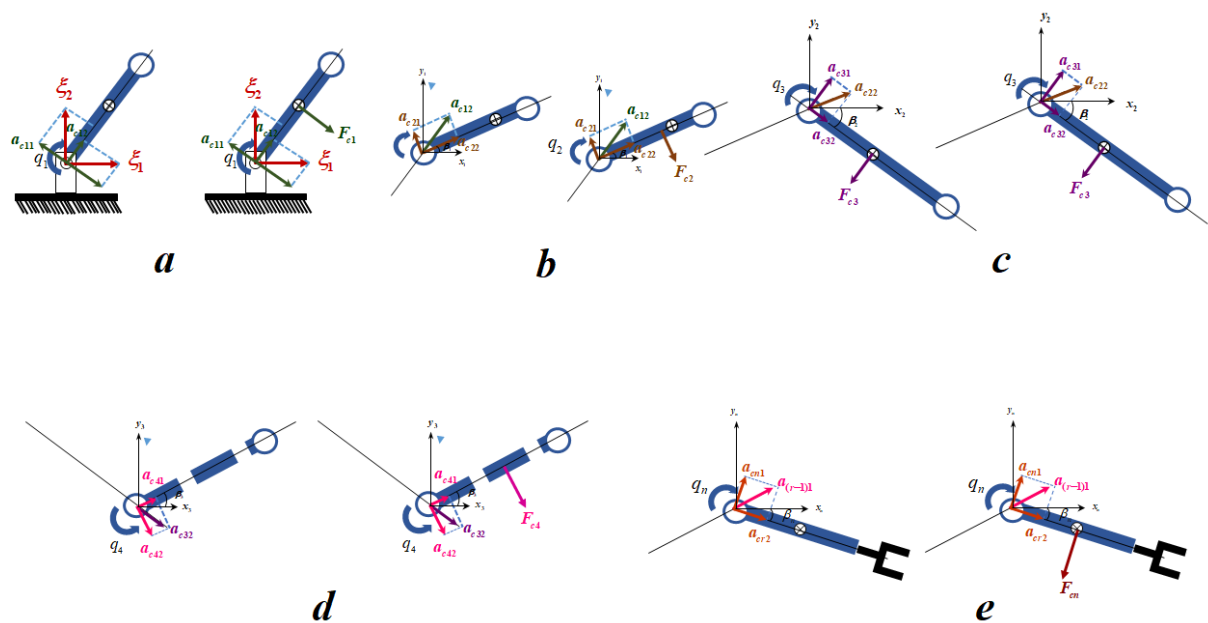


FIGURE 2.4 – Propagation des accélérations aléatoires.

B. Deuxième étape : Décomposition de a_{c12} (Figure 2.2.b)

L'accélération a_{c12} se décompose également en une composante normale a_{c21} et une composante tangentielle a_{c22} définie par :

$$\begin{aligned} a_{c21} &= -a_{c12} \sin(\eta_1); \\ a_{c22} &= a_{c12} \cos(\eta_1); \end{aligned} \quad (2.79)$$

avec $\eta_1 = q_2 - q_1$. L'expression (2.79) devient :

$$\begin{aligned} a_{c21} &= -(\xi_1 \cos(q_1) + \xi_2 \sin(q_1)) \sin(q_2 - q_1); \\ a_{c22} &= (\xi_1 \cos(q_1) + \xi_2 \sin(q_1)) \cos(q_2 - q_1). \end{aligned} \quad (2.80)$$

Comme dans **A.**, a_{c21} va créer une force d'inertie stochastique F_2 tandis que a_{c22} va se propager le long du segment 3. Ainsi, on a :

$$\begin{aligned} F_2 &= -m_2 a_{c21}; \\ F_2 &= m_2 (\xi_1 \cos(q_1) + \xi_2 \sin(q_1)) \sin(q_2 - q_1); \\ F_2 &= m_2 \beta_1 \xi_1 \cos(q_1) + m_2 \beta_1 \xi_2 \sin(q_1); \end{aligned} \quad (2.81)$$

avec $\beta_1 = \sin(q_2 - q_1)$ et $\gamma_1 = \cos(q_2 - q_1)$.

C. Troisième étape : Décomposition de a_{c22} (Figure 2.2.c)

L'accélération a_{c22} se décompose également en une composante normale a_{c31} et une composante tangentielle a_{c32} qui sont respectivement :

$$\begin{aligned} a_{c31} &= -a_{c22} \sin(\eta_2); \\ a_{c32} &= a_{c22} \cos(\eta_2); \end{aligned} \quad (2.82)$$

avec $\eta_2 = q_3 - q_2$. En remplaçant a_{c22} par son expression, on obtient :

$$\begin{aligned} a_{c31} &= -(\xi_1 \cos q_1 + \xi_2 \sin q_1) \cos(q_2 - q_1) \sin(q_3 - q_2), \\ a_{c32} &= (\xi_1 \cos q_1 + \xi_2 \sin q_1) \cos(q_2 - q_1) \cos(q_3 - q_2). \end{aligned} \quad (2.83)$$

Comme dans les étapes précédentes, l'accélération a_{c31} va également créer une force d'inertie stochastique F_3 tandis que a_{c32} va se propager le long du segment 4.

$$\begin{aligned} F_3 &= -m_3 a_{c31} \\ F_3 &= -m_3 (\xi_1 \cos q_1 + \xi_2 \sin q_1) \cos(q_2 - q_1) \sin(q_3 - q_2) \\ F_3 &= m_3 \beta_2 \xi_1 \cos q_1 + m_3 \beta_2 \xi_2 \sin q_1 \end{aligned} \quad (2.84)$$

avec $\beta_2 = \cos(q_2 - q_1) \sin(q_3 - q_2)$ et $\gamma_2 = \cos(q_2 - q_1) \cos(q_3 - q_2)$

D. Quatrième étape : Décomposition de a_{c32} (Figure 2.2.d)

Le même phénomène se répète pour a_{c32} qui se décompose également en une composante normale a_{c41} et une composante tangentielle a_{c42} qui sont respectivement :

$$\begin{aligned} a_{c41} &= -a_{c32} \sin(\eta_3), \\ a_{c42} &= a_{c32} \cos(\eta_3), \end{aligned} \quad (2.85)$$

avec $\eta_3 = q_4 - q_3$. En remplaçant a_{c32} par son expression, on obtient :

$$\begin{aligned} a_{c41} &= -(\xi_1 \cos q_1 + \xi_2 \sin q_1) \cos(q_2 - q_1) \cos(q_3 - q_2) \sin(q_4 - q_3), \\ a_{c42} &= (\xi_1 \cos q_1 + \xi_2 \sin q_1) \cos(q_2 - q_1) \cos(q_3 - q_2) \cos(q_4 - q_3). \end{aligned} \quad (2.86)$$

L'accélération a_{c41} crée également une force d'inertie stochastique F_4 définie par :

$$\begin{aligned} F_4 &= -m_4 a_{c41}, \\ F_4 &= -m_4 (\xi_1 \cos q_1 + \xi_2 \sin q_1) \cos(q_2 - q_1) \cos(q_3 - q_2) \sin(q_4 - q_3), \\ F_4 &= m_4 \beta_3 \xi_1 \cos q_1 + m_3 \beta_3 \xi_2 \sin q_1. \end{aligned} \quad (2.87)$$

avec $\beta_3 = \cos(q_2 - q_1) \cos(q_3 - q_2) \sin(q_4 - q_3)$ et $\gamma_3 = \cos(q_2 - q_1) \cos(q_3 - q_2) \cos(q_4 - q_3)$.

I. ième étape : Décomposition de a_{ci2} (Figure 2.2.d)

D'une manière générale, l'accélération ${}^i a_{ci-1,2}$ dans le $i^{\text{ème}}$ segment se décompose en deux composantes définis par :

$$\begin{aligned} a_{ci1} &= -(\xi_1 \cos q_1 + \xi_2 \sin q_1) \beta_i(q), \\ a_{ci2} &= (\xi_1 \cos q_1 + \xi_2 \sin q_1) \gamma_i(q), \end{aligned} \quad (2.88)$$

avec $\beta_i(q) = \prod_{k=2}^{i-1} \cos(q_k - q_{k-1}) \sin(q_i - q_{i-1})$; $\gamma_i(q) = \prod_{k=2}^{i-1} \cos(q_k - q_{k-1}) \cos(q_i - q_{i-1})$
et $i = 3, \dots, n$.

La force d'inertie stochastique créée dans le $i^{\text{ème}}$ segment est définie par :

$$\begin{aligned} F_{ci} &= m_i \times a_{ci2}, \\ F_{ci} &= m_i \cos(q_1) \beta_i(q) \xi_1 + m_i \sin(q_1) \beta_i(q) \xi_2. \end{aligned} \quad (2.89)$$

La combinaison des forces d'inertie stochastiques de chaque segment va d'après les travaux

de Spong [53], générer un couple stochastique généralisé définit par :

$$\begin{aligned}\tau_{ci} &= \sum_{j=1}^n \left[-F_{cj} \sin(q_j) \frac{\partial x_j}{\partial q_i} + F_{cj} \cos(q_j) \frac{\partial y_j}{\partial q_i} \right]; \\ \tau_{ci} &= F_{ci} l_{ci} + \sum_{j=1+i}^n F_{cj} l_i \cos(q_j - q_i);\end{aligned}\tag{2.90}$$

avec :

- l_{ci} , la distance entre l'extrémité inférieur i et son centre de masse ;
- l_i , la longueur du segment i ;
- m_i , la masse du segment i ;
- x_i, y_i les coordonnées du segment i par rapport à la base,
- $i = 1, \dots, n$.

En développant l'expression (2.89), on obtient l'expression du couple généralisé stochastique définit par :

$$\tau_{ci} = \Gamma_{i1}(q) \xi_1(q) + \Gamma_{i2}(q) \xi_2(q), \quad i = 1, \dots, n\tag{2.91}$$

avec :

$$\begin{aligned}\Gamma_{11}(q) &= m_1 l_{c1} \sin(q_1) + \sum_{j=2}^n m_j l_1 \cos(q_1) \beta_j(q) \cos(q_j - q_1), \\ \Gamma_{12}(q) &= -m_1 l_{c1} \cos(q_1) + \sum_{j=2}^n m_j l_1 \sin(q_1) \beta_j(q) \cos(q_j - q_1), \\ \Gamma_{i1}(q) &= m_i l_{ci} \cos(q_1) \beta_i(q) + \sum_{j=i+1}^n m_j l_i \cos(q_1) \beta_j(q) \cos(q_j - q_1), \\ \Gamma_{i2}(q) &= m_i l_{ci} \sin(q_1) \beta_i(q) + \sum_{j=i+1}^n m_j l_i \sin(q_1) \beta_j(q) \cos(q_j - q_1).\end{aligned}$$

Ainsi, le couple stochastique généralisé peut se mettre sous la forme

$$\tau_{ci} = \Gamma(q) \xi.\tag{2.92}$$

Ce couple τ_{ci} va s'additionner au couple τ du RMI. On obtient ainsi le modèle dynamique d'un RMI soumis aux perturbations aléatoires suivant

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = \tau + \Gamma(q) \xi,\tag{2.93}$$

avec :

- $\ddot{q} \in \mathbb{R}^n$, le vecteur d'accélération des articulations ;
- $M(q) \in \mathbb{R}^{n \times n}$, la matrice d'inertie ;
- $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$, La matrice de Coriolis ;

- $g(q) \in \mathbb{R}^n$, le couple gravitationnel ;
- $\tau \in \mathbb{R}^n$, le couple ;
- $\xi = [\xi_1, \xi_2, \dots, \xi_r]^T$, le vecteur d'accélération aléatoires ;
- $\Gamma(q) = (\Gamma_{ij}(q))_{n \times r}$, la matrice d'excitation aléatoire.

En plus des Propriétés 1.4.1 ; 1.4.2 ; 1.4.3 et les Lemmes 1.4.1 ; 1.4.2 de la Section 1.4, nous considérons l'Hypothèse 2.3.1 qui stipule que l'espérance probabiliste des accélérations aléatoires qui représentent la valeur moyenne pondérée des accélérations aléatoires admet une borne supérieure.

Hypothèse 2.3.1. *Le processus $\xi(t)$ est continu, stationnaire et \mathcal{F}_t -adapté, s'il existe une constante $\alpha > 0$ telle que pour tout $t \geq t_0$, [48]*

$$\sup_{t_0 \leq s \leq t} \mathbb{E} \|\xi(s)\|^2 \leq \alpha. \quad (2.94)$$

2.3.3 Linéarisation du couple stochastique généralisée τ_c

Afin d'appliquer notre commande adaptative hybride, nous démontrons que le vecteur $\Gamma(q) \Gamma^T(q) y$ utilisé dans les travaux Cui [13, 16, 49] pour éliminer les perturbations aléatoires, peut se mettre sous la forme paramétrique linéaire tel que

$$\Gamma(q) \Gamma^T(q) y = Q(q, y) \Theta. \quad (2.95)$$

Pour cela, nous savons que le vecteur des couples stochastiques généralisées est décrit par [13]

$$\tau_{cj} = \sum_{i=1}^n \sum_{s=1}^r \left(F_{ci}(s) \frac{\partial x_{ci}(s)}{\partial q_j} \right), \quad (2.96)$$

où $F_{ci}(s)$ et $x_{ci}(s)$ représentent le $s^{\text{ième}}$ élément respectivement de la Force d'inertie stochastique et de la position du centre de masse. Définissons ${}^0H_i \in \mathbb{R}^{(r+1) \times (r+1)}$, la matrice de transformation homogène décrivant la relation entre la base du robot et le $i^{\text{ième}}$ segment ; $\check{x}_{ci} = [{}^i x_{ci}^T, 1]^T$ la position du centre de masse du $i^{\text{ième}}$ segment avec le ${}^i x_{ci} \in \mathbb{R}^r$ et $\check{F}_{ci} = [{}^i F_{ci}^T, 0]^T$ la force inertielle stochastique du $i^{\text{ième}}$ segment avec ${}^i F_{ci} = -m_i {}^i a_{ci2}$. Le vecteur des couples stochastiques généralisés de (2.96) peut être réécrit comme suit

$$\begin{aligned} \tau_{cj} &= \sum_{i=1}^n \left({}^0H_i \check{F}_{ci} \right)^T \left(\frac{\partial} {\partial q_j} {}^0H_i \check{x}_{ci} \right), \\ &= \sum_{i=1}^n \check{F}_{ci}^T {}^0H_i^T \left(\frac{\partial} {\partial q_j} {}^0H_i \right) \check{x}_{ci}. \end{aligned} \quad (2.97)$$

D'après la définition de ${}^i a_{ci2}$, il existe une matrice $\Omega_i(q) \in \mathbb{R}^{(r+1) \times (r+1)}$ telle que

$$\check{F}_{ci} = m_i \Omega_i(q) \check{\xi}, \quad (2.98)$$

avec $\check{\xi} = [\xi^T, 1]^T$.

On sait que pour un robot à articulation rotoïde $\frac{\partial}{\partial q_j} {}^0 H_i = 0_{(r+1) \times (r+1)}$, pour $j > i$. En utilisant ce dernier terme et en mettant (2.98) dans (2.97), il résulte que

$$\begin{aligned} \tau_{cj} &= \sum_{i=j}^n m_i \check{\xi}^T \Omega_i^T {}^0 H_i^T \left(\frac{\partial}{\partial q_j} {}^0 H_i \right) \check{x}_{ci} \\ &= \sum_{i=j}^n \left(m_i \check{x}_{ci}^T \left(\frac{\partial}{\partial q_j} {}^0 H_i \right)^T {}^0 H_i \Omega_i \right) \check{\xi} \\ &= [\Gamma_{j \times r}, 1] \check{\xi}, \end{aligned} \quad (2.99)$$

avec

$$[\Gamma_{j \times r}, 1] = \sum_{i=j}^n \left(m_i \check{x}_{ci}^T \left(\frac{\partial}{\partial q_j} {}^0 H_i \right)^T {}^0 H_i \Omega_i \right), \quad (2.100)$$

et $\Gamma_{j \times r}$ le $j^{\text{ième}}$ segment de $\Gamma(q)$.

A partir de (2.100), nous obtenons la formule explicite des éléments de $\Gamma(q)$

$$\Gamma_{js} = \sum_{i=j}^n m_i \check{x}_{ci}^T \left[\left(\frac{\partial}{\partial q_j} {}^0 H_i \right)^T {}^0 H_i \Omega_i \right]_{(r+1) \times s}, \quad (2.101)$$

avec $1 \leq j \leq n$, et $1 \leq s \leq r$.

Cependant, pour un robot à articulation rotoïde, ${}^0 H_i$ dépend linéairement des paramètres cinématiques. Par conséquent, nous pouvons conclure que Γ_{js} dépend linéairement des paramètres dynamiques et cinématiques. Définissons maintenant $v, y \in \mathbb{R}^n$ tels que $v = \Gamma(q) \Gamma^T(q) y$. Le $j^{\text{ième}}$ élément de v est donné par

$$v_j = \sum_{s=1}^n \sum_{k=1}^r \Gamma_{jk} \Gamma_{sk} y_s, \quad (2.102)$$

avec y_s le $s^{\text{ième}}$ élément de y .

Puisque Γ_{js} dépend linéairement des paramètres dynamiques et cinématiques, nous pouvons facilement conclure que v_j de (2.102) dépend linéairement des paramètres cinématiques et dynamiques. Ce dernier point signifie que v peut être linéairement paramétré. En d'autres termes, nous pouvons trouver une matrice $Q(q, y) \in \mathbb{R}^{n \times \bar{m}}$ et un vecteur des paramètres cinématiques du robot $\Theta \in \mathbb{R}^{\bar{m}}$ tel que $\Gamma(q) \Gamma^T(q) y = Q(q, y) \Theta$.

2.3.4 Présentation de l'algorithme de commande

Nous savons que la forme générale du contrôleur interne d'un RMIACIII est (En référence à l'étude faite dans la Section 2.2)

$$\tau = -K_d(\dot{q} - \dot{q}_c) + \Phi \left(q, \dot{q}, q_c, \int_0^t q(s)ds, \int_0^t q_c(s)ds \right), \quad (2.103)$$

avec

- \dot{q}_c , la commande de vitesse articulaire (contrôleur externe);
- K_d , la matrice de gain de rétroaction inconnue;
- $\Phi \in \mathbb{R}^{n \times n}$, la partie inconnue du contrôleur interne.

Nous définissons ainsi :

- $\tilde{x} = x(t) - x_{des}(t)$, l'erreur de suivi dans l'espace des tâches.
- $J^\dagger(q) = J(q)^T [J(q)J(q)^T]^{-1}$, la pseudo inverse de la matrice jacobienne $J(q)$.

Le calcul de la commande en vitesse \dot{q}_c nécessite la vitesse de référence suivante

$$\dot{q}_r = \hat{J}^\dagger(q) [\dot{x}_{des} - \varrho_0 \psi(\tilde{x})], \quad (2.104)$$

avec :

- $\hat{J}^\dagger(q) = \hat{J}(q)^T [\hat{J}(q)\hat{J}(q)^T]^{-1}$, l'estimation de $J^\dagger(q)$;
- $\hat{J}(q)$, L'estimation de $J(q)$;
- $\varrho_0 = \text{diag}(\varrho_{01}, \varrho_{02}, \dots, \varrho_{0r})$, la matrice de gain diagonale définie positive.

$\psi(\tilde{x}) \in \mathbb{R}^r$ est un vecteur de performance défini par

$$\psi(\tilde{x}) = \varrho_1 \times \text{diag} \left(\left[\exp(\tilde{x}_1^2 - \varpi_1^2), \dots, \exp(\tilde{x}_r^2 - \varpi_r^2) \right] \right) \tilde{x}, \quad (2.105)$$

où :

- \tilde{x}_j est le $j^{\text{ème}}$ élément de \tilde{x} ;
- $\varpi_j, j = 1, 2, \dots, r$ sont les contraintes de performance spécifiées par l'utilisateur;
- ϱ_1 un paramètre de conception positif.

Les paramètres ϖ_j représentent les erreurs maximales de suivi dans l'espace de tâche que le contrôleur peut tolérer. Lorsque l'erreur de suivi \tilde{x}_j est hors de la plage attendue $[-\varpi_j, \varpi_j]$ (c'est-à-dire $\tilde{x}_j^2 > \varpi_j^2$), en se basant sur la propriété de la fonction exponentielle,

le terme $\varrho_0 \psi(\tilde{x})$ augmente pour imposer la convergence de l'erreur de suivi vers zéro. En conséquence, le taux de convergence augmente lorsque les contraintes de performances de suivi sont violées. Définissons, l'erreur de suivi des vitesses articulaires par

$$z = \dot{q} - \dot{q}_r, \quad (2.106)$$

La dérivée de (2.106) est :

$$\dot{z} = \ddot{q} - \ddot{q}_r. \quad (2.107)$$

Cependant, on sait que

$$\begin{aligned} M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) &= \tau + \Gamma(q) \xi, \\ M(q) \ddot{q} &= \tau - (C(q, \dot{q}) \dot{q} + G(q)) + \Gamma(q) \xi. \end{aligned} \quad (2.108)$$

En multipliant (2.107) par $M(q)$ et en substituant (2.108) et (2.103) dans le résultat obtenu on obtient

$$\begin{aligned} M(q) \dot{z} &= M(q) \ddot{q} - M(q) \ddot{q}_r; \\ M(q) \dot{z} &= \tau - (C(q, \dot{q}) \dot{q} + G(q)) + \Gamma(q) \xi - M(q) \ddot{q}; \\ M(q) \dot{z} &= -K_d (\dot{q} - \dot{q}_c) + \Phi(*) - (C(q, \dot{q}) \dot{q} + G(q)) + \Gamma(q) \xi - M(q) \ddot{q}; \\ M(q) \dot{z} &= -K_d (\dot{q} - \dot{q}_c) - C(q, \dot{q}) \dot{q} - G(q) - M(q) \ddot{q} + \Phi(*) + \Gamma(q) \xi. \end{aligned} \quad (2.109)$$

Ainsi, (2.109) peut se mettre sous la forme

$$M(q) \dot{z} = -K_d (\dot{q} - \dot{q}_c) - C(q, \dot{q}) z + \Gamma(q) \xi + \Psi(Z), \quad (2.110)$$

avec :

$$\begin{aligned} \Psi(Z) &= -[M(q) \ddot{q}_r + C(q, \dot{q}) \dot{q}_r + G(q) - \Phi(*)]; \\ \text{et} & \\ Z &= \left[q, \dot{q}, \dot{q}_r, \ddot{q}_r, q_c, \int_0^t q(s) ds, \int_0^t q_c(s) ds \right]^T. \end{aligned} \quad (2.111)$$

Explorons ainsi l'erreur de suivi dans l'espace des tâches. Nous exploitons le fait que la vitesse dans l'espace des tâches \dot{x} et la vitesse des articulations \dot{q} sont liées par le MCD suivant $\dot{x} = J(q) \dot{q}$. On sait que $J(q) \dot{q} = Y(q, \dot{q}) \theta$. Par conséquent, la dérivée de \tilde{x} donne

la dynamique d'erreur suivante

$$\begin{aligned}
 \dot{\tilde{x}} &= \dot{x} - \dot{x}_{des}, \\
 &= J(q)\dot{q} - \dot{x}_{des}, \\
 &= Y(q, \dot{q})\tilde{\theta} + Y(q, \dot{q})\hat{\theta} - \dot{x}_{des}, \\
 &= Y(q, \dot{q})\tilde{\theta} + \hat{J}(q)\dot{q} - \dot{x}_{des} \\
 &= -\varrho_0\psi(\tilde{x}) + Y(q, \dot{q})\tilde{\theta} + \hat{J}(q)z,
 \end{aligned} \tag{2.112}$$

avec :

- $\tilde{\theta} = \theta - \hat{\theta}$, l'erreur paramétrique .
- $\hat{\theta}$, l'estimation du vecteur des paramètres cinématiques idéaux θ .

Les paramètres cinématiques et dynamiques étant incertains, le vecteur $\Psi(Z)$ est incertain. $\Psi(Z)$ représente les incertitudes déterministes tandis que $\Gamma(q)\xi$ représente les incertitudes stochastiques. De plus, K_d est inconnu car l'utilisateur ne connaît pas la structure du contrôleur interne. Nous allons considérer que K_d est une matrice diagonale. Par conséquent, trouver \dot{q}_c qui permet de faire face aux incertitudes mentionnées ci-dessus tout en assurant la stabilité du système en boucle fermée, est un problème majeur dans la commande des RMIACIII. Pour résoudre ce problème, la loi de commande des vitesses suivante est proposée

$$\dot{q}_c = \dot{q}_r - \hat{\Upsilon}\vartheta, \tag{2.113}$$

avec :

$$\vartheta = \hat{\beta}\bar{U} + \hat{J}^T(q)\tilde{x} + \frac{1}{4\varrho_2}Q(q, z)\hat{\Theta}, \tag{2.114}$$

où :

- $\hat{\Upsilon} \in \mathbb{R}^{n \times n}$, l'estimation de K_d^{-1} .
- ϱ_2 , le paramètre de conception positif;
- $\hat{\beta} \in \mathbb{R}$, l'estimation de β .

Le paramètre β est utilisé pour borner le modèle RBF NN de la fonction $z^T\Psi(Z)$. Il est définis comme suit

$$\beta = \max\{\delta_{\Psi_1}, \delta_{\Psi_2}, \dots, \delta_{\Psi_n}, \|W_{\Psi_1}\|, \dots, \|W_{\Psi_n}\|\}. \tag{2.115}$$

Le vecteur $\bar{U} \in \mathbb{R}^n$ est choisi de la manière suivante

$$\bar{U} = \left[U_1 \tanh\left(\frac{U_1 z_1}{\varepsilon_1}\right), \dots, U_n \tanh\left(\frac{U_n z_n}{\varepsilon_n}\right) \right]^T, \tag{2.116}$$

avec $\varepsilon_i > 0$ un paramètre de conception, et z_i le $i^{\text{ème}}$ élément de z , $i = 1, 2, \dots, n$.

La fonction U_i est définie par

$$U_i = \|S_i(\bar{Z})\| + 1, \quad (2.117)$$

avec $\bar{Z} = [q, \dot{q}_r]^T$.

Les estimés des paramètres adaptatifs θ , β , Θ , et Υ sont définis par les lois d'adaptations suivantes :

$$\dot{\hat{\theta}} = \lambda_1 Y^T(q, \dot{q}) \tilde{x} - \gamma_1 \hat{\theta}, \quad (2.118)$$

$$\dot{\hat{\beta}} = \lambda_2 z^T \bar{U} - \gamma_2 \hat{\beta}, \quad (2.119)$$

$$\dot{\hat{\Theta}} = \frac{\lambda_3}{4\varrho_2} Q^T(q, z) z - \gamma_3 \hat{\Theta}, \quad (2.120)$$

$$\dot{\hat{\Upsilon}}_i = \lambda_{4i} \vartheta_i z - \gamma_{4i} \hat{\Upsilon}_i, \quad (2.121)$$

où

- $\hat{\Upsilon}_i$, la $i^{\text{ème}}$ colonne de la matrice $\hat{\Upsilon}$;
- ϑ_i , la $i^{\text{ème}}$ élément de ϑ , $i = 1, 2, \dots, n$;
- les paramètres λ_j , γ_j , λ_{4i} et γ_{4i} , $j = 1, 2, 3$, sont des paramètres de conception positifs.

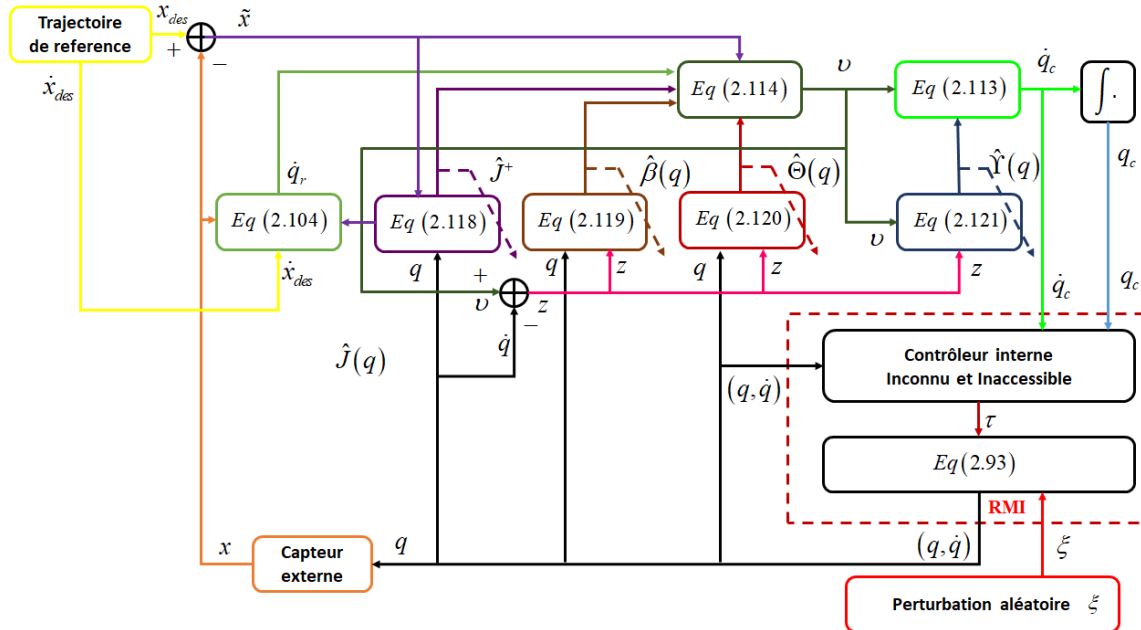


FIGURE 2.5 – Vue d'ensemble du schéma de contrôle proposé.

L'algorithme de la loi de commande proposée se résume dans la Figure 2.5.

L'étude de la stabilité du système bouclé décrit dans la Figure 2.5 est l'objet de la section suivante.

2.3.5 Analyse de la stabilité

En présence des vibrations aléatoires, le système décrit dans la Figure 2.5 est stable si l'espérance de l'erreur de suivi dans l'espace de tâche $\mathbb{E}\|\tilde{x}(t)\|^2$ est bornée. Cela se traduit par le Théorème 2.3.1. L'analyse de la stabilité revient à vérifier ce théorème.

Théorème 2.3.1. *Considérons le modèle aléatoire (2.93) d'un RMIACIII à articulation rotoïde à n -DLL soumis à des vibrations aléatoires. Pour tout signal de référence $x_{des}(t)$ admettant au moins une dérivé, la loi de commande 2.113 avec les lois de mise à jour 2.118, 2.119, 2.120 et 2.121 assurent que :*

- (1) *Le système bouclé (Figure 2.5) admet une solution unique globale qui garantit la stabilité SGUUB ;*
- (2) *Tous les signaux du système bouclé (Figure 2.5) sont bornée ;*
- (3) *L'erreur de suivi dans l'espace des tâches $\tilde{x}(t) = x(t) - x_{des}(t)$ doit satisfaire à :*

$$\lim_{t \rightarrow \infty} \mathbb{E}\|\tilde{x}(t)\|^2 \leq \frac{2(\rho_2\alpha + \rho_2)}{\rho_3}. \quad (2.122)$$

Preuve

Les signaux du système bouclé (Figure 2.5) sont z , \tilde{x} , $\tilde{\theta} = \theta - \hat{\theta}$, $\tilde{\beta} = \beta - \hat{\beta}$, $\tilde{\Theta} = \Theta - \hat{\Theta}$, et $\tilde{\Upsilon}_i$ la $i^{\text{ième}}$ colonne de la matrice $\tilde{\Upsilon} = K_d^{-1} - \hat{\Upsilon}$. Par conséquent, nous choisissons le candidat de Lyapunov $V(t) \in \mathbb{R}$ suivant

$$V(t) = \frac{1}{2}z^T M(q)z + \frac{1}{2}\tilde{x}^T \tilde{x} + \frac{1}{2\lambda_1}\tilde{\theta}^T \tilde{\theta} + \frac{1}{2\lambda_2}\tilde{\beta}^2 + \frac{1}{2\lambda_3}\tilde{\Theta}^T \tilde{\Theta} + \frac{1}{2} \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \tilde{\Upsilon}_i. \quad (2.123)$$

En dérivant (2.123) et en utilisant (2.110), (2.112) et la Propriété 1.4.1, nous obtenons

$$\begin{aligned} \dot{V}(t) = & -z^T K_d (\dot{q} - \dot{q}_c) z + z^T \Gamma(q) \xi + z^T \Psi(Z) - \rho_0 \tilde{x}^T \psi(\tilde{x}) + \tilde{x}^T Y(q, \dot{q}) \tilde{\theta} + \tilde{x}^T \hat{J}(q) z \\ & + \frac{1}{\lambda_1} \tilde{\theta}^T \dot{\tilde{\theta}} + \frac{1}{\lambda_2} \tilde{\beta} \dot{\tilde{\beta}} + \frac{1}{\lambda_3} \tilde{\Theta}^T \dot{\tilde{\Theta}} + \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \dot{\tilde{\Upsilon}}_i. \end{aligned} \quad (2.124)$$

Les paramètres β , θ , Θ et K_d^{-1} de la $i^{\text{ième}}$ colonne de K_d^{-1} étant des constantes, nous avons donc les inégalités suivantes

$$\dot{\tilde{\beta}} = -\dot{\hat{\beta}}, \quad \dot{\tilde{\theta}} = -\dot{\hat{\theta}}, \quad \dot{\tilde{\Theta}} = -\dot{\hat{\Theta}}, \quad \dot{\tilde{\Upsilon}}_i = -\dot{\hat{\Upsilon}}_i. \quad (2.125)$$

En appliquant l'inégalité de Young à $z^T \Gamma(q) \xi$ et par la suite la forme linéaire paramétrée de $\Gamma^T(q) \Gamma(q) z$, nous obtenons les inégalités suivantes

$$\begin{aligned} z^T \Gamma(q) \xi &\leq \frac{1}{4\varrho_2} z^T \Gamma(q) \Gamma^T(q) z + \varrho_2 \|\xi\|^2, \\ &\leq \frac{1}{4\varrho_2} z^T Q(q, z) \Theta + \varrho_2 \|\xi\|^2. \end{aligned} \quad (2.126)$$

Le vecteur $\Psi(Z)$ étant inconnu, par conséquent, ses éléments $\Psi_i(Z)$ sont approximés par le modèle RBF NN suivant

$$\Psi_i(Z) = W_{\Psi_i}^T S_i(Z) + \delta_{\Psi_i}(Z), \quad (2.127)$$

avec $|\delta_{\Psi_i}(Z)| \leq \epsilon_{\Psi_i}$, $i = 1, 2, \dots, n$.

Les expressions (2.115), (2.116), (2.117) et (2.127), nous permettent de borner la fonction non linéaire $z^T \Psi(Z)$.

En effet :

$$\begin{aligned} z^T \Psi(Z) &= \sum_{i=1}^n z_i [W_{\Psi_i}^T S_i(Z) + \delta_{\Psi_i}(Z)], \\ &\leq \sum_{i=1}^n |z_i| [\|W_{\Psi_i}^T\| \|S_i(Z)\| + |\delta_{\Psi_i}(Z)|], \\ &\leq \beta \sum_{i=1}^n |z_i| [\|S_i(Z)\| + 1], \\ &\leq \beta \sum_{i=1}^n |z_i| [\|S_i(\bar{Z})\| + 1], \\ &\leq \beta \sum_{i=1}^n |z_i U_i| \\ &\leq \beta z^T \bar{U} + 0.2785 \times \beta \sum_{i=1}^n \epsilon_i. \end{aligned} \quad (2.128)$$

En substituant (2.103), (2.125), (2.126), et (2.128) dans (2.124), on obtient

$$\begin{aligned} \dot{V}(t) &\leq -z^T K_d (\dot{q} - \dot{q}_c) + \frac{1}{4\varrho_2} z^T Q(q, z) \Theta + \varrho_2 \|\xi\|^2 + \beta z^T \bar{U} + 0.2785 \times \beta \sum_{i=1}^n \epsilon_i \\ &\quad - \varrho_0 \tilde{x}^T \psi(\tilde{x}) + \tilde{x}^T Y(q, \dot{q}) \tilde{\theta} + \tilde{x}^T \hat{J}(q) z - \frac{1}{\lambda_1} \tilde{\theta}^T \dot{\hat{\theta}} - \frac{1}{\lambda_2} \tilde{\beta} \dot{\hat{\beta}} - \frac{1}{\lambda_3} \tilde{\Theta}^T \dot{\hat{\Theta}}, \\ &\quad - \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{Y}_i^T K_d \dot{\hat{Y}}_i, \\ &\leq -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) - z^T K_d \hat{Y} \vartheta + \frac{1}{4\varrho_2} z^T Q(q, z) \Theta + \varrho_2 \|\xi\|^2 + \beta z^T \bar{U} \\ &\quad + 0.2785 \times \beta \sum_{i=1}^n \epsilon_i + \tilde{\theta}^T Y(q, \dot{q}) \tilde{x} + z^T \hat{J}(q) \tilde{x} - \frac{1}{\lambda_1} \tilde{\theta}^T \dot{\hat{\theta}} - \frac{1}{\lambda_2} \tilde{\beta} \dot{\hat{\beta}} \\ &\quad - \frac{1}{\lambda_3} \tilde{\Theta}^T \dot{\hat{\Theta}} - \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{Y}_i^T K_d \dot{\hat{Y}}_i, \end{aligned} \quad (2.129)$$

Ainsi l'on a

$$\begin{aligned}
 \dot{V}(t) \leq & -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) - z^T K_d \hat{\Upsilon} \vartheta + \varrho_2 \|\xi\|^2 + 0.2785 \times \beta \sum_{i=1}^n \varepsilon_i \\
 & + z^T \left[\hat{\beta} \bar{U} + \hat{J}(q) \tilde{x} + \frac{1}{4\varrho_2} Q(q, z) \hat{\Theta} \right] + \frac{1}{\lambda_1} \tilde{\theta}^T \left[\lambda_1 Y(q, \dot{q}) \tilde{x} - \dot{\hat{\theta}} \right], \\
 & + \frac{1}{\lambda_2} \tilde{\beta} \left[\lambda_2 z^T \bar{U} - \dot{\hat{\beta}} \right] - \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \dot{\hat{\Upsilon}}_i + \frac{1}{\lambda_3} \tilde{\Theta}^T \left[\frac{\lambda_3}{4\varrho_2} Q(q, z) z - \dot{\hat{\Theta}} \right].
 \end{aligned} \tag{2.130}$$

La substitution de (2.104), (2.111), (2.112), et (2.113) dans (2.130), donne

$$\begin{aligned}
 \dot{V}(t) \leq & -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) + \varrho_2 \|\xi\|^2 + z^T \left[\mathbf{1}_{n \times n} - K_d \hat{\Upsilon} \right] \vartheta + \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \hat{\theta} \\
 & + \frac{\gamma_2}{\lambda_2} \tilde{\beta} \hat{\beta} + \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \hat{\Theta} - \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \dot{\hat{\Upsilon}}_i + 0.2785 \times \beta \sum_{i=1}^n \varepsilon_i.
 \end{aligned} \tag{2.131}$$

Le théorème de Young nous permet d'avoir les inégalités suivantes :

$$\begin{aligned}
 \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \hat{\theta} & \leq -\frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \tilde{\theta} + \frac{\gamma_1}{\lambda_1} \|\theta\|^2, \\
 \frac{\gamma_2}{\lambda_2} \tilde{\beta} \hat{\beta} & \leq -\frac{\gamma_2}{\lambda_2} \tilde{\beta}^2 + \frac{\gamma_2}{\lambda_2} \beta^2, \\
 \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \hat{\Theta} & \leq -\frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \tilde{\Theta} + \frac{\gamma_3}{\lambda_3} \|\Theta\|^2.
 \end{aligned} \tag{2.132}$$

En substituant (2.132) dans (2.131) et en utilisant le fait que $\mathbf{1}_{n \times n} - K_d \hat{\Upsilon} = K_d \tilde{\Upsilon}$, il en découle que

$$\begin{aligned}
 \dot{V}(t) \leq & -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) - \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \tilde{\theta} \\
 & - \frac{\gamma_2}{\lambda_2} \tilde{\beta}^2 - \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \tilde{\Theta} + z^T K_d \tilde{\Upsilon} \vartheta \\
 & - \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \dot{\hat{\Upsilon}}_i + \varrho_2 \|\xi\|^2 + \rho_1,
 \end{aligned} \tag{2.133}$$

$$\text{avec : } \rho_1 = \frac{\gamma_1}{\lambda_1} \|\theta\|^2 + \frac{\gamma_2}{\lambda_2} \beta^2 + \frac{\gamma_3}{\lambda_3} \|\Theta\|^2 + 0.2785 \times \beta \sum_{i=1}^n \varepsilon_i.$$

Puisque $\tilde{\Upsilon}$ est une matrice et z un vecteur, le terme $z^T K_d \tilde{\Upsilon} \vartheta$ peut être réécrit de la manière suivante

$$z^T K_d \tilde{\Upsilon} \vartheta = \sum_{i=1}^n \tilde{\Upsilon}_i^T \vartheta_i K_d z, \tag{2.134}$$

avec ϑ_i le $i^{\text{ième}}$ élément de ϑ , $i = 1, 2, \dots, n$.

En substituant (2.121) et (2.134) dans (2.133), on obtient

$$\begin{aligned}
 \dot{V}(t) &\leq -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) - \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \tilde{\theta} - \frac{\gamma_2}{\lambda_2} \tilde{\beta}^2 \\
 &\quad - \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \tilde{\Theta} + \sum_{i=1}^n \frac{1}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \left[\lambda_{4i} \vartheta_i z - \hat{\Upsilon}_i \right] \\
 &\quad + \varrho_2 \|\xi\|^2 + \rho_1, \\
 &\leq -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) - \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \tilde{\theta} - \frac{\gamma_2}{\lambda_2} \tilde{\beta}^2 \\
 &\quad - \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \tilde{\Theta} - \sum_{i=1}^n \frac{\gamma_{4i}}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \tilde{\Upsilon}_i \\
 &\quad + \sum_{i=1}^n \frac{\gamma_{4i}}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d K_{di}^{-1} + \varrho_2 \|\xi\|^2 + \rho_1.
 \end{aligned} \tag{2.135}$$

Notons que $K_d K_{di}^{-1}$ est un vecteur dont le $i^{\text{ième}}$ élément est égal à 1, alors que tous les autres éléments sont nuls. Par conséquent, l'égalité suivante s'applique

$$\left(K_d K_{di}^{-1} \right)^T \left(K_d K_{di}^{-1} \right) = 1, \quad \forall i = 1, 2, \dots, n. \tag{2.136}$$

D'après (2.136) et l'inégalité de Young, il s'en-suit que

$$\sum_{i=1}^n \frac{\gamma_{4i}}{\lambda_{4i}} \tilde{\Upsilon}_i^T K_d K_{di}^{-1} \leq \sum_{i=1}^n \frac{\gamma_{4i}}{2\lambda_{4i}} \tilde{\Upsilon}_i^T \tilde{\Upsilon}_i + \sum_{i=1}^n \frac{\gamma_{4i}}{2\lambda_{4i}}. \tag{2.137}$$

En substituant (2.137) dans (2.135), nous obtenons :

$$\begin{aligned}
 \dot{V}(t) &\leq -z^T K_d z - \varrho_0 \tilde{x}^T \psi(\tilde{x}) - \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \tilde{\theta} - \frac{\gamma_2}{\lambda_2} \tilde{\beta}^2 \\
 &\quad - \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \tilde{\Theta} - \sum_{i=1}^n \frac{\gamma_{4i}}{\lambda_{4i}} \tilde{\Upsilon}_i^T \left(K_d - \frac{1}{2} I_{n \times n} \right) \tilde{\Upsilon}_i \\
 &\quad + \varrho_2 \|\xi\|^2 + \rho_2,
 \end{aligned} \tag{2.138}$$

$$\text{avec } \rho_2 = \sum_{i=1}^n \frac{\gamma_{4i}}{2\lambda_{4i}} + \rho_1.$$

Dans cette étude, K_d est supposé inconnu de l'utilisateur. Par conséquent, le signe de $K_d - \frac{1}{2} I_{n \times n}$ n'est pas connu a priori. Néanmoins, il faut remarquer que les gains de rétroaction K_{dii} sont choisis de manière suffisamment grand pour réduire le temps de réponse et l'erreur de suivi. Par conséquent, dans la plupart des cas pratiques, $K_{dii} \gg \frac{1}{2}$ pour permettre à ce que les matrices K_d et $K_d - \frac{1}{2} I_{n \times n}$ soient définies positives [80, 81]. Cela nous permet pour tout $y \in \mathbb{R}^n$, d'avoir les inégalités suivantes :

$$0 \leq \lambda_{\min}(K_d) y^T y \leq y^T K_d y \leq \lambda_{\max}(K_d) y^T y, \tag{2.139}$$

$$0 \leq y^T \left(K_d - \frac{1}{2} I_{n \times n} \right) y. \tag{2.140}$$

La propriété décroissance de la fonction $\exp(x)$ nous permet de déduire que

$$-\varrho_0 \tilde{x}^T \psi(\tilde{x}) \leq -\tilde{x}^T \text{diag}([\varrho_{01} \varrho_1 \exp(-\varpi_1^2), \varrho_{0r} \varrho_1 \exp(-\varpi_r^2)]) \tilde{x}. \quad (2.141)$$

En introduisant (2.139), (2.140) et (2.141) dans (2.138), nous obtenons

$$\begin{aligned} \dot{V}(t) &\leq -\lambda_{\min}(K_d) z^T z - \frac{\gamma_1}{\lambda_1} \tilde{\theta}^T \tilde{\theta} - \frac{\gamma_2}{\lambda_2} \tilde{\beta}^2 + \varrho_2 \|\xi\|^2 \\ &\quad - \frac{\gamma_3}{\lambda_3} \tilde{\Theta}^T \tilde{\Theta} - \sum_{i=1}^n \frac{\gamma_{4i}}{\lambda_{4i}} \tilde{\Upsilon}_i^T \left(K_d - \frac{1}{2} I_{n \times n} \right) \tilde{\Upsilon}_i \\ &\quad + \rho_2 - \tilde{x}^T \text{diag}([\varrho_{01} \varrho_1 \exp(-\varpi_1^2), \dots, \varrho_{0r} \varrho_1 \exp(-\varpi_r^2)]) \tilde{x}. \end{aligned} \quad (2.142)$$

À partir de (2.139) et (2.140), et du fait que la matrice d'inertie $M(q)$ soit définie positive, pour tout $y \in \mathbb{R}^n$, il existe une constante positive ρ_3 qui satisfait les trois contraintes suivantes

$$\rho_3 \leq \min \left\{ 2\varrho_{0j} \varrho_1 \exp(-\varpi_j^2), 2\gamma_k \right\}, \quad (2.143)$$

$$\frac{\rho_3}{2} \lambda_{\max}(M(y)) \leq \lambda_{\min}(K_d), \quad (2.144)$$

$$\frac{\rho_3}{2} y^T K_d y \leq \gamma_{4i} y^T \left(K_d - \frac{1}{2} I_{n \times n} \right) y, \quad (2.145)$$

avec $k = 1, 2, 3$, $j = 1, 2, \dots, r$, et $i = 1, 2, \dots, n$. En utilisant (2.123), (2.143), (2.144), (2.145) et la Propriété 1.4.1, (2.142) devient

$$\begin{aligned} \dot{V}(t) &\leq -\frac{\rho_3}{2} z^T M(q) z - \frac{\rho_3}{2\lambda_1} \tilde{\theta}^T \tilde{\theta} - \frac{\rho_3}{2\lambda_2} \tilde{\beta}^2 \\ &\quad - \frac{\rho_3}{2\lambda_3} \tilde{\Theta}^T \tilde{\Theta} - \sum_{i=1}^n \frac{\rho_3}{2\lambda_{4i}} \tilde{\Upsilon}_i^T K_d \tilde{\Upsilon}_i \\ &\quad - \frac{\rho_3}{2} \tilde{x}^T \tilde{x} + \varrho_2 \|\xi\|^2 + \rho_2 \\ &\leq -\rho_3 V(t) + \varrho_2 \|\xi\|^2 + \rho_2. \end{aligned} \quad (2.146)$$

En considérant l'espérance dans (2.146) et en utilisant l'Hypothèse 2.3.1, on a

$$\begin{aligned} \mathbb{E} \dot{V}(t) &\leq -\rho_3 \mathbb{E} V(t) + \varrho_2 \mathbb{E} \|\xi\|^2 + \rho_2 \\ &\leq -\rho_3 \mathbb{E} V(t) + \varrho_2 \alpha + \rho_2. \end{aligned} \quad (2.147)$$

Soit ν un vecteur défini par $\nu = [z^T, \tilde{x}^T, \tilde{\theta}^T, \tilde{\beta}, \tilde{\Theta}^T, \tilde{\Upsilon}_1^T, \dots, \tilde{\Upsilon}_n^T]^T$, le candidat de Lyapounov peut être borné comme suite

$$a_1 \|\nu\|^2 \leq V(t) \leq a_2 \|\nu\|^2, \quad (2.148)$$

avec

$$a_1 = \frac{1}{2} \min \left\{ \lambda_{\min}(M(q)), 1, \frac{1}{\lambda_k}, \frac{\lambda_{\min}(K_d)}{\max_{i=1,\dots,n}(\lambda_{4i})} \right\}, \quad (2.149)$$

et

$$a_2 = \frac{1}{2} \max \left\{ \lambda_{\max}(M(q)), 1, \frac{1}{\lambda_k}, \frac{\lambda_{\max}(K_d)}{\min_{i=1,\dots,n}(\lambda_{4i})} \right\}, \quad (2.150)$$

et $k = 1, 2, 3$. D'après (2.146), (2.148) et la Propriété 2.2.3 de [48], le système en boucle fermée a une solution globale unique, et son état a un gain asymptotique de second ordre. De plus, (2.146) et (2.148) signifient que $V(t) < \infty$, et $\dot{V}(t) < \infty$. Par conséquent, d'après le Théorème de Fubini, nous avons

$$\int_{t_1}^t \mathbb{E} \dot{V}(s) ds = \mathbb{E} \int_{t_1}^t \dot{V}(s) ds = \mathbb{E} V(t) - \mathbb{E} V(t_1). \quad (2.151)$$

Ce qui implique que

$$\mathbb{E} \frac{dV(t)}{dt} = \frac{d\mathbb{E}V(t)}{dt}. \quad (2.152)$$

En substituant (2.152) dans (2.147), en multipliant l'inégalité résultante par $\exp(\rho_3 t)$ et en intégrant les deux côtés sur $[0, t]$, il résulte que

$$\begin{aligned} \mathbb{E}V(t) &\leq \left(V(0) - \frac{\varrho_2 \alpha + \rho_2}{\rho_3} \right) \exp(-\rho_3 t) + \frac{\varrho_2 \alpha + \rho_2}{\rho_3}, \\ &\leq V(0) \exp(-\rho_3 t) + \frac{\varrho_2 \alpha + \rho_2}{\rho_3}. \end{aligned} \quad (2.153)$$

En considérant l'espérance dans (2.148), en utilisant (2.153) et l'inégalité de Chebyshev, on obtient

$$\begin{aligned} \lim_{\rho_4 \rightarrow \infty} \sup_{t > 0} P \{ \|\nu(t)\| > \rho_4 \} &\leq \lim_{\rho_4 \rightarrow \infty} \frac{\sup_{t > 0} \mathbb{E} \|\nu(t)\|^2}{\rho_4^2}, \\ &\leq \lim_{\rho_4 \rightarrow \infty} \frac{V(0) \exp(-\rho_3 t)}{a_1 \rho_4^2} \\ &\quad + \lim_{\rho_4 \rightarrow \infty} \frac{\varrho_2 \alpha + \rho_2}{a_1 \rho_3 \rho_4^2}, \\ &= 0. \end{aligned} \quad (2.154)$$

L'équation (2.154) signifie que $\nu(t)$ est **une variation bornée**. Par conséquent, on peut conclure que tous les signaux aléatoires du système en boucle fermée (Figure 2.5) sont **bornés**. Ainsi, on obtient

$$\begin{aligned} \lim_{t \rightarrow \infty} \mathbb{E} \|\tilde{x}(t)\|^2 &\leq \lim_{t \rightarrow \infty} 2\mathbb{E}V(t), \\ &\leq \lim_{t \rightarrow \infty} \left[2V(0) \exp(-\rho_3 t) + \frac{2(\varrho_2 \alpha + \rho_2)}{\rho_3} \right], \\ &\leq \frac{2(\varrho_2 \alpha + \rho_2)}{\rho_3}. \end{aligned} \quad (2.155)$$

On peut donc conclure que le système est stable.

La borne $\frac{2(\varrho_2\alpha+\varrho_2)}{\rho_3}$ peut devenir petite avec un choix approprié des paramètres de conception $\varpi_k, \gamma_j, \lambda_j, \varepsilon_i, \gamma_{4i}, \lambda_{4i}, \varrho_{0k}, \varrho_1$, et ϱ_2 , avec $i = 1, 2, \dots, n$, $j = 1, 2, 3$, et $k = 1, 2, \dots, r$.

avec $\rho_2 = \sum_{i=1}^n \frac{\gamma_{4i}}{2\lambda_{4i}} + \rho_1$, $\rho_1 = \frac{\gamma_1}{\lambda_1} \|\theta\|^2 + \frac{\gamma_2}{\lambda_2} \beta^2 + \frac{\gamma_3}{\lambda_3} \|\Theta\|^2 + 0.2785 \times \beta \sum_{i=1}^n \varepsilon_i$. ρ_3 est choisi en fonction des contraintes des expressions (2.143), (2.144) et (2.145). Nous pouvons donc conclure que la commande proposée garantie la stabilité du système bouclé.

2.4 Description des plates-formes expérimentales

Les lois de commande élaborées dans les sections précédentes seront mises en œuvre sur deux robots afin de tester leur efficacité. Ces plates-formes sont présentées dans cette section.

2.4.1 Description du Scorbot ER-4U

Le robot Scorbot ER-4U (Figure 2.6), distribué par la société Intelitek, est un robot anthropomorphe à 5 degrés de liberté. Son contrôleur interne est inconnu et inaccessible. Ses articulations sont de type rotoïdes. Elles sont actionnées par des moteurs à courant continu à aimants permanents (PMDC). Ces moteurs sont équipés d'encodeurs optiques en quadrature ainsi que d'interrupteurs de fin de course. Le système de transmission comprend des réducteurs de vitesse, des poulies et des courroies. Les mouvements du Scorbot ER-4U sont décrits dans le Tableau 2.1 et ses paramètres dynamiques dans le Tableau 2.2. La Figure 2.7 montre une vue d'ensemble du dispositif expérimental avec les annotations des principaux composants.

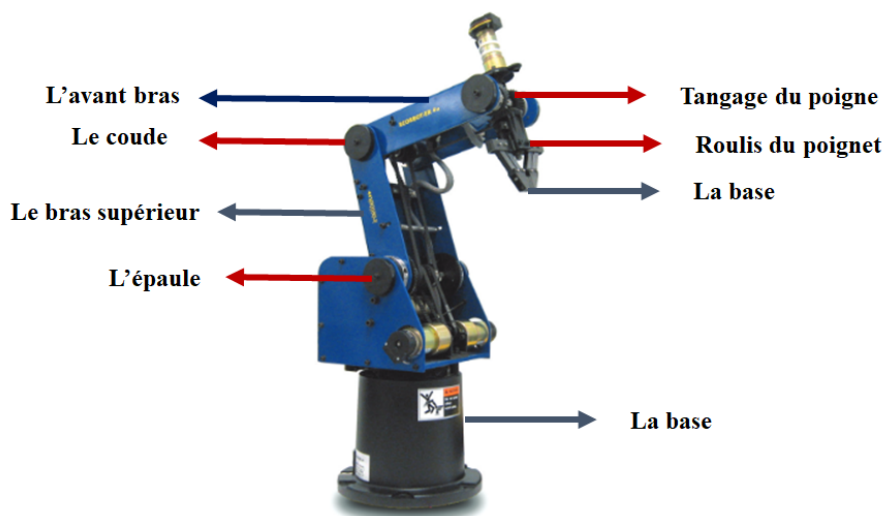


FIGURE 2.6 – Scorbot ER-4u.

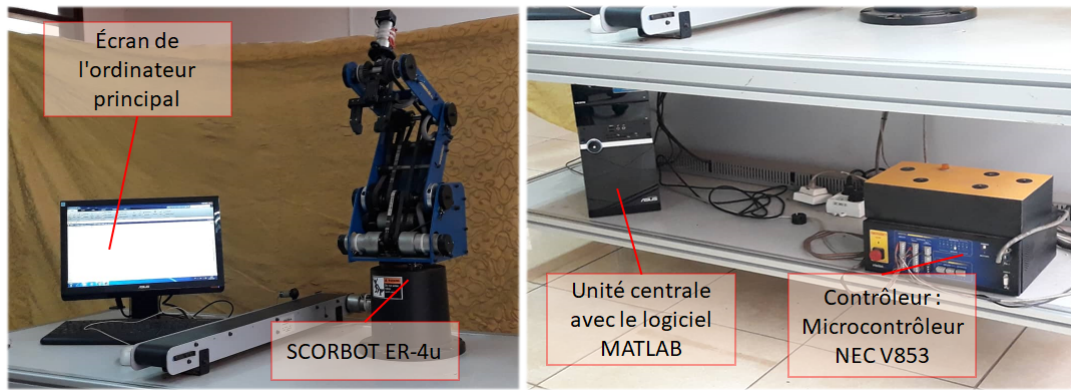


FIGURE 2.7 – Vue d'ensemble du dispositif expérimental avec l'annotation des principaux composants.

TABLE 2.1 – Description des mouvements du Scorbot ER-4U

N° de l'axe	Nom du segment	Mouvement effectué
1	Base	Fait pivoter le corps du robot
2	L'épaule (Shoulder)	Lève et abaisse le bras supérieur
3	Le coude (Elbow)	Lève et abaisse l'avant-bras
4	Tangage du poignet (Wrist Pitch)	Lève et abaisse l'effecteur
5	Roulis du poignet (Roll Pitch)	Rotation de l'effecteur

TABLE 2.2 – Paramètres du SCORBOT ER-4U

Segment n°	Paramètres [SI units]	Masse des segments [kg]
1	$d_1 = 0.3490$, $l_1 = 0.016$, $\alpha_1 = \pi/2$, $I_{xx} = 866.66 \times 10^{-6}$, $I_{yy} = 1616.665 \times 10^{-6}$ $I_{zz} = 1950 \times 10^{-6}$, $I_{xy} = 647.99 \times 10^{-6}$	2
2	$l_2 = 0.2210$, $I_{xx} = 799.976 \times 10^{-6}$, $I_{yy} = 17499.6 \times 10^{-6}$, $I_{zz} = 17499.6 \times 10^{-6}$	3
3	$l_3 = 0.2210$, $I_{xx} = 799.978 \times 10^{-6}$, $I_{yy} = 12499.6 \times 10^{-6}$, $I_{zz} = 12499.6 \times 10^{-6}$	3
4	$l_4 = 0$, $\alpha_4 = \pi/2$ rad, $I_{xx} = 0.6 \times 10^{-3}\text{kg.m}^2$, $I_{yy} = 0.6 \times 10^{-3}\text{kg.m}^2$, $I_{zz} = 0.3 \times 10^{-3}\text{kg.m}^2$	1.7
5	$l_5 = 0.15$, $I_{xx} = 0.9 \times 10^{-3}\text{kg.m}^2$, $I_{yy} = 0.9 \times 10^{-3}\text{kg.m}^2$, $I_{zz} = 0.2 \times 10^{-3}\text{kg.m}^2$	0.8

2.4.2 Description du robot UR5

L'UR5 (Figure 2.8) est un robot industriel série, léger, collaboratif et flexible à 6 DDL de type RRR. Il est fabriqué par Universal Robot. Il est destiné aux applications industrielles de taille moyenne et offre une flexibilité optimale. Sa portée maximale avec une charge utile maximale de 5 kg est de 850 mm. Il est conçu selon la norme EN ISO 10218-1 de 2006, ce qui lui permet de collaborer avec les opérateurs. Sa structure cinématique est généralement subdivisée en quatre parties : la base, le bras supérieur, le bras inférieur et le poignet. La base est reliée au bras supérieur par les deux premières articulations, et

le bras supérieur est relié au bras inférieur par la troisième articulation. Le poignet est formé des articulations numéros 4, 5 et 6.

Contrairement aux RMIs traditionnels, le système de commande de l'UR5 est doté de plusieurs mécanismes de sécurité intégrés, dont le dispositif d'arrêt de protection qui se déclenche en cas d'application d'une force externe de 150 N sur le robot. Les paramètres cinématiques et dynamiques de l'UR5 sont indiqués respectivement aux Tableau 3.3 et Tableau 3.4.



FIGURE 2.8 – Robot UR5.

TABLE 2.3 – Paramètre de Denavit Hartenberg de l'UR5

Segment n°	$\theta[n^\circ]$	a [mm]	d [mm]	$\alpha[n^\circ]$
1	q_1	0	89.16	$+90^\circ$
2	q_2	452	0	0
3	q_3	392.25	0	0
4	q_4	0	109.15	$+90^\circ$
5	q_5	0	94.56	-90°
6	q_6	-	82.3	0

TABLE 2.4 – Paramètres d'inertie de l'UR5

Segment n°	Position du centre de masse [mm]	Masse [kg]
1	[0, - 25.61, 1.93]	3.7
2	[212.5, 0, 113.36]	8.393
3	[212.5, 0, 113.36]	2.33
4	[0, - 1.8, 16.34]	1.219
5	[0, 1.8, 16.34]	1.219
6	[0, 0, - 1.159]	0.1879

2.5 Conclusion

Dans ce chapitre, nous avons d'abord élaboré deux lois de commande pour les RMIA-CII évoluant dans des environnements déterministes et aléatoires. Nous avons ensuite présenté les différents robots sur lesquels nous effectuerons des expériences. Nous avons proposé une commande capable de garantir le suivi de trajectoire d'un RMI lorsque l'architecture de commande interne est inconnue et inaccessible, ainsi qu'une commande capable de rejeter les perturbations aléatoires. Au chapitre 3, nous effectuerons des tests et des simulations afin d'évaluer leur efficacité par rapport à celles proposées dans la littérature existante.

Résultats et discussions

3.1 Introduction

Dans le chapitre précédent, nous avons formulé les lois de commande pour les RMIs à architecture de commande interne inconnue et inaccessible, opérant dans des contextes déterministes et aléatoires respectivement. Dans celui-ci, nous présentons les résultats des simulations et des expériences effectuées. Nous les comparons à ceux de la littérature pour mettre en évidence la pertinence de nos travaux de recherche.

3.2 Commande dans l'espace des tâches d'un robot manipulateur industriel à architecture de commande interne inconnue et inaccessible évoluant en environnement déterministe

L'objectif de cette section est de démontrer l'efficacité de la stratégie de commande proposée dans la section 2.2 par le biais de simulations et d'expériences effectuées sur le Scorbot ER-4U et l'UR5. Pour cela, nous avons utilisé le logiciel MATLAB avec un pas d'échantillonnage de 0.001 seconde. La position cartésienne de l'effecteur du robot est obtenue grâce à une fonction intégrée dans la bibliothèque MATLAB. La trajectoire de référence est définie par (3.1) et le modèle RBF NN a 100 nœuds (i.e., $\iota = 100$), avec des centres uniformément espacés de $[-3.14, 3.14] \times [-20, 20]$ et une largeur $\eta = 2$.

$$x_{des} = \left[0.2 + 0.1 \cos\left(\frac{\pi}{12.5}t\right), 0.2 + 0.1 \sin\left(\frac{\pi}{12.5}t\right), 0.4 + 0.2 \sin\left(\frac{\pi}{12.5}t\right) \right]^T. \quad (3.1)$$

3.2.1 Résultats des simulations effectuées sur le modèle numérique du Scrobot ER-4U

Pour montrer la pertinence de nos travaux, nous avons effectué une étude comparative avec les travaux de Wang et al. [11] qui s'appuient sur l'Hypothèse H2 stipulant que la structure du contrôleur interne est connue, mais non modifiable. Nous démontrons que nos travaux offrent de meilleures performances de suivi sans poser cette hypothèse. Dans le cadre de cette étude, nous considérons que le contrôleur interne du Scrobot ER-4U est un régulateur PI avec un mécanisme de compensation de la gravité défini par (3.2) :

$$u = g(q) - \bar{K}_p(q - q_c) - \bar{K}_d(\dot{q} - \dot{q}_c), \quad (3.2)$$

avec $\bar{K}_p = 40I_{3 \times 3}$ et $\bar{K}_d = 20I_{3 \times 3}$.

L'algorithme développé dans les travaux de Wang et al. [11] est décrit par les Équations (3.3) et (3.4) suivantes

$$\dot{q}_c + \hat{\kappa}q_c = \dot{q}_r + \hat{\kappa}q_r + \text{diag}[\hat{w}]Y_d(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\hat{a}_d, \quad (3.3)$$

avec

$$\begin{aligned} \dot{q}_r &= J^T(q) \left[J(q)J^T(q) \right]^{-1} \dot{x}_d - \gamma J^T(q) (x - x_d), \\ \dot{\hat{w}} &= -\Lambda \text{diag} [Y_d(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\hat{a}_d] s, \\ \dot{\hat{w}}_I &= \Lambda_I \text{diag} [q - q_r] s, \\ \dot{\hat{a}}_d &= -\Gamma_d Y_d^T(q, \dot{q}, \dot{q}_r, \ddot{q}_r) s, \\ \hat{\kappa} &= \text{diag} [\hat{w}_I]. \end{aligned} \quad (3.4)$$

Les paramètres de conception et les valeurs initiales de l'algorithme de Wang et al. [11] sont : $\gamma = \text{diag} ([50, 50, 100])$, $\Gamma_d = I_{17}$, $\Lambda = 0.01I_3$, $\Lambda_I = 0.2I_3$, $\hat{a}_d(0) = 0_{17}$ et $\hat{w}_I(0) = 2 \times \mathbf{1}_{3 \times 1}$.

Nous considérons que les paramètres cinématiques sont connus avec précision. Nous utilisons la loi de commande (2.22) et les lois d'adaptation (2.26) , (2.27) et (2.28). Toutefois, $\hat{J}^\dagger(q)$ est remplacé par $J^\dagger(q)$. Les paramètres de conception sont $k = 2$, $\bar{\varepsilon}_1 = \bar{\varepsilon}_2 = \bar{\varepsilon}_3 = 10$, $\varpi_i = 0.001$, $\gamma_d = \gamma_{W_i} = 0.001$, $\Gamma_d = 10$, $\Gamma_{W_i} = 5$, et $\alpha = \text{diag} ([50, 50, 100])$. Les valeurs initiales choisies sont $q(0) = [0.6, 0.4, 0.2]^T$, $\dot{q}(0) = \mathbf{0}_{3 \times 1}$, $\hat{\theta}_d(0) = 0.04$ et $\hat{W}(0) = 10I_{3 \times 3}$.

A. Présentation des résultats des simulations

Les résultats des simulations sont présentés dans la Figure 3.1. On y observe les erreurs

cartésiennes de suivi de la trajectoire. On remarque que le robot suit avec succès la trajectoire souhaitée. Les deux contrôleurs présentent toutefois des erreurs de suivi élevées pendant la phase transitoire (c'est-à-dire $t \leq 0,05$ seconde). La valeur maximale de l'erreur de suivi est de $|\tilde{x}(t)| \leq 0.04m$. Cependant, comme le montrent les courbes en zoom, le contrôleur proposé présente un meilleur temps de réponse. En régime permanent, le contrôleur proposé présente une erreur maximale de $8 \times 10^{-4}m$, alors que le contrôleur de Wang et al. [11] présente une erreur maximale de $4 \times 10^{-3}m$.

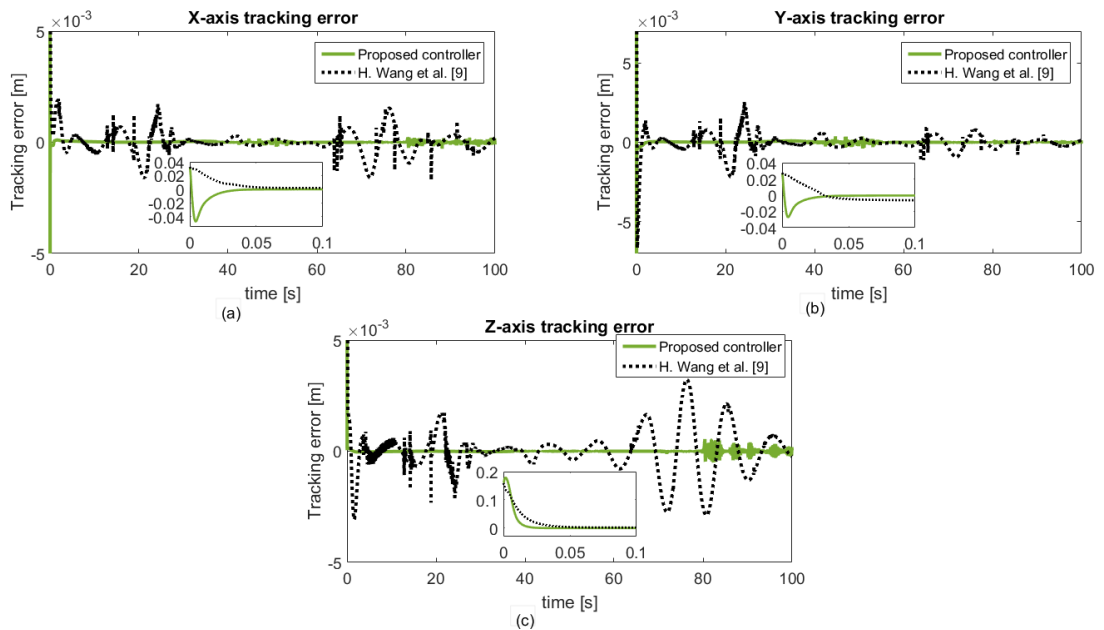


FIGURE 3.1 – Résultats des simulations : (a) Erreurs de suivi sur l'axe X, (b) Erreurs de suivi sur l'axe Y et (c) Erreurs de suivi sur l'axe Z.

B. Analyse et interprétation des résultats des simulations

Les deux contrôleurs présentent de faibles performances en régime transitoire, car aucun d'eux ne possède d'étage de robustesse spécifique durant cette phase. Par contre, en régime permanent, le contrôleur proposé présente un gain de performance de 20% par rapport au contrôleur de Wang et al. [11]. Ce gain de performance est dû aux deux principales caractéristiques du contrôleur proposé :

- il utilise moins de paramètres adaptatifs, ce qui signifie une réduction des potentielles erreurs résultantes des valeurs initiales erronées ;
- le contrôleur proposé est capable de respecter les performances prescrites par l'utilisateur, grâce à la fonction $\zeta(x)$.

Le contrôleur proposé présente non seulement de meilleures performances en suivi de trajectoire, mais il génère aussi moins de fluctuations en régime permanent. Ces perfor-

mances sont obtenues sans qu'il soit nécessaire de poser une hypothèse simplificatrice concernant la structure du contrôleur interne du robot. Les résultats satisfaisants obtenus lors de la simulation sont attestés par une série d'expériences. Nous présentons les résultats de ces expériences dans la section suivante.

3.2.2 Résultats des expériences effectuées sur le robot Scrobot ER-4U

Les expériences effectuées visent à prouver que la stratégie de commande proposée dans nos travaux offre de meilleurs résultats que ceux rencontrés dans la littérature. Pour des raisons de simplicité et de comparaison, nous ne modifions pas l'orientation de l'effecteur. Pour cela, l'angle de roulis du poignet q_5 du Scrobot ER-4U est inchangé (c'est-à-dire, $q_5(0) = 0$ rad) par conséquent, son signal de commande sera toujours nul, $\dot{q}_{c5}(t) = 0$ rad/s.

Nous commencerons par effectuer une série d'expériences pour choisir les meilleurs paramètres de conception. Par la suite, nous procéderons à une étude comparative entre nos résultats et ceux de la commande adaptative basée sur un observateur, développée par Wang et al. [11]. Les paramètres de l'algorithme de Wang et al. [11] sont identiques à ceux de la Section 3.2.1.

A. Analyse de l'influence des paramètres de conception sur les performances de l'algorithme de commande

Il est important de choisir de meilleurs paramètres de la loi de commande pour garantir un meilleur suivi de trajectoire. En effet, comme cela a été observé dans la Section 2.3.2, l'algorithme de commande proposé utilise plusieurs paramètres. Ceux-ci influencent considérablement ses performances. Nous allons pour cela effectuer plusieurs expériences afin de déterminer les bons paramètres de conception.

Les paramètres qui influencent les performances de l'algorithme de commande en termes de diminution du temps de réponse et de réduction des erreurs cartésiennes sont ϖ , k , et α . Plus précisément, ϖ et k améliorent les performances pendant la phase transitoire, tandis que α permet de réduire les erreurs de suivi en régime permanent. Différentes valeurs de ϖ_i , k et α_{ii} , $i = \{1, 2, 3\}$ sont utilisées pour atteindre les objectifs fixés. Les autres paramètres restent inchangés pendant les tests : $\bar{\varepsilon}_i = 5$, $\Gamma_d = 10$, $\gamma_d = \gamma_J = \gamma_{W_i} = 0,001$ et $\Gamma_J = \Gamma_{W_i} = 3$ avec $i = \{1, \dots, 5\}$. Les valeurs initiales sont $q(0) = [0.6, 0.4, 0.2, 0, 0]^T$,

$$\dot{q}(0) = \mathbf{0}_{5 \times 1}, \hat{\theta}_d(0) = 0.04, \hat{\theta}_J(0) = 0.5 \times \mathbf{1}_{4 \times 1}, \text{ et } \hat{W}(0) = \text{diag}(10 \times \mathbf{1}_{5 \times 1}).$$

A.1. Analyse de l'influence de ϖ

Afin de choisir le bon paramètre ϖ , nous attribuons des valeurs fixes à k et α ($k = 1$ et $\alpha = \text{diag}([2, 2, 2])$) et faisons varier celui de ϖ entre $\{10^{-3}, 0.5, 1\}$. Nous choisirons une de ces trois valeurs. Les Figures 3.2 et 3.3 représentent les résultats de la première série d'expériences. La Figure 3.2 présente les erreurs cartésiennes de la position de l'effecteur. Le Tableau 3.5 résume les valeurs maximales, minimales (de l'ordre de 10^{-6}) et la convergence observée sur chaque axe suivant les différentes valeurs de ϖ . La Figure 3.3 montre les signaux de commande $\dot{q}_{cj}(t)$, $j = 1, 2, 3, 4$ et le Tableau 3.6 indique les amplitudes maximales et minimales des différents signaux de commande.

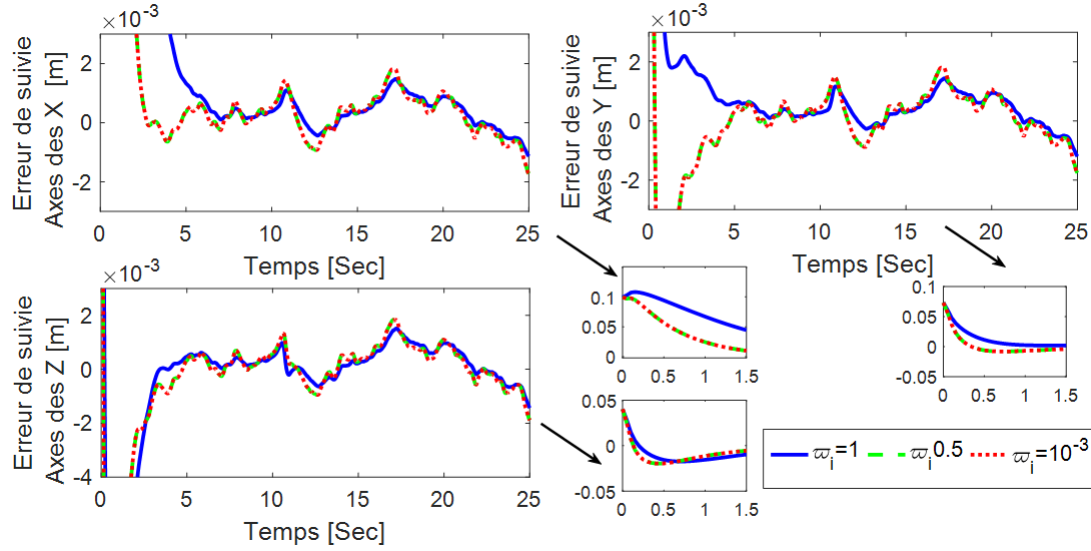


FIGURE 3.2 – Erreurs de suivi sur les axes X,Y,Z pour $\varpi_i = \{10^{-3}, 0.5, 1\}$, $i = 1, 2, 3$.

TABLE 3.1 – Valeurs Max,Min de l'erreur de suivi sur les axes X,Y,Z pour $\varpi_i = \{10^{-3}, 0.5, 1\}$, $i = 1, 2, 3$

Erreur de suivi	$\varpi = 10^{-3}$			$\varpi = 0.5$			$\varpi = 1$		
	Max	Min	Réponse	Max	Min	Réponse	Max	Min	Réponse
X	99.32	-1.72	R	99.32	-1.72	R	108	-1.134	L
Y	73.19	-8.06	R	73.19	-8.05	R	73.19	-1.20	L
Z	40.17	-19.65	L	40.17	-19.65	L	40.17	-17.35	R

D'après la Figure 3.2 et le Tableau 3.5, nous obtenons le meilleur temps de réponse lorsque $\varpi_i = 10^{-3}$ et $\varpi_i = 0.5$. Ce temps est compris entre $0 \leq t \leq 1.5$ seconde (selon les axes). On constate que les erreurs de suivi sont relativement identiques pour les trois valeurs de ϖ_i . L'erreur de suivi maximale est l'ordre de $|\tilde{x}_i(t)| \leq 2 \times 10^{-3}m$. L'erreur de suivi la plus faible est obtenue pour $\varpi_i = 10^{-3}$. Nous pouvons donc conclure que de

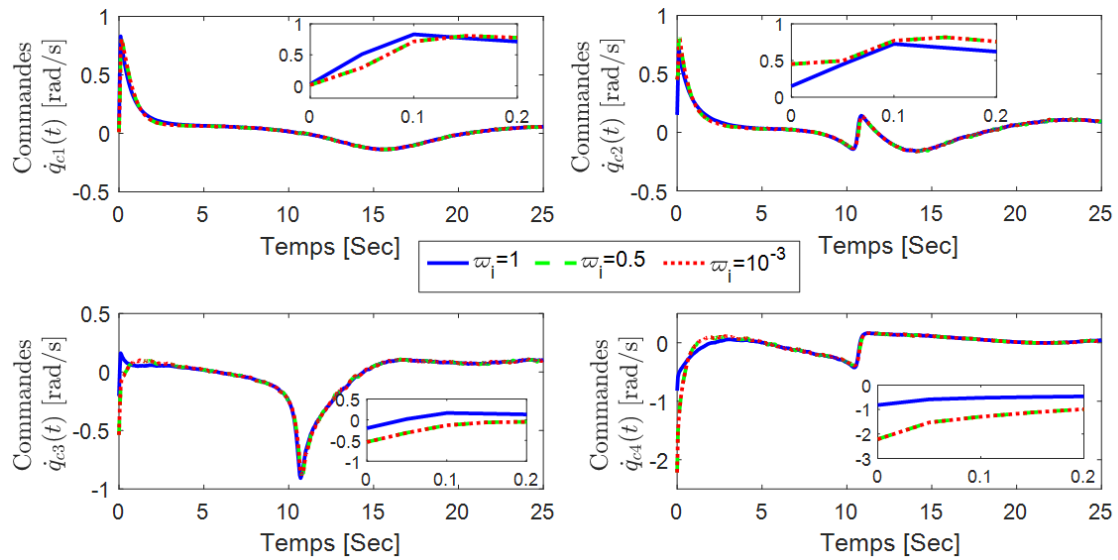


FIGURE 3.3 – Signaux de commande des vitesses \dot{q}_{cj} , $j = \{1, 2, 3, 4\}$, $\varpi_i = \{10^{-3}, 0.5, 1\}$, $i = 1, 2, 3$.

TABLE 3.2 – Commandes de vitesse \dot{q}_{cj}

Signaux de commande	$\varpi = 10^{-3}$		$\varpi = 0.5$		$\varpi = 1$	
	Max	Min	Max	Min	Max	Min
\dot{q}_{c1}	0.805	-0.1367	0.8051	-0.136	0.8305	-0.1367
\dot{q}_{c2}	0.8169	-0.1634	0.8166	-0.1634	0.7261	-0.1599
\dot{q}_{c3}	0.1105	-0.8818	0.1618	-0.9064	0.1105	-0.8817
\dot{q}_{c4}	0.167	-2.218	0.8641	-2.212	0.1638	-0.817

meilleurs résultats sont obtenus lorsque $\varpi_i = 10^{-3}$. De plus, on constate effectivement que ϖ_i n'affecte pas de manière significative les performances du contrôleur pendant le régime permanent.

la Figure 3.3 et le Tableau 3.6 montrent que les signaux de commande $\dot{q}_{cj}(t)$ sont identiques pendant le régime permanent, mais l'amplitude la plus élevée est obtenue pour $\varpi_i = 10^{-3}$ et $\varpi_i = 0,5$ pendant phase transitoire. La Figure 3.2 permet d'observer que les signaux de commande éliminent les perturbations durant la phase transitoire lorsque l'amplitude du vecteur de contrainte $\zeta(x)$ augmente. Néanmoins, il est essentiel de réduire les valeurs de k et α pour éviter la saturation des actionneurs.

A.2. Analyse de l'influence de k

La deuxième série d'expériences permet de choisir la meilleur valeur du paramètre k . Pour cela, on fixe les paramètres α et ϖ_i et l'on fait varier le paramètre k . Conformément aux résultats de la première série, $\varpi_i = 10^{-3}$ et nous fixons $\alpha = \text{diag}([2, 2, 2])$. Nous allons faire varier k entre $\{1, 2, 3\}$.

Les Figures 3.4 et 3.5 présentent respectivement les erreurs cartésiennes de suivi de

trajectoire et les signaux de commande des vitesses. Les données pertinentes des différentes figures sont contenus respectivement dans les Tableaux 3.7 et 3.8. Ces données sont respectivement de l'ordre de 10^{-6} et de 10^{-3} rad/s .

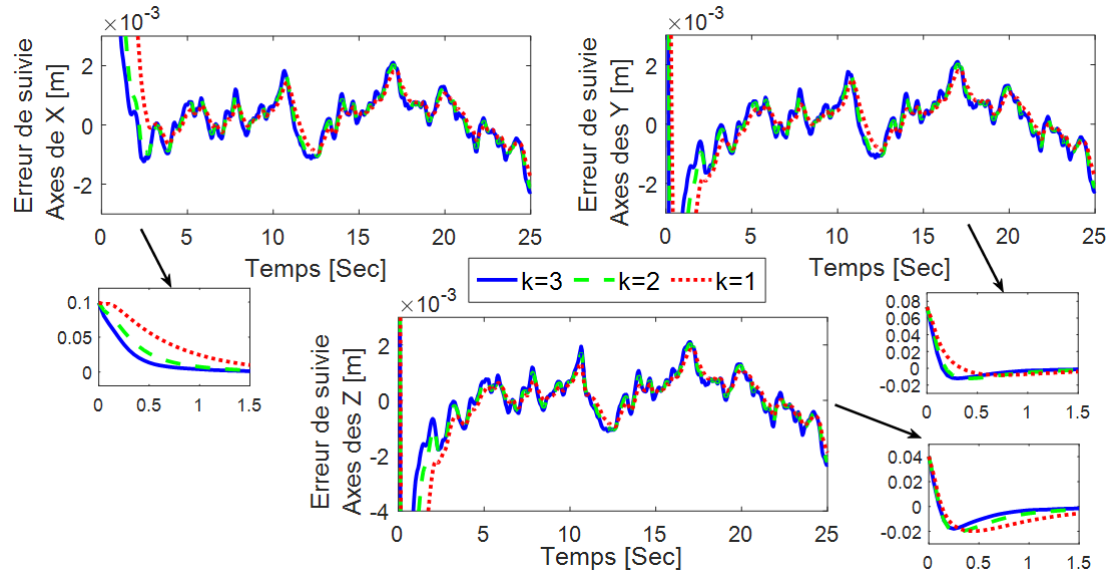


FIGURE 3.4 – Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $k = \{1, 2, 3\}$

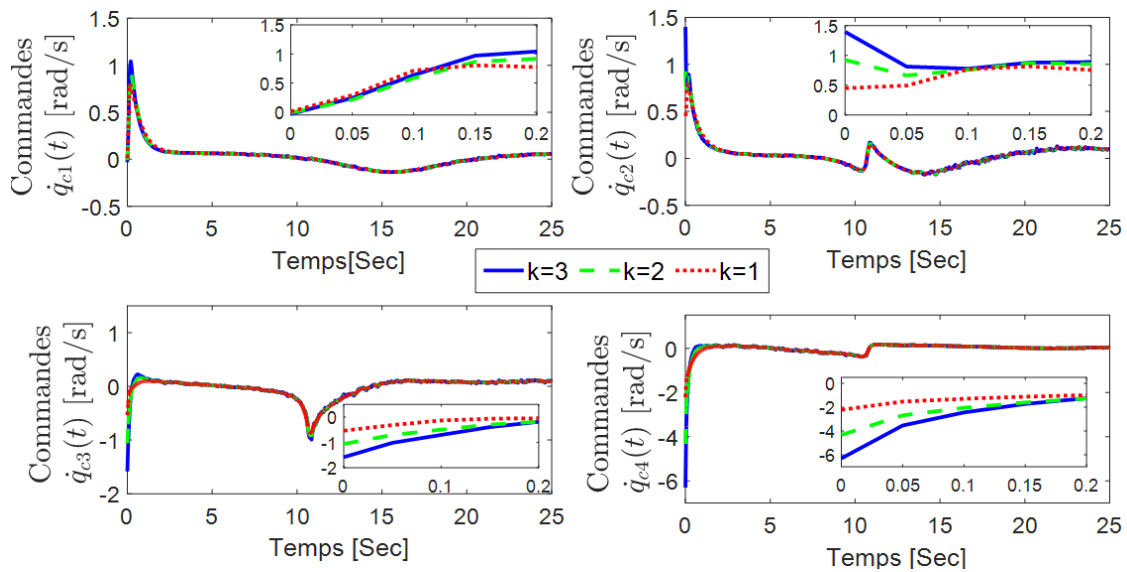


FIGURE 3.5 – Résultats expérimentaux pour $k = \{1, 2, 3\}$, commandes de vitesse \dot{q}_{cj} , $j = \{1, 2, 3, 4\}$

À partir de la Figure 3.4 et du Tableau 3.7, on observe que le système présente un meilleur temps de réponse lorsque $k = 3$. On constate également que les erreurs de suivi présentent le même dépassement (amplitudes maximales) en régime permanent, pour les trois valeurs de k . Cela indique que ce paramètre n'affecte pas les performances de suivi en régime permanent. D'après la Figure 3.5 et le Tableau 3.8, on constate que les amplitudes

TABLE 3.3 – Valeurs Max,Min de l'erreur de suivi sur les axes X,Y,Z pour $k = 1, 2, 3$

Erreur de suivi	$k = 1$			$k = 2$			$k = 3$		
	Max	Min	Réponse	Max	Min	Réponse	Max	Min	Réponse
X	99.32	-1.72	R	99.32	-2.105	R	99.32	-2.273	R
Y	73.19	-8.06	R	73.19	-12.04	R	73.19	-12.02	R
Z	40.17	-19.65	L	40.17	-20.3	L	40.17	-17.05	R

 TABLE 3.4 – Commandes de vitesse \dot{q}_{cj} pour $k = 1, 2, 3$.

Signaux de commande	$k = 1$		$k = 2$		$k = 3$	
	Max	Min	Max	Min	Max	Min
\dot{q}_{c1}	805	-136.7	918	-136.6	1024	-136.7
\dot{q}_{c2}	816.9	-163.4	926.4	-170.1	1398	-175.8
\dot{q}_{c3}	110.5	-881.8	166.8	-1063	221.8	-1585
\dot{q}_{c4}	164	-2218	167.7	-4326	170.9	-6297

des signaux de commande sont plus élevées lorsque $k = 3$ pendant la phase transitoire. Cela signifie que pour cette valeur de k , les signaux de commande peuvent hors de la plage de vitesse des actionneurs. Ces résultats nous permettent de confirmer que le vecteur de contraintes $\zeta(x)$ ne modifie pas les performances de suivi en régime permanent, mais qu'il améliore les performances durant la phase transitoire. Le paramètre $k = 1$ est donc le meilleur choix, car son erreur de suivi et l'amplitude des signaux de commande sont relativement faibles.

A.3. Choix du l'influence de α

L'objectif de cette série d'expérience est de choisir la meilleure valeur du paramètre α . Comme dans les expériences précédentes qui ont permis le choix de k et ϖ , les paramètres k et ϖ_i sont fixes et α va varier entre $\{1, 2, 4\}$. Conformément aux résultats obtenus précédemment, $k = 1$ et $\varpi_i = 10^{-3}$.

Les Figures 3.6 et 3.7 présentent respectivement les erreurs cartésiennes de position de l'effecteur et les signaux de commandes en vitesses. Les données pertinentes des différentes Figures sont contenues respectivement dans les Tableaux 3.9 (de l'ordre de 10^{-6}) et 3.10 (de l'ordre de 10^{-3} rad/s .)

 TABLE 3.5 – Valeurs Max/Min de l'erreur de suivi sur les axes X,Y,Z pour $\alpha_{ii} = \{1, 2, 4\}$.

Erreur de suivi	$\alpha_{ii} = 1$			$\alpha_{ii} = 2$			$\alpha_{ii} = 4$		
	Max	Min	Réponse	Max	Min	Réponse	Max	Min	Réponse
X	99.32	-2.10	R	99.32	-1.72	L	105	-1.31	R
Y	73.19	-12.04	L	73.19	-8.061	R	73.19	-1.61	L
Z	40.17	-20.3	R	40.17	-19.65	R	40.17	-17.73	R

La Figure 3.6 et le Tableau 3.9 montrent qu'on obtient un meilleur temps de réponse

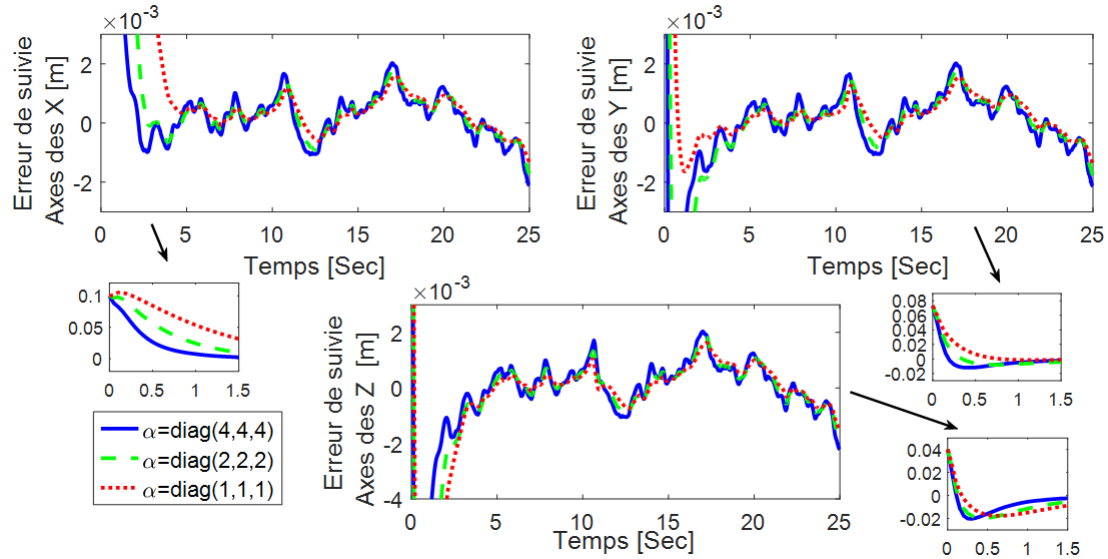


FIGURE 3.6 – Erreurs de suivi sur l’axe X, l’axe Y et l’axe Z pour $\alpha_{ii} = \{1, 2, 4\}$.

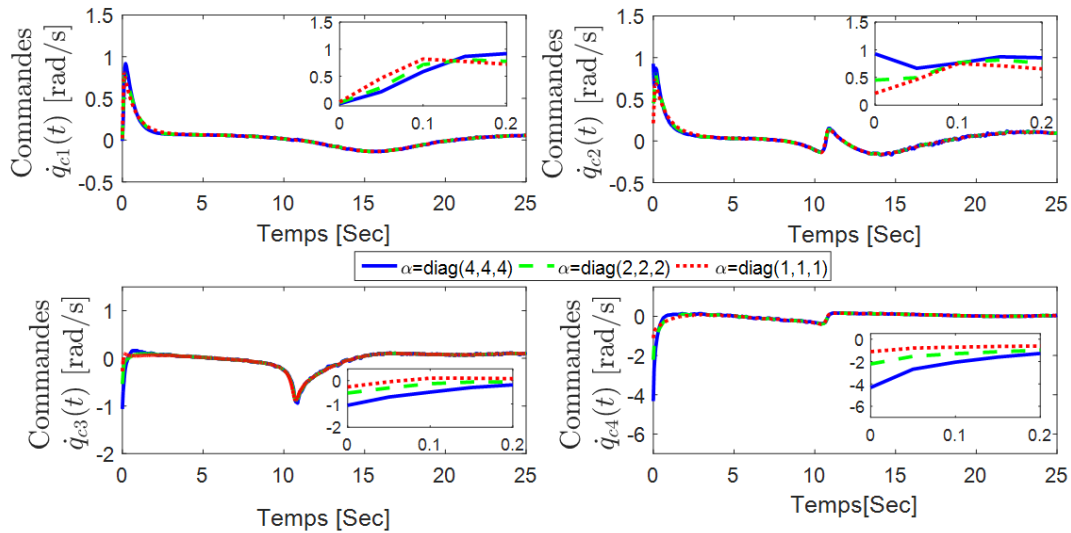


FIGURE 3.7 – Commandes de vitesse \dot{q}_{c_j} , $j = \{1, 2, 3, 4\}$ pour $\alpha_{ii} = \{1, 2, 4\}$.

lorsque $\alpha_{ii} = 4$. Cependant, la Figure 3.7 montre que pour cette valeur de α_{ii} , les signaux de commande des vitesses sont élevés. Ces résultats montrent qu’une valeur élevée de α_{ii} permet d’obtenir de bonnes performances pendant la phase transitoire et une précision acceptable pendant le régime permanent, mais peut entraîner la saturation des actionneurs. Cependant, il est bien connu qu’il n’est pas possible d’obtenir le compromis requis entre un signal de commande de faible amplitude, la rapidité, la stabilité et la précision en variant un seul paramètre. D’après les Figures 3.6 et 3.7, on observe que pour $k = 1$, $\alpha = \text{diag}([1, 1, 1])$, et $\varpi_i = 10^{-3}$, le contrôleur atteint les meilleures performances de suivi. En effet, pour ces valeurs, le temps de réponse est supérieur à 3,7 secondes. La plus petite valeur de l’erreur de suivi est alors $|\tilde{x}_i(t)| \leq 17 \times 10^{-4}m$, tandis que celle de la commande est $|\dot{q}_{c_j}(t)| \leq 1,2rad/s$.

TABLE 3.6 – Valeurs Max/Min des signaux de Commandes de vitesse \dot{q}_{cj} pour $\alpha_{ii} = \{1, 2, 4\}$.

Signaux de commande	$\alpha_{ii} = 1$		$\alpha_{ii} = 2$		$\alpha_{ii} = 4$	
	Max	Min	Max	Min	Max	Min
\dot{q}_{c1}	918	-136.6	805	-136.7	818	-136.7
\dot{q}_{c2}	926.4	-170.1	816.9	-163.4	751.9	-160.4
\dot{q}_{c3}	166.8	-1063	1105.5	-881.8	106.5	-885.2
\dot{q}_{c4}	167.7	-4326	164	-2218	162.9	-1113

les trois séries expériences montrent qu'un compromis doit être trouvé entre le temps de réponse souhaité, la précision nécessaire et les caractéristiques des actionneurs (vitesse maximale, fréquence, etc.). De plus, ces expériences montrent que :

- Plus ϖ_i est petit, meilleur est le temps de réponse ;
- Plus α_{ii} et k sont grands, plus les signaux de commande ont une amplitude maximale ce qui peut provoquer la saturation des actionneurs.

Les paramètres de conception étant choisis, nous allons dans la sous-section suivante, présenter les résultats des expériences obtenues et faire une étude comparative.

B. Étude comparative

Dans cette section, nous présentons les résultats de la comparaison des performances du contrôleur proposé et celui de Wang et al. [11].

Nous choisissons $\varpi_i = 0,001$, $k = 1$, et $\alpha = \text{diag}([1, 1, 1])$. Pour le contrôleur de Wang et al. [11] les paramètres de conception sont les suivants : $\gamma = \text{diag}([5, 5, 10])$, $\Gamma_d = I_{54}$, $\Lambda = 0.01I_5$, $\Lambda_I = 0.2I_5$, $\hat{a}_d(0) = 0_{54}$ et $\hat{w}_I(0) = 2 \times \mathbf{1}_{5 \times 1}$.

Les résultats obtenus sont décrits dans les Figures 3.8, 3.9 et 3.10. Elles représentent respectivement le suivi de trajectoire, les erreurs cartésiennes de position de l'effecteur et les signaux de commandes de vitesse de chaque actionneur.

Le Tableau 3.7 donne les valeurs maximales et minimales de l'erreur cartésienne et le temps de réponse. Ces données sont de l'ordre de 10^{-6} pour les valeurs maximales et minimales et 10^{-3} pour le temps de réponse. Le Tableau 3.8 indique les valeurs maximales et minimales des signaux de commandes.

Les résultats contenus dans le Tableau 3.11 nous indiquent que l'algorithme de commande proposé présente des erreurs de suivi faibles sur les axes X et Y. Ces erreurs sont respectivement de 1.315×10^{-6} et 1.642×10^{-6} . Tandis que pour l'approche de Wang et al. [11], c'est sur l'axe Z que l'on a l'erreur de suivi la plus faible qui est de l'ordre de 10.45×10^{-6} . En outre, le temps de réponse est relativement faible sur les axes X et Y

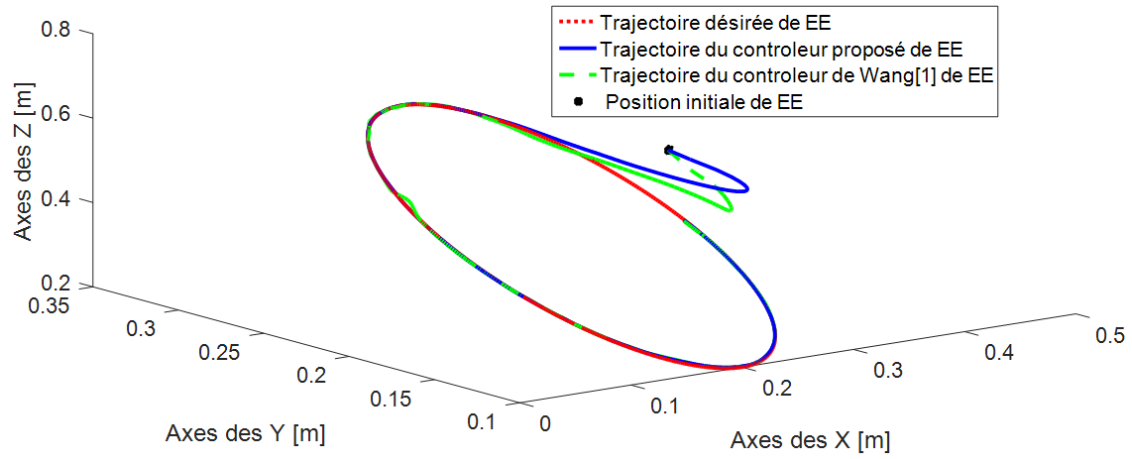


FIGURE 3.8 – Résultat de l'expérience : Suivi de trajectoire de l'effecteur.

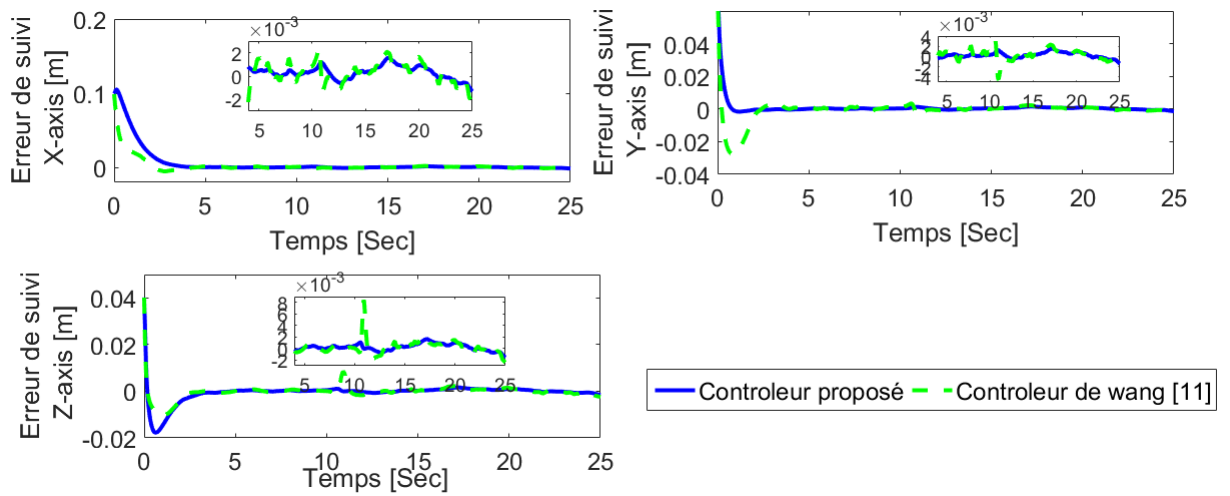


FIGURE 3.9 – Résultat de l'expérience, Erreurs de suivi sur chaque axe.

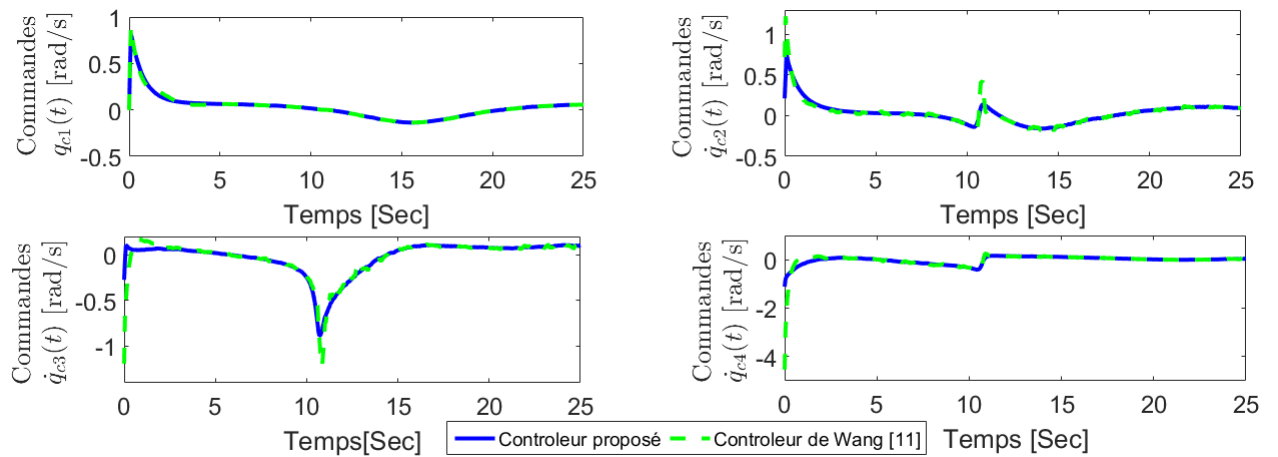


FIGURE 3.10 – Résultat de l'expérience, Signaux de commande.

pour l'algorithme proposé alors que pour celui de Wang et al. [11], on observe un temps de réponse faible sur l'axe de Z. En plus des erreurs de suivi et le temps réponse, les amplitudes des signaux de commande sont plus faibles dans l'algorithme proposé par rapport à celui de Wang et al. [11] (Voir Tableau 3.12). Cela signifie que le contrôleur proposé

TABLE 3.7 – Valeurs Max/Min de l’erreur de suivi sur les axes X,Y,Z et temps de réponse des différents contrôleurs.

Erreur de suivi	Contrôleur Proposé			Contrôleur de Wang [11]		
	Max	Min	Tr	Max	Min	Tr
X	105	-1.315	3.7	99.32	-5.20	4.25
Y	73.19	-1.642	2.4	73.19	-27.58	4.3
Z	40.17	-17.73	3.25	40.17	-10.45	3

TABLE 3.8 – Valeurs Max/Min des commandes de vitesse

Erreur de suivi	Contrôleur Proposé		Contrôleur de Wang [11]	
	Max	Min	Max	Min
\dot{q}_{c1}	818.3	-136.7	894	-136
\dot{q}_{c2}	755.19	-160.4	816.9	-185.9
\dot{q}_{c3}	106.5	-885.2	116.6	-25.53
\dot{q}_{c4}	162.9	-1113	252.8	4528

est moins exposé aux phénomènes de saturation des actionneurs. Les Figures 3.8, 3.9 et 3.10 attestent les observations faites précédemment. En effet, pour l’algorithme proposé par Wang et al. [11], la valeur maximale des signaux de commande pendant la phase transitoire est de $4.5rad/s$, tandis qu’elle est égale à $1.1rad/s$ pour l’algorithme proposé. L’erreur de suivi est $|\tilde{x}_i(t)| \leq 0,1m$ pendant la phase transitoire et $|\tilde{x}_i(t)| \leq 0,008m$ pendant le régime permanent pour l’algorithme de Wang et al. [11] alors que pour l’algorithme proposé, les erreurs de suivi sont $|\tilde{x}_i(t)| \leq 0,1m$ pendant la phase transitoire et $|\tilde{x}_i(t)| \leq 0,002m$ pendant le régime permanent.

D’après les résultats obtenus, nous pouvons conclure que l’approche de commande développée dans cette thèse qui résout la problématique liée à la commande des RMIs à architecture de commande interne inconnu et inaccessible, présente de meilleures performances en suivi de trajectoire, rapidité, stabilité et précision. En outre, elle est simple, facile à implémenter par rapport à celles rencontrées dans la littérature. Elle n’utilise pas de régresseur dynamique ou de capteur (ou observateur) de vitesse et ne dépend pas de la structure du contrôleur interne. Dans la section suivante, nous allons présenter les résultats des expériences effectuées sur l’UR5.

3.2.3 Résultats des expériences effectuées sur le robot UR5

Dans cette section, nous allons effectuer des expériences sur le robot UR5. L’objectif de cette étude est de démontrer que quelque soit la structure du contrôleur interne, l’algorithme est capable d’assurer de bonnes performances. Pour cette étude, nous choisissons

les trajectoires désirées suivantes :

$$\begin{aligned} x_d &= [x_0 + c_1 - c_1 \cos \omega_0 t, y_0 - c_2 \sin(\omega_0 t), z_0 - c_3 \sin^{12}(\omega_0 t)], \\ \dot{x}_d &= [c_1 \omega_0 \sin \omega_0 t, -c_2 \omega_0 \cos(\omega_0 t), z_0 - 12\omega_0 c_3 \cos(\omega_0 t) \sin^{11}(\omega_0 t)], \end{aligned} \quad (3.5)$$

avec $x_0 = -0,222$, $y_0 = -0,1$, $z_0 = 0,4$ les positions initiales, $\omega_0 = \pi/25$ la pulsation et $c_1 = 0.5$, $c_2 = 0.5$, $c_3 = -0,2$ les rayons.

Les paramètres du contrôleur sont : $\varepsilon_0 = 10$, $a_0 = 8 \text{ diag}([1; 1; 1])$, $a_1 = 0,2$; $a_2 = 0,02$; $\lambda_2 = 0.02$, $\beta = 0,4$, $\gamma_i = 0$ et $\lambda_{4i} = 1 \times 10^{-6}$ pour $i = 1 \dots 6$, $K_d = 10 \text{ diag}[1; 1; 1; 1; 1; 1]$, $\theta = 0,2$, $K_d = 10 \text{ diag}[1; 1; 1; 1; 1; 1]$, $\varpi_j = 0,001$ pour $j = 1 \dots 3$, $\dot{\Upsilon} = \text{zero}[6, 6]$.

A. Présentation des résultats

Les performances en suivi de trajectoire sont représentées dans les Figures 3.11 et 3.13. Elles décrivent respectivement le suivi de trajectoire et les erreurs cartésienne de suivi. Les signaux de commande de chaque actionneur sont représentés par la Figure 3.12.

Le Tableau 3.9 donne les valeurs maximales et minimales de l'erreur de suivi exprimées en mètre. Le Tableau 3.10 indique les valeurs maximales et minimales des signaux de commandes *rad/s*.

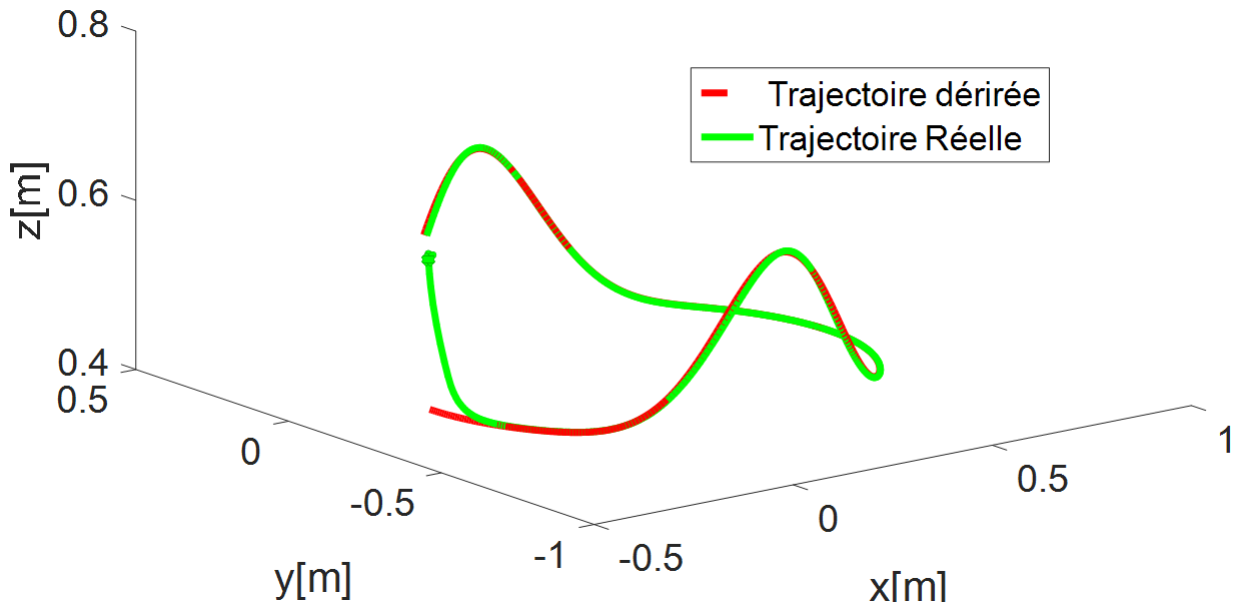


FIGURE 3.11 – Résultat de l'expérience effectué sur l'UR5 : Suivi de trajectoire

Ces résultats montrent que l'effecteur du robot converge rapidement vers la trajectoire de référence et la suit parfaitement. Les erreurs de suivi sur chaque axe sont très faibles. Les amplitudes des signaux de commande sont également faibles. Cela garantit la protection

des actionneurs.

TABLE 3.9 – Valeurs Max/Min des erreurs de suivi sur les axes

Erreur de suivi sur les axes	Valeurs Max	Valeurs Min
e_x	$3,64 \times 10^{-4}$	-0,154
e_y	0,00069	-0,1984
e_z	0,2166	$-9,64 \times 10^{-4}$

TABLE 3.10 – Valeurs Max/Min des angles et commandes de vitesse

Articulation	Position des angles		Commande de vitesse	
	Max	Min	Max	Min
1	-2,416	-5,9965	0.1752	-0.382
2	-0,8134	-1,9587	0.0912	-0.8633
3	1,9769	-0,7665	1.42	-0.1798
4	-0,8531	-1,4205	0.6848	-0.0213
5	-0,8233	-1,7103	0.0787	-0.1901
6	-0,7836	-0,7837	3.8999×10^{-4}	-3.8644×10^{-4}

B. Analyse et interprétation des résultats

On observe une erreur de suivi importante sur l'axe Z durant la phase transitoire, et faible sur les axes X et Y . Cependant, on constate que cette erreur converge très rapidement. Ceci est dû à la fonction de contrainte de performance qui entre en action lorsque l'on sort de la plage de performance prescrite. Cette convergence exponentielle et globale nous permet d'obtenir un temps de réponse de l'ordre de $t_r = 2,36s$. En régime permanent, nous obtenons également des erreurs de suivi relativement faibles (Figure 3.13). Ceci résulte du choix judicieux des paramètres de la loi de commande, mais aussi du nombre réduit de paramètres du réseau de neurones, qui permet aux lois d'adaptation d'avoir un temps de calcul faible.

Les résultats des simulations et des expériences effectuées sur les robots Scrobot et UR5 témoignent de la capacité de l'approche de commande proposée à assurer la stabilité SGUUB du système bouclé, quelle que soit la structure du contrôleur interne. Ils montrent également que, dans un environnement déterministe, l'approche de commande permet de résoudre les problèmes liés à la structure fermée et inaccessible du contrôleur interne. La section suivante sera consacrée à l'évaluation de l'approche de commande proposée pour les problématiques de contournement de la structure fermée et d'inaccessibilité du contrôleur interne, ainsi que pour le rejet des perturbations aléatoires lorsque le robot évolue dans un environnement non déterministe.

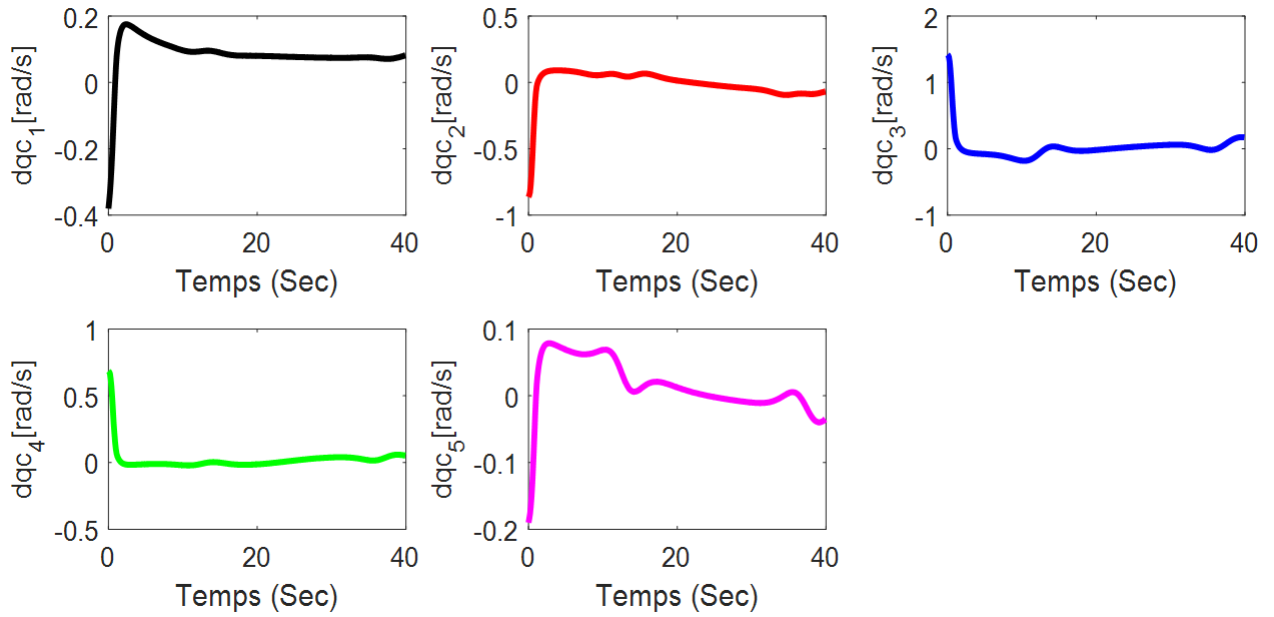


FIGURE 3.12 – Résultat de l'expérience effectuée sur l'UR5 : Vitesse de commande.

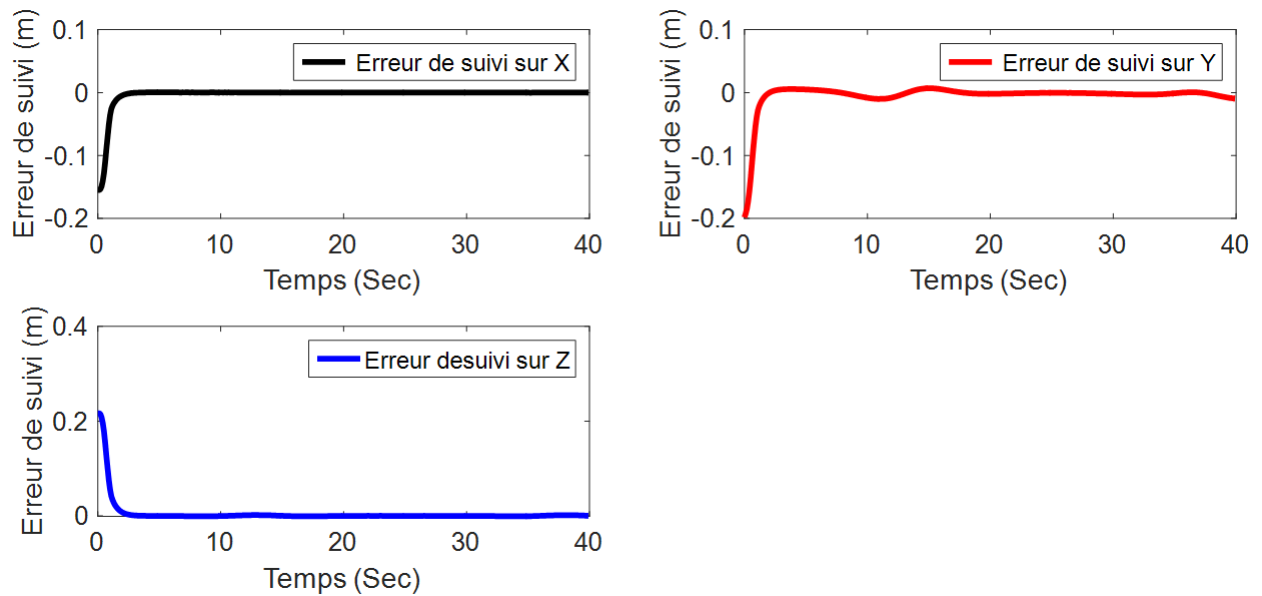


FIGURE 3.13 – Résultat de l'expérience effectuée sur l'UR5 : Erreur de suivi

3.3 Commande neuro adaptative dans l'espace de tâche des RMIACIIIs évoluant dans un environnement aléatoire

Afin de démontrer la supériorité de l'algorithme proposé dans la Section 2.3, nous allons le comparer à l'algorithme de Cui et al. [14] basé sur l'observateur de vitesse. Les tests ont été effectués sur les modèles numériques des robots SCORBOT ER-4U et UR5. La trajectoire cartésienne désirée et le RBFNN sont identiques à ceux définis dans la 3.2.

3.3.1 Résultats des simulations effectuées sur le Scorbot ER-4U

Nous ferons également deux séries de simulation. La première série vise à choisir des bons paramètres et la seconde série montre la supériorité de loi de commande proposée. Pour des nécessités d'étude comparative, on suppose que le contrôleur interne du robot est un contrôleur de vitesse PI défini par (3.6).

$$\tau = -K_d(\dot{q} - \dot{q}_c) - K_i(q - q_c), \quad (3.6)$$

avec $K_d = K_i = \text{diag}(10 \times \mathbf{1}_{5 \times 1})$.

Les vibrations aléatoires ξ_1 , ξ_2 et ξ_3 (Voir Figure 3.14) sont définies comme dans [14]

$$b_i \dot{\xi}_i(t) = -\xi_i(t) + w_i(t), \quad \xi_i(0) = 0, \quad i = 1, 2, 3, \quad (3.7)$$

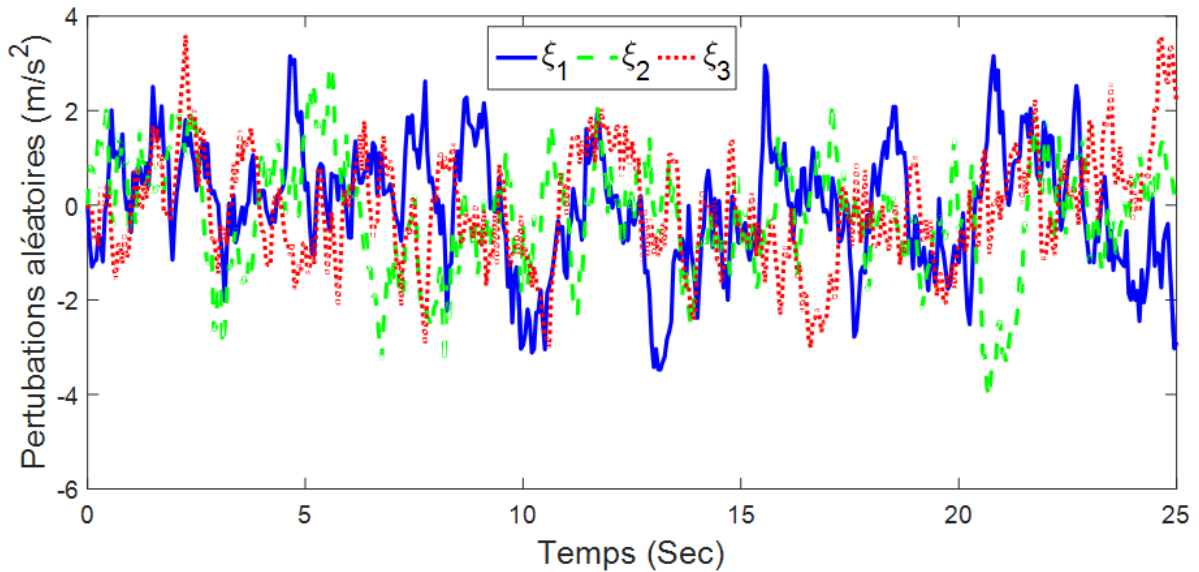


FIGURE 3.14 – Perturbations aléatoires ξ_1 , ξ_2 , et ξ_3 .

avec, $b_i > 0$, $w_i(t)$ un bruit blanc à valeur moyenne nulle de puissance A_i et t_c est la période d'échantillonnage. La valeur moyenne quadratique du processus stationnaire ξ_i est $\mathbb{E}\|\xi_i(t)\|^2 = \frac{A_i}{\pi b_i} \arctan\left(\frac{\pi b_i}{50 t_c}\right)$. Pour cette étude, nous choisissons $b_i = 0,5$, $t_c = 0,05$ et $A_i = 2$, ce qui donne $\mathbb{E}\|\xi_i(t)\|^2 = 0,7143$. En propageant les accélérations aléatoires ξ_1 , ξ_2 et ξ_3 comme démontré à la section 2.3, sur le robot Scorbot ER-4U, nous obtenons la matrice d'excitation des dynamiques aléatoires suivante (2.101) :

$$\begin{aligned}
\Gamma_{21}(q) &= -\cos(q_1) (-m_3 l_{c3} \cos(q_2) \sin(q_3) + m_3 l_3 \cos(q_2) \sin(q_3) + \\
&\quad m_3 l_2 \sin(q_3) \cos(q_3) \cos(q_2) + m_2 l_{c2} \sin(q_2) - m_2 l_2 \sin(q_2)) \\
\Gamma_{22}(q) &= -\sin(q_1) (-m_3 l_{c3} \cos(q_2) \sin(q_3) + m_3 l_3 \cos(q_2) \sin(q_3) + \\
&\quad m_3 l_2 \sin(q_3) \cos(q_3) \cos(q_2) + m_2 l_{c2} \sin(q_2) - m_2 l_2 \sin(q_2)) \\
\Gamma_{23}(q) &= -m_2 \cos(q_2) l_2 + m_2 l_{c2} \cos(q_2) + m_3 l_{c3} \sin(q_2) \sin(q_3) - \\
&\quad m_3 l_3 \sin(q_2) \sin(q_3) - m_3 l_2 \sin(q_3) \cos(q_3) \sin(q_2)
\end{aligned} \tag{3.8}$$

$$\begin{aligned}
\Gamma_{31}(q) &= -m_3 (l_3 - l_{c3}) \cos(q_1) \cos(q_2) \sin(q_3) \\
\Gamma_{32}(q) &= -m_3 (l_3 - l_{c3}) \sin(q_1) \cos(q_2) \sin(q_3) \\
\Gamma_{33}(q) &= -m_3 (l_3 - l_{c3}) \sin(q_2) \sin(q_3),
\end{aligned} \tag{3.9}$$

$$\begin{aligned}
\Gamma_{11}(q) &= -m_1 l_{c1} \sin(q_1) + m_1 l_1 \sin(q_1) \\
&\quad + m_4 l_3 \cos(q_1) \cos(q_2)^2 \cos(q_3)^2 \sin(q_4) \\
&\quad - m_4 l_3 \cos(q_1) \cos(q_2) \cos(q_3) \sin(q_4) \sin(q_2) \sin(q_3) \\
&\quad + m_4 l_2 \cos(q_1) \cos(q_2)^2 \cos(q_3) \sin(q_4) \\
&\quad + m_4 l_1 \cos(q_1) \cos(q_2) \cos(q_3) \sin(q_4) \\
\Gamma_{12}(q) &= m_1 l_{c1} \cos(q_1) - m_1 l_1 \cos(q_1) \\
&\quad + m_4 l_3 \sin(q_1) \cos(q_2)^2 \cos(q_3)^2 \sin(q_4) \\
&\quad - m_4 l_3 \cos(q_2) \cos(q_3) \sin(q_4) \sin(q_2) \sin(q_3) \sin(q_1) \\
&\quad + m_4 l_2 \sin(q_1) \cos(q_2)^2 \cos(q_3) \sin(q_4) \\
&\quad + m_4 l_1 \cos(q_2) \cos(q_3) \sin(q_4) \sin(q_1) \\
\Gamma_{13}(q) &= m_4 \sin(q_4) \sin(q_2) \cos(q_3) (l_3 \cos(q_2) \cos(q_3) \\
&\quad - l_3 \sin(q_2) \sin(q_3) + l_2 \cos(q_2) + l_1),
\end{aligned} \tag{3.10}$$

avec $\Gamma_{i1} = \Gamma_{i2} = \Gamma_{i3} = 0$, pour $i = 4, 5$. À partir de (3.10), (3.8), (3.9) et (2.102), le régresseur $Q(q, z) \in \mathbb{R}^{5 \times 28}$, et le vecteur des paramètres cinématiques $\Theta \in \mathbb{R}^{28}$ se calcule aisément.

A. Analyse de l'influence des paramètres de commande sur les performances de l'algorithme de commande

Les paramètres importants qui influencent et améliorent les performances de suivi de trajectoire de l'algorithme développé (Voir Figure 2.5) sont

- ϖ_j , améliore les performances de suivi durant la phase transitoire.
- ϱ_2 , améliore les performances de suivi durant le régime permanent.

Nous allons effectuer une série d'expérience visant à analyser leur influence sur les

performances de l'algorithme de commande. Les autres paramètres sont choisis tel que :

$\varepsilon_i = 5$, $\lambda_2 = 10$, $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 0.01$, $\varrho_0 = \text{diag}(\mathbf{1}_{3 \times 1})$, $\varrho_1 = 1$ et $\lambda_1 = \lambda_{4_i} = 3$, $i = 1, 2, \dots, 5$. Les valeurs initiales sont prises comme suit $q(0) = [0, 6, 0, 4, 0, 2, 0, 0]^T$, $\dot{q}(0) = \mathbf{0}_{5 \times 1}$, $\hat{\beta}(0) = 0,04$, $\hat{\theta}(0) = 0.5 \times \mathbf{1}_{4 \times 1}$, $\Theta(0) = 0.4 \times \mathbf{1}_{28 \times 1}$ et $\hat{\Upsilon}(0) = \text{diag}(10 \times \mathbf{1}_{5 \times 1})$.

A.1. Analyse de l'influence de ϱ_2

L'objectif de cette Section est de choisir le paramètre ϱ_2 . Pour cela, nous fixons $\varpi_j = 0,001$, $j = 1, 2, 3$, tandis que trois valeurs différentes pour ϱ_2 sont utilisées, à savoir $\varrho_2 = \{0.01, 0.5, 1\}$.

Les Figures 3.15 et 3.16 représentent respectivement les erreurs de suivi sur chaque axes et les commandes en vitesse des différentes articulations. Les valeurs maximales et minimales des erreurs de suivi ainsi que les commandes de vitesse sont indiquées respectivement dans les Tableaux 3.15 et 3.16. Elles sont de l'ordre de 10^{-3} .

TABLE 3.11 – Valeurs de l'erreur de suivi sur les axes pour $\varrho_2 = \{0.01, 0.5, 1\}$.

Erreur de suivi	$\varrho_2 = 0.01$			$\varrho_2 = 0.5$			$\varrho_2 = 1$		
	Max	Min	Réponse	Max	Min	Réponse	Max	Min	Réponse
X	99.32	-620×10^{-5}	R	101.8	-1.255	L	105	-1.381	L
Y	73.19	-566×10^{-5}	R	73.19	-0.877	L	73.19	-0.754	L
Z	40.17	-3.979	R	40.17	-15.52	L	40.17	-17.05	L

TABLE 3.12 – Commande de vitesse \dot{q}_{cj} pour $\varrho_2 = \{0.01, 0.5, 1\}$.

Signaux de commande	$\varrho_2 = 0.01$		$\varrho_2 = 0.5$		$\varrho_2 = 1$	
	Max	Min	Max	Min	Max	Min
\dot{q}_{c1}	1106	-138.7	896.3	-136.1	864.1	-137.9
\dot{q}_{c2}	1091	-185.9	787.3	-168	763.5	-165.9
\dot{q}_{c3}	129	-719.6	112.3	-855.6	113.3	-885.2
\dot{q}_{c4}	139.2	-1113	151.7	-1113	153.7	-1113

Nous observons dans la Figure 3.15 que pendant la phase transitoire (c'est-à-dire $t \leq 2 \text{Sec}$), le temps de réponse est faible pour $\varrho_2 = 1$ et élevé pour $\varrho_2 = 0,01$. L'erreur de suivi la plus important est obtenu pour $\varrho_2 = 1$. Durant le régime permanent, on observe que $|\tilde{x}_j(t)| \leq 2 \times 10^{-3}m$ pour $\varrho_2 = 1$, et $|\tilde{x}_j(t)| \leq 2 \times 10^{-4}m$ pour $\varrho_2 = 0.01$.

Dans la Figure 3.16, on observe que pendant la phase transitoire, la valeur la plus élevée des signaux de commande (soit $|\dot{q}_{ci}(t)| = 1,1 \text{rad/s}$) est obtenue pour $\varrho_2 = 0.01$. Ces observations indiquent que les performances de suivi sont meilleures lorsque ϱ_2 est

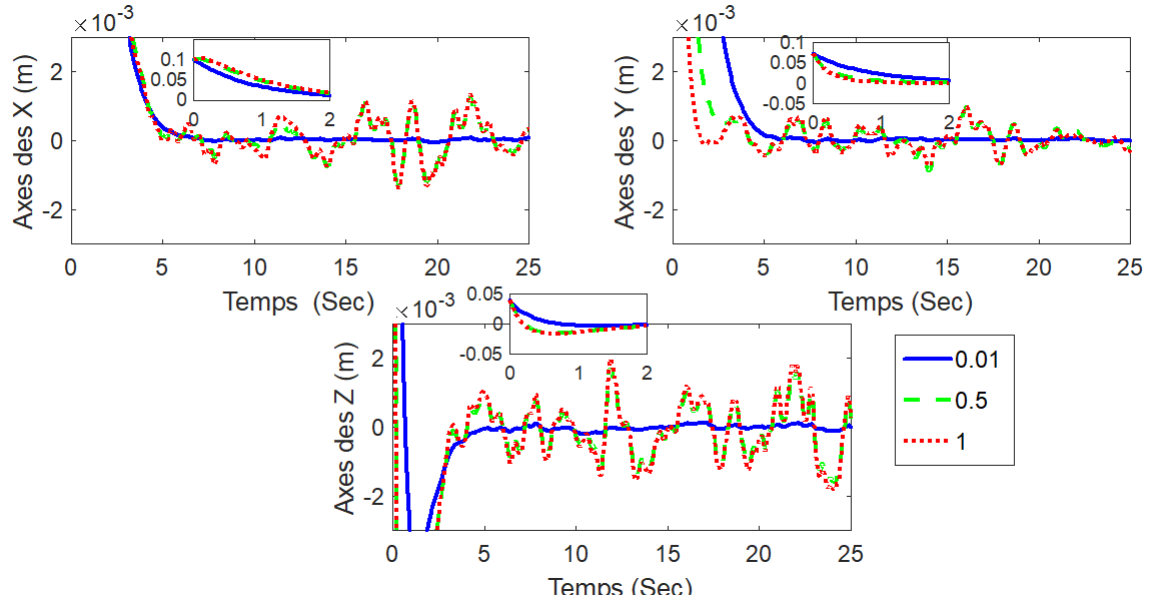


FIGURE 3.15 – Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $\varrho_2 = \{0.01, 0.5, 1\}$.

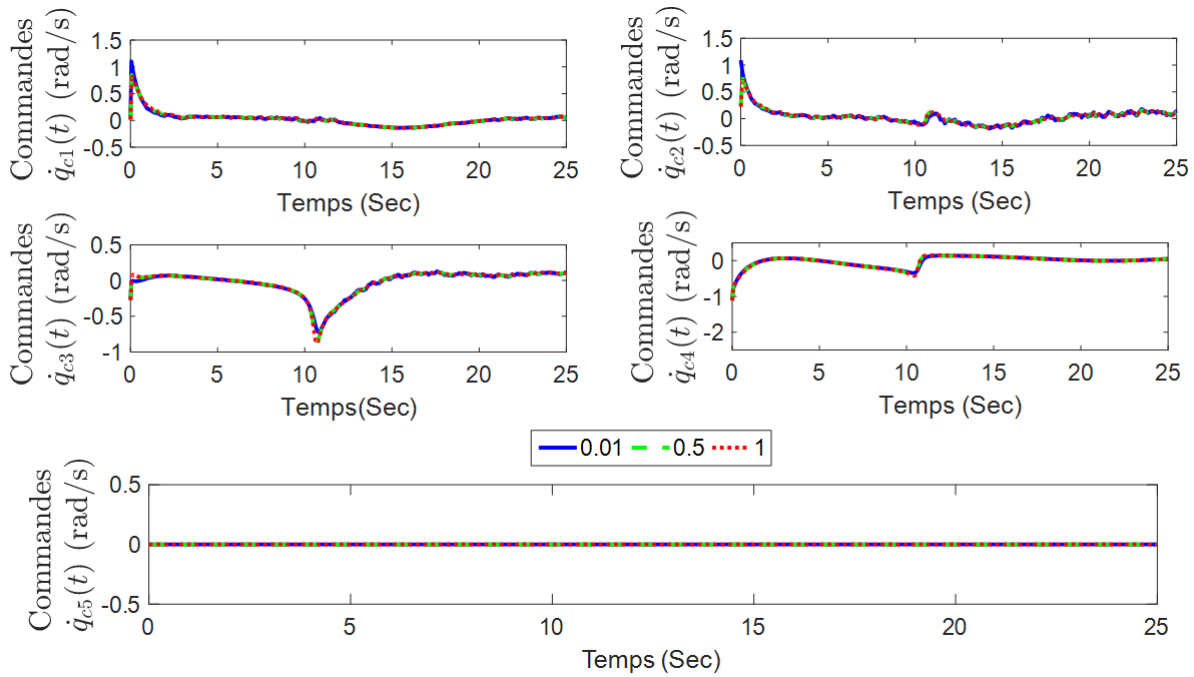


FIGURE 3.16 – Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$ pour $\varrho_2 = \{0.01, 0.5, 1\}$, Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$.

faible. Cependant, une faible valeur de ϱ_2 induit des commandes en vitesse importantes pendant la phase de transition. Ces résultats sont compatibles avec les attentes. Car, d'après (2.122), une faible valeur de ϱ_2 entraîne une faible valeur de $\mathbb{E} \|\tilde{x}(t)\|^2$. Tandis que d'après (2.114), une faible valeur de ϱ_2 peut entraîner des signaux de commande élevés. Par conséquent, ϱ_2 doit être choisi en fonction de la vitesse maximale admissible de l'actionneur.

A.2. Analyse de l'influence de ϖ_j

Les expériences entreprises dans cette partie permettent de choisir le paramètre ϖ_j . Pour cela, nous fixons à $\varrho_2 = 0,5$ et choisissons trois valeurs différentes de ϖ_j à savoir, $\varpi_j = \{10^{-3}, 0,2, 2\}$. Nous allons également observer l'erreur de suivi et les signaux de commandes afin de déterminer le bon paramètre.

Les Figures 3.17 et 3.18 représentent respectivement comme dans le cas précédents les erreurs de suivi et les signaux de commande. Les Tableaux 3.17 et 3.18 donnent les amplitudes maximales et minimales.

TABLE 3.13 – Valeurs de l'erreur de suivi sur les axes X,Y,Z pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3$.

Erreur de suivi	$\varpi_j = 0.001$			$\varpi_j = 0.2$			$\varpi_j = 2$		
	Max	Min	Réponse	Max	Min	Réponse	Max	Min	Réponse
X	101.8	-1.2555	R	102.2	-1.256	R	124	3.64	L
Y	73.19	-0.876	R	73.19	0.883	R	73.19	3.82	L
Z	40.17	-15.52	R	40.17	-15.49	R	40.17	-16.39	L

On observe que l'erreur de suivi la plus faible (soit $|\tilde{x}_j(t)| \leq 1,2 \times 10^{-3}m$) est obtenue pour $\varpi_j = 0,001$.

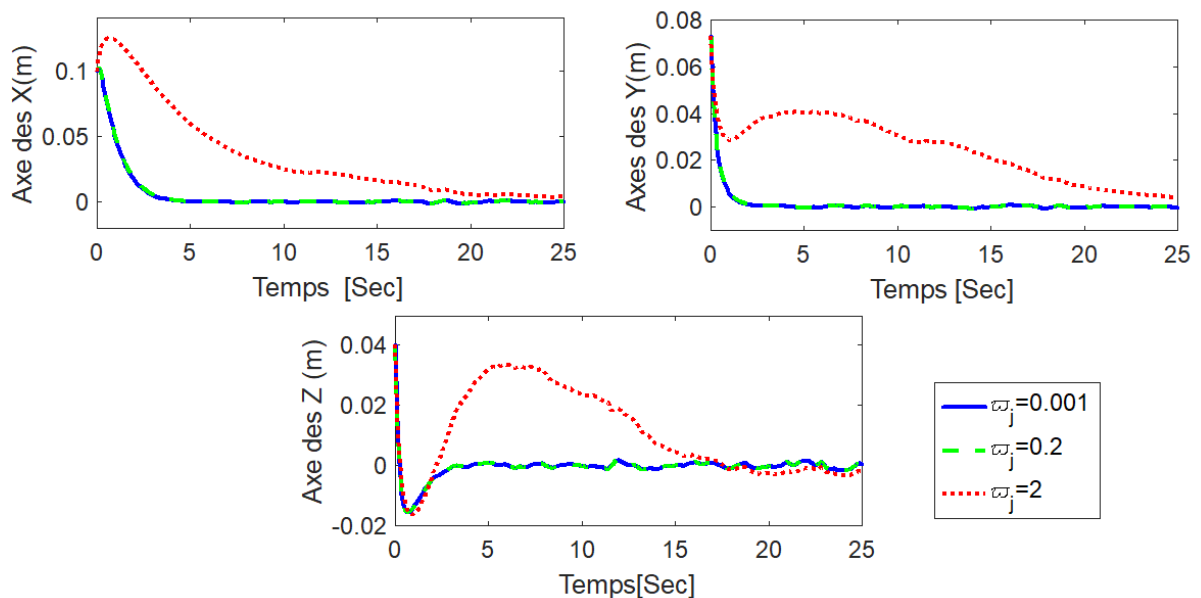


FIGURE 3.17 – Erreurs de suivi sur l'axe X, l'axe Y et l'axe Z pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3$.

D'après la Figure 3.18 et le Tableau 3.18 les signaux de commande sont plus élevés pour $\varpi_j = 2$ en régime permanent, mais le signal de commande le plus élevé est observé

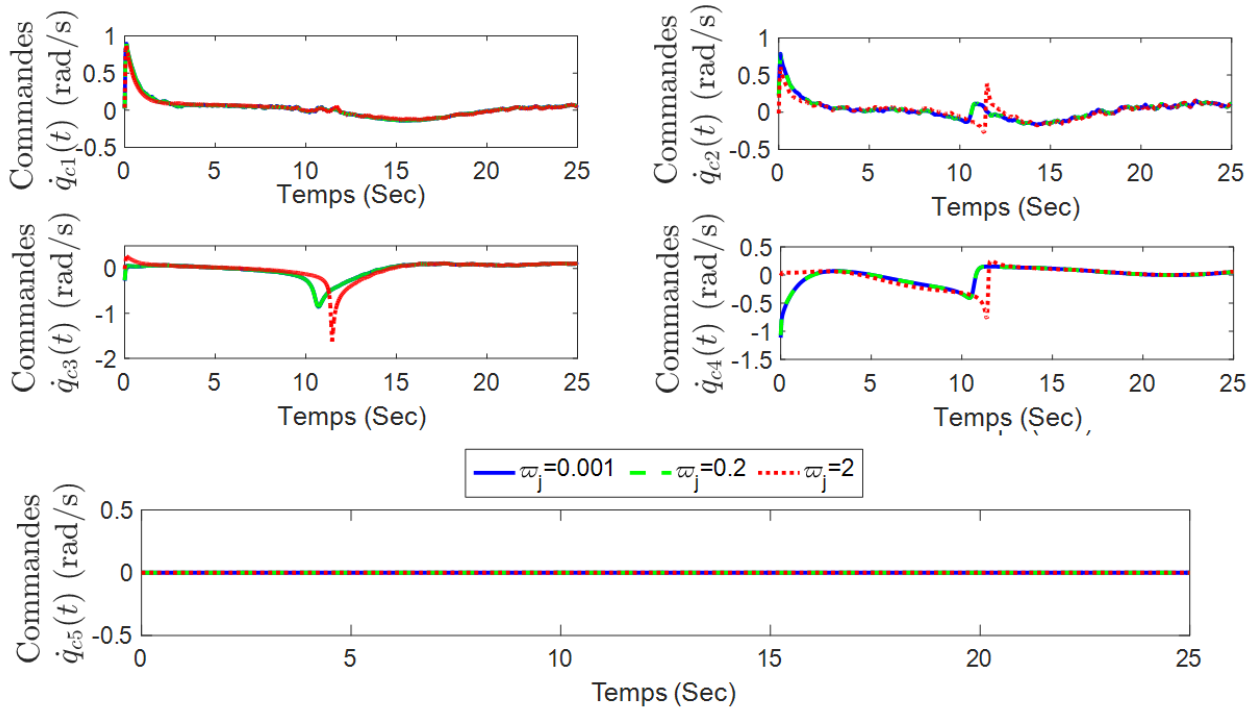


FIGURE 3.18 – Résultats de simulation pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3$; Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$.

TABLE 3.14 – Commandes de vitesse \dot{q}_{cj} pour $\varpi_j = \{0.001, 0.2, 2\}$, $j = 1, 2, 3, 4$

Signaux de commande	$\varpi_j = 0.001$		$\varpi_j = 0.2$		$\varpi_j = 2$	
	Max	Min	Max	Min	Max	Min
\dot{q}_{c1}	896.3	-138.1	895	-138	870.4	-127.4
\dot{q}_{c2}	787.3	-168	782.1	-168	617.1	-267.7
\dot{q}_{c3}	112.3	-855.6	112.3	-858.7	276.6	-1626
\dot{q}_{c4}	151.7	-1113	151.8	-1069	258.1	-765.9

pour $\varpi_j = 0,001$ durant la phase transitoire. Ces résultats montrent qu’une faible valeur de ϖ_j conduit à de bonnes performances en régime permanent et transitoire avec un signal de commande important pendant la phase transitoire. Ces résultats confirment l’efficacité du rejet des perturbations réalisé par $\psi(\tilde{x})$. En effet, lorsque la contrainte $\tilde{x}_j^2 \leq \varpi_j^2$ n’est pas respectée (principalement rencontrée en phase transitoire), $\psi(\tilde{x})$ augmente le gain de contrôle $\varrho_1\varrho_{0j}$ pour améliorer le rejet des perturbations. Cependant, ϖ_j doit être choisi en fonction de la vitesse maximale admissible de l’actionneur.

Pour tous les tests qui vont suivre, nous adoptons les paramètres de conceptions suivants $\varpi_j = 0.001$, $j = 1, 2, 3$, et $\varrho_2 = 0.01$.

B. Étude comparative

Dans cette sous-section, nous allons comparer nos résultats avec ceux de Cui et al. [14]. Les paramètres de l’algorithme de Cui et al [14] sont définis tels-que : $\gamma = \text{diag}([5, 5, 10])$,

$$\Gamma_d = I_{54}, \Lambda = 0.01I_5, \Lambda_I = 0.2I_5, \hat{a}_d(0) = 0_{54} \text{ et } \hat{w}_I(0) = 2 \times \mathbf{1}_{5 \times 1}.$$

La Figure 3.20 présente le suivi de trajectoire de l'effecteur des différents algorithmes (l'algorithme proposé et celui de Cui et al. [14]). La Figure 3.19 montrent les erreurs de suivi cartésiennes et la Figure 3.21 montrent les signaux de commande en vitesses des actionneurs. Les données pertinentes de ces différentes Figures sont contenues dans les Tableaux 3.19 et 3.20. Nous observons que l'algorithme de commande proposé a le temps de réponse le plus petit ($t_r \leq 4.5 \text{Sec}$) et l'erreur de suivi de trajectoire la plus faible ($|\tilde{x}_j(t)| \leq 1,5 \times 10^{-4} m$). Pour l'algorithme de Cui et al. [14], l'erreur de suivi est $|\tilde{x}_j(t)| \leq 2 \times 10^{-3} m$ et le temps de réponse de $t_r = 11.75 \text{ sec}$

TABLE 3.15 – Valeurs Max/Min de l'erreur de suivi sur les axes X,Y,Z et temps de réponse des différents contrôleurs

Erreur de suivi	Contrôleur Proposé			Contrôleur de Cui [49]		
	Max	Min	Tr	Max	Min	Tr
X	99.32	-620.1×10^{-5}	3.5	99.32	-5.628	4.95
Y	73.19	-566.5×10^{-5}	3.4	73.19	-28.31	4.45
Z	40.17	-3.979	4.45	40.17	-10.47	11.75

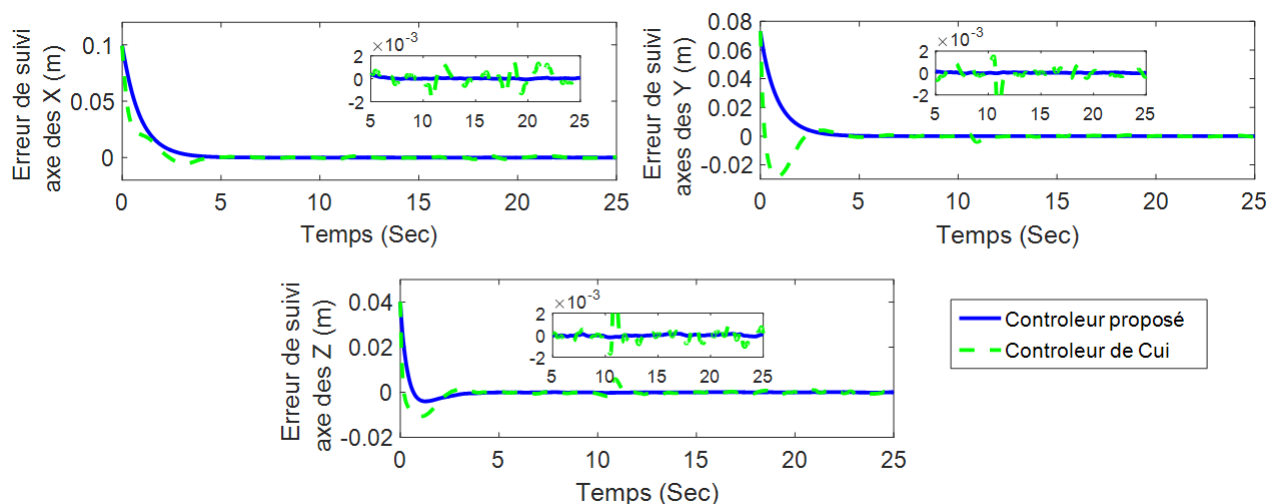


FIGURE 3.19 – Erreurs de suivi sur chaque axe.

C. Interprétations des résultats

Le Tableau 3.20 et la Figure 3.21 nous permettent de confirmer que l'algorithme de commande développé offre de meilleurs signaux de commande (amplitude relativement faible) par rapport à celui de Cui et al. ([14]). Cela signifie que notre approche de commande permet aux actionneurs de consommer moins d'énergie et d'éviter le problème de

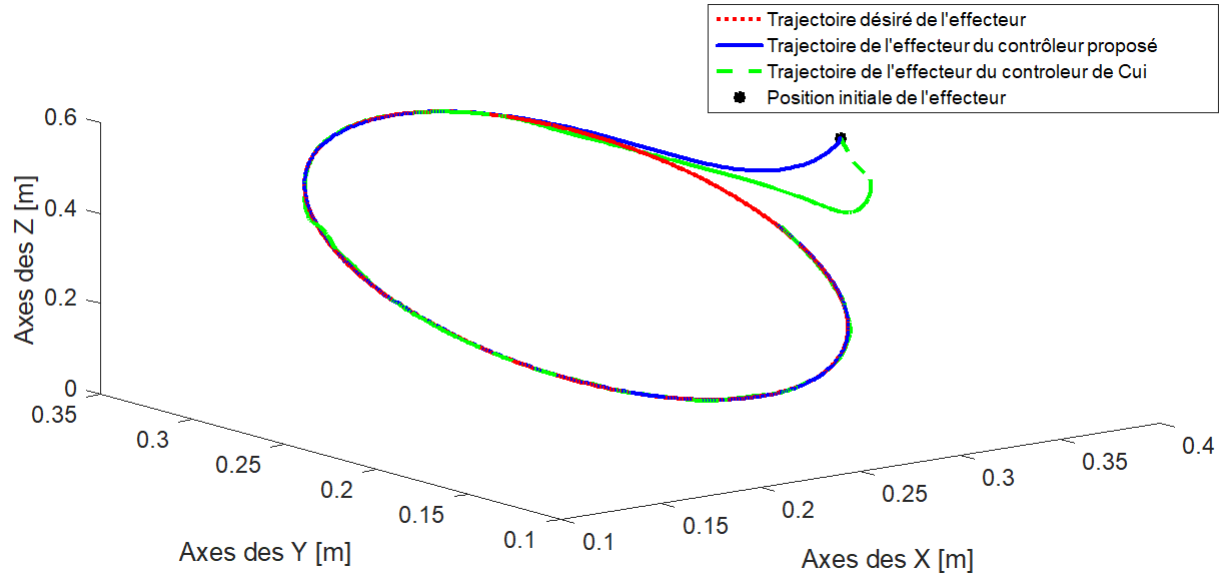


FIGURE 3.20 – Suivi de Trajectoire.

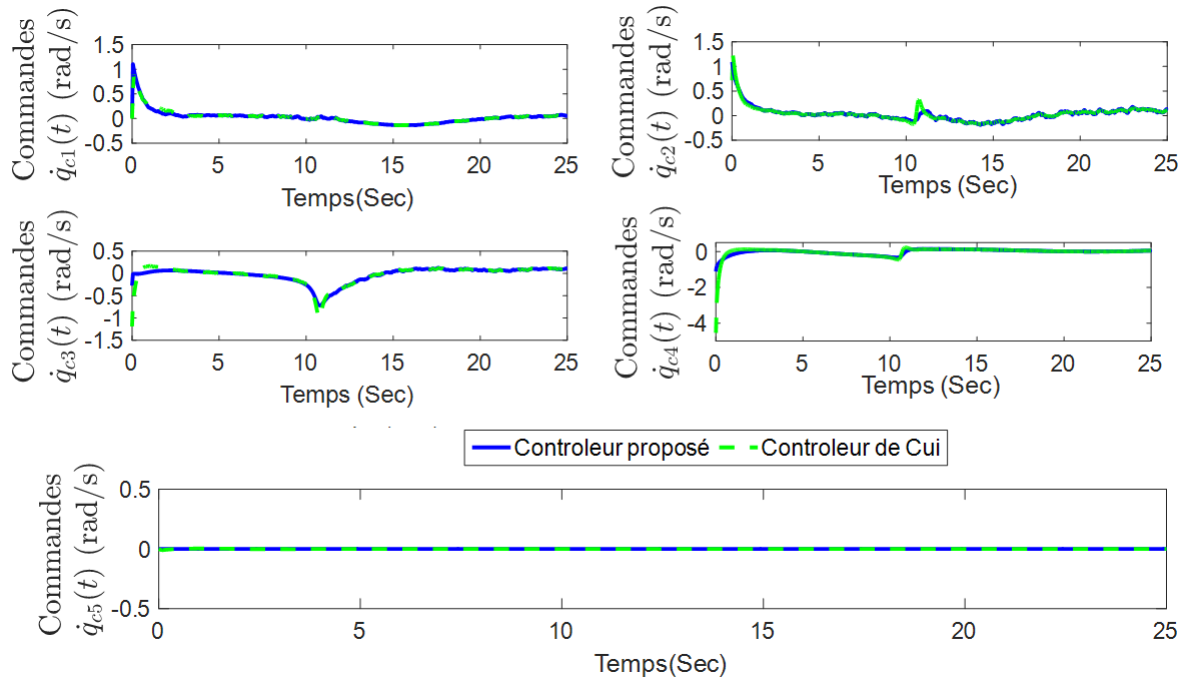


FIGURE 3.21 – Résultats de l'étude comparative, Commandes de vitesse \dot{q}_{ci} , $i = 1, 2, \dots, 5$ pour les deux contrôleurs.

TABLE 3.16 – Signaux de commandes.

Signaux de commande	Contrôleur Proposé		Contrôleur de Cui [49]	
	Max	Min	Max	Min
\dot{q}_{c1}	1106	-138.7	890.5	-137.8
\dot{q}_{c2}	1091	-185.9	1221	-172.8
\dot{q}_{c3}	129	-719.6	167.7	-1191
\dot{q}_{c4}	139.2	-1113	227.8	-4528

saturation. Le Tableau 3.19 et les Figures 3.19 attestent également que l'algorithme de commande développé présente des meilleures performances de suivi par rapport à celui de Cui et al. [14]. En effet, nous obtenons un meilleur temps de réponse de l'ordre de $t_r \leq 4sec$, ainsi que une faible valeur de l'erreur de suivi $|\tilde{x}_j(t)| \leq 1,5 \times 10^{-4}m$. Ainsi, notre approche est plus rapide, plus stable, plus précise et moins sensible aux perturbations aléatoires.

Les bonnes performances observées de l'algorithme de commande proposé pour le rejet des perturbations aléatoires peuvent se justifier par le fait que contrairement à l'algorithme de commande de suivi basé sur l'observateur [14], notre contrôleur rejette les perturbations déterministes et stochastiques. De plus, l'algorithme permet à l'utilisateur de choisir les performances de suivi à travers le vecteur $\psi(\tilde{x})$ et plus précisément grâce au paramètre ϖ_j . En effet, notre approche de commande permet de maintenir la contrainte $\tilde{x}_j(t)^2 \leq \varpi_j^2$, ce qui améliore la précision du suivi même en présence de perturbations aléatoires importantes. Enfin, contrairement à la méthode de Cui et al. [14], le contrôleur proposé ne dépend pas de la structure du contrôleur interne, ce qui permet de l'utiliser avec n'importe quel robot manipulateur industriel.

3.3.2 Résultats des simulations effectuées sur le robot UR5

Les performances de la commande proposée (Figure 2.9) ont été démontrées dans la section précédente. Dans celle-ci, nous allons évaluer la capacité de l'algorithme proposé à maintenir les mêmes performances lorsque la structure du contrôleur interne est différente. Les trajectoires de référence et les paramètres du contrôleur sont identiques à ceux de la Section 3.2.3.

A.Présentation des résultats

La Figure 3.22 décrit le suivi de trajectoire tandis que la Figure 3.23 représente les erreurs de suivi. Les vitesses de commande sont présentées dans la Figure 3.24. Les valeurs maximales et minimales des erreurs de suivi exprimées en mètres, sont indiquées dans le Tableau 3.17 . Celles des positions angulaires et des commandes en vitesse exprimées respectivement en degrés et rad/s sont indiquées dans le Tableau 3.18.

B. Analyse et interprétation des résultats

Les résultats obtenus montrent que la loi de commande définie par (2.113) et les lois d'adaptation (2.118)-(2.120) assurent de bonnes performances de suivi de trajectoire. Les

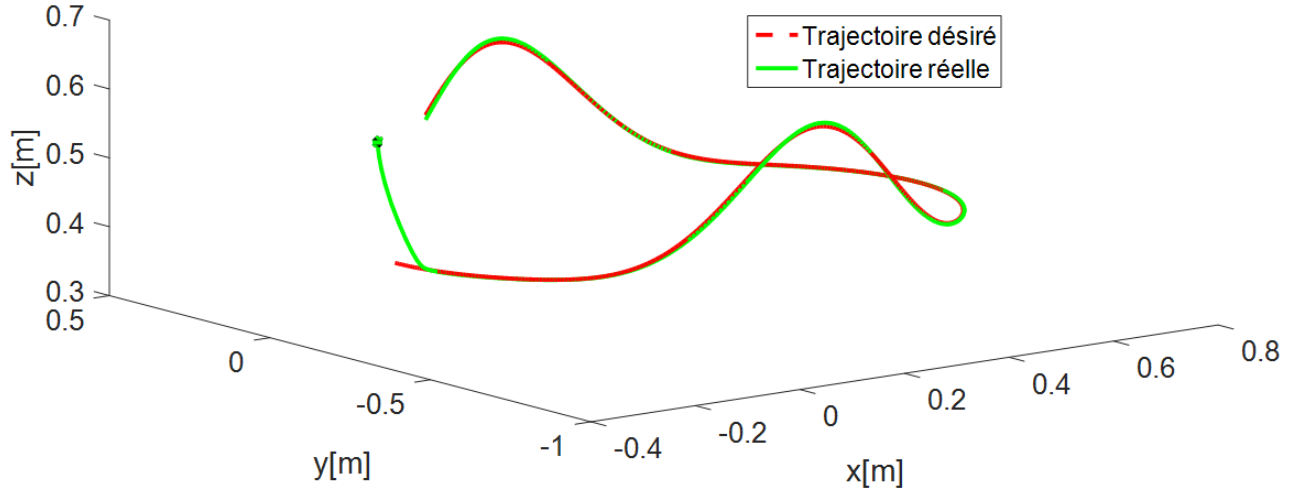


FIGURE 3.22 – Suivi de Trajectoire

TABLE 3.17 – Valeurs Max/Min des erreurs de suivi sur les axes.

Erreur de suivi sur les axes	Valeurs Max	Valeurs Min
e_x	0,1548	0,00700
e_y	0,1983	0,0080
e_z	0,2167	0,0061

TABLE 3.18 – Valeurs Max/Min des signaux de commande

Articulation	Position angulaires		Commande en vitesse	
	Max	Min	Max	Min
1	-2,4227	-6,0202	0.2017	-0.4706
2	-0,8085	-1,9767	0.0904	-0.9353
3	2,0136	-0,7711	1.6161	-0.1780
4	-0,8275	-1,4205	0.7700	-0.0206
5	-0,6919	-1,7349	0.1095	-0.2642
6	-0,7836	-0,7837	3.904×10^{-6}	-3.9372×10^{-6}

Figures 3.22 et 3.23 ainsi que le Tableau 3.21 nous indiquent un temps de réponse relativement très faible de l'ordre $t_r \leq 1.5ms$ ce qui prouve la rapidité de la commande. En plus, nous obtenons des erreurs suivi relativement très faibles (Tableau 3.21) donc la plus élevée est $0,1267 m$ obtenue sur l'axe des Z durant la phase transitoire et la plus faible, soit $0,00700 m$ obtenue sur l'axe des X en régime permanent.

Les performances de suivi sont obtenues en présence d'incertitudes paramétriques, cinématiques et dynamiques, ainsi que de perturbations aléatoires. Les Figures 3.23 et 3.24 ainsi que le Tableau 3.22 nous permettent de certifier que la commande proposée permet de résoudre le problème de saturation des moteurs. En effet, nous obtenons des amplitudes de commande des actionneurs relativement faibles (voir le Tableau 3.22)).

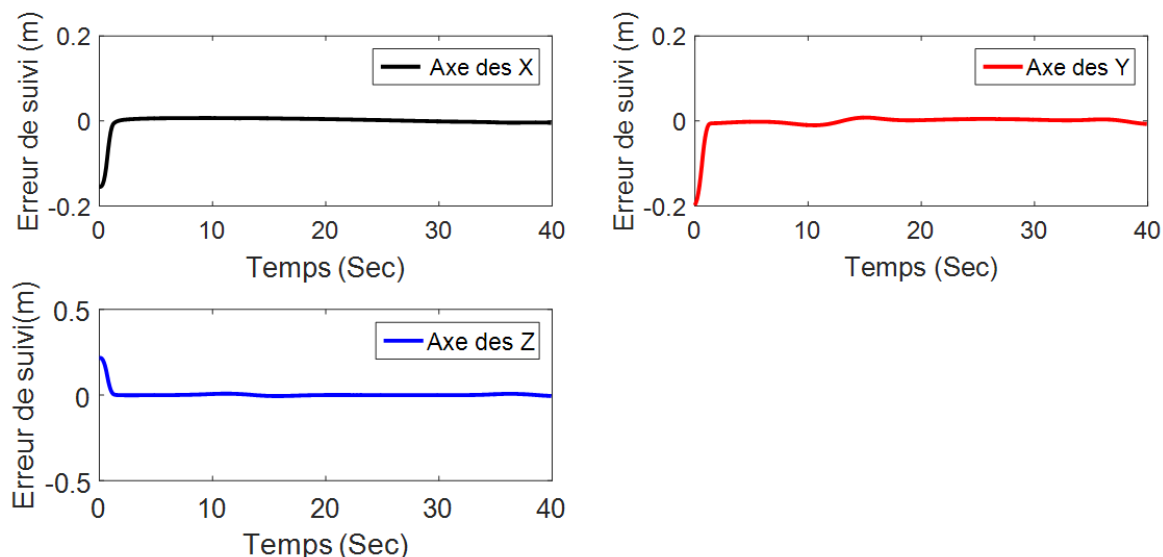


FIGURE 3.23 – Erreurs de suivi.

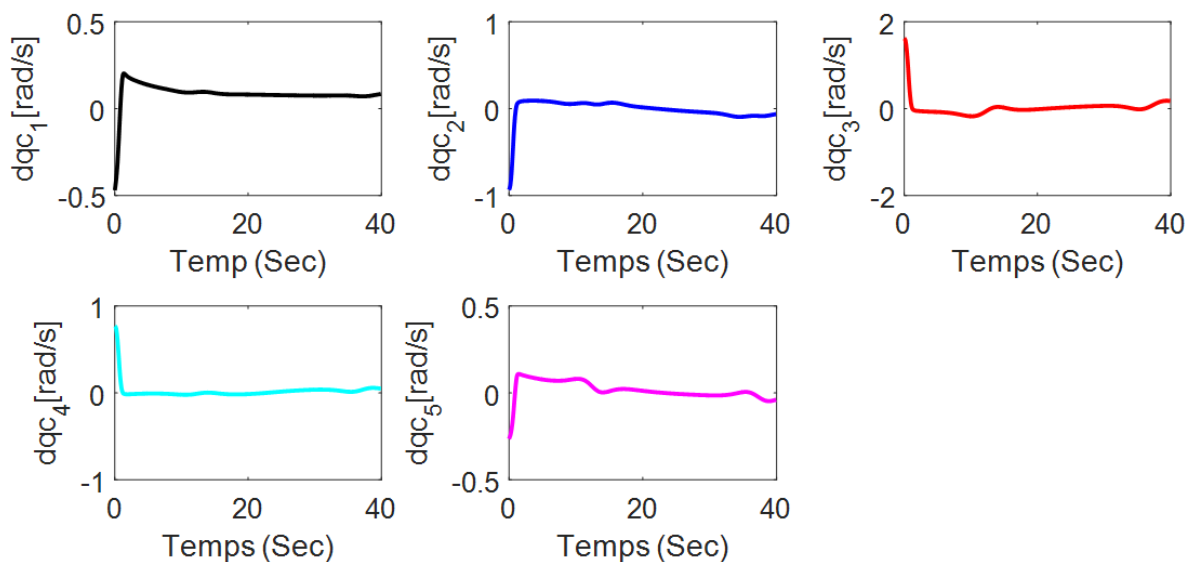


FIGURE 3.24 – Vitesses de commande

Cela est rendu possible grâce à l'efficacité du vecteur de performance de contrainte $\psi(\tilde{x})$ dans \mathbb{R}^r .

3.4 Conclusion

L'objectif de ce chapitre était de démontrer la supériorité des performances des stratégies de commande des RMIACIIs développées dans le chapitre 2 par rapport à celles de la littérature à travers des simulations et expériences.

Les résultats obtenus en environnement déterministe attestent l'efficacité de la commande proposée. Nous avons également réalisé une comparaison des performances avec

les travaux de Wang et al. ([11]). Nous avons constaté clairement que la commande proposée présente de meilleures performances. Nous avons obtenu une erreur de suivi de $|\tilde{x}| = 8 \times 10^{-4}$ et un temps de réponse de $t_r = 0.05s$ lors des simulations (uniquement sur le Scorbot). Les expériences réalisées respectivement sur le Scorbot et l'UR5 présentent des performances impressionnantes. Nous avons obtenu une erreur maximale de $|\tilde{x}| \leq 0.002m$ et un temps de réponse de l'ordre de $t_r \leq 3,7s$ lors des expériences effectuées sur le SCORBOTS. Les performances obtenues sur l'UR5 sont : $|\tilde{x}| \leq 0.000364m$ et $t_r \leq 2,3s$.

Les résultats de la commande des RMIACIII évoluant en environnement aléatoire sont également très satisfaisants par rapport à ceux de la littérature. Ils ont été comparés à ceux de Cui [14]. Nous avons obtenu une erreur de suivi de l'ordre de $1,5 \times 10^{-4}m$ et un temps de réponse inférieur à $4s$ lors des simulations effectuées sur le SCORBOTS. Des performances similaires ont été obtenues lors des tests sur l'UR5.

Les résultats montrent que les algorithmes proposés assurent une stabilité SGUUB pour un RMIACIII soumis aux perturbations déterministes et aléatoires.

Conclusion générale

L'élaboration de stratégies de commande de RMIs capables de répondre aux nouvelles exigences de la robotique industrielle actuelle et future, notamment l'exécution de tâches complexes dans des environnements à la fois déterministes et aléatoires, est le travail de recherche entrepris dans cette thèse. Les tâches complexes auxquelles les RMIs sont destinés, ainsi que les nouveaux domaines d'utilisation des RMIs caractérisés par la présence de vibrations aléatoires et la concurrence prédominante entre les fabricants, sont des facteurs probants de dégradation des performances de suivi de trajectoires. Cela est dû à deux causes majeures : l'inaccessibilité des contrôleurs internes des RMIs et la présence de perturbations stochastiques. Les algorithmes de commande élaborés dans cette thèse permettent de garantir un suivi de trajectoire fiable, une meilleure précision, un temps de réponse rapide, le rejet des perturbations aléatoires et déterministes, et assurent la stabilité globale du système bouclé.

Le premier problème traité par nos travaux est l'élaboration d'une loi de commande de suivi de trajectoire des RMIs à architecture de commande interne inconnue et inaccessible. Pour atteindre cet objectif, nous proposons un contrôleur externe RBF NN adaptatif. La commande proposée est indépendante de la structure du contrôleur interne, n'utilise pas de régresseur dynamique et ne nécessite pas la mesure de la vitesse dans l'espace de tâche ainsi que des accélérations des articulations du robot.

Le deuxième problème abordé dans nos travaux concerne l'élimination des perturbations aléatoires et déterministes des RMIACIIs. Pour cela, nous proposons une commande adaptative hybride qui combine une méthode adaptative directe pour le rejet des perturbations stochastiques, et une méthode adaptative indirecte par réseaux de neurones récurrents (RBF NN) pour le rejet des perturbations déterministes. Contrairement aux approches développées dans les articles de référence [16, 49], nous avons démontré que $\Gamma(q) \Gamma^T(q) y = Q(q, y) \Theta$, ce qui nous a permis de développer une commande insensible aux erreurs de modélisation. Les résultats obtenus démontrent l'effectivité de l'algorithme de commande proposé.

Les résultats obtenus et présentés dans ce manuscrit ouvrent plusieurs perspectives. Notamment :

- Adapter l'algorithme de commande proposé aux robots à articulation flexible, continue, etc. qui sont très utilisés pour réaliser des tâches complexes en environnements aléatoires. Pour cela, il faudra tenir compte de certaines dynamiques fortement influencées par les perturbations aléatoires à l'exemple des dynamiques électriques et la flexibilité des articulations.
- Il serait également intéressant de considérer dans les futurs travaux les contraintes d'état des RMIACIII. Cela va permettre de considérer certains phénomènes non linéaires tels que la saturation des moteurs, le backlash, etc. qui dégradent considérablement les performance des moteurs.
- Les travaux menés jusqu'à présent sur la modélisation et la commande des robots manipulateurs évoluant en environnement aléatoire, notamment [13], [14], [16] considèrent uniquement les signaux stochastiques générés par les vibrations du sol. Cependant les phénomènes physiques tels que le vent, la résistance de l'air, etc. impactent considérablement les performances du robot et peuvent détruire certains éléments du robot tels que les actionneurs, les capteurs, etc. Il serait donc intéressant de faire des études notamment sur la modélisation du robot en tenant compte des actions de ces phénomènes physiques sur le robot et de proposer des stratégies de commandes adéquates dans un tel contexte. Les commandes tolérantes aux fautes seront très conseillées.

Bibliographie

- [1] Mesrob, *Distinguished Figures in Mechanism and Machine Science*. Springer, 2018.
- [2] IFR, “History of industrial robots : Milestones of technology and commercialization,” in *International Federation of Robotics*, 2012. Tech. Rep.
- [3] C. Toroa, I. Barandiarana, and J. Posadaa, “A perspective on knowledge based and intelligent systems implementation in industrie 4.0,” in *19th International Conference on Knowledge Based and Intelligent Information and Engineering Systems*, vol. 60, p. 362 – 370, Elsevier, 2015.
- [4] G. Schuh, T. Potente, C. Wesch-Potente, A. R. Weber, and J.-P. Prote, “Collaboration mechanisms to increase productivity in the context of industrie 4.0,” in *Robust Manufacturing Conference*, vol. 60, p. 362 – 370, Elsevier, 2014.
- [5] G. Luhn, G. Luhn, K. Bartl, J. Postel, T. Stevens, and M. Zinner, “Real-time information base as key enabler for manufacturing intelligence and “industrie 4.0,” *ASMC*, 2015.
- [6] Wang and Hanlei, “Adaptive visual tracking for robotic systems without image-space velocity measurement,” *Automatica*, vol. 55, pp. 294–301, 2015.
- [7] Wang and Hanlei, “Adaptive control of robot manipulators with uncertain kinematics and dynamics,” *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 948–954, 2016.
- [8] R. Kelly and J. Moreno, “Manipulator motion control in operational space using joint velocity inner loops,” *Automatica*, vol. 41, no. 8, pp. 1423–1432, 2005.
- [9] K. Camarillo, R. Campa, V. Santibáñez, and J. Moreno-Valenzuela, “Stability analysis of the operational space control for industrial robots using their own joint velocity pi controllers,” *Robotica*, vol. 26, no. 6, pp. 729–738, 2008.

- [10] F. Sanfilippo, L. I. Hatledal, H. Zhang, M. Fago, and K. Y. Pettersen, “Controlling kuka industrial robots : Flexible communication interface jopenshowvar,” *IEEE robotics & automation magazine*, vol. 22, no. 4, pp. 96–109, 2015.
- [11] H. Wang, W. Ren, C. C. Cheah, Y. Xie, and S. Lyu, “Dynamic modularity approach to adaptive control of robotic systems with closed architecture,” *IEEE Transactions on Automatic Control*, vol. 65, no. 6, pp. 2760–2767, 2019.
- [12] Mvogo, Medzo, Melingui, Essimbi, and Merzoukid, “Task-space control for industrial robot manipulators with unknown inner loop control architecture,” *franklin institute*, vol. 12, pp. 6256–6310, juin 2022.
- [13] Cui, Wu, Xie, and Shi, “Modeling and adaptive tracking for a class of stochastic lagrangian control systems,” *Automatica*, vol. 49, no. 3, pp. 770–779, Mar. 2013.
- [14] Cui, Xie, and Wu, “Dynamics modeling and tracking control of robot manipulators in random vibration environment,” *IEEE Transactions on Automatic Control*, vol. 58, no. 6, pp. 1540–1545, Jun. 2013.
- [15] M. Hui-Fang, D. Na, and Z.-J. Zhang, “Adaptive output-feedback control for stochastic robot system based on neural network,” in *2015 34th Chinese Control Conference (CCC)*, IEEE, pp. 1934–1768, 2015. Hangzhou, China.
- [16] M. Cui and Z. Wu, “Trajectory tracking of flexible joint manipulators actuated by dc-motors under random disturbances,” *Journal of the Franklin Institute*, vol. 356, no. 16, pp. 9330–9343, 2019.
- [17] C. Medzo, J. MVOGO, A. MELINGUI, and R. MERZOUKI, “Adaptive neural task space control for robot manipulators with unknown and closed control architecture under random vibrations,” *IEEE ACCESS*, vol. 10, pp. 60765–60777, 2022.
- [18] A. V. Le, P. Veerajagadheswar, P. T. Kyaw, M. V. J. Muthugala, M. R. Elara, M. Kuma, and N. H. K. Nhan, “Towards optimal hydro-blasting in reconfigurable climbing system for corroded ship hull cleaning and maintenance,” *Expert Systems with Applications*, vol. 170, no. 4, p. 114519, May 2021.
- [19] P. V, L. V, K. T, and R. E. Mohan, “A self-evaluating hydro-blasting reconfigurable robot for ship hull maintenance,” *IEEE Access*, vol. 8, no. 1, pp. 193790–193800, Oct. 2020.
- [20] A. Gasparetto and L. Scalera, “A brief history of industrial robotics in the 20th century,” *Advances in Historical StudiesAd*, vol. 8, no. 24, pp. 24–35, 2019.

-
- [21] J. Craig, *Introduction to Robotics : Mechanics and Control*, vol. 2 edition. Pearson Education, 1989.
- [22] R. Vaccaro and S. Hill, “A joint-space command generator for cartesian control of robotic manipulators,” *Journal of Robotics and Automation*, vol. 40, no. I, pp. 70–76, 1988.
- [23] V. Kumar, Hoeller, David, Sundaralingam, Balakumar, T. Jonathan, Birchfield, and Stan, “Joint space control via deep reinforcement learning,” 11 2020.
- [24] S. moon Hur, Sang-Rok, and Yonghwan, “Joint space torque controller based on time-delay control with collision detection,” *International Workshop on Intelligent Robots and Systems (IROS)*, 2014.
- [25] R. Hu and P. Moller, “Independent joint control : Estimation and compensation of coupling and friction effects in robot position control,” *Journal of Intelligent and Robotic Systems*, vol. 15, pp. 41–51, 1996.
- [26] T.Hsia, T.Lasky, and Z. Guo, “Robust independent joint controller design for industrial robot manipulators,” *Transactions on Industrial Electronics*, vol. 38, no. 1, pp. 21 – 25, 1996.
- [27] W. Chung, L.-C. Fu, and S.-H. Hsu, *Motion Control*, pp. 133–159. 01 2008.
- [28] P. Song, *Robotic Manipulator Control : Fundamentals of Task Space Design*. Ph.d. dissertation, Depart ment of Mechanical Engineering, 1997. Toronto.
- [29] S. S. Ge, C. C. Hang, S. Member, and L. C. Woon, “Adaptive neural network control of robot manipulators in task space,” *IEEE Transactions on Industrial Electronics*, vol. 44, pp. 895–904, DECEMBER 1997.
- [30] X. Li and C. C. Cheah, “Global task-space adaptive control of robot,” *Automatica*, vol. 49, no. 1, pp. 52–69.
- [31] Garcia, Rodolfo, and V. Parra-Vega, “Task-space neuro-sliding mode control of robot manipulators under jacobian uncertainties,” *International Journal of Control Automation and Systems*, vol. 9, pp. 895–904, 10 2011.
- [32] M. W. Xinwu Liang, Xinhuan Huang, “Adaptative task-space tracking control of robot without task-space and joint space velocity measurement,” *Transaction on Robotics and Automation*, vol. 20, no. 1, pp. 21 – 25, 2010.

- [33] Q. Wang and C. Sun, “Adaptive consensus of multiagent systems with unknown high-frequency gain signs under directed graphs,” *IEEE Transactions on Systems, Man, and Cybernetics : Systems*, vol. 50, no. 6, pp. 2181–2186, 2018.
- [34] Q. Wang, H. E. Psillakis, C. Sun, and F. L. Lewis, “Adaptive nn distributed control for time-varying networks of nonlinear agents with antagonistic interactions,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 6, pp. 2573–2583, 2020.
- [35] A. Izadbakhsh and S. Khorashadizadeh, “Robust task-space control of robot manipulators using differential equations for uncertainty estimation,” *Robotica*, vol. 35, no. 9, pp. 1923–1938, 2017.
- [36] Liu and Y. Chen, “Task-space coordination control of bilateral human–swarm systems,” *Journal of the Franklin Institute*, vol. 352, no. 1, pp. 311–331, 2015.
- [37] K. Dogan, E. Tatlicioglu, E. Zergeroglu, and K. Çetin, “Learning control of robot manipulators in task space,” *Asian Journal of Control*, vol. 20, no. 3, pp. 1003–1013, 2018.
- [38] Galicki and Mirosław, “Finite-time trajectory tracking control in a task space of robotic manipulators,” *Automatica*, vol. 67, pp. 165–170, 2016.
- [39] Muñoz-Vázquez, A. Jonathan, Sánchez-Torres, J. Diego, Gutiérrez-Alcalá, Salvador, and Jiménez-Rodríguez, “Predefined-time robust contour tracking of robotic manipulators,” *Journal of the Franklin Institute*, vol. 356, no. 5, pp. 2709–2722, 2019.
- [40] R. Gholipour and M. M. Fateh, “Adaptive task-space control of robot manipulators using the fourier series expansion without task-space velocity measurements,” *Measurement*, vol. 123, pp. 285–292, 2018.
- [41] B. Xiao and S. Yin, “Exponential tracking control of robotic manipulators with uncertain dynamics and kinematics,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 689–698, 2018.
- [42] H. Wang, T. Jiang, S. Xu, and Y. Xie, “On the robust outer loop control of robotic systems,” in *2017 36th Chinese Control Conference (CCC)*, pp. 1260–1264, IEEE, 2017.
- [43] Feng, Zhi, Hu, and Guoqiang, “A unified inner-/outer-loop distributed design for task-space robust cooperative tracking of uncertain networked manipulators under directed graphs,” *International Journal of Robust and Nonlinear Control*, vol. 32, no. 9, pp. 5652–5672, 2022.

- [44] Liu and Wu, “Modelling and adaptive tracking control for flexible joint robots with random noises,” *International Journal of Control*, vol. 87, no. 12, pp. 2499–2510, Jun. 2014.
- [45] Z. Wu, S. Wang, and M. Cui, “Tracking controller design for random nonlinear benchmark system,” *Journal of the Franklin Institute*, vol. 354, no. 1, pp. 360–371, Jan. 2017.
- [46] W. Sun, W. Yuan, J. Zhang, and Q. Sun, “Adaptive tracking control for a class of manipulator systems with state constraints and stochastic disturbances,” *Mathematical Problems in Engineering*, vol. 2018, Jun. 2018.
- [47] W. Sun, S. Diao, S.-F. Su, and Y. Wu, “Adaptive fuzzy tracking for flexible-joint robots with random noises via command filter control,” *Information Sciences*, vol. 575, no. 2021, pp. 116–132, 2021.
- [48] Z. Wu, “Stability criteria of random nonlinear systems and their applications,” *IEEE Transactions on Automatic*, vol. 60, no. 4, pp. 1038–1049, 2015.
- [49] M. Cui, C. Yang, and Z. Wu, “Global trajectory tracking of a class of manipulators without velocity measurements in random surroundings,” *International Journal of Control*, vol. 356, no. 16, pp. 1–10, 2021.
- [50] G. Shen, S. Xu, C. Chen, and Q. Wang, “Dynamic modeling and vibration analysis of a 6-dofs industrial robot considering joint flexibility,” *Conference on Real-time Computing and Robotics*, vol. xx, no. x, pp. 1–7, 2021.
- [51] F. Temurtas, H. Temurtas, and N. Yumusakr, “Application of neural generalized predictive control to robotic manipulators with a cubic trajectory and random disturbance,” *Robotics and Autonomous Systems*, vol. 54, no. x, pp. 74–83, 2006.
- [52] W. Su, B. Niu, H. Wang, and W. Qi, “Adaptive neural network asymptotic tracking control for a class of stochastic nonlinear systems with unknown control gains and full state constraints,” *International Journal of Adaptive Control and Signal Processing*, vol. 35, no. 10, pp. 2007–2024, Jul. 2021.
- [53] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*, vol. 3. New York : Wiley, 2006.
- [54] B. Siciliano, L. Sciavicco, L. Villani, , and G. Oriolo, *Robotics : Modelling, Planning and Control*, vol. 3. London Limited : Springer-Verlag, 2009.
- [55] W. Khalil, *Modeling, Identification and control of robots*. 2004.

- [56] W. Khalil and D. Etienne, *identification, Modélisation et commande des robots manipulateurs*. 1999.
- [57] H. Lee, S. Kim, and H. J. Kim, “Control of an aerial manipulator using on-line parameter estimator for an unknown payload,” in *International Conference on Automation Science and Engineering (CASE)*, pp. 1260–1264, IEEE, 2015.
- [58] W. Khalil and E. Dombre, *Modeling, identification and control of robots*. 2004.
- [59] K. Bouzgou, *Contribution à l’architecture, la modélisation et la commande d’un bras manipulateur aérien*. Ph.d. dissertation, sciences et technologies de l’information et de la communication (STIC), 2021. Paris-Saclay.
- [60] J. Craig, *Introduction to Robotics : Mechanics and Control*, vol. 3 edition. Pearson Education, 2005.
- [61] D. Pieper, *The Kinematics of manipulators under computer control*. Phd thesis, Department of Computer Science, 1968. Stanford.
- [62] Paul, *Robot manipulators : mathematics, programming and control*. 1981.
- [63] S. Lee and M. Ziegler, *A geometric approach in solving the inverse kinematics of PUMA robots*, vol. 16. Symp on Industrial Robots : In Proc. 13th Int, pages 1–18, 1983.
- [64] M. Renaud, *Contribution a la modélisation et à la commande dynamiques des robots manipulateurs*. Phd thesis, September 1980. Toulouse.
- [65] Z. Wu, S. Wang, and M. Cui, “Tracking controller design for random nonlinear benchmark system,” *Journal of the Franklin Institute*, vol. 354, no. 1, pp. 360–371, Jan. 2017.
- [66] D. Etienne, *Analyse et modélisation des robots manipulateurs*. Lavosier, 2001.
- [67] J. J.-B. Mvogo, J. B. Mbede, A. Melingui, and B. E. Zobo, “Robust adaptive control for robot manipulators : Support vector regression-based command filtered adaptive backstepping approach,” *Robotica*, vol. 36, no. 4, pp. 516–534, 2018.
- [68] W. M. Elawady, “An adaptive second order sliding mode inverse kinematics approach for serial kinematic chain robot manipulators,” *robotic*, vol. 9, no. 4, 2020.
- [69] D. S. Broomhead and D. Lowe, *Radial basis functions, multi-variable functional interpolation and adaptive networks*. 1988.
- [70] D. Broomhead, D. S. and Lowe, *Multivariable functional interpolation and adaptive networks*. 1988.

- [71] B. M. D, *Radial Basis Functions*, vol. Cambridge. Cambridge University Press, 2003.
- [72] A. Borgi, *Apprentissage supervise par génération de règles :le système SUCRAGE*. 1999.
- [73] Z. Liu, B. Chen, and C. Lin, “Adaptive neural backstepping for a class of switched nonlinear system without strict-feedback form,” *IEEE Transactions on Systems, Man, and Cybernetics : Systems*, vol. 47, no. 7, pp. 1315–1320, 2016.
- [74] M. M. Polycarpou and P. A. Ioannou, “A robust adaptive nonlinear control design,” in *1993 American control conference*, pp. 1365–1369, IEEE, 1993.
- [75] H. K. Khalil, *Nonlinear Systems*, vol. 3rd edition. New Jerse : Prentice Hall, Upper Saddle River, 2002.
- [76] P. P. Müllhaupt, *Introduction à l’Analyse et à la Commande des Systèmes Non Linéaire*. Ph.d. dissertation, Presses Polytechniques, 2009. Lausanne.
- [77] D. Hadjer, *Equations différentielles stochastique*. Master dissertation, Département de Mathématiques, 2020. Algérie.
- [78] S. Nadjla, *Les solutions fortes d’EDS coefficients Lipschitziens en dimension finie*. Master dissertation, Département de Mathématiques, Juin 2020. Algérie.
- [79] N. Rabiaa, *Sur les équations différentielles ordinaires aléatoires* *Sur les équations différentielles ordinaires aléatoires*. Master dissertation, Département de Mathématiques, Juin 2021. Algérie.
- [80] W. He, C. Xue, X. Yu, Z. Li, and C. Yang, “Admittance-based controller design for physical human–robot interaction in the constrained task space,” *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1937–1949, 2020.
- [81] R. Datouo, J. J.-B. M. Ahanda, A. Melingui, F. Biya-Motto, and B. E. Zobo, “Adaptive fuzzy finite-time command-filtered backstepping control of flexible-joint robots,” *Robotica*, vol. 39, no. 6, pp. 1081–1100, 2021.