

Université Mohamed Khider – Biskra
Faculté des Sciences et de la Technologie
Département de Génie Électrique
Réf:



جامعة محمد خيضر - بسكرة
كلية العلوم والتكنولوجيا
قسم الهندسة الكهربائية
المرجع:

Thèse présentée en vue de l'obtention
du diplôme de
Doctorat LMD
Spécialité: Automatique

Commande Robuste Non-linéaire d'un Quadrotor

Présentée par:

HASSENI Seif El Islam

Soutenue publiquement le: 27/10/2020

Devant le jury composé de:

Dr. Mohamed BOUMEHRAZ	Professeur	Président	Université de Biskra
Dr. Latifa ABDOU	Professeur	Rapporteur	Université de Batna 2
Dr. Souad CHAOUCH	Professeur	Examineur	Université de Batna 2
Dr. Abdelghani CHELIHI	Maitre de Conférences A	Examineur	Université de Constantine 1
Dr. Mostefa Mohamed TOUBA	Maitre de Conférences A	Examineur	Université de Biskra

2019/2020

à mes très chers parents

à ma chère grand-mère

à mes frères et sœurs

Je dédie ce travail

Seif El Islam HASSENI

Remerciements:

Je remercie très profondément ma directrice de thèse *Dr. Latifa ABDOU*, Professeur à l'université Batna 2, de m'avoir proposé un sujet très intéressant, et surtout de m'avoir dirigé et encouragé tout au long de la réalisation de ce travail.

Je tiens à remercier le *Dr. Mohamed BOUMEHRAZ*, Professeur à l'université de Biskra d'avoir accepté de présider mon jury de soutenance. Mes remerciements vont également au *Dr. Souad CHAOUCH*, Professeur à l'université Batna 2, au *Dr. Abdelghani CHELIHI*, Maitre de conférences A à l'université Constantine 1 et au *Dr. Mostefa Mohamed TOUBA*, Maitre de conférences A à l'université de Biskra d'avoir accepté de faire partie de mon jury de soutenance.

Enfin, je n'oublierais pas de remercier tous les membres du laboratoire Modélisation des Systèmes Energétiques (LMSE), Biskra.

Résumé: Cette thèse étudie les stratégies de la commande robuste pour résoudre le suivi de trajectoire d'un mini drone. Celui-ci est de type quadrotor, qui est caractérisé par six degrés de liberté. En premier lieu, nous avons développé des correcteurs basés sur des techniques linéaires et nonlinéaires classiques (PID, Backstepping et Mode Glissant). En suite, nous nous sommes focalisé sur les méthodes robustes basées sur la norme H_∞ . Les approches utilisées sont; LPV robuste avec exploitation des avantages de la forme LFT sur la représentation d'un système non-linéaire comme un système LPV. Ce correcteur est exploité pour résoudre le problème de stabilisation rotationnel robuste, et le problème est interprété à des matrices d'inégalité linéaires (LMI) et qui sont résolues par le théorème du petit gain, avec une structure de commande conçue à la forme en cascade. La deuxième méthode robuste utilisée est H_∞ non-linéaire. Nous avons commandé les degrés de liberté actionnés par leur couplage dynamique avec les degrés de liberté sous-actionnés, ainsi le problème est interprété par des équations optimales et un terme d'adaptation est ajouté afin de compenser les incertitudes paramétriques. Les résultats de simulation présentés considèrent la robustesse du quadrotor vis-à-vis des bruits, des perturbations et des incertitudes.

Mots-clés: Quadrotor; Commande Robuste; PID; Algorithmes Évolutionnaires; Backstepping; Mode Glissant; Systèmes Sous-Actionnés; LPV; H_∞ ; Réseaux de Neurones.

Abstract: This thesis investigates the strategies of robust control to solve the trajectory tracking of a mini drone. The considered system is a quadrotor, characterized by six degrees of freedom. Firstly, we developed such controllers based on classical linear and nonlinear techniques (PID, Backstepping and Sliding Mode). Next, we focused on the robust methods based on H_∞ norm. The used approaches are; robust LPV with using the advantages of LFT form by representing a nonlinear system as an LPV one. This controller is used to solve the robust rotational stabilization problem is the problem is interpreted to linear matrices inequalities (LMI) which are solved by small gain theorem and the control structure is designed as cascade form. The second robust approach used is the non-linear H_∞ . We controlled the actuated degrees of freedom with their dynamic coupling with the under-actuated degrees of freedom. The problem is interpreted by nonlinear optimal equations and an adaptive term is added in order to compensate the parametric uncertainties. The presented simulation results consider the robustness against noises, disturbances and parametric uncertainties.

Keywords: Quadrotor; Robust Control; PID; Evolutionary Algorithms; Backstepping; Sliding Mode; Under-Actuated Systems; LPV; H_∞ ; Neural Networks.

ملخص: تتطرق هذه الأطروحة إلى تقنيات التحكم المتين لحل مشكلة إتباع المسار لطائرة بدون طيار، هذه الطائرة من نوع رباعية المرواح، التي تتميز بست درجات للحرية. بداية، قمنا بتطوير بعض المتحكمات التقليدية الخطية والغير خطية (PID، Backstepping، Sliding Mode). بعد ذلك، ركزنا على أساليب التحكم المتينة المعتمدة على المعيار H_∞ . الطرق المستخدمة كالتالي: التحكم المتين عن طريق الأنظمة الخطية متغيرة المعاملات (LPV) وباستخدام مميزات LFT الخاصة بتمثيل النظام الغير خطي كنظام خطي متغير المعاملات، إستخدمنا هذا المتحكم لحل اشكالية الإستقرار المتين، ترجمت الإشكالية إلى متراجحات من نوع LMI وتم حلها بواسطة نظرية التقوية الصغيرة. الطريقة الثانية وهي H_∞ الغير خطية، حيث نتحكم في درجات الحرية الناشطة مع تشابكها بدرجات الحرية الغير ناشطة. إشكالية التحكم المتين تم ترجمتها إلى معادلات غير خطية لتحكم أمثل. تم إضافة عبارة لتحكم مكيف من أجل تعويض عدم الدقة في معاملات النظام. نتائج المحاكاة تظهر متانة في مقابل التسويش في قياس المتغيرات والإضطرابات وعدم دقة معالم النظام.

الكلمات المفتاحية: طائرة رباعية المرواح; التحكم المتين; PID; الخوارزميات التطوري; Backstepping; Sliding Mode; الأنظمة تحت نشاط; الأنظمة الخطية متغيرة المعاملات; H_∞ ; الشبكات العصبية.

Table des Matières:

Table des Matières	5
Liste des Figures	8
Liste des Tableaux	10
Abréviations	11
Introduction Générale	13
Chapitre I: <i>Etat de l'Art sur les Quadrotors; Commande et Applications</i>	17
I.1 Introduction	18
I.2 Les Quadrotors dans la Robotique	18
I.2.1 Les Applications des Quadrotors	18
I.2.2 Les Avantages	19
I.2.3 Les Défis	20
I.3 Etat de l'Art sur la Commande du Quadrotor	21
I.3.1 Les Systèmes de Type Quadrotor	21
I.3.2 Méthodes de Commande du Quadrotor	22
I.3.3 Suivi Précis de La Trajectoire	24
I.3.4 L'Aérodynamique du Quadrotor	26
I.4 Conclusion	29
Chapitre II: <i>État de l'Art sur Les Systèmes LPV et H_∞ Non-Linéaires</i>	30
II.1 Introduction	31
II.2 Systèmes LPV	31
II.2.1 Approximation des Systèmes Non-Linéaires	31
II.2.2 Représentation des Systèmes LPV	32
II.2.3 Commande des Systèmes LPV	33
II.2.4 Applications	35
II.3 La Commande H_∞ Non-Linéaire et les Systèmes Sous-Actionnés	35
II.3.1 La Commande H_∞ Non-Linéaire	36
II.3.2 Les Systèmes Mécaniques Sous-Actionnés	37
II.4 Conclusion	39
Chapitre III: <i>Amélioration Des Performances de la Commande Linéaire par l'utilisation des Algorithmes Évolutionnaires</i>	40
III.1 Introduction	41

III.2 Modélisation du Quadrotor	41
III.3 Les Algorithmes Évolutionnaires	45
III.3.1 Applications des Algorithmes Évolutionnaires sur la Commande	45
III.3.2 Les Caractéristiques des Algorithmes Sélectionnés	50
III.4 Commande Linéaire (PID) de Quadrotor	55
III.4.1 Structure de la Commande	55
III.4.2 Résultats de Simulation	57
III.5 Conclusion	64
Chapitre IV: Commande du Quadrotor par les Techniques Non-Linéaires	65
IV.1 Introduction	66
IV.2 Commande Non-Linéaire du Quadrotor	66
IV.3 Commande par Backstepping Intégral	67
IV.4 Commande par Mode Glissant	76
IV.5 Résultats de Simulation	83
IV.6 Conclusion	86
Chapitre V: Commande Robuste du Quadrotor Basée sur la Technique LPV	87
V.1 Introduction	88
V.2 Représentation et Commande des Systèmes LPV	88
V.2.1 Représentation des Systèmes LPV	88
V.2.2 Commande des Systèmes LPV	92
V.3 Stabilité des Systèmes LPV	94
V.3.1 Stabilité des Systèmes LPV Génériques et Polytopiques	94
V.3.2 Stabilité des Systèmes LPV de Forme LFT	96
V.4 Stabilisation LPV Robuste du Quadrotor	100
V.4.1 Sélection des Paramètres	101
V.4.2 Synthèse du Correcteur	102
V.5 Suivi de Trajectoire et Simulation	104
V.5.1 Suivi de Trajectoire	104
V.5.2 Résultats de Simulation	106
V.6 Commande d'un Système Sous-Actionné; Pendule Inversé	110
V.6.1 Représentation LFT- LPV du Pendule Inversé	111
V.6.2 Sélection des Paramètres des Fonctions de Pondération	112
V.6.3 Stabilité Étendue de 2DOF	114
V.6.4 Résultats de Simulation	115

V.7 Conclusion	117
Chapitre VI: Commande Robuste du Quadrotor par la Technique H_∞ Non-Linéaire	118
VI.1 Introduction	119
VI.2 Commande par Couple Calculé	119
VI.3 Modélisation Complète du Quadrotor	121
VI.3.1 La Cinématique du Quadrotor	122
VI.3.2 Les Équations d'Euler-Lagrange	124
VI.4 Commande H_∞ Non-Linéaire du Quadrotor	126
VI.4.1 Transformation de Modèle pour le Robot	127
VI.4.2 H_∞ Non-Linéaire	131
VI.4.3 Calcul des Forces/Couples	133
VI.4.4 Résultats de Simulation	135
VI.5 Commande H_∞ Non-Linéaire Adaptative	137
VI.5.1 Adaptation par Paramétrage Linéaire	139
VI.5.2 Adaptation par Réseaux de Neurones	142
VI.5.3 Résultats de Simulation	147
VI.6 Conclusion	151
Conclusion Générale et Perspectives	152
Bibliographie	154

Liste des Figures:

Figure I.1 Des images acquis d'une drone de surveillance des feux (Merino et al., 2011)	19
Figure I.2 Les coordonnées de la formation Δ leader-follower (Mulgaonkar et al., 2015)	20
Figure I.3 Le plateforme quadrotor du groupe <i>Robotics and Perception</i> (Fässler, 2018)	22
Figure I.4 Un quadrotor en équilibrant un pendule inversé (Hehn et D'Andrea, 2013)	25
Figure I.5 Un schéma montré les pales inclinés (Bangura et al., 2016)	27
Figure I.6 Schéma de la traînée du rotor (Omari et al., 2013)	29
Figure III.1 Schéma du quadrotor	41
Figure III.2 Structure de commande du quadrotor	56
Figure III.3-a Évolution de la fonction-objectif pour l'altitude (z)	58
Figure III.3-b Évolution de la fonction-objectif du roulis-tangage (φ, θ)	58
Figure III.3-c Évolution de la fonction-objectif du lacet (ψ)	59
Figure III.3-d Évolution de la fonction-objectif de x - y	59
Figure III.4 Le suivi de la trajectoire en 3D	62
Figure III.5 Les réponses des coordonnées de translation	62
Figure III.6 Les erreurs des coordonnées de translation	62
Figure III.7 Les réponses des coordonnées de rotation	63
Figure III.8 Les erreurs des coordonnées de rotation	63
Figure IV.1 Structure en cascade de la commande non-linéaire du quadrotor	67
Figure IV.2 Les réponses des coordonnées de translation	84
Figure IV.3 Les réponses des coordonnées de rotation	84
Figure IV.4 Le suivi de la trajectoire en 3D	85
Figure IV.5 Les commandes appliquées	85
Figure V.1 Plan du polytope de l'exemple V.1	90
Figure V.2 Plan du polytope de l'exemple V.2	91
Figure V.3 Représentation LFT d'un système LPV	92
Figure V.4 Structure de commande par séquençement de gains	93
Figure V.5 Représentation d'un correcteur LPV sous forme LFT	94
Figure V.6 La boucle fermée système-correcteur d'un système LPV sou forme LFT	97
Figure V.7 Schéma LPV-LFT d'attitude du quadrotor	101
Figure V.8 La boucle fermée système-correcteur d'attitude du quadrotor	104
Figure V.9 La structure de commande du quadrotor	106

Figure V.10 Les réponses des coordonnées de translation avec des incertitudes sur les inerties	107
Figure V.11 Les réponses des coordonnées d'orientation avec des incertitudes sur les inerties	107
Figure V.12 La trajectoire en 3D	108
Figure V.13 Une vue horizontale de la trajectoire	108
Figure V.14 Les signaux de commande	108
Figure V.15 La trajectoire en 3D du IBS-LPV et BS-LPV	109
Figure V.16 Les réponses de coordonnées de translation du IBS-LPV et BS-LPV	109
Figure V.17 Vue générale du véhicule pendule personnel	110
Figure V.18 Schéma géométrique du pendule inversé sur chariot	111
Figure V.19 L'interconnexion en boucle fermée du système-correcteur	113
Figure V.20 La boucle fermée du système-correcteur sous-actionné	115
Figure V.21 Les réponses de l'angle (θ), la vitesse (V) et la force (F) avec des incertitudes sur la masse M	116
Figure V.22 Les réponses de l'angle (θ), la vitesse (V) et la force (F) avec différents angles initiaux	116
Figure VI.1 Schéma de commande par couple calculé	120
Figure VI.2 Schéma général de la structure du quadrotor	121
Figure VI.3 Les trois rotations d'Euler roulis φ , tangage θ et lacet ψ	122
Figure VI.4 La structure de commande H_∞ non-linéaire utilisée	127
Figure VI.5 Suivi de la trajectoire en 3D (1 ^{ère} trajectoire)	136
Figure VI.6 Réponses des coordonnées de la position (1 ^{ère} trajectoire)	137
Figure VI.7 Réponses des coordonnées de rotation (1 ^{ère} trajectoire)	137
Figure VI.8 La structure de commande adaptative utilisée	138
Figure VI.9 Schéma du réseau de neurone pour l'adaptation de $F_z, \Delta E_1(x_e, B)$	145
Figure VI.10 Schéma du réseau de neurone pour l'adaptation de $\tau_\varphi, \Delta E_2(x_e, B)$	146
Figure VI.11 Schéma du réseau de neurone pour l'adaptation de $\tau_\theta, \Delta E_3(x_e, B)$	146
Figure VI.12 Schéma du réseau de neurone pour l'adaptation de $\tau_\psi, \Delta E_4(x_e, B)$	147
Figure VI.13 Suivi de la trajectoire en 3D (2 ^{ème} trajectoire)	148
Figure VI.14 Réponses des coordonnées de la position (2 ^{ème} trajectoire)	149
Figure VI.15 Réponses des coordonnées de rotation (2 ^{ème} trajectoire)	149
Figure VI.16 Les entrées de commande (2 ^{ème} trajectoire)	150

Liste des Tableaux:

Tableau III.1 Les paramètres du OS4	44
Tableau III.2 Les paramètres des algorithmes	54
Tableau III.3 Valeurs de la fonction-objectif (ISE)	59
Tableau III.4 Les gains des correcteurs PID optimaux	60
Tableau III.5 Le critère ISE de la trajectoire	63
Tableau IV.1 Les paramètres du correcteur Backstepping intégral	76
Tableau V.1 Les intervalles des paramètres variés	103
Tableau V.2 Les valeurs nominales du pendule inversé sur chariot	111
Tableau V.3 Les intervalles des paramètres variés	112
Tableau VI.1 Les valeurs nominales du quadrotor utilisé	135
Tableau VI.2 Les coefficients de la matrice d'adaptation A	141
Tableau VI.3 Comparaison de l'ISE entre les trois correcteurs	150
Tableau VI.4 Comparaison de l'ISEF entre les trois correcteurs	150
Tableau VI.5 Comparaison entre les différents correcteurs	151

Abréviations:

BMW	La technique Meilleur-Accoupler-Movais (Best-Mate-Worst)
BS	La technique BackStepping
CFD	Diagramme de Flux de Commande (Control Flow Diagram)
CS	La Recherche du Coucou (Cuckoo Search)
CTC	Commande par Couple Calculé (Computed Torque Control)
DE	Évolution Différentielle (Differential Evolution)
DOF	Degré de Liberté (Degrees Of Freedom)
EA	Algorithmes Évolutionnaires (Evolutionary Algorithms)
ES	Stratégies Évolutionnaire (Evolutionary Strategies)
ESC	Régulateur Electronique du Vitesse (Electronic Speed Controller)
FO	Ordre Fractionnel (Fractional Order)
FTSMC	Commande par Mode Glissant Terminal Rapide (Fast Terminal Sliding Mode Control)
GA	Algorithmes Génétiques (Genetic Algorithms)
H.	Norme d'espace de Hardy (Hardy space norm)
HJ	Hamilton Jacobi
HJBI	L'équation Hamilton Jacobi Bellman Isaacs
IAE	Intégrale de la valeur Absolue de l'Erreur (Integral of Absolute magnitude of the Error)
IBS	BackStepping Intégral (Integral BackStepping)
IBS-H_∞ NL	Integral BackStepping H_∞ Non-Linéaire
IBS-H_∞ NL Adapt.	Integral BackStepping H_∞ Non-Linéaire Adaptative
IBS-H_∞ NL Adapt.-RN	Integral BackStepping H_∞ Non-Linéaire Adaptative Réseau de Neurones
IQC	Contrainte Quadratique Intégrale (Integral Quadratic Constraint)
ISE	Intégrale de l'Erreur Carrée (Integral of the Squared Error)
ISEF	Intégral d'Effort Carré (Integral Square Effort)
ITAE	Intégrale de Temps multiplié par la valeur Absolue de l'Erreur (Integral of Time multiplied by Absolute Error)
ITSE	Intégrale de Temps multiplié par l'Erreur Carrée (Integral of Time multiplied by the Squared Error)
LFT	Transformation Fractionnel Linéaire (Linear Fractional Transformation)

LMI	Inégalité des Matrices Linéaires (Linear Matrices Inequality)
LPV	Linéaire à Paramètres Variés (Linear Parameter Varying)
LQG	Commande Linéaire Quadratique Gaussienne (Linear Quadratic Gaussian)
LQR	Commande Linéaire Quadratique (Linear Quadratic Regulator)
LTI	Linéaire Invariant dans le Temps (Linear Time Invariant)
MPC	Commande Prédicative (Model Predictive Control)
MSE	Moyenne de l'Erreur Carrée (Mean Squared Error)
NP	Non-déterministe Polynomial (Non-deterministic Polynomial-time)
OS4	Robot Étiré de Vol Stationnaire Omnidirectionnel (Omnidirectional Stationary Flying Outstretched Robot)
PDE	Équations aux Dérivées Partielles (Partial Differential Equations)
PID	Proportionnel Intégral Dérivé (Proportional Integral Derivative)
PSO	Optimisation par Essaims Particulaires (Particle Swarm Optimization)
SMC	Commande Mode Glissant (Sliding Mode Control)
SO	Orthogonal Spécial (Special Orthogonal)
TSM	Commande par Mode Glissant Terminal (Terminal Sliding Mode)
UAV	Véhicules Aériens Sans-Pilote (Unmanned Aerial Vehicles)
VTOL	Décollage et Atterrissage Vertical (Vertical Take Off and Landing)

Introduction Générale:

Le développement des véhicules aériens sans pilote (*Unmanned Aerial Vehicles*: UAV) a suscité un grand intérêt pour la commande automatique au cours des dernières décades. De nombreux domaines de la commande et de la robotique ont été exploités pour améliorer les performances de ce type de systèmes. Les UAV ont été utilisés dans des domaines aussi bien militaires que civils, en se concentrant sur des tâches telles que la recherche et le sauvetage, l'exploration des routes, la sécurité, l'inspection, et la cinématographie aérienne, ainsi que des manœuvres acrobatiques. Les UAV sont surtout utiles lorsque ces tâches souhaitées sont exécutées dans des environnements dangereux et inaccessibles.

La construction d'un véhicule aérien autonome était un rêve pour beaucoup de chercheurs, qui étaient limités par les restrictions imposées par la déficience du matériel existant. Cependant, ce qui a vraiment permis la construction réussie des véhicules aériens autonomes, c'est les progrès technologiques réalisés dans les actionneurs et les capteurs à petite échelle, appelés systèmes micro-électromécaniques (MEMS), ainsi que les données de stockage et de traitement de l'énergie.

La mise au point des systèmes de commande pour ces véhicules n'est pas insignifiante, vue que les UAV ont un comportement non-linéaire et variable dans le temps et sont affectés par les perturbations aérodynamiques. De plus, ils sont généralement soumis à une dynamique non-modélisée et à des incertitudes paramétriques. Cela signifie que les lois de commande classiques linéaires et monovariabiles peuvent avoir une certaine limitation, provoquant une instabilité lorsque le système fonctionne dans des conditions éloignées de l'équilibre. Par conséquent, des stratégies de commande avancées sont nécessaires pour obtenir de bonnes performances en vol autonome ou au moins pour aider au pilotage du véhicule, avec une maniabilité et une robustesse élevées en ce qui concerne les perturbations.

La plupart de ces systèmes sont des systèmes mécaniques sous-actionnés (c'est-à-dire qu'ils ont moins d'entrées de commande que de degrés de liberté) qui apportent une complexité et un défi à la commande. Les techniques développées pour les robots entièrement actionnés ne peuvent pas être appliquées directement à ces types de systèmes mécaniques, car la plupart des systèmes sous-actionnés ne sont pas entièrement linéarisables. C'est pourquoi, les techniques de modélisation non-linéaire et la théorie moderne de la commande non-linéaire sont généralement utilisées pour réaliser des vols autonomes très performants et dans des conditions de vol spécifiques, telles que le vol stationnaire et de l'atterrissage / décollage.

En effet, les UAV avec une configuration à quatre moteurs (le quadrotor) ont été mis en évidence ces dernières années et ceci dans de nombreux travaux. Ces derniers ont permis de constater que les systèmes réalisés présentent certaines caractéristiques avantageuses par rapport à celles de l'hélicoptère standard, telles que:

- Le mouvement du quadrotor est basé sur quatre rotors, ce qui permet de réduire la taille de chaque rotor et de maintenir ou d'augmenter la capacité de charge totale par rapport à un hélicoptère à un rotor principal.
- Ces véhicules (les quadrotors) ne nécessitent pas de liaisons mécaniques pour agir sur les hélices. Cela réduit la conception, la maintenance et le coût du véhicule.
- Ces véhicules sont bien intéressants pour une utilisation à l'intérieur en raison de l'usage des moteurs électriques à la place des moteurs à combustion, ce qui évite la pollution de l'air liée à la combustion des déchets.
- Les quadrotors sont basés sur le concept VTOL (*Vertical Take Off and Landing*). Ils tentent d'atteindre un vol stationnaire et stable en utilisant les forces d'équilibre produites par quatre rotors.

Les avantages précédents, ajoutés à sa grande maniabilité, permettent au quadrotor des décollages et atterrissages, ainsi que des vols dans des environnements difficiles et dangereux.

Objectif:

Cette thèse présente une contribution au développement et à la mise en œuvre des stratégies de commande robustes en vue de résoudre le problème de suivi de trajectoire des véhicules aériens autonomes. Le véhicule utilisé est un mini-drone de type quadrotor caractérisé par un système mécanique sous-actionné.

Le premier objectif formulé ici est l'obtention d'un modèle dynamique approprié de l'hélicoptère à quadrimoteurs en tenant compte des objectifs de conception de commande, en plus d'une considération à un compromis entre complexité et réalisme. Le modèle sera basé sur des lois physiques pour obtenir une représentation du comportement du véhicule en présence de plusieurs sources d'incertitude. Le premier modèle est donné sous forme simplifiée alors que le deuxième modèle présente un modèle détaillé du fait que la méthode exploitée est fortement basée sur le modèle.

Étant donné que le quadrotor est un système mécanique sous-actionné, un moyen courant de suivre la trajectoire d'un UAV consiste à utiliser des stratégies de commande en cascade. Par conséquent,

des structures en cascade sont utilisées pour commander deux sous-systèmes: le rotationnel et le translationnel. Les techniques de commande utilisées dans chaque boucle seront basées à la fois sur la technique linéaire classique PID avec une optimisation des paramètres au moyen d'algorithmes évolutionnaire, sur l'approche backstepping et aussi sur l'approche par mode glissant. En outre, la technique backstepping est combinée par l'approche basée sur la théorie des systèmes linéaires à paramètres variables (LPV) afin d'obtenir une commande robuste en boucle fermée où le correcteur H_∞ LPV garantit la robustesse du sous-système de rotation.

Cependant, concernant les systèmes sous-actionnés, il faut souligner que les stratégies de commande en cascade possèdent un inconvénient qu'on n'a pas pu démontrer la stabilité de l'ensemble de la boucle fermée tient un comportement stable. Pour éviter d'utiliser des structures en cascade, une stratégie de commande basée sur la technique H_∞ non-linéaire appliquée aux systèmes mécaniques sous-actionnés sera réalisée. L'objectif est d'obtenir une loi de commande qui garantit la robustesse du problème de suivi de trajectoire du quadrotor sans recourir à des stratégies en cascade.

Un autre problème à résoudre est l'amélioration de la robustesse du correcteur H_∞ non-linéaire conçu pour les systèmes mécaniques. Cette loi de commande est calculée en tenant compte du fait que toutes les incertitudes qui affectent le système sont des perturbations externes. Cependant, cette hypothèse n'est en effet pas très réaliste. Par conséquent, pour résoudre ce problème, une solution permettant de renforcer la loi de commande non-linéaire H_∞ sera présentée, où un terme d'adaptation supplémentaire est calculé à l'aide de propriété de paramétrage linéaire et des estimateurs à base des réseaux de neurones.

Plan de travail:

La thèse est organisée comme suit:

- Chapitre 1 surligne sur l'état de l'art des projets récents du système du quadrotor et résume les applications des quadrotors ainsi que les avantages et les inconvénients de l'utilisation des quadrotors et les défis relevés dans les travaux présentés.
- Chapitre 2, les différentes stratégies appliquées et exploitées dans cette thèse vont présenter avec leur état de l'art dans la littérature; la commande robuste des systèmes linéaires à paramètres variables (LPV) et la commande robuste H_∞ non-linéaire.
- Chapitre 3 traite des stratégies de commande classique linéaire en cascade pour effectuer le suivi de trajectoire du véhicule, ce qui met en évidence la poursuite de l'amélioration continue des

performances. Dans ce chapitre, le modèle dynamique du quadrotor est effectué et puis une structure de commande PID est développée, les paramètres des correcteurs sont optimisés par quatre algorithmes méta-heuristique (GA, ES, DE et CS).

- Chapitre 4 traite des stratégies de commande classiques non-linéaires en cascade pour effectuer le suivi de trajectoire du quadrotor. Dans ce chapitre, deux techniques de commande sont développées; backstepping intégral et mode glissant, et finalement un teste de robustesse est effectué.

- Chapitre 5 présente la commande robuste par la technique LPV qui a été utilisé pour stabiliser le sous-système de rotation. Pour le suivi de trajectoire, le correcteur backstepping a été utilisé pour le sous-système de translation et par conséquent, fournit les signaux désirés des angles.

- Chapitre 6 présente le développement d'un correcteur H_∞ non-linéaire. Un terme d'adaptation supplémentaire est calculé à l'aide de propriété de paramétrage linéaire et des estimateurs à base des réseaux de neurones afin d'améliorer le comportement du suivi de la trajectoire.

Chapitre I:

*Etat de l'Art sur les Quadrotors; Commande et
Applications*

I.1 Introduction:

Au cours des dernières décades, les quadrotors ont acquis une énorme popularité dans la communauté de la robotique, chez les amateurs et sur le marché commercial. Ils sont des plateformes aériennes de conception mécanique simple pouvant être assemblées à partir de composants bon marché adaptés à un large éventail d'applications. Les quadrotors sont généralement construits à partir de quatre moteurs avec des hélices à pas fixe disposées dans une configuration en croix où tous les axes des moteurs sont parallèles les uns aux autres et où les hélices voisines tournent dans des directions opposées. Les quatre forces poussées des rotors générées peuvent être commandées individuellement en modifiant la vitesse du moteur respectif. Cela permet au quadrotor de produire une force poussée collective perpendiculaire au plan des rotors et des couples le long des trois axes du corps.

I.2 Les Quadrotors dans la Robotique:

Cette section résume les applications des quadrotors en plus des avantages et des inconvénients de leur utilisation, ainsi que les défis relevés dans les travaux présentés en littérature.

I.2.1 Les Applications des Quadrotors:

Du fait que les drones, en particulier les quadrotors, sont devenus abordables et fiables il y a environ une décade, leur nombre d'applications a augmenté très rapidement, couvrant désormais pratiquement toutes les tâches qui impliquent le déplacement, par exemple, de capteurs ou de charges utiles dans des endroits difficiles à atteindre, voire inaccessibles pour les humains et les véhicules de sol. Les marchés les plus importants pour les drones sont ceux de l'inspection et de l'agriculture. Les drones sont utilisés avec succès pour l'inspection de lignes électriques, de ponts, de pipelines (Ozaslan et al., 2017), de chemins de fer et de nombreuses autres infrastructures. Dans l'agriculture, les drones sont utilisés pour l'analyse des sols, surveillance, plantation, et pulvérisation des cultures (Das et al., 2015). En outre, les drones sont utilisés pour les missions de recherche et de sauvetage, où ils sont utilisés pour la cartographie et l'analyse des catastrophes ou même pour la recherche de victimes (Murphy et al., 2014) ou la sécurité et la surveillance (Perez et al., 2012; Pasqualetti et al., 2012; Merino et al., 2011). En plus, les drones sont largement utilisés par les amateurs, comme c'est le cas de la photographie aérienne, qui est également devenue un grand marché. À l'avenir, les drones devraient être largement utilisés pour la livraison de colis, pour lesquels, déjà aujourd'hui, de nombreux investissements et recherches sont en cours. De plus, des drones devraient être déployés pour des missions de construction (Alejo et al., 2013; Jimenez-Cano et al., 2013).

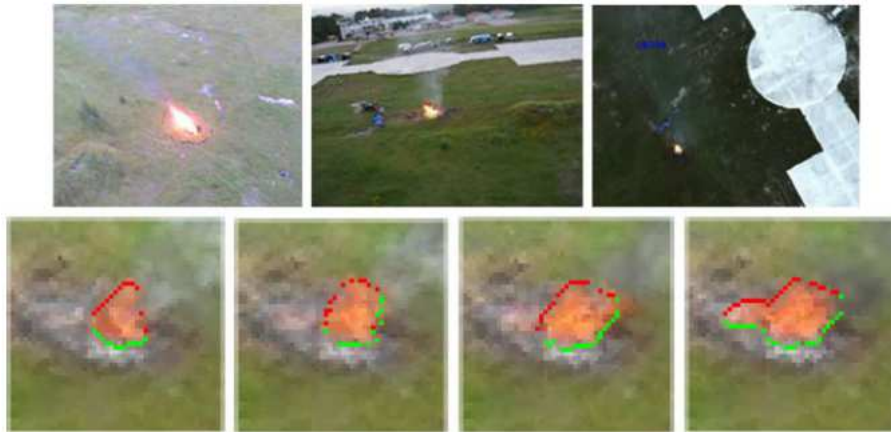


Figure I.1 Des images acquies d'une drone de surveillance des feux (Merino et al., 2011)

En résumé, la plupart des applications des quadrotors sont destinées à des tâches passives, c'est-à-dire au déplacement des capteurs aux emplacements souhaités dans un espace 3D. De telles tâches sont difficiles et coûteuses à accomplir sans drones, mais conviennent également particulièrement aux quadrotors. Il existe donc un besoin énorme d'exploiter davantage le potentiel des quadrotors autonomes. Pour rendre les quadrotors encore plus utiles et plus rentables, il est essentiel d'augmenter leur niveau d'autonomie et la rapidité avec laquelle ils peuvent fonctionner, car cela réduit le nombre d'opérateurs requis et accélère l'achèvement de leur tâche.

I.2.2 Les Avantages:

L'un des principaux avantages des quadrotors par rapport aux autres types de véhicules aériens de petite taille et, ce qui explique leur popularité, réside dans leur conception mécanique simple et robuste. Ils sont construits à partir d'un cadre rigide en forme de croix, qui peut être conçu léger mais très résistant aux chocs. Comparés aux autres multi-rotors, où tous les rotors sont parallèles, les quadrotors utilisent un nombre minimal de moteurs pour commander les degrés de liberté, ils, par conséquent, constituent la configuration la plus simple de ces types de multi-rotors. Les moteurs à hélices sont les seules pièces mobiles du véhicule. Étant donné que ces hélices ont généralement un pas de pale fixe, leur force poussée peut être commandée en modifiant la vitesse du moteur, ce qui est effectué par les correcteurs de vitesse électroniques (ESC). Par conséquent, cette configuration ne nécessite pas de pièces mécaniques complexes, telles qu'un plateau en biais utilisé sur les hélicoptères. Cela les rend particulièrement adaptés à la recherche, car ils sont peu coûteux et offrent une grande flexibilité en termes de taille et de poids, ce qui est nécessaire pour utiliser de nouveaux capteurs et configurations, comme c'est souvent le cas dans les expériences de recherche.

En plus de leur conception simple et robuste, les quadrotors ont des capacités de vol favorables par rapport aux autres types de véhicules aériens. Les quadrotors peuvent décoller et atterrir verticalement et ne nécessitent donc qu'un espace minimal pour le démarrage et l'atterrissage. Ces engins peuvent rester en vol sans avoir à se déplacer, ce qui est nécessaire pour de nombreuses tâches, telles que l'inspection aérienne, la surveillance ou la photographie. En plus du vol stationnaire, les quadrotors peuvent être conçus pour avoir une structure rigide et un rapport poussée / poids très élevé, ce qui leur permet de réaliser des manœuvres agiles.

En résumé, les quadrotors conviennent parfaitement à la recherche en robotique aérienne car ils sont simples et peu coûteux à concevoir, robustes contre les collisions, peuvent décoller et atterrir à la verticale, tout en restant très agiles.

I.2.3 Les Défis:

Les quadrotors sont des systèmes instables, c'est-à-dire qu'un vol stationnaire stable ne peut pas être obtenu en appliquant une entrée constante aux quatre moteurs. Pour stabiliser un quadrotor, les entrées de chaque moteur doivent être commandées à une fréquence suffisamment élevée, ce qui dépend de ses caractéristiques dynamiques. Le quadrotor de petite taille est plus agile, du fait que l'accélération angulaire maximale est inversement proportionnelle à la taille du quadrotor, comme c'est indiqué dans (Mulgaonkar et al., 2015).

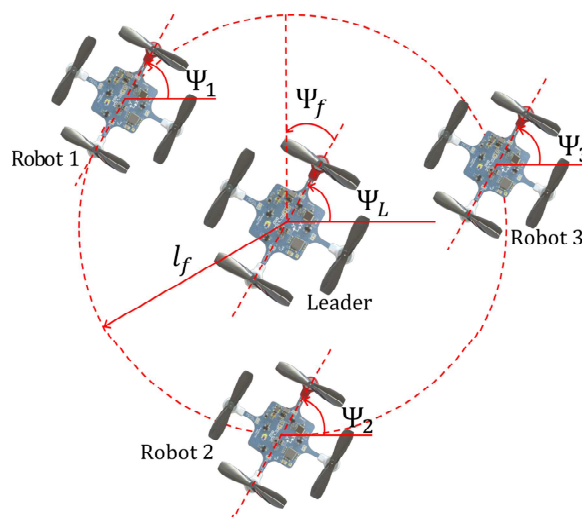


Figure I.2 Les coordonnées de la formation Δ leader-follower (Mulgaonkar et al., 2015)

Cela pourrait être exploité pour un vol agile sauf qu'il nécessite également des fréquences de commande plus élevées pour faire face à la dynamique la plus rapide du véhicule. En outre, le quadrotor de petite taille est moins efficace en raison de sa très petite surface de disque du rotor, qui

décroît quadratiquement avec la taille du quadrotor. Pour ces raisons, il est difficile de trouver la configuration optimale d'un quadrotor pour une tâche donnée, car il s'agit d'un compromis entre le temps de vol réalisable, l'agilité ainsi que les capteurs et les ordinateurs pouvant être transportés.

Les quadrotors sont des systèmes sous-actionnés, car ils disposent de quatre entrées de commande pour commander leurs six degrés de liberté. Ce n'est pas un problème pour le vol libre puisque seuls quatre degrés de liberté indépendants doivent être commandés. Cependant, les quadrotors ne sont pas adaptés aux interactions avec leur environnement car ils ne sont pas capables d'exercer des forces indépendamment de leur orientation. Cela peut entraîner des problèmes lors du décollage d'une surface inclinée par exemple, et doit être pris en compte dans la procédure de décollage (Tognon et al., 2016).

Par conséquent, pour les interactions avec l'environnement, les véhicules aériens pouvant exercer des forces sur plusieurs axes sont généralement utilisés, tels que les hélicoptères (Kondak et al., 2013) ou les hexarotors à hélices inclinées (Ryll et al., 2017). En outre, lors de vols agiles proches des limites physiques d'un quadrotor, des moteurs peuvent être saturés, ce qui entraîne une incapacité à commander les quatre degrés de liberté indépendants.

I.3 Etat de l'Art sur la Commande du Quadrotor:

I.3.1 Les Systèmes de Type Quadrotor:

On a des systèmes de recherche sur la commande du quadrotor et l'estimation de l'état des drones ont été mis au point dans de nombreux groupes de recherche du monde entier. En particulier, les travaux de (Lupashin et al., 2014; Michael et al., 2010) conçus pour la recherche sur la commande des quadrotors ont servi de base et d'inspiration au développement du système à quadrotors du groupe *Robotics and Perception* (Fässler, 2018). Le système présenté dans (Grabe et al., 2013) insiste beaucoup sur la flexibilité d'utilisation de quadrotors avec différentes interfaces pour les entrées d'utilisateur, différentes méthodes de commande et différentes estimations d'état. Le travail de (Fässler, 2018) a également influencé le système qui vise à utiliser des quadrotors avec des estimations d'état provenant d'un système de capture de mouvement ou d'algorithmes visuelles.



Figure I.3 Le plateforme quadrotor du groupe *Robotics and Perception* (Fässler, 2018)

I.3.2 Méthodes de Commande du Quadrotor:

La structure de commande la plus courante pour les quadrotors consiste en deux boucles de commande, l'une pour la position et l'autre pour l'attitude. Une telle architecture est présentée dans tous les systèmes aux quadrotors mentionnés ci-dessus. La boucle de commande de position de haut niveau (extérieur) implémente généralement une loi de commande PD sur la position et la vitesse avec des termes d'anticipation pour compenser la gravité et les accélérations à partir d'une trajectoire de référence. La sortie de cette boucle de régulation est une accélération désirée qui génère l'attitude désirée. L'attitude désirée est ensuite commandée par une boucle de commande de bas niveau (intérieur) basée sur l'hypothèse que la dynamique d'attitude d'un quadrotor est beaucoup plus rapide que sa dynamique de position et de vitesse. Une étude des représentations d'attitude et de leur aptitude à commander l'attitude d'un corps rigide est présentée dans (Chaturvedi et al., 2011). Dans la littérature de la commande du quadrotor, l'attitude est représentée sous la forme d'un quaternion unitaire ou d'une matrice de rotation. Sur la base de ces représentations, il est possible de concevoir des correcteurs d'attitude globalement asymptotiquement stables. Les méthodes de commande du quadrotor consistant en une boucle de commande de position par PD de haut niveau et en une boucle de commande d'attitude par PD de bas niveau basée sur des matrices de rotation, sont présentées dans (Mellinger et Kumar, 2011; Lee et al., 2010). Les correcteurs d'attitude ayant des propriétés similaires basées sur les quaternions unitaires sont présentés dans (Fresk et Nikolakopoulos, 2013; Bangura et al., 2014). Étant donné que pour toutes ces méthodes, la stabilité asymptotique du correcteur de position de haut niveau et du correcteur d'attitude de bas niveau peut être montrée, le système interconnecté avec les deux boucles est également asymptotiquement stable (Khalil, 1996).

Souvent, la commande d'attitude ne peut pas être réalisée par une seule boucle de commande en raison du matériel utilisé avec plusieurs unités de traitement. En règle générale, deux unités de

traitement sont intégrées aux quadrotors, où un ordinateur de bord effectue une estimation d'état et une commande de haut niveau, et un micro-contrôleur moins puissant mais en temps réel est utilisé pour les niveaux bas de commande. Comme la commande d'attitude nécessite une estimation des états qui n'est disponible que sur l'ordinateur de bord. Pour obtenir une commande précise, des entrées de moteur rapides et à faible temps de latence sont nécessaires, ce qui ne peut être obtenu que par un micro-contrôleur en temps réel. Par conséquent, la commande l'attitude d'un quadrotor est souvent réalisée par deux boucles de commande en cascade; Par exemple dans (Kumar et Gil, 2017; Brescianini et al., 2013a), la première boucle commande l'attitude à basse fréquence et la deuxième boucle commande les vitesses à haute fréquence. Ce n'est pas préférable à une seule boucle de commande, mais elle est souvent nécessaire en raison de la configuration matérielle utilisée.

Dans (Mellinger et Kumar, 2011), il a été montré que le modèle commun d'un quadrotor sans tenir compte des effets de traînée du rotor est différentiellement plat lors du choix de sa position et de sa rotation en tant que sorties plates. En outre, ce travail a présenté un algorithme de commande qui calcule les entrées de poussée et des couples à partir des erreurs de position, de vitesse et d'orientation mesurées. Avec cette méthode, des manœuvres agiles avec des vitesses de plusieurs mètres par seconde ont été réalisées. Dans (Ferrin et al., 2011), la propriété de planéité différentielle d'un hexarotor prenant l'orientation désirée en tant qu'entrées ont été exploitées pour calculer les termes de *feed-forward* utilisés dans un correcteur LQR. L'orientation désirée était alors commandée par une boucle de commande séparée de bas niveau, ce qui permet également l'exécution de manœuvres en vol à des vitesses de plusieurs mètres par seconde. La propriété de planéité différentielle a été démontrée et utilisée pour la commande également pour de nombreuses configurations différentes de véhicules aériens, tels que les véhicules aériens sous-actionnés avec un nombre quelconque de bras manipulateurs parallèles différents fixés à leur centre de gravité (Yüksel et al., 2016). Dans (Fässler, 2018) l'auteur a utilisé cette propriété pour calculer les termes de *feed-forward* qui sont ensuite appliqués par un correcteur de position.

Dans (Mueller et D'Andrea, 2015), il est démontré que la position d'un quadrotor peut être stabilisée malgré la perte totale des trois hélices au maximum. Pour stabiliser l'orientation en cas de défaillance du rotor, il convient de prendre en compte plus d'hélices, tels que les hexarotors et les hélices potentiellement inclinées qui conviennent parfaitement comme plates-formes sûres avec redondance du rotor (Michieletto et al., 2017). Dans (Monteiro et al., 2016), une méthode d'allocation de commande partielle qui donne la priorité à l'application des couples désirés et la force poussée permet de traiter les entrées infaisables avant d'appliquer les commandes des rotors.

Les schémas de commande les plus communs composés de boucles en cascade pour la position et l'attitude, plusieurs autres méthodes de commande pour commander un quadrotor ont été proposées dans la littérature. Parmi ces méthodes, on trouve les régulateurs linéaires quadratiques (LQR), les correcteurs prédictifs (MPC) et les techniques de commande optimales. Les correcteurs basés sur LQR peuvent être implémentés en tant que correcteurs à retour d'état complet pour la commande de position et la commande d'attitude (Reyes-Valeria et al., 2013). De plus, il a été démontré que les méthodes LQR étaient appropriées pour commander des systèmes à plusieurs corps, tels que des quadrotors portant une charge en élingue (Decrousaz et al., 2015) ou pour équilibrer un pendule inversé (Hehn et D'Andrea, 2011; Brescianini et al., 2013b). Dans (Kamel et al., 2015), un correcteur d'attitude prédictif de modèle rapide est proposé, qui, associé à un régulateur quadratique linéaire de haut niveau, permet de suivre les trajectoires agressives. Des correcteurs prédictifs ont également été utilisés en tant que correcteurs de haut niveau avec une boucle de commande d'attitude séparée (Raffo et al., 2008; Mueller et D'Andrea, 2013; Bangura et Mahony, 2014). De plus, les méthodes de génération rapide de trajectoires peuvent être exploitées selon la commande prédictive, en appliquant uniquement la première commande résultant d'une trajectoire planifiée, puis en reprogrammant une nouvelle trajectoire pour la prochaine itération pour la boucle de commande (Hehn et D'Andrea, 2015). Des méthodes de commande optimale ont été utilisées dans (Ritz et al., 2011; Hehn et al., 2012) pour trouver des trajectoires optimales dans le temps pour la transition entre deux états. La structure d'une manœuvre de transition est obtenue par le principe du minimum de Pontryagin, puis un algorithme numérique est proposé pour résoudre le problème des valeurs limites induit par le principe du minimum permettant de calculer les manœuvres pour des états initiaux et finaux arbitraires. Basée sur un modèle quadrotor à deux dimensions, cette méthode trouve les trajectoires *bang-bang* dans la commande de force poussée et *bang-singulier* dans la commande de la vitesse. Un algorithme de commande optimale itératif qui trouve simultanément une trajectoire et un correcteur de stabilisation est proposé dans (Decrousaz et al., 2015). Les capacités de cet algorithme sont démontrées en effectuant un quadrotor avec une charge pour laisser passer la charge en étant suspendu vers le bas, ainsi que par des tâches allant au but avec des défaillances à un ou deux rotors.

I.3.3 Suivi Précis de La Trajectoire:

Il est de plus en plus difficile de suivre des trajectoires avec un quadrotor tenant une augmentation de la vitesse en raison d'erreurs de modélisation, et qui deviennent plus pertinentes à des vitesses et accélérations plus élevées. Une façon de surmonter cette limitation consiste à répéter une trajectoire donnée et à apprendre de manière itérative comment contrôler les entrées qui doivent être adaptées afin d'améliorer la précision du suivi de trajectoire. Dans (Lupahsin et D'Andrea, 2012), les multi-

retournements en boucle ouverte et les manœuvres de translation optimales dans le temps ont été apprises en les répétant et en adaptant les paramètres de trajectoire. Dans (Hehn et D'Andrea, 2014), un schéma d'apprentissage itératif au domaine des fréquences pour l'apprentissage de manœuvres périodiques avec des quadrotors est présenté. Le schéma d'apprentissage proposé prend les erreurs de commande en entrée et génère un décalage de point de consigne qui améliore la précision du suivi de trajectoire.



Figure I.4 Un quadrotor en équilibrant un pendule inversé (Hehn et D'Andrea, 2013)

En outre, un procédé de mise à l'échelle temporelle permet le transfert de manœuvres apprises à différentes vitesses d'exécution via une prédiction du changement de perturbation. Les erreurs de suivi obtenues après l'apprentissage d'une trajectoire sous forme infini (∞) étaient similaires aux écarts entre différentes tests de la trajectoire. Cette approche a également été appliquée pour voler à des cercles rapides avec un quadrotor tout en équilibrant un pendule inversé dans (Hehn et D'Andrea, 2013). Contrairement à l'apprentissage de la correction d'entrée, dans (Schoellig et al., 2012a), les paramètres de mouvement d'une trajectoire périodique sont ajustés pour améliorer le suivi temporel. Les paramètres de mouvement sont soit estimés en-ligne ou hors-ligne avant le vol pour éviter les transitoires initiaux. De même, (Schoellig et al., 2012b) estime les perturbations dues aux erreurs de modélisation le long d'une trajectoire désirée à l'aide d'un filtre de Kalman, sur la base des perturbations estimées. Pour surmonter le problème de ne pas pouvoir généraliser à différentes trajectoires, (Hamer et al., 2013) montre que les dynamiques principales du processus de commande d'apprentissage itératif peuvent être capturées par une carte linéaire, qui peut ensuite être utilisée pour améliorer l'initialisation des trajectoires d'apprentissage non vues. Une approche

différente pour apprendre à améliorer le suivi pour des trajectoires invisibles est adoptée dans (Li et al., 2017; Zhou et al., 2017), où un réseau de neurones profond est utilisé comme bloc additionnel à un correcteur de retour d'état classique. Le réseau de neurones profond est formé hors-ligne sur des données enregistrées à partir d'un ensemble de trajectoires d'entraînement et permet une réduction significative de l'erreur de suivi de trajectoire.

Contrairement à l'apprentissage des entrées de commande qui compensent les erreurs de modélisation, la considération d'un modèle dynamique plus précis dans le correcteur devrait obtenir des résultats similaires.

L'amélioration du modèle dynamique présente l'avantage qu'une fois les paramètres du modèle identifiés, toute trajectoire peut être parcourue avec plus de précision sans l'apprentissage itératif.

Dans (Yu et al., 2015), les correcteurs PID en cascade sont conçus et améliorés à l'aide de prédicteurs de Smith afin d'intégrer la dynamique des moteurs pour une commande complète d'attitude du quadrotor sur $SO(3)$ (*Special Orthogonal group*, $SO(3) = \mathbb{R} \in \mathbb{R}^{3 \times 3}$). Un correcteur d'attitude LQR pour un seul axe, étendu avec une dynamique des moteurs du premier ordre, est présenté dans (Kawai et Uchiyama, 2016; Wang et al., 2016). L'intégration d'effets de traînée aérodynamique dans le modèle dynamique d'un quadrotor sera décrite dans la section suivante.

I.3.4 L'Aérodynamique du Quadrotor:

Grâce à la théorie bien établie des hélicoptères de grande taille, l'influence des effets aérodynamiques sur la dynamique des quadrotors a été considérée depuis le début de la recherche sur les quadrotors (Pounds et al., 2006; Hoffmann et al., 2007). Depuis lors, la modélisation de ces effets pour les quadrotors a été intensément étudiée à partir de la théorie bien établie des hélicoptères de taille normale. Dans (Bangura et al., 2016), des modèles très détaillés des forces aérodynamiques des rotors utilisés sur des quadrotors à petite échelle ont été dérivés en utilisant la théorie de la quantité de mouvement et la théorie des éléments de pale.

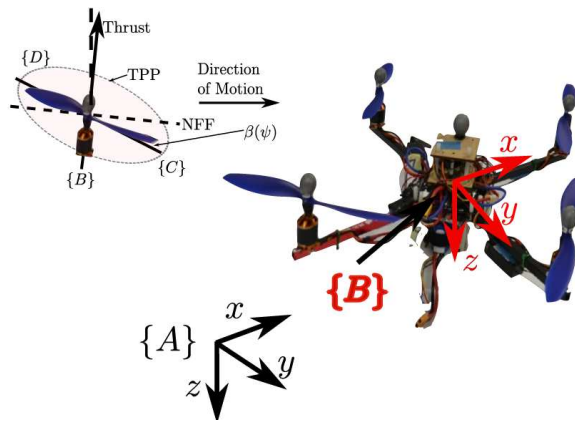


Figure I.5 Un schéma montré les pales inclinés (Bangura et al., 2016)

Dans (Bristeau et al., 2009; Martin et Salaün, 2010), les hélices ont été modélisées avec la théorie des éléments de pale montrant que la force principale résultante est linéaire en vitesse de translation. Ces travaux mettent en évidence l'influence de l'emplacement vertical du centre de gravité d'un quadrotor sur la stabilité de celui-ci, ainsi que ses conséquences sur le retour du correcteur. En plus de l'étude de l'aérodynamique des quadrotors en vol vertical et en avant, (Powers et al., 2012) étudie également les effets de proximité tels que les effets au sol et ceux des véhicules voisins. Dans (Schulz et al., 2015; Gill et D'Andrea, 2017), un quadrotor attaché au plafond est utilisé pour identifier des paramètres aérodynamiques tels que des coefficients de traînée linéaires, ainsi que des variations de force poussée dues à la vitesse. Grâce à ça, le quadrotor atteint des vitesses très élevées sur un cercle horizontal malgré l'espace disponible limité. Un cadre d'estimation permettant d'identifier les paramètres de modèle d'un véhicule à rotors multiples, y compris les paramètres aérodynamiques, est proposé dans (Burri et al., 2017). Ce travail montre que la traînée est négligeable à une vitesse inférieure à environ 5 m/s , bien qu'elle soit inférieure aux effets de traînée linéaires du rotor au cours des expériences menées dans une vitesse de vent maximale de 15 m/s .

Le modèle dynamique d'un quadrotor utilisé dans (Fässler, 2018) est basé sur ces travaux et utilise notamment le modèle présenté dans (Kai et al., 2017), y compris les effets de traînée de rotor linéaires. Ces effets de traînée de rotor proviennent de la traînée induite des rotors, qui, comme suggéré dans (Mahony et al., 2012), sont combinés en effets linéaires dans un modèle dynamique à paramètre global, grâce à leur expression mathématique équivalente.

Déjà, les travaux de (Bristeau et al., 2009; Martin et Salaün, 2010; Bangura et al., 2016) suggéraient que les modèles dynamiques de quadrotors incorporant des effets de traînée linéaires pourraient être utilisés pour l'estimation de la vitesse, où (Bristeau et al., 2009; Martin et Salaün, 2010) le montrent

expérimentalement. Dans (Leishman et al., 2014), un modèle du quadrotor incluant des effets de traînée linéaires est utilisé pour améliorer l'estimation d'état avec un filtre de Kalman étendu. Ce travail montre que l'estimation de l'attitude et de la vitesse peut être améliorée même en utilisant uniquement une unité de mesure inertielle et aucune mesure de position.

De même, (Burri et al., 2015) a proposé un schéma d'estimation d'état basé sur un modèle pour les micro-véhicules aériens basé sur un filtre de Kalman étendu. En utilisant uniquement les mesures d'une unité de mesure inertielle et d'un capteur de pression barométrique, ce schéma d'estimation maintient la vitesse estimée limitée, ce qui est crucial lorsqu'on utilise cette estimation d'état comme un support si aucune estimation de position n'est disponible. En plus de l'estimation des vitesses horizontales, (Allibert et al., 2014) étend l'estimation à la mesure de la vitesse de corps fixe en exploitant leurs travaux antérieurs sur la modélisation aérodynamique de la performance du rotor et la mesure de la puissance mécanique fournie au moyeu du rotor. Ce travail propose à un observateur non-linéaire d'estimer l'attitude et la vitesse linéaire fixée au corps à l'aide d'une unité de mesure inertielle et de correcteurs de vitesse électroniques qui mesurent la vitesse et le couple du rotor.

Il existe plusieurs approches pour incorporer des effets aérodynamiques dans la conception des commandes de quadrotors. Dans (Bangura et Mahony, 2017), les régulateurs de vitesse électroniques permettent d'obtenir une régulation de force poussée précise grâce à un modèle de la puissance aérodynamique générée par un rotor à pas fixe soumis à des perturbations dues au vent, ce qui réduit l'erreur de suivi de trajectoire d'un quadrotor.

Cependant, cette méthode nécessite un matériel spécifique et un étalonnage de longue durée par rapport à d'autres algorithmes (Franchi et Mallet, 2017) qui permettent une commande de la force poussée grâce à une commande rapide et précise de la vitesse des moteurs. Les effets de traînée de rotor ont été pris en compte dans les méthodes de commande pour les véhicules à multi-rotors (Kai et al., 2017; Omari et al., 2013), où le problème de commande a été simplifié en décomposant la force de traînée du rotor en un composant indépendant de l'orientation du véhicule et la direction de la force poussée, ce qui aboutit à une expression explicite de la direction de force poussée désirée. Tandis que (Kai et al., 2017) modélisent la traînée du rotor pour qu'elle soit proportionnelle à la racine carrée de la force poussée, laquelle est proportionnelle à la vitesse du rotor et à la relation physique correcte. (Omari et al., 2013) modélisent la traînée du rotor pour qu'elle soit proportionnelle à la poussée. Dans (Svacha et al., 2017), un modèle de poussée raffiné et un schéma de commande tenant compte de la traînée du rotor dans le calcul de la commande poussée et de l'orientation désirée sont présentés.

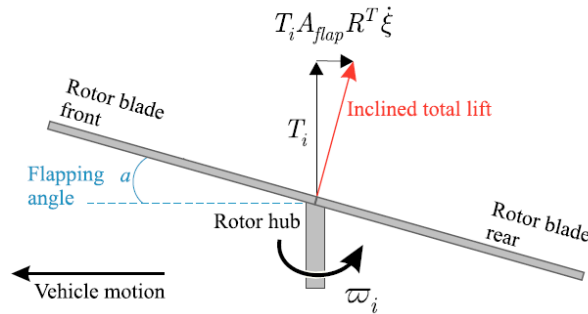


Figure I.6 Schéma de la traînée du rotor (Omari et al., 2013)

Cependant, ce schéma n'utilise pas de termes anticipés sur les vitesses et les accélérations angulaires, ce qui ne permet pas un suivi de trajectoire parfait. En plus de la commande de poussée et de l'orientation désirée, le schéma de commande de (Bangura, 2017) calcule les vitesses et les accélérations angulaires désirées en tenant compte de la traînée du rotor, mais nécessite des estimations de l'accélération du quadrotor, qui ne sont généralement pas disponibles.

En effet, le modèle dynamique d'un quadrotor soumis aux effets de traînée de rotor linéaires développé dans (Kai et al., 2017) est différentiellement plat dans sa position et dans sa direction. Cette propriété est ensuite utilisée pour calculer la force poussée de référence exacte, l'orientation, la vitesse du corps, et les accélérations angulaires directement à partir d'une trajectoire de référence à suivre, qui sont ensuite utilisées comme termes de *feed-forward* dans le correcteur (Fässler, 2018). Il en résulte une loi de commande relativement simple et intuitive qui permet théoriquement un suivi parfait des trajectoires à grande vitesse.

I.4 Conclusion:

Dans ce chapitre, nous avons abordé les quadrotors, leur commande et leur applications, ce qui a permis de constater que de domaine des mini drones, en particulier les quadrotors a connu un intérêt croissant par, soit les chercheurs ou les ingénieurs. Ce qui est aux évolutions des instruments de commande, de communication et des capteurs. Cependant, les quadrotors impliquent des défis en raison des caractéristiques de leur système dynamique; instable, couplé, non-linéaire, complexe, sensible, et sous-actionné. Par conséquent, on trouve des efforts qui sont effectués afin de surmonter ces difficultés par plusieurs techniques de commande; des méthodes linéaires, des techniques non-linéaires, optimales, et même des approches basées sur l'intelligence artificielle. Dans le chapitre prochain, nous allons fournir un état de l'art sur différentes techniques de commande robuste linéaire et non-linéaire.

Chapitre II:

État de l'Art sur Les Systèmes LPV et H_∞ Non- Linéaires

II.1 Introduction:

Nous avons considéré précédemment les différents projets concernant la commande du quadrotor, dont les caractéristiques dynamiques nécessitent une analyse particulière, afin de surmonter les difficultés rencontrées et de développer les techniques de commande adéquates. Dans ce chapitre, on présente un état de l'art des différentes stratégies de commande robuste présentées en littérature et qui seront appliquées et exploitées dans cette thèse. Premièrement, la commande robuste des systèmes linéaires à paramètres variables (LPV) qui a été choisie pour représenter une équivalence des systèmes non-linéaires, et par la suite, la commande robuste H_∞ non-linéaire basée sur la résolution des équations différentielles qui est appliquée et développée pour les robots manipulateurs.

II.2 Systèmes LPV:

L'objectif de cette section est de présenter les méthodes principales de représentation des systèmes linéaires à paramètres variables (*Linear-Parameter-Varying*: LPV) et de mettre en évidence leur capacité à représenter une classe vaste des systèmes dynamiques.

II.2.1 Approximation des Systèmes Non-Linéaires:

Les systèmes linéaires à paramètres variables (LPV) sont des systèmes dynamiques linéaires dont le modèle mathématique dépend des paramètres qui changent leurs valeurs dans le temps. Généralement, on considère que ces paramètres sont liés et prenant des valeurs dans un ensemble définie. Les paramètres peuvent être utilisés pour approximer la dynamique non-linéaire, intégrer de manière systématique des parties variant dans le temps, ou même introduire des degrés de liberté supplémentaires pouvant s'avérer utiles pour une perspective de conception.

Chaque fois que les systèmes LPV sont considérés comme des approximations de systèmes non-linéaires, les paramètres sont des fonctions des états du système. Ce type particulier de systèmes LPV est appelé systèmes quasi-LPV, parfois abrégé q-LPV (Shin, 2002; He et Yang, 2006; Jung et Glover, 2006; Wei et del Re, 2007; White et al., 2007).

Une difficulté supplémentaire importante est que les systèmes non-linéaires admettent généralement plusieurs approximations / représentations LPV, et trouver la plus précise n'est pas une tâche facile pour certaines méthodes d'approximation (Shin, 2002; Bruzelius et al., 2004; Mehendale et Grigoriadis, 2004). Une autre difficulté liée à la commande des systèmes quasi-LPV obtenue à partir de la linéarisation d'un système non-linéaire, réside dans la présence de termes de couplage cachés pouvant conduire à un système à boucle fermée instable même si le système LPV à boucle

fermée correspondant est stable. Des informations supplémentaires sur ces termes de couplage sont décrites dans (Lawrence et Rugh, 1995).

II.2.2 Représentation des Systèmes LPV:

Différents paradigmes de modélisation LPV ont été présentés dans la littérature. Étant donné que la forme générale du LPV découle directement d'une analyse et d'une commande robuste.

La formulation générique pour les systèmes LPV est la plus naturelle puisque les systèmes LPV sont directement pris, cette formule est trouvée dans (Apkarian et Tuan, 1998). La forme polytopique offre un moyen élégant et pratique de représenter et d'analyser les systèmes LPV et les systèmes incertains, plusieurs exemples ont été proposés dans les références suivantes; (Apkarian et al., 1995; Geromel et Colaneri, 2006; Borges et Peres, 2006; Oliveira et al., 2007; Chesi et al., 2009). Contrairement aux systèmes LPV génériques pour lesquels on ne suppose a priori aucune dépendance particulière vis-à-vis des paramètres, les systèmes polytopiques sont, par contre, explicitement représentés comme une combinaison convexe de systèmes LTI. Cette propriété structurelle peut être exploitée pour obtenir des résultats de stabilité et de stabilisation facilement vérifiables à l'aide des techniques d'optimisation convexes. Même si les systèmes polytopiques peuvent conduire à des descriptions inexacts pour certaines classes de systèmes LPV, ils restent intéressants du point de vue théorique et informatique, car certaines propriétés du système LPV considéré peuvent être connectées à celles des sous-systèmes, à savoir les systèmes extrêmes situés aux sommets du système polytope. Dans certains cas, la stabilité du système LPV polytopique peut être liée à la stabilité des sous-systèmes.

Les systèmes LPV sous forme LFT sont des systèmes LPV exprimés sous forme d'interconnexions de deux sous-systèmes. L'abréviation LFT signifie transformation linéaire fractionnaire (*Linear Fractional Transformation*) et est la procédure de reformulation utilisée pour exprimer les systèmes LPV / incertains sous forme d'interconnexions à deux blocs. Cette procédure est présentée dans l'analyse et la commande robuste, pour lesquelles elle a été principalement développée. L'intérêt porté aux interconnexions entre les systèmes, remonte au milieu des années 70 lorsque les chercheurs ont obtenu des résultats inexplicables sur la commande optimale des avions et des sous-marins. Ils ont commencé à supposer des incertitudes pour expliquer ces résultats. Ils ont ensuite décidé d'étudier les problèmes de robustesse dans la commande optimale et ont jeté les bases de la théorie de la commande robuste à réussir. L'idée principale derrière la LFT est de réécrire un système complexe en tant qu'interconnexion d'une partie "simple et agréable" et d'une partie "compliquée et gênante". La partie agréable doit posséder des propriétés convenables telles que la linéarité, l'invariance dans le temps, etc. La partie gênante, en revanche, contient généralement des

termes variables dans le temps, des non-linéarités, une dynamique à dimensions infinies, etc. Le système global est ensuite analysé comme un paradigme des systèmes interconnectés pour lequel de nombreux outils spécifiques ont été développés; critère de Popov pour les non-linéarités statiques délimitées par secteur (Khalil, 1996), concepts de gain tels que gain L_2 (Zames, 1966), concepts de valeurs singulières (Zhou et Doyle, 1998), la théorie de la dissipativité et ses applications (Packard, 1994; Apkarian et Gahinet, 1995; Scherer, 2001; Scherer et Köse, 2012), séparation topologique (Iwasaki et Hara, 1998) et contraintes quadratiques intégrales (*Integral Quadratic Constraints: IQC*) (Pfifer et Seiler, 2015).

L'intérêt de cette formulation, dans la forme LPV, a été montré dans (Packard, 1994; Apkarian et Gahinet, 1995) où une formulation convexe de la conception des correcteurs à séquençement de gains «*gain scheduling*» avec des performances H_∞ a été décrite. Depuis lors, de nombreux travaux ont été consacrés à ce problème (Apkarian et Adams, 1998; Scherer, 2001; Scherer et Köse, 2012).

Pour être complet, il semble important de mentionner la description entrée/sortie des systèmes LPV. Ce mode de représentation est beaucoup moins répandu que les trois précédents et peut être utilisé pour l'identification et la conception de commandes (Tóth, 2008; Tóth, 2010; Cerone et al., 2012).

II.2.3 Commande des Systèmes LPV:

Les outils d'analyse des systèmes LPV sont directement hérités d'une analyse robuste et d'une théorie d'une commande robuste, l'intégralité de la théorie LPV n'est révélée que lorsque l'objectif principal est la conception. La différence principale avec la commande robuste réside dans le fait que les paramètres sont supposés connus ou mesurables. Les paramètres peuvent être utilisés dans la loi de commande de manière programmée, ce qui donne lieu à des correcteurs LPV à séquençement de gains. Il est toutefois important de montrer que le séquençement de gains existait avant la théorie LPV, plusieurs exemples ont été proposés dans la référence; (Leith et Leithead, 2000). Les techniques de séquençement de gains LPV traitent le problème de manière directe et globale, alors, le correcteur est construit à partir d'une famille de correcteurs linéaires locaux conçus à l'aide des méthodes linéaires invariantes dans le temps. Les premières idées du séquençement de gains pour le mode LPV ont été proposées par (Shamma, 1988), et les travaux subséquentes ont été présentés dans (Shamma et Athans, 1991; Shamma et Athans, 1992). À l'époque, la difficulté principale était l'absence d'une théorie générale permettant d'analyser la stabilité des systèmes LPV et de concevoir des lois de commande basées sur LPV (Shamma et Athans, 1992).

Comme prévu, un cadre approprié pour l'analyse et la commande des systèmes LPV est né d'idées de commande robuste, telles que la commande H_∞ et l'utilisation des inégalités matricielles (*Linear Matrices Inequalities: LMI*). Les techniques d'optimisation robustes modernes ont considérablement renforcé ce cadre en fournissant un moyen rigoureux de traiter avec LMIs dépendantes des paramètres. Les travaux de (Leith et Leithead, 2000; Rugh et Shamma, 2000) donnent un aperçu de l'état de la technique en 2000, tandis que le livre de (Mohammadpour et Scherer, 2012) fournit des développements plus récents de la théorie des systèmes et de la commande LPV.

La loi de commande la plus simple qui puisse être conçue est le retour d'état. Il s'agit d'une extension immédiate du retour d'état LTI au réglage LPV; la matrice du correcteur est désormais fonction des paramètres. Ce correcteur est le plus facile à concevoir, mais il nécessite la connaissance de l'état complet du système à mettre en œuvre.

En outre, la classe des correcteurs du retour de sortie est très facile à mettre en œuvre car la commande est calculée directement à partir de la sortie mesurée. La difficulté principale réside toutefois dans la difficulté d'obtenir des conditions favorables à la conception de telles lois de commande. Notez que certaines occurrences de ce problème sont connues pour être NP-difficile, (Blondel et Tsitsiklis, 1997; Fu, 2004). Pourtant, il existe certaines méthodes pour les concevoir, du moins dans le cadre robuste (Prempain et Postlethwaite, 2005; Henrion et Lasserre, 2006; Apkarian et Noll, 2006), dont certains peuvent être étendus au cas LPV.

Une classe très importante des correcteurs est la classe des correcteurs à retour de sortie dynamique. Cette classe des correcteurs a été étudiée dans (Packard, 1994; Apkarian et Gahinet, 1995; Scherer, 2001; Scherer et Köse, 2012). Il est intéressant de noter que lorsque l'ordre du correcteur est égal à l'ordre du processus, le problème de conception s'avère admettre des solutions convexes dans plusieurs configurations. Lorsque le correcteur est d'ordre réduit, le problème est connu pour être NP-difficile en raison de la présence d'une contrainte de rang (non-convexe) dans les conditions de stabilisation (Scherer et al., 1997).

Les lois de commande basées sur l'observateur peuvent être comprises comme un cas particulier du retour de sortie dynamique. Cependant, la structure de l'observateur ne permet pas toujours de déduire des conditions de synthèse exactes et convexes. C'est par exemple le cas lorsque les gains de l'observateur et du correcteur doivent être déterminés en même temps, via une seule condition LMI. Cependant, une conception séparée entraîne toujours des conditions de synthèse convexes (Heemels et al., 2010).

II.2.4 Applications:

On donne ici une liste presque compréhensive des applications de la modélisation et de la commande LPV dans la littérature. Dans (Wei et del Re, 2007), le modèle et la commande du circuit de ventilation des moteurs diesel en vue de réduire le carburant polluant est abordé. La commande des éléments dans les moteurs diesel est considéré dans (He et Yang, 2006; Jung et Glover, 2006; Gauthier et al., 2007), où le débit d'air, l'injection de carburant et/ou le groupe moteur sont commandés. Les moteurs à combustion turbocompressés sont considérés dans (Kominék et al., 2012). Les systèmes LPV sont appliqués à la modélisation et au commande des moteurs à double flux, les actionneurs électromagnétiques sont considérés dans (Forrai et al., 2007), tandis qu'une application robotique est présentée dans (Kwiatkowski et Werner, 2005). Dans (Liu et al., 2006), la commande LPV est appliquée au régulateur du système d'alimentation. Les éoliennes sont prises en compte dans (Daher-Adegas et al., 2012). Dans (White et al., 2007), la commande LPV est appliquée à la synthèse des pilotes automatiques des missiles. Dans (Lu et al., 2006), la commande d'attitude d'une avion F-16 en réponse aux ordres du pilote pour différents angles d'attaque est traité; les avions et les vaisseaux spatiaux sont également pris en compte dans (Corti et Lovera, 2012; Hughes et Wu, 2012; Seiler et al., 2012). La modélisation et la commande des suspensions des véhicules LPV sont présentés dans (Do et al., 2012; Poussout-Vassal et al., 2012), tandis que la commande globale du châssis (commande d'attitude) est traité dans (Szabó et al., 2012). D'autres applications dans l'automobile sont disponibles dans (Kwiatkowski et al., 2009; Senname et al., 2013). La détection et l'isolement des défaillances à l'aide des techniques LPV a été effectuée dans (Rodrigues et al., 2013). Les observateurs LPV ont été appliqués pour estimer la température des cellules dans des batteries dans (Debert et al., 2013). La commande de la circulation est considéré dans (Luspay et al., 2012). La commande des systèmes des données échantillonnées asynchrones est traitée de manière LPV dans (Robert et al., 2006). Les méthodes LPV sont appliquées pour commander les systèmes à temporisation temporelle dans (Briat et al., 2009). Enfin, la commande LPV basée sur l'observateur des équations aux dérivées partielles non-linéaires peut être trouvée dans (Hashemi et Werner, 2011).

II.3 La Commande H_∞ Non-Linéaire et les Systèmes Sous-Actionnés:

Dans cette thèse, le problème de suivi de trajectoire d'un quadrotor sera abordé, l'objectif principal étant d'améliorer la robustesse des stratégies de commande lorsque le véhicule vole en présence des perturbations externes, de dynamiques non-modélisées et d'incertitudes paramétriques.

II.3.1 La Commande H_∞ Non-Linéaire:

La plupart des stratégies de commande classiques ne tiennent pas compte des incertitudes sur les paramètres ni des perturbations externes. Cependant, les drones sont constamment affectés par les incertitudes des modèles et les rafales de vent, qui peuvent facilement déstabiliser le véhicule.

Un choix approprié pour rejeter ces perturbations est la théorie de la commande H_∞ non-linéaire. Les premiers efforts pour étendre le problème de commande H_∞ aux systèmes non-linéaires ont été faits dans les années 1980. Dans (Ball et al., 1987), le problème des systèmes non-linéaires à temps discret a été formulé et des solutions acceptables ont été trouvées à l'aide de la série de Volterra. La solution pour les systèmes à temps continu non-linéaires a été fournie par van der Schaft dans (van der Schaft, 1992).

L'objectif de la théorie H_∞ est de réaliser un rapport borné entre l'énergie des signaux d'erreur et celle des signaux de perturbation. En général, l'approche non-linéaire de cette théorie considère une équation différentielle partielle (*Partial Differential Equations: PDE*) de Hamilton-Jacobi (HJ), qui remplace l'équation de Riccati dans le cas de la formulation de commande H_∞ linéaire. La solution de la commande H_∞ non-linéaire peut être obtenue par deux approches: la théorie différentielle (Doyle et al., 1989; Başar et Bernhard, 2008) et la théorie des systèmes dissipatifs (van der Schaft, 2000). Le problème principal dans le cas non-linéaire est l'absence de méthode générale pour résoudre cette PDE HJ. Par conséquent, des solutions analytiques doivent être résolues pour chaque cas particulier.

En raison de la difficulté à trouver des solutions analytiques, certains travaux utilisent des méthodes numériques permettant d'intégrer de telles équations, par exemple les approximations de Galerkin et les séries de Taylor (Beard et al., 1997; Beard et McLain, 1998; Beard et al., 1998; Hardt et al., 2000).

Étant donné que l'intérêt principal est de travailler avec des modèles de systèmes mécaniques obtenus via la formulation d'Euler-Lagrange, il est possible de trouver des solutions de la commande H_∞ non-linéaire minimisant les forces qui agissent sur le système, comme exposé dans (Johansson, 1990). Dans cet article, l'auteur a proposé une solution au problème de la commande optimale de H_2 non-linéaire dans les systèmes mécaniques complètement actionnés. Depuis ce travail, une quantité énorme de modifications a été apportée à la formulation des correcteurs non-linéaires H_2 , H_∞ et H_2/H_∞ pour les systèmes mécaniques (Chen et al., 1994; Feng et Postlethwaite, 1994; Chen et al., 1997; Postlethwaite et Bartoszewicz, 1998). Une étude sur la commande robuste

des manipulateurs des robots a été présentée (Sage et al., 1999). On y trouve un aperçu de la commande non-linéaire H_∞ appliquée aux systèmes mécaniques.

Une solution paramétrée globale explicite au problème de la commande optimale H_∞ à retour d'état, formulée sous la forme d'un jeu *min-max*, a été développée dans (Chen et al., 1994). Cette solution traite le cas particulier des systèmes mécaniques actionnés formulés par les équations d'Euler-Lagrange en utilisant l'équation d'erreur d'état proposée dans (Johansson, 1990) et les propriétés dynamiques des systèmes mécaniques. La même année (Feng et Postlethwaite, 1994) ont proposé une approche similaire pour le correcteur H_∞ à retour d'état non-linéaire pour les systèmes robotiques, la variable coût prenant en compte le couplage entre les variables du correcteur et la loi de commande, qui donne plus de degrés de liberté à la conception de la commande. En outre, une loi de commande H_∞ non-linéaire avec un schéma adaptatif a été présenté pour améliorer la robustesse de l'ensemble du système.

Une stratégie de commande des systèmes mécaniques actionnés tenant compte de l'équation dynamique d'erreur de suivi a été proposée, l'intégrale de l'erreur de position étant ajoutée au vecteur d'erreur (Ortega et al., 2005). Dans cette stratégie, une commande H_∞ non-linéaire a été appliquée. Cette stratégie fournit, par le biais d'une solution analytique, une loi de commande variant dans le temps, fortement dépendante du modèle et similaire aux résultats obtenus avec les procédures de linéarisation à retour d'état. Les conditions pour formuler le correcteur sous la forme d'un PID non-linéaire ont été établies.

Certains travaux utilisant des correcteurs non-linéaires H_2 , H_∞ et H_2/H_∞ ont été établis dans le domaine de l'aéronautique. La théorie de la commande non-linéaire H_∞ a été utilisée pour concevoir une loi d'orientation des missiles tridimensionnelle (Yang et Chen, 2001). On trouve aussi qu'un missile tactique poursuivant des objectifs de manœuvre dans un espace tridimensionnel a été résolu en utilisant une loi de guidage H_∞ non-linéaire basée sur un modèle flou (Chen et al., 2002). Une commande latérale mixte flou H_2/H_∞ adaptative des systèmes des missiles non-linéaires à perturbations a été proposée (Chen et al., 2003). Un hélicoptère de laboratoire à double rotors a été commandé à l'aide d'un correcteur non-linéaire L_2 basée sur un modèle d'ordre réduit des hélices (López-Martínez et al., 2007).

II.3.2 Les Systèmes Mécaniques Sous-Actionnés:

Les UAV sont généralement des systèmes mécaniques sous-actionnés, et le quadrotor n'est pas différent, car il dispose de six degrés de liberté et de quatre actions de commande. Les systèmes mécaniques sous-actionnés apparaissent dans plusieurs applications telles que les robots

aérospatiaux et sous-marins, les systèmes mobiles, les systèmes flexibles, les robots de marche, de gymnastique. Selon (Olfati-Saber, 2001), la propriété de sous-actionné des systèmes sous-actionnés est due à quatre raisons: la dynamique du système, la conception visant à réduire le coût ou certains objectifs pratiques, la défaillance de l'actionneur, et l'imposition artificielle pour créer des systèmes non-linéaires de complexité réduite dans le but de mieux comprendre la commande des systèmes sous-actionnés d'ordre élevé.

Dans le domaine de la commande des systèmes mécaniques sous-actionnés, une contribution importante a été présentée dans (Spong, 1994), où les auteurs ont utilisé la linéarisation partielle à retour d'état pour linéariser les degrés de liberté non-actionnés.

La commande du mouvement des systèmes mécaniques sous-actionnés est souvent difficile en raison des contraintes non-holonomiques sur l'accélération générée par la sous-actionnement, ce qui est impossible de régler tous les degrés de liberté du système au même instant dans la position souhaitée. Dans (Wichlund et al., 1995), des propriétés de commande de la dynamique de véhicules sous-actionnés (véhicules sous-marins, hélicoptères, avions, etc.) ont été étudiées. Le système mécanique sous-actionné ne peut être commandé ou stabilisé à aucun point d'équilibre pour les conditions initiales. Dans (Reyhanoglu et al., 1999), les propriétés de commandabilité et d'observabilité du système mécanique sous-actionné avec des contraintes non-holonomiques du second ordre ont été dérivées. On trouvera une étude intéressante sur les systèmes mécaniques sous-actionnés dans (Spong, 1998).

Le suivi de la position des systèmes mécaniques sous-actionnés a été effectué dans plusieurs publications utilisant des correcteurs non-linéaires L_2 . Dans (Toussaint et al., 2000), un modèle non-linéaire de navire sous-actionné a été commandé au moyen d'une loi de commande H_∞ de suivi par retour d'état en présence de perturbations et de mesures d'états bruités. Dans (Siqueira et Terra, 2004a), les auteurs ont établi une commande non-linéaire H_∞ pour les manipulateurs sous-actionnés, comme une extension de celui proposé par (Chen et al., 1994). Dans (Siqueira et Terra, 2004b), un correcteur H_∞ a également été mis au point lorsque le manipulateur sous-actionné est sujet de changements de configuration. Dans (Siqueira et al., 2006), des correcteurs non-linéaires obtenus par H_2 , H_∞ , H_2/H_∞ , des problèmes d'optimisation ont été appliqués aux manipulateurs sous-actionnés par redondance d'activation. Dans (He et Han, 2008), une commande de retour d'accélération a été appliquée aux véhicules autonomes non-linéaires totalement actionnés ou sous-actionnés en utilisant la théorie H_∞ . En outre, les résultats de simulations pour un suivi de trajectoire d'hélicoptère ont été présentés.

II.4 Conclusion:

On a présenté dans ce chapitre; la représentation et la commande des systèmes LPV, et la commande robuste (H_∞ non-linéaire) des systèmes mécaniques sous-actionnés, afin d'introduire aux chapitres suivants (5 et 6) où le développement des correcteurs robustes a consacré à la commande d'un système aérien autonome, le quadrotor. Dans les deux chapitres suivants (3 et 4), des approches classiques seront développées et appliquées sur le quadrotor.

Chapitre III:

Amélioration Des Performances de la Commande Linéaire par l'utilisation des Algorithmes Évolutionnaires

III.1 Introduction:

Dans ce chapitre, la commande d'un quadrotor par la méthode classique linéaire PID est établie. En premier lieu, nous allons extraire le modèle dynamique de l'organe à partir des lois physiques et aérodynamiques par le formalisme d'Euler-Lagrange. Par la suite, nous allons réaliser une commande du système en boucle fermée par le correcteur PID. Du fait, que le système est non-linéaire et sous-actionné, l'ajustement des paramètres du correcteur est effectué à partir des algorithmes évolutionnaires d'optimisation; les algorithmes génétiques (GA), les stratégies évolutionnaires (ES), l'algorithme à évolution différentielle (DE), et aussi avec l'algorithme du cuckoo search (CS) (Hasseni et al., 2019). À la fin de ce chapitre, la simulation des dynamiques du quadrotor sera présentée en utilisant l'outil Matlab/Simulink, avec les résultats atteints pour diverses situations.

III.2 Modélisation du Quadrotor:

Le mouvement du quadrotor est lié à ses coordonnées de translation (x , y et z), ce qui constitue l'altitude de l'engin et à ses coordonnées de rotation (φ : roulis, θ : tangage and ψ : lacet), ce qui constitue son attitude, comme c'est représenté sur la Figure III.1, qui décrit une vue générale du quadrotor. Ce dernier possède quatre rotors, dont chaque deux rotors tournent dans le même sens. Le roulis présente la rotation selon l'axe x , le tangage présente la rotation selon l'axe y et le lacet présente la rotation selon l'axe z . Le modèle mathématique du quadrotor par l'approche Euler-Lagrange sera présenté dans ce qui suit.

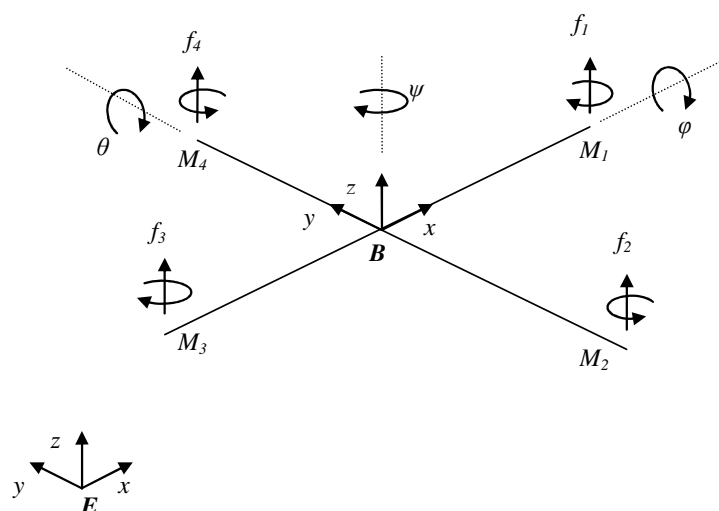


Figure III.1 Schéma du quadrotor

La loi de Lagrange est donnée comme suit (Spong et al., 2006):

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \Gamma_i \quad (\text{III.1})$$

Où:

L : Terme de Lagrange.

q_i : Variable mesurée, présentant chaque variable de sortie: x, y, z, φ, θ et ψ .

Γ_i : Vecteur des forces et couples non-conservés: $F_x, F_y, F_z, \tau_\varphi, \tau_\theta$ et τ_ψ .

$$L = E_{ct} + E_{cr} - E_p \quad (\text{III.2})$$

E_{ct} : Energie cinétique de translation.

E_{cr} : Energie cinétique de rotation.

E_p : Energie potentielle.

On obtient alors:

$$L = \frac{m}{2} \cdot \mathbf{V}^2 + \frac{1}{2} \mathbf{J} \boldsymbol{\Omega}^2 - mgz \quad (\text{III.3})$$

Tel que:

\mathbf{V} : Vecteur des vitesses de translation.

$\boldsymbol{\Omega}$: Vecteur des vitesses de rotation.

\mathbf{J} : Matrice symétrique de l'inertie.

m : Masse du quadrotor.

g : Constante de gravité.

$$\boldsymbol{\Omega} = \begin{pmatrix} \dot{\varphi} - \dot{\psi} \sin \theta \\ \dot{\theta} \cos \varphi + \dot{\psi} \sin \varphi \cos \theta \\ \dot{\psi} \cos \varphi \cos \theta - \dot{\theta} \sin \varphi \end{pmatrix} \quad (\text{III.4})$$

$$\mathbf{J} = \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix} \quad (\text{III.5})$$

I_x, I_y, I_z : L'inertie sur l'axe x , l'axe y et l'axe z respectivement.

Finalement, les forces et couples non-conservés sont présentés par:

$$\Gamma = \begin{pmatrix} F_x \\ F_y \\ F_z \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} = \begin{pmatrix} (c\varphi s\theta c\psi + s\varphi s\psi)b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ (c\varphi s\theta s\psi - s\varphi c\psi)b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ (c\varphi c\theta)b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ lb(\omega_4^2 - \omega_2^2) \\ lb(\omega_3^2 - \omega_1^2) \\ d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{pmatrix} \quad (\text{III.6})$$

Où:

s et c dénotent le *sin* et le *cos* respectivement.

b : Coefficient de portance.

d : Coefficient de trainée.

l : Longueur du bras.

ω_i : Vitesse angulaire de $i^{\text{ème}}$ rotor.

J_r : Inertie du rotor.

Pour plus des détails, les références (Bouabdallah, 2007; Carrillo et al., 2013) peuvent être consultées. En combinant (III.1) et (III.6) nous avons:

$$\begin{cases} \ddot{x} = \frac{1}{m}(c\varphi s\theta c\psi + s\varphi s\psi)U_1 \\ \ddot{y} = \frac{1}{m}(c\varphi s\theta s\psi - s\varphi c\psi)U_1 \\ \ddot{z} = \frac{1}{m}(c\varphi c\theta)U_1 - g \\ \ddot{\varphi} = \frac{(I_y - I_z)}{I_x}\dot{\theta}\dot{\psi} - \frac{J_r\Omega_r}{I_x}\dot{\theta} + \frac{l}{I_x}U_2 \\ \ddot{\theta} = \frac{(I_z - I_x)}{I_y}\dot{\varphi}\dot{\psi} + \frac{J_r\Omega_r}{I_y}\dot{\varphi} + \frac{l}{I_y}U_3 \\ \ddot{\psi} = \frac{(I_x - I_y)}{I_z}\dot{\varphi}\dot{\theta} + \frac{1}{I_z}U_4 \end{cases} \quad (\text{III.7})$$

Le quadrotor est commandé par les vitesses angulaires des rotors (ω_1 ω_2 ω_3 et ω_4), alors le vecteur des commandes est exprimé comme une fonction des vitesses angulaires comme suit:

$$\begin{cases} U_1 = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ U_2 = b(\omega_4^2 - \omega_2^2) \\ U_3 = b(\omega_3^2 - \omega_1^2) \\ U_4 = d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{cases} \quad (\text{III.8})$$

$$\Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4 \quad (\text{III.9})$$

Les entrées du system sont, la commande d'altitude (U_1), la commande du roulis (U_2), la commande du tangage (U_3) et la commande du lacet (U_4).

Le vecteur des états X est donné par:

$$X = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \varphi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T \quad (\text{III.10})$$

Le vecteur des entrées U est donné par:

$$U = [U_1, U_2, U_3, U_4]^T \quad (\text{III.11})$$

Les points d'équilibre de (III.7) satisfont:

$$f(X_{eq}, U_{eq}) = 0 \quad (\text{III.12})$$

Où:

$$X_{eq} = [x_{eq}, \dot{x}_{eq}, y_{eq}, \dot{y}_{eq}, z_{eq}, \dot{z}_{eq}, \varphi_{eq}, \dot{\varphi}_{eq}, \theta_{eq}, \dot{\theta}_{eq}, \psi_{eq}, \dot{\psi}_{eq}]^T \quad (\text{III.13})$$

$$U_{eq} = [U_{1eq}, U_{2eq}, U_{3eq}, U_{4eq}]^T \quad (\text{III.14})$$

La résolution de (III.12), résulte en les points stationnaires:

$$X_{eq} = [x_{ss}, 0, y_{ss}, 0, z_{ss}, 0, 0, 0, 0, 0, \psi_{ss}, 0]^T \quad (\text{III.15})$$

$$U_{eq} = [g, m, 0, 0, 0]^T \quad (\text{III.16})$$

Où: $x_{ss}, y_{ss}, z_{ss}, \psi_{ss} \in \mathbf{R}$, nous notons *ss* signifie *steady state* (régime permanent).

En outre, des limites sont appliquées sur les angles de roulis (φ) et de tangage (θ) pour éviter la non-commandabilité:

$$-\frac{\pi}{2} \leq \varphi, \theta \leq \frac{\pi}{2} \quad (\text{III.17})$$

Dans ce travail, les paramètres de simulation du quadrotor sont pris depuis le projet (OS4) (Bouabdallah, 2007), ils sont présentés sur le Tableau III.1:

Tableau III.1 Les paramètres du OS4

Paramètre	Description	Valeur	Unité
m	Masse	0.65	Kg
l	Longueur de bras	0.23	m
b	Coefficient de portance	$3.13 \cdot 10^{-5}$	N.s^2
d	Coefficient de trainée	$7.5 \cdot 10^{-7}$	N.m.s^2

I_x	Inertie de rotation selon l'axe x	$7.5 \cdot 10^{-3}$	Kg.m^2
I_y	Inertie de rotation selon l'axe y	$7.5 \cdot 10^{-3}$	Kg.m^2
I_z	Inertie de rotation selon l'axe z	$1.3 \cdot 10^{-2}$	Kg.m^2
J_r	Inertie de rotor	$6 \cdot 10^{-5}$	Kg.m^2
g	Constante de gravité	9.8	m.s^{-2}

III.3 Les Algorithmes Évolutionnaires:

Nous allons présenter un état de l'art d'application des algorithmes évolutionnaires, ensuite des descriptions et les caractéristiques des algorithmes inspirés de la nature; les algorithmes génétiques, les stratégies évolutionnaires, l'algorithme d'évolution différentielle et l'algorithme cuckoo search, qui seront utilisés dans cette partie pour l'optimisation les paramètres des correcteurs linéaires PID.

III.3.1 Applications des Algorithmes Évolutionnaires sur la Commande:

L'application des algorithmes évolutionnaires (*Evolutionary Algorithms: EA*) aux techniques de commande peut être globalement classée en deux domaines: la conception et l'analyse hors ligne et l'adaptation et le réglage en ligne. Dans les applications hors ligne, les EAs peuvent être utilisés comme moteur de recherche et d'optimisation, par exemple pour sélectionner les lois de commande appropriées pour une installation connue afin de satisfaire des critères de performance donnés ou pour rechercher les réglages de paramètres optimaux pour une structure de correcteur particulière. Dans l'adaptation en ligne, les EAs peuvent être utilisés comme une mécanisme d'apprentissage pour identifier les caractéristiques de systèmes inconnus ou non-stationnaires ou pour le réglage adaptatif de correcteurs pour des systèmes connues ou inconnues.

Dans cette section, certaines applications récentes des EAs sur la commande sont considérées lorsque les méthodes conventionnelles sont inappropriées, problématiques ou indisponibles.

A/ Conception du système de commande:

Une des applications les plus courantes des EAs est l'optimisation paramétrique. Dans la conception des systèmes de commande, de nombreuses tâches peuvent être exécutées dans un cadre d'optimisation. En particulier, dans la conception des systèmes de commande par l'optimisation des paramètres, la solution finale peut être sensible à l'estimation initiale de la solution ou il peut exister un certain nombre de combinaisons de réglages de paramètres qui produisent l'action de commande souhaitée. Les optimiseurs classiques peuvent donc ne trouver que des solutions sous-optimales, car la recherche est forcée de ne prendre en compte que de petites régions de l'espace de recherche

influencées par l'estimation initiale. Les EAs sont capables d'échantillonner l'ensemble de l'espace de la solution et doivent produire des solutions de nature plus globale. De plus, les approches évolutionnaires ont le potentiel de trouver simultanément des solutions dans de nombreux domaines différents de l'espace de recherche. Ainsi, davantage d'information sur la nature du problème de conception peuvent être obtenues au cours de la recherche, ce qui donne potentiellement un processus de conception plus éclairé.

Les EAs se sont révélés être une stratégie efficace dans la conception hors-ligne des correcteurs par l'optimisation paramétrique. (Jogindar-Singh et al., 2017) ont présenté comment les EAs peuvent être utilisés dans la sélection et le réglage des paramètres du correcteur PID d'un servomoteur. (Zhang et al., 2017) ont montré comment les EA peuvent être utilisés pour résoudre des problèmes de planification de trajectoire des robots dans des environnements stationnaires et non-stationnaires.

B/ Optimisation multi-objectif:

Les problèmes d'ingénierie de commande nécessitent très rarement l'optimisation d'une fonction-objectif unique. Cela, il existe généralement un certain nombre des objectifs de conception concurrents qui doivent être atteints simultanément. Conventionnellement, les membres de l'ensemble de solutions Pareto-optimal sont recherchés par la résolution d'un problème de programmation non-linéaire formulé de manière appropriée. Un certain nombre des approches sont utilisées, notamment les méthodes de ϵ -contrainte, de somme pondérée et de réalisation des objectifs (Chircop et Zammit-Mangion, 2013). Cependant, de telles approches nécessitent l'expression précise d'un ensemble de poids et des objectifs généralement mal compris. Si la surface de compromis entre les objectifs de conception doit être mieux comprise, une application répétée de telles méthodes sera nécessaire. De plus, les méthodes de programmation non-linéaires ne peuvent pas gérer correctement la multi-modalité et les discontinuités dans l'espace fonctionnel et ne peuvent donc que produire des solutions locales.

Les EAs peuvent potentiellement devenir une méthode puissante pour l'optimisation multi-objectif. En incluant l'ingénieur de commande automatique dans le processus de conception en tant que décideur, l'évaluation environnementale peut être guidée. Les compromis entre les critères de conception et leurs interactions peuvent être examinés de près et les connaissances et l'expérience des ingénieurs peuvent être utilisées pour prendre une décision en connaissance de cause sur la base des exigences de conception plutôt que des propriétés des fonctions-objectif (Nazarahari et al., 2019).

C/ Commande robuste:

Une approche de la conception des correcteurs robustes consiste à attribuer une structure propre. (Tamilchelvan et Arounassalame, 2016) ont mis en évidence une approche par les GA sur le système de conversion d'énergie éolienne. Leur approche a été appliquée à la conception d'un correcteur PI d'ordre fractionnel (FO-PI) et utilise une représentation réelle des valeurs pour minimiser une fonction-objectif basée sur l'intégral de temps multiplié par l'erreur absolue (ITAE). La robustesse de FO-PI est comparée avec PI classique, les résultats de la simulation montrent la robustesse du correcteur FO-PI dans la commande de la tension de sortie, même en présence de variation du vent due à la fluctuation de la vitesse du vent.

Les EAs utilisent une représentation structurée (Do et al., 2011; Vu et al., 2017) qui permet de rechercher des fonctions de pondération et d'identifier simultanément les paramètres appropriés. Le problème de conception considéré était un système de suspension pour un véhicule à lévitation magnétique. Il consistait à trouver les fonctions de pondération appropriées pour définir les fonctions de transfert en boucle ouverte tout en satisfaisant l'optimisation de H_{∞} de performance explicite en boucle fermée.

D/ Systèmes d'identification:

De nombreux problèmes de commande, de traitement du signal et d'apprentissage automatique peuvent être considérés comme un problème d'identification, dans le but de déterminer un modèle approprié à partir d'un ensemble de données d'entrée-sortie. Le modèle résultant peut ensuite être utilisé pour la prédiction et la commande d'un système « boîte noire ». Bien qu'il existe de nombreuses méthodes éprouvées pour l'identification de systèmes linéaires, dans la pratique, la plupart des systèmes du monde réel sont non-linéaires. La complexité supplémentaire associée à l'identification de système non-linéaire, en particulier en l'absence des informations initiales ou de détails de la structure du modèle, dans une certaine mesure il a contribué au manque relatif d'attention accordée à ce domaine particulier. Une approche réussie de ce problème est la méthode de régression orthogonale par la méthode des moindres carrés (Molugaram et Shanker-Rao, 2017) pour trouver un ensemble approprié de termes non-linéaires pour le système. Cependant, la complexité et la croissance combinatoire de l'espace de recherche font qu'une recherche exhaustive n'est pas toujours réalisable et son application est limitée.

(Dub et Stefek, 2014) ont mis en œuvre une identification de système non-linéaire en utilisant PSO. PSO est utilisé pour estimer les paramètres du modèle (fonction de transfert) d'un moteur à courant continu. Les données du système sont présentées par la tension d'armature comme entrée, et la vitesse de rotation comme sortie. Cette approche a été utilisée avec les données originales bruitées,

ainsi que filtrées. Les tests ont présenté que les résultats ont été améliorées si une distribution logarithmique des positions initiales était appliquée.

Ainsi, l'algorithme génétique a le potentiel d'être plus efficace pour identifier une structure de modèle appropriée et donc de nature plus générale. La stratégie d'identification génétique peut être rendue plus efficace en permettant à la recherche d'être effectuée sur un nombre variable de termes. Plutôt que d'utiliser une fonction-objectif basée sur la longueur minimale du descripteur, le problème peut être exprimé sous la forme « multi-objectif », prenant ainsi en considération le nombre de termes et la variance simultanément. Cette approche est préférable car l'impact du nombre de termes dans le modèle ne masquera pas directement la performance résiduelle d'un modèle. Pour accueillir une population composée des individus constitués d'un nombre variable de termes, une représentation structurée peut être appropriée (Dasgupta et McGregor, 1992). (Alsakini et al., 2016) montrent comment les algorithmes génétiques peuvent être appliqués à l'identification de systèmes non-linéaires des turbo générateurs.

E/ Systèmes d'intégration:

Le problème de la conception des correcteurs offrant des performances optimales, ceux-ci doivent toutefois être intégrés à l'ensemble du système. Par exemple, dans les implémentations de correcteur de moteur à turbine à gaz, les problèmes d'algorithme de commande, il existe des autres problèmes importants de réalisation et d'implémentation de matériel. Celles-ci incluent des considérations de conception telles que l'emplacement, le poids et les types d'interface pour les transducteurs et les actionneurs, ainsi que la conception des règles de redondance, de tolérance aux pannes et de diagnostic des pannes. Tous ces problèmes ont une incidence sur l'optimisation de l'intégration et il est souhaitable d'examiner comment elles interagissent les unes avec les autres en tant que processus de conception unique. Un moteur moderne typique comporte plus de 160 interfaces combinées avec un bus de données et de faisceaux de câbles dédiés pour transmettre des signaux de données discrets et continus. Le placement des composants est dicté par l'environnement, les exigences de refroidissement de l'électronique, la fiabilité, la sécurité et un certain nombre des autres critères. L'intégration des systèmes a donc pour objectif d'envisager l'intégration du système de contrôle avec l'usine et les exigences d'interface avec des autres systèmes en interaction, par exemple, la cellule, afin d'améliorer la sécurité, l'efficacité, le poids, la fiabilité et les coûts d'exploitation.

Le problème de l'intégration des systèmes est commun à de nombreuses applications industrielles et peut être posé dans un cadre multi-objectif. Cependant, il a généralement été constaté que ses propriétés le rendaient impropre aux techniques d'optimisation conventionnelles. Par exemple, la

variable de décision est un hybride de paramètres discrets et continus et les objectifs de conception couvrent un intervalle large de mesures de la performance du correcteur aux considérations mécaniques. De plus, on peut s'attendre à ce que les fonctions-objectif soient en général hautement non-linéaires et présentent un degré élevé d'interaction entre elles. L'utilisation de méthodes évolutionnaires a le potentiel de rechercher de telles surfaces complexes de fonction-objectif, en incorporant les connaissances des concepteurs à la fois dans la formulation et la résolution du problème (voir, par exemple, (Suresh et Meenakumari, 2016)).

F/ Commande adaptative et temps-réel:

Les tentatives d'utilisation des GA dans des applications de commande en temps réel rencontrent deux problèmes majeurs: le temps d'exécution générationnel et la production de lois de commande satisfaisantes à chaque génération. Le problème du temps d'exécution générationnel peut être résolu par des algorithmes génétiques parallèles. Les GAs incrémentiels ne produisent qu'une ou deux individus enfants à chaque génération et présentent donc l'avantage de réduire le temps de cycle de génération et les besoins en mémoire, mais ils peuvent ne pas produire des nouveaux individus ou satisfaisants.

La tâche de produire une loi de commande satisfaisante à chaque génération est le problème le plus difficile à résoudre. Par exemple, un correcteur relativement peu sensible peut avoir un grand nombre de réglages de paramètres appropriés qui aboutissent à une loi de commande satisfaisante. Ainsi, il est possible que le GA produise des lois de commande successives qui s'interprètent par des modifications trop importantes des paramètres du correcteur, entraînant probablement des problèmes de stabilité. Cependant, un certain nombre de programmes réussis basés sur des évaluations environnementales ont été développés. (Jaen-Cuellar et al., 2013) ont mis au point une approche génétique de réglage des correcteurs PID numériques qui serait beaucoup plus simple à mettre en œuvre que les techniques d'optimisation sous contrainte, même dans le cas des systèmes simples. Des autres approches adaptatives ont utilisé PSO pour mettre en œuvre des systèmes de commande PID numérique d'un moteur (voir, par exemple, (Yazgan et al., 2019)).

(Chen et al., 2016) montrent comment les EAs peuvent être utilisés pour concevoir des correcteurs de logique floue adaptatifs et non-adaptatifs pour un système dynamique. Dans cette approche, le GA est utilisée pour optimiser les fonctions d'appartenance du correcteur. Il est avancé que les règles de commande ont tendance à rester constantes, même dans un large éventail de conditions, et que les fonctions d'appartenance devraient être adaptées. Pour l'optimisation des fonctions d'appartenance (Valdez et al., 2018) utilisent les algorithmes ACO et PSO, et un système de classificateur flou est utilisé pour acquérir et modifier de bons ensembles de règles en ligne. Dans

leur schéma, la commande adaptative est fournie par l'adaptation constante du jeu des règles pour répondre à la dynamique changeante du problème.

III.3.2 Les Caractéristiques des Algorithmes Sélectionnés:

A/ Les Algorithmes Génétiques:

L'algorithme génétique (Genetic Algorithm: GA), considéré comme étant l'un des anciens algorithmes évolutionnaires est en effet c'est l'algorithme le plus populaire dans les applications d'ingénierie. Il est basé sur le mécanisme de la sélection naturelle des systèmes biologiques, où leurs opérateurs sont des opérateurs biologiques; croisement, mutation, et sélection où ils sont appliqués sur chaque population. La population est divisée à des individus, les bons individus ont une chance à survivre et transférer leurs caractéristiques à la génération suivante. Chaque individu est appelé *chromosome* et représente une solution dont nous considérons les individus comme des solutions réelles dans ce travail.

L'opération d'évaluation (*Evaluer*) est effectuée selon la fonction-objectif, où nous avons besoin de classer tout les individus réels de la population afin d'effectuer le croisement Best-Mate-Worst (Ranjitham et Shankar-Kumar, 2016). Le croisement est effectué après la sélection de type déterministe. Ce croisement (*Recombiner_BMW*) est présenté par l'expression suivante:

$$\begin{cases} x_i^{t+1} = \alpha x_i^t + (1 - \alpha) x_j^t \\ x_j^{t+1} = \alpha x_j^t + (1 - \alpha) x_i^t \end{cases} \quad (\text{III.18})$$

La mutation est appliquée de façon aléatoire, et implémentée avec un taux (p_m) sur les populations d'enfants après la fin de l'opération de croisement. Plusieurs types de mutations ont été proposés en littérature parmi lesquelles, on utilise la mutation Gaussienne (Deb et Deb, 2014). Cette dernière est effectuée selon la formule suivante:

$$x_i^{t+1} = x_i^{t+1} + N(0, \sigma) \quad (\text{III.19})$$

Finalement, l'opération *Remplacer*, permet de remplacer les individus mauvais de la génération précédente par les individus mieux de la génération actuelle afin de garder la taille de la population comme l'initiale.

La procédure est illustrée dans l'Algorithme 3.1.

Algorithme 3.1: Algorithme Génétique

Initialiser des solutions aléatoirement;
TantQue max_Generation n'est pas atteint
 Evaluer chaque solution;
 Classer les solutions;
 Recombiner_BMW par paires de parents;
 Mutation avec un taux p_m ;
 Remplacement déterministe;
Fin TantQue

B/ Les Stratégies Évolutives:

Les stratégies évolutives (Evolutionary Strategies; ES) sont des algorithmes qui ont été proposés avant les algorithmes génétiques, initialement pour des problèmes d'optimisation des systèmes continus. De même que les GA, leurs opérateurs sont: le croisement (ou combinaison), la mutation et le remplacement (Hansen et al., 2015). L'avantage principal d'une ES est l'auto-adaptation de la commande des paramètres dans la tâche de la mutation. Plusieurs types de stratégies existent en littérature, parmi lesquelles on trouve la stratégie $(\mu+\lambda)$ avec codage réel des individus (Hansen et al., 2015). Ce type de ES est utilisée dans ce travail. Nous allons prendre λ individus de la population des enfants recueillis avec μ meilleurs individus de la population parente, et nous allons les considérer comme étant la nouvelle génération. Contrairement au GA, dans le croisement, nous n'avons pas besoin de classer la population, parce que la combinaison est implémentée aléatoirement entre les λ individus.

L'opération du croisement, *Recombiner*, est implémentée de la même façon que celle dans le cas du GA (III.18) en sélectionnant λ individus réels aléatoirement de la génération parente (Wright, 1991). L'une des caractéristiques principales de l'ES est l'auto-adaptation de la mutation, que nous allons utiliser plus tard dans ce chapitre. La mutation auto-adaptative log-normal est donnée par:

$$\sigma^{t+1} = \sigma^t e^{(\sqrt{n})^{-1} \cdot N(0,1)} \quad (\text{III.20})$$

$$x_i^{t+1} = x_i^t + N(0, \sigma^{t+1}) \quad (\text{III.21})$$

Où

σ^t et σ^{t+1} : sont l'écart-type de la génération actuelle et suivante respectivement.

$e^{(\sqrt{n})^{-1} \cdot N(0,1)}$: est le taux de mutation avec $n=2$.

$N(0, 1)$: la loi de probabilité normale de moyenne 0 et l'écart-type 1.

$N(0, \sigma^{t+1})$: la loi de probabilité normale de moyenne 0 et l'écart type σ^{t+1} .

Dans la dernière opération, *Remplacer*, nous prenons les λ individus de la génération des enfants et les meilleurs μ individus de la génération des parents en les transférant à la génération suivante.

La procédure est illustrée dans *Algorithme 3.2*.

Algorithme 3.2: Stratégies Evolutionnaires

Initialiser des solutions aléatoirement;
TantQue *max_Generation n'est pas atteint*
 Générer de nouvelles solutions;
 Recombinaison des paires de parents;
 Réaliser une mutation;
 Classer les solutions de la génération parente;
 Remplacer la nouvelle génération ($\mu+\lambda$);
Fin TantQue

C/ Algorithme d'Évolution Différentielle:

L'algorithme d'évolution différentielle (Differential Evolution; DE) a été créé par Storn et Price en 1997 (Storn et Price, 1997). Il est considéré comme l'algorithme évolutionnaire le plus récent, qui est en plus similaire au GA basé sur les deux processus de croisement et de mutation (Price et Storn, 2005). Bien qu'il utilise les termes évolutionnaires; i.e. croisement, mutation, et sélection, il ne suit aucune évolution naturelle. Cependant, les opérateurs sont implémentés par opérations mathématiques; un croisement différentiel et mutation différentiel. Ces opérations peuvent être exponentiels ou distribution binomial (Yang, 2014).

Comme nous avons mentionné précédemment, le croisement et la mutation sont différentiels. Premièrement, la mutation est implémenté par *DE/Best/1/Bin* qui dénote que nous générons un vecteur donneur v_i pour chaque $i^{\text{ème}}$ individu. Le vecteur de base est le meilleur vecteur dans la population actuelle, *Best*. La mutation suit une distribution binomiale, *Bin*, entre deux autres vecteurs aléatoires (Yang, 2014):

$$v_i^{t+1} = x_b^t + F (x_p^t - x_q^t) \quad (\text{III.22})$$

Où x_b est la meilleure solution (*Best*), x_p et x_q sont des solutions aléatoires différentes de x_i , F est le poids ou le coefficient différentiel. Chaque solution est un vecteur de taille d , où le croisement binomial différentiel, *Crois_Diff*, est implémenté pour chaque $j^{\text{ème}}$ élément d'une solution i . en générant un vecteur uniformément distribué r_i et J_r et selon la probabilité de croisement, C_r , et le nouveau vecteur est manipulé comme suit:

$$w_{i,j}^{t+1} = \begin{cases} v_{i,j}^{t+1} & \text{si } r_i \leq C_r \text{ ou } j = J_r \\ x_{i,j}^t & \text{ailleurs} \end{cases} \quad (\text{III.23})$$

Finalement, la tache de la sélection est atteinte par:

$$x_i^{t+1} = \begin{cases} w_i^{t+1} & \text{si } f(w_i^{t+1}) < f(x_i^t) \\ x_i^t & \text{ailleurs} \end{cases} \quad (\text{III.24})$$

Où $f(.)$ est la fonction-objectif.

La procédure est illustrée dans *Algorithme 3.3*.

Algorithme 3.3: *Évolution Differentielle*

Initialiser des solutions aléatoires;

TantQue *max_Generation n'est pas atteint*

Evaluer chaque solution;

Générer un vecteur donneur par "DE/Best/1/Bin"

Crois_Diff concernant C_r

Sélectionner le meilleur entre l'ancienne solution et la solution d'essai;

Fin TantQue

D/ Cuckoo Search:

Cuckoo Search (CS) est un algorithme intelligent basé sur l'essaim, et qui a été développé par (Yang et Deb, 2009). Il est inspiré du comportement de l'oiseau (coucou) en posant ses œufs. Le coucou ne construit pas son nid, mais sélectionne un nid d'autres espèces et leur permet de couvrir son petit. Nous utilisons une méthode simple à appliquer l'algorithme; un nid correspond à un coucou et aussi à un œuf. La procédure standard balance entre une marche aléatoire locale et globale. La marche globale est implémentée comme la recherche de coucou d'un nouveau nid par le vol Lévy (Yang et Deb, 2009). La marche locale est présentée comme; une probabilité de p_a des cas que les oiseaux hôtes découvrent que les œufs ne sont pas les leurs et ils les abandonnent, alors le coucou cherche de nouveau un autre nid (solution) loin du nid actuel (la solution actuelle). Dans ce travail, nous utilisons un CS amélioré, qui est développé par (Ouaarab et al., 2014), où, ce n'est pas tous les coucous qui font la marche globale, mais seulement, une probabilité p_c de coucous intelligents cherchent de nouvelles solutions dans d'autres zones par le vol Lévy.

La fraction p_c des coucous intelligents, qui cherchent de meilleurs solutions par le vol Lévy est comme suit:

$$x_i^{t+1} = x_i^t + \alpha L(\lambda) \quad (\text{III.25})$$

Où L est la fonction de vol Lévy (Yang et Deb, 2013), α est un facteur scalaire. Après l'obtention de la nouvelle solution (solution d'essai), nous la comparons avec une autre solution aléatoire (j), si la nouvelle est mieux, nous remplaçons la plus mauvaise (j) par la nouvelle solution. Il y a une probabilité p_c que les œufs sont abandonnés par les oiseaux hôtes, alors le coucou cherche un nouveau nid dans la zone locale comme suit:

$$x_i^{t+1} = \begin{cases} x_i^t + s(x_p^t - x_q^t) & \text{si } r > p_a \\ x_i^t & \text{ailleurs} \end{cases} \quad (\text{III.26})$$

x_p et x_q sont des solutions aléatoires différentes de x_i , s est un facteur scalaire.

La procédure est illustrée dans l'Algorithme 3.4.

Algorithm 3.4: Improved Cuckoo Search

Initialiser des solutions aléatoires;

TantQue $max_Generation$ n'est pas atteint

Recherche par fraction (p_c);

Obtenir un coucou aléatoirement par le vol Lévy;

Evaluer sa fonction-objectif F_i ;

Obtenir un nid aléatoirement (j)

si F_i est mieux que F_j

Remplacer le mauvais nid (j) par le nid d'essai (i);

Fin si

Fraction (p_a) des mauvaises nids sont abandonnés et d'autres sont construits;

Fin TantQue

Les paramètres communs sont, $max_Generation$ est 30, la taille de la population est 50. Les autres sont présentés dans le Tableau III.2.

Tableau III.2 Les paramètres des algorithmes

GA		ES		DE		CS amélioré	
Par.	Val.	Par.	Val.	Par.	Val.	Par.	Val.
$crois.$	BMW	p_m	0.3	F	0.2	p_c	0.5
p_m	0.04	λ	36	C_r	0.6	α	0.1
α	0.33	μ	14			λ	1.5
σ	0.4	σ^0	0.3			p_a	0.25
		n	2			s	0.2

III.4 Commande Linéaire (PID) de Quadrotor:

III.4.1 Structure de la Commande:

L'objectif ici est la conception des correcteurs classiques (PID) pour le suivi de la trajectoire du quadrotor. Les paramètres du correcteur vont être ajustés par les différents algorithmes inspirés de la nature et qui ont été présentés de façon minutieuse précédemment. Cependant, le quadrotor est un robot à 6DOF (six degrés de liberté) sauf que la structure du PID est mono variable. À cet effet, nous développons un ensemble de PID pour toutes les variables de sorties du quadrotor. La loi de commande générale du correcteur PID est donnée par:

$$U_q = k_p(q_d - q) + k_d(\dot{q}_d - \dot{q}) + k_i \int (q_d - q) dt \quad (\text{III.27})$$

Où: $\mathbf{q} = [x \ y \ z \ \varphi \ \theta \ \psi]$. La génération de la loi de commande pour chaque variable de \mathbf{q} sera présentée ultérieurement. Le quadrotor est commandé par les vitesses angulaires des rotors, mais le rotor n'a pas une réponse temps réel. Nous devons donc considérer la dynamique de l'actionneur comme celle de la dynamique (Rotor Dynamic; RD) du projet OS4 (Bouabdallah, 2007).

$$RD(s) = \frac{\omega_i}{\omega_{di}} = \frac{0.936}{0.178 s + 1} \quad (\text{III.28})$$

Nous notons, ω_{di} et ω_i pour la vitesse désirée et actuelle du $i^{\text{ème}}$ rotor. Nous avons extrait les forces et couples actuelles du couplage entre les vitesses des rotors suivant la formule (III.8), en considérant que les correcteurs génèrent les entrées désirées ($U_{d1}, U_{d2}, U_{d3}, U_{d4}$). Chaque rotor obtient son signal d'entrée (ω_{di}) de ($U_{d1}, U_{d2}, U_{d3}, U_{d4}$) par le découplage présenté par:

$$\begin{cases} \omega_{d1} = \sqrt{\frac{1}{4b} U_{d1} - \frac{1}{2b} U_{d3} + \frac{1}{4d} U_{d4}} \\ \omega_{d2} = \sqrt{\frac{1}{4b} U_{d1} - \frac{1}{2b} U_{d2} - \frac{1}{4d} U_{d4}} \\ \omega_{d3} = \sqrt{\frac{1}{4b} U_{d1} + \frac{1}{2b} U_{d3} + \frac{1}{4d} U_{d4}} \\ \omega_{d4} = \sqrt{\frac{1}{4b} U_{d1} + \frac{1}{2b} U_{d2} - \frac{1}{4d} U_{d4}} \end{cases} \quad (\text{III.29})$$

Comme il est présenté dans (III.7), le modèle du quadrotor est fortement couplé et sous-actionné; où les coordonnées x et y n'ont pas d'entrées directes. Il y a deux types d'interactions: la première est celle entre l'altitude (z) et les angles d'attitude (φ, θ); pour l'éliminer nous remplaçons U_{d1} par U'_{d1} avec un facteur adaptatif ($k = \frac{1}{\cos \varphi \cos \theta}$);

$$U'_{d1} = \frac{1}{\cos \varphi \cos \theta} U_{d1} \quad (\text{III.30})$$

La deuxième interaction est présente entre les variables de translation et de rotation. x et y sont des degrés de liberté n'ont pas des commandes (entrées) réelles. Dans ce cas, nous utilisons la structure de commande en cascade. En outre, nous générons les entrées virtuelles (u_x, u_y) , puis elles sont utilisées pour générer les angles d'attitude désirés (φ_d, θ_d) . Nous avons extrait (u_x, u_y) du modèle mathématique (III.7) où $(u_x = \ddot{x}, u_y = \ddot{y})$; alors, la transformation de translation à l'attitude désirée (φ_d, θ_d) est présentée comme suit:

$$\begin{cases} \varphi_d = \frac{m}{U_{d1}} (u_x \sin \psi - u_y \cos \psi) \\ \theta_d = \frac{m}{U_{d1}} (u_x \cos \psi + u_y \sin \psi) \end{cases} \quad (\text{III.31})$$

Dans ce cas, nous avons deux boucles, la boucle intérieure qui contient les boucles de l'altitude (z) et les trois angles (φ, θ, ψ) avec une commande directe, et la boucle extérieure qui contient la boucle des positions de translation (x, y). La structure de la stratégie de commande utilisée est présentée en Figure III.2. Nous allons concevoir l'ensemble des correcteurs pour les variables de la boucle intérieure et puis ceux des variables de la boucle extérieure où la boucle intérieure va considérer comme une partie du modèle. Par la suite, nous allons appliquer cette stratégie de commande en utilisant les différents algorithmes évolutionnaires (GA, ES, DE et CS). Une comparaison entre ces algorithmes sera faite en termes de performances dynamiques.

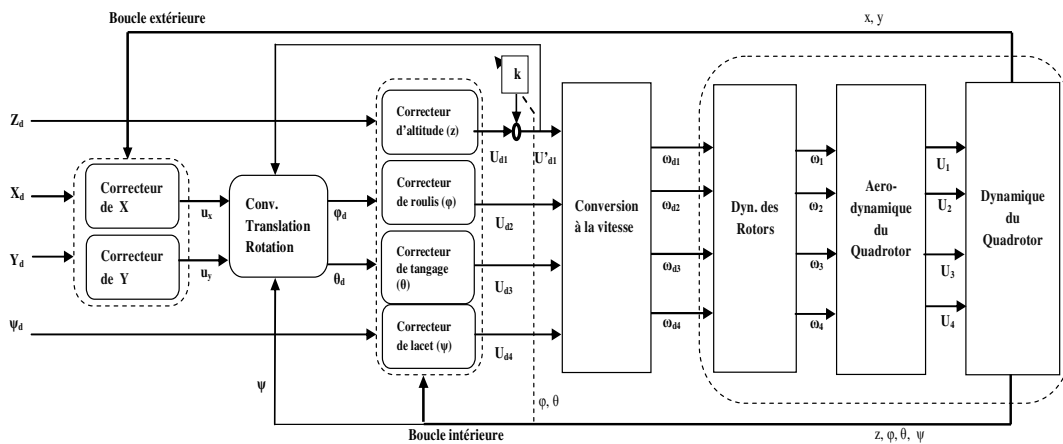


Figure III.2 Structure de commande du quadrotor

Dans le domaine de l'automatique, pour évaluer tel correcteur, les caractéristiques qui généralement être pris sont le premier dépassement, le temps de réponse, le temps de montée, etc. Il y a des autres indices de la performance dépendent sur l'erreur dynamique sont utilisés plutôt que combiner deux ou plus caractéristiques dans un objectif multi-variable, parmi lesquelles on trouve:

- La moyenne de l'erreur carrée (*MSE: Mean Squared Error*)
- L'intégrale du temps multiplié par la valeur absolue de l'erreur (*ITAE: Integral of Time multiplied by Absolute Error*)
- L'intégrale de la valeur absolue de l'erreur (*IAE: Integral of Absolute magnitude of the Error*)
- L'intégrale de l'erreur carrée (*ISE: Integral of the Squared Error*)
- L'intégrale du temps multiplié par l'erreur carrée (*ITSE: Integral of Time multiplied by the Squared Error*).

L'ISE est une mesure indirecte de la vitesse de réponse, il est l'un des indices de performance les plus utilisés (Rhimian et Tavazoei, 2014; Shahemabadi et al., 2013). Par l'expérience de (Jain et Nigam, 2008), l'ISE a des avantages comparé à un autre critère incluant le temps, par exemple l'ITAE, où le critère ITAE dans lequel on ne peut pas éviter le grand dépassement avec un petit temps de montée dans la réponse indicielle du système, le critère ISE a un dépassement et un temps de montée acceptables comparé avec l'ITAE.

La fonction fitness est donnée par:

$$fitness = ISE = \int_0^{\tau} e(t)^2 dt \quad (III.32)$$

Où $e(t)$ est l'erreur entre le signal de référence est l'état actuelle de chaque degré de liberté $e(t) = q_r(t) - q(t)$ et $q = [x \ y \ z \ \varphi \ \theta \ \psi]$. En raison de la nature du système, multi-variables, le processus d'ajustement des paramètres des correcteurs est implémenté pour chaque degré de liberté par les algorithmes évolutionnaires mentionnés dans la section III.3.2 (les algorithmes génétiques GA, stratégies évolutionnaires ES, évolution différentielle DE, et cuckoo search amélioré CS).

III.4.2 Résultats de Simulation:

Dans l'ingénierie, plusieurs travaux ont utilisés des études comparatives entre différentes méthodes afin d'améliorer la qualité des résultats selon un certain nombre des performances. Parmi ces travaux, on trouve des études comparatives effectuées sur la base de problèmes d'optimisation par des algorithmes inspirés de la nature (Sivalingam et al., 2017; Acampora et al., 2014; Klempka et Filipowicz, 2017; Rout et al., 2018). L'objectif de ce travail est de présenter les résultats atteints par l'ajustement des paramètres des correcteurs PID en utilisant des algorithmes inspirés de la nature; GA, ES, DE et CS amélioré. Ces algorithmes stochastiques sont appliqués afin d'obtenir les paramètres optimaux des correcteurs du quadrotor.

Comme nous avons déjà mentionné dans la section III.2, le roulis (φ) et le tangage (θ) ont la même dynamique, alors on considère que ses correcteurs ont les mêmes paramètres dans la procédure d'optimisation, également la même chose en rapport avec les coordonnées des mouvements de translation x et y . Nous avons réalisé la procédure de réglage des correcteurs des différents variables de sorties (z , roulis-tangage, lacet et x - y) par les quatre algorithmes d'optimisation sélectionnés, selon la fonction d'objectif sélectionnée (ISE) qui est présenté dans la section III.4.1.

La Figure III.3 représente l'évolution de l'ISE de chaque variable de sortie par les quatre algorithmes selon les références sélectionnées et présentées dans le Tableau III.3. On présente dans le même tableau les valeurs (ISE) des meilleurs correcteurs obtenus de chaque algorithme. Le tableau III.4 présente les paramètres optimaux des correcteurs obtenus par chaque algorithme.

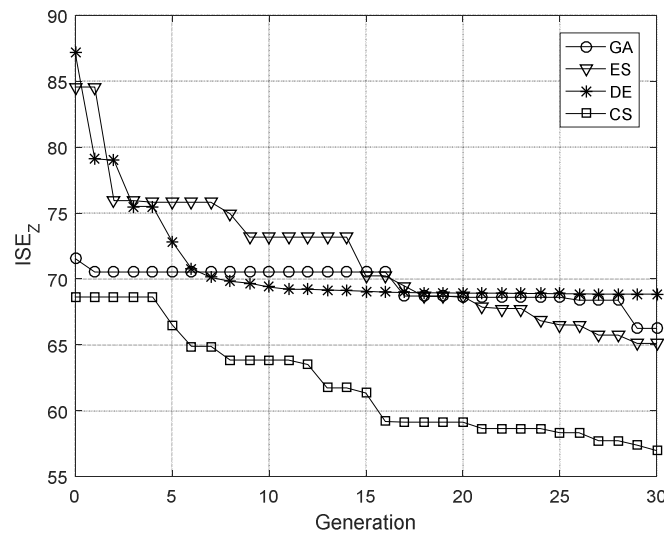


Figure III.3-a Évolution de la fonction-objectif pour l'altitude (z)

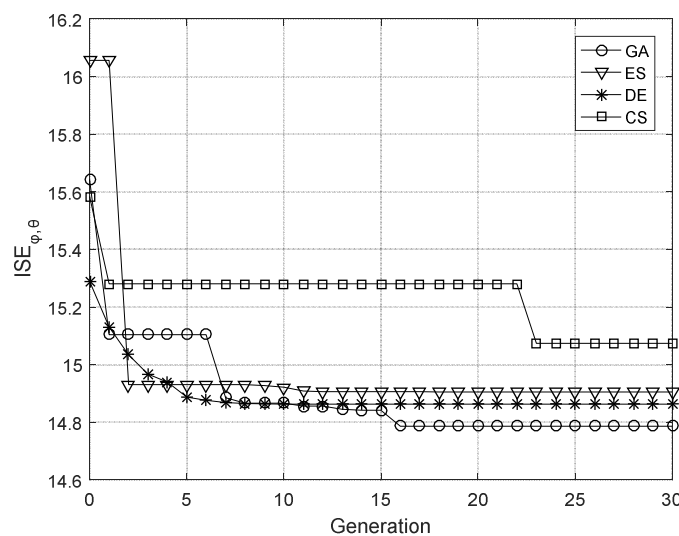


Figure III.3-b Évolution de la fonction-objectif du roulis-tangage (φ, θ)

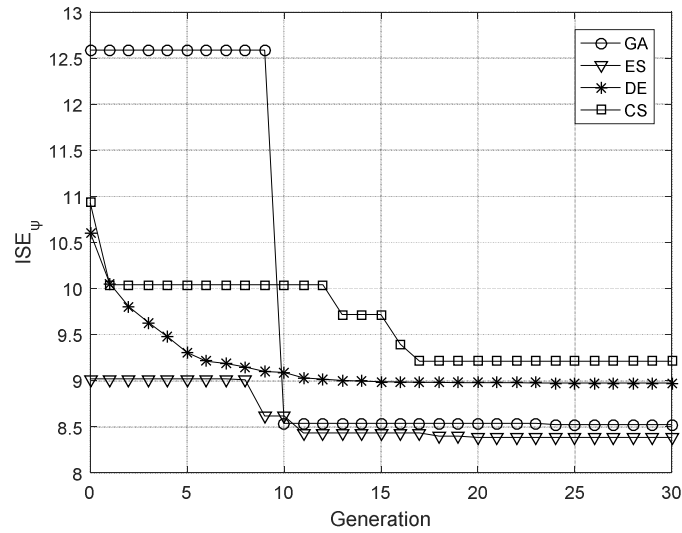


Figure III.3-c Évolution de la fonction-objectif du lacet (ψ)

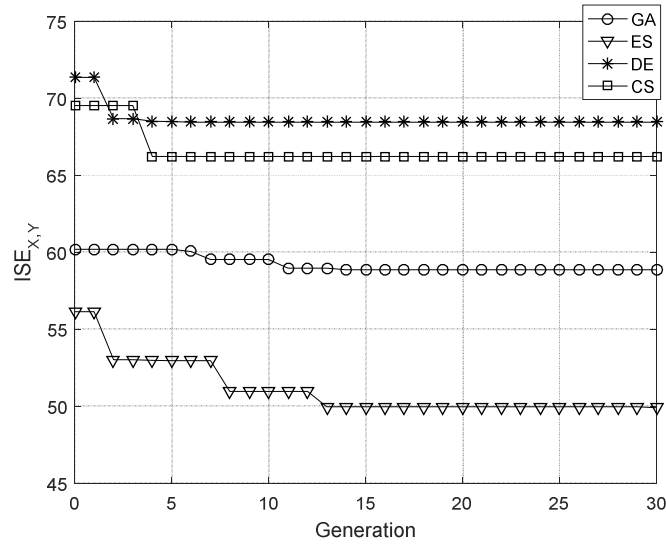


Figure III.3-d Évolution de la fonction-objectif de $x-y$

Tableau III.3 Valeurs de la fonction-objectif (ISE)

Variable	Référence	GA	ES	DE	CS
z	$2 m$	66.2466	65.1213	68.8395	56.9583
φ, θ	$1 rad$	14.7867	14.9053	14.8625	15.0745
ψ	$1 rad$	8.5226	8.3853	8.9753	9.2107
x, y	$2 m$	58.8647	49.9717	68.4469	66.2255

Tableau III.4 Les gains des correcteurs PID optimaux

Variable		GA	ES	DE	CS
z	K_p	9.6135	7.7762	10.9604	8.2018
	K_i	4.0704	3.8678	4.1644	4.6874
	K_d	4.8897	3.8457	4.5612	6.0404
φ, θ	K_p	0.4521	0.4801	0.6114	0.5567
	K_i	0	0	0	0
	K_d	0.3622	0.5857	0.2982	0.3377
ψ	K_p	0.4637	0.4967	0.143	0.1345
	K_i	0	0	0	0
	K_d	0.1872	0.2021	0.0727	0.0832
x, y	K_p	4.81	8.0178	3.1309	3.0888
	K_i	0	0.15	0	0
	K_d	0.89	1.7179	1.1918	1.1464

Il faut mentionner dans ce cas que nous ne pouvons pas ajuster les paramètres des correcteurs de x - y sauf si nous réglons premièrement les correcteurs des roulis-tangage parce que x - y sont des degrés de libertés sous-actionnés. Les correcteurs qui correspondent, génèrent seulement des signaux de commande virtuelles qui à leur tour génèrent les roulis et tangages désirés. Alors, les correcteurs des roulis-tangage sont considérés comme une partie du système à commander, pour cette raison, les courbes d'évolution du critère (ISE) des correcteurs de x - y sont séparés (Figure III.3-d); la conception des correcteurs x - y par chaque algorithme (GA, ES, DE et CS) est basée sur les correcteurs propres des roulis-tangage.

Selon les résultats présentés précédemment, il est clair que nous avons obtenu un meilleur correcteur pour l'altitude par l'utilisation du CS, les meilleurs correcteurs des roulis-tangage sont obtenus par le GA, et enfin les meilleurs correcteurs du lacet et de x - y sont obtenus par l'ES.

L'évaluation précédente est réalisée selon les références (Tableau III.3) pour ajuster les paramètres des correcteurs (Tableau III.4). Après l'obtention des correcteurs, nous allons les tester pour le suivi de la trajectoire. Il faut mentionner ici, que nous examinons les correcteurs obtenus par chaque algorithme séparément.

Afin de tester les correcteurs sur une dynamique dure, nous avons suggéré la trajectoire suivante: le quadrotor commence de la position initiale $(x_0, y_0, z_0)=(0, -4, 0)$ et monte à 2 m dans l'axe z , dans l'instant 10 s le quadrotor fait une trajectoire en spirale et tourne autour de lui-même pendant 15 seconds; $x_d = 4 \sin\left(\frac{2\pi}{15} t\right)$, $y_d = -4 \cos\left(\frac{2\pi}{15} t\right)$, $z_d = 0.1 t$, $\psi_d = \pi \sin\left(\frac{2\pi}{15} t\right)$.

La Figure III.4 présente la simulation du suivi de trajectoire en 3D, et la Figure III.5 présente les réponses dynamiques de position de translation (x , y et z). Afin de présenter la précision de la réponse du système, la Figure III.6 montre les erreurs des coordonnées de translation dans les derniers 10 seconds. Concernent les angles, la Figure III.7 montre les réponses des coordonnées d'orientation (φ , θ et ψ). La Figure III.8 montre les erreurs des coordonnées d'orientation dans les derniers 10 seconds.

Dans la Figure III.4, qui présente la simulation en 3D de la translation, nous pouvons remarquer que les courbes des correcteurs liées au GA et à l'ES nous donnent un meilleur suivi de la trajectoire. Ce qui est confirmé par les réponses de translation de la Figure III.5.

Dans la Figure III.6, l'erreur minimale de x et y (10 *cm* dans les pics) est observée dans la courbe de l'ES. Où la plus mauvaise (20 *cm*) est observée dans la courbe du DE. Pour l'altitude (z) aussi, les meilleurs correcteurs de suivi de la trajectoire sont ceux du GA et de l'ES, alors que le mauvais résultat est celui trouvé par l'application du DE.

Dans la Figure III.7, représentant les réponses d'orientation, nous observons que les meilleures réponses des roulis-tangage (φ , θ) sont celles du GA et du CS. Dans la réponse du lacet (ψ), la courbe de l'ES est la meilleure. Plus précisément, la Figure III.8 qui présente les erreurs dans les derniers 10 seconds, la mauvaise erreur est observée sur la courbe du DE avec un maximum d'erreur de 0.2 *rad* pour le roulis et de 0.15 *rad* de tangage.

À cause de la sensibilité notée dans les réponses de rotation du système commandé par le correcteur de l'ES, on constate un meilleur suivi dans les coordonnées de translation pour le correcteur de l'ES et puis celui du GA, ce qui est confirmé par les résultats du tableau III.5 qui présente le critère (ISE) de cette trajectoire. La meilleure stabilisation dans la rotation est observée pour le correcteur GA, comme nous avons conclu précédemment.

À cause de la bonne stabilisation de la rotation par le GA et le suivi de translation acceptable, nous pouvons conclure ici que le meilleur correcteur d'une façon générale, est celui régler par le GA.

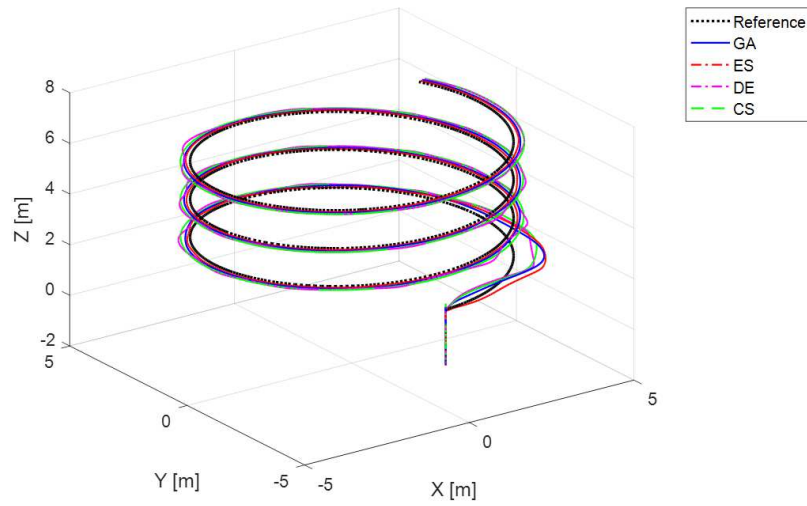


Figure III.4 Le suivi de la trajectoire en 3D

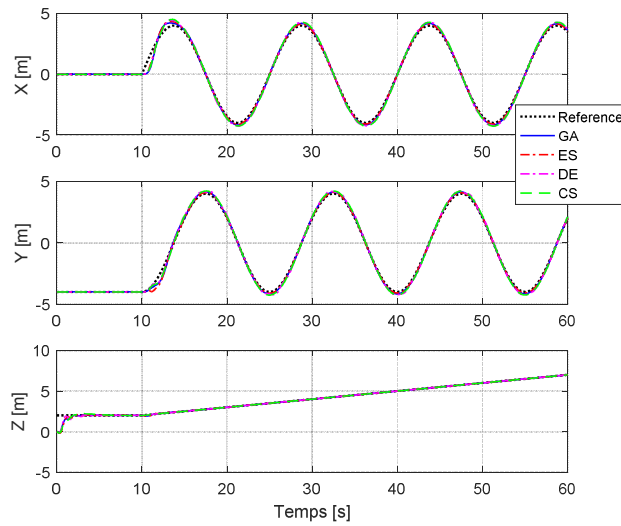


Figure III.5 Les réponses des coordonnées de translation

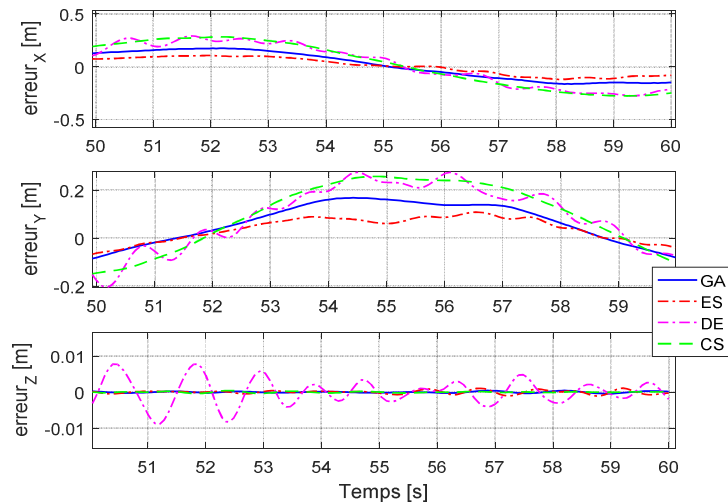


Figure III.6 Les erreurs des coordonnées de translation

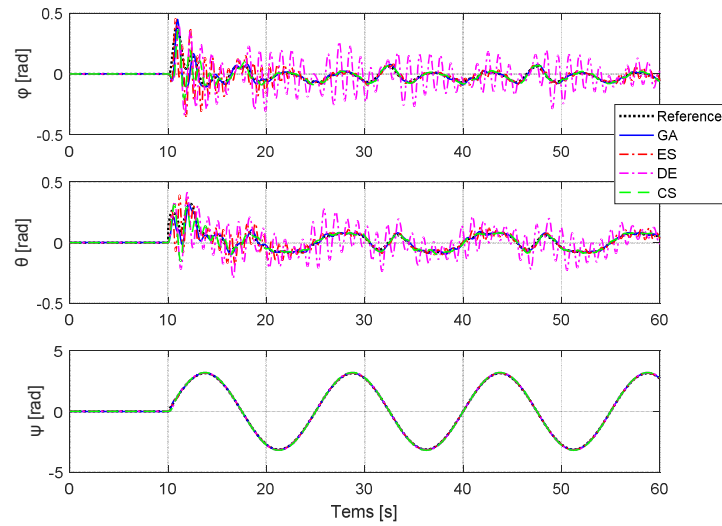


Figure III.7 Les réponses des coordonnées de rotation

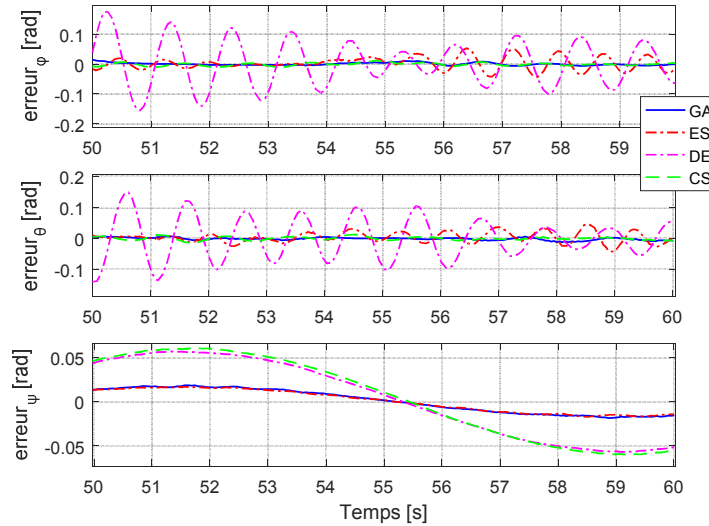


Figure III.8 Les erreurs des coordonnées de rotation

Tableau III.5 Le critère ISE de la trajectoire

	GA	ES	DE	CS
x	36.5709	32.1692	48.8904	53.5110
y	17.2932	21.6567	23.9190	25.2884
z	74.6024	75.4515	69.7439	79.8864
φ	0.9595	4.6924	8.7895	0.9665
θ	0.7733	9.1578	6.3536	1.1624
ψ	0.3705	0.4192	2.2186	2.2467

III.5 Conclusion:

Dans ce chapitre, en premier lieu nous avons établi la modélisation dynamique du quadrotor. En suite, la structure de commande en cascade a été conçue afin de découpler les degrés de liberté contrôlés (z , φ , θ et ψ) et les degrés de liberté sous-actionnées (x et y). Nous avons opté dans ce chapitre, à commander le quadrotor par la technique linéaire PID où leurs paramètres ont été sélectionnés par des algorithmes évolutionnaires; GA, ES, DE et CS où la fonction-objectif sélectionnée était ISE. Les variables étaient déjà découplées, alors, la procédure de la sélection des paramètres de correcteur, optimisation, a été faite de manière décentralisé. Enfin et d'après les résultats trouvés, nous avons constaté que les algorithmes génétiques ont permis de fournir le meilleur correcteur testé sur le suivi de la trajectoire. Par la suite, nous allons commander le quadrotor par développer des lois de commande non-linéaires.

Chapitre IV:

Commande du Quadrotor par les Techniques Non-Linéaires

IV.1 Introduction:

Le quadrotor est classé parmi les systèmes non-linéaires, ainsi, nous allons considérer dans ce chapitre la commande de ce mini-drone par des méthodes classiques non-linéaires. En premier lieu, nous allons présenter la représentation d'état du modèle dynamique. Ensuite, nous allons établir en détails le développement des commandes non-linéaires communes, qui sont le Backstepping et le Mode Glissant. A la fin, un test de robustesse sera effectué sur les approches de commande développées sur le quadrotor et la simulation des dynamiques sera montrée à partir de l'outil Matlab/Simulink.

IV.2 Commande Non-Linéaire du Quadrotor:

Le modèle du quadrotor est effectivement compliqué et sous-actionné. Comme nous avons mentionné précédemment, les degrés de liberté x et y n'ont pas de commandes réelles, nous devons alors considérer des commandes virtuelles (u_x et u_y), et appliquer une structure de commande en cascade comme nous l'avons effectué précédemment avec le PID (Figure IV.1).

Nous considérons les états de notre système:

$$x_1 = z, x_2 = \dot{z}, x_3 = x, x_4 = \dot{x}, x_5 = y, x_6 = \dot{y}, x_7 = \varphi, x_8 = \dot{\varphi}, x_9 = \theta, x_{10} = \dot{\theta}, x_{11} = \psi, x_{12} = \dot{\psi}$$

Le modèle (III.7) peut être récréé $\dot{X} = f(X, U)$ et détaillé comme suit:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{pmatrix} x_2 \\ \frac{1}{m}(\cos x_7 \cos x_9)U_1 - g \\ x_4 \\ \frac{1}{m}u_x U_1 \\ x_6 \\ \frac{1}{m}u_y U_1 \\ x_8 \\ \frac{(I_y - I_z)}{I_x}x_{10}x_{12} - \frac{J_r \Omega_r}{I_x}x_{10} + \frac{l}{I_x}U_2 \\ x_{10} \\ \frac{(I_z - I_x)}{I_y}x_8x_{12} + \frac{J_r \Omega_r}{I_y}x_8 + \frac{l}{I_y}U_3 \\ x_{12} \\ \frac{(I_x - I_y)}{I_z}x_8x_{10} + \frac{1}{I_z}U_4 \end{pmatrix} \quad (\text{IV.1})$$

Où:

$$\begin{cases} u_x = \cos\varphi \sin\theta \cos\psi + \sin\varphi \sin\psi \\ u_y = \cos\varphi \sin\theta \sin\psi - \sin\varphi \cos\psi \end{cases} \quad (\text{IV.2})$$

Finalement, après la génération des commandes virtuelles (u_x et u_y) comme elles sont représentées sur la Figure IV.1, nous pouvons générer les valeurs désirées du roulis (φ_d) et du tangage (θ_d).

À partir des équations (IV.2), nous avons:

$$\begin{cases} \varphi_d = \sin^{-1}(u_x \sin\psi - u_y \cos\psi) \\ \theta_d = \sin^{-1}\left(\frac{u_x \cos\psi + u_y \sin\psi}{\cos\varphi}\right) \end{cases} \quad (\text{IV.3})$$

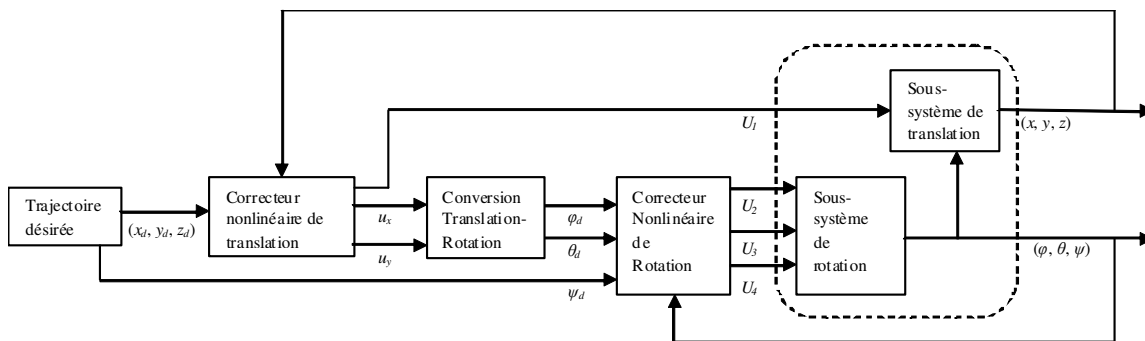


Figure IV.1 Structure en cascade de la commande non-linéaire du quadrotor

IV.3 Commande par Backstepping Intégral:

La commande par Backstepping est une méthode de conception de commande non-linéaire qui a été considérée comme une alternative à la linéarisation par retour. L'idée principale de cette méthode est de partitionner l'ensemble du système en nombres de sous-systèmes en cascade (Tan et al. 2016). En conséquence, les états du premier sous-système ont servi de variables de commande pour le sous-système suivant. Les sous-systèmes étant connectés en cascade, l'entrée du premier sous-système provient automatiquement de la sortie du deuxième sous-système.

L'entrée idéale du deuxième sous-système est calculée de manière similaire à celle du premier sous-système. Les entrées désirées pour tous les sous-systèmes sont calculées de manière séquentielle jusqu'à l'arrivée du dernier sous-système. À titre d'exemple, l'entrée désirée pour le dernier sous-système donne l'expression de l'entrée de commande réelle et, en conséquence, les concepteurs

peuvent mettre en œuvre une loi de retour d'état pour l'ensemble du système non-linéaire. Cette approche constitue souvent une caractéristique avantageuse lors de la conception de la loi de commande pour les systèmes dynamiques non-linéaires complexes.

Le correcteur Backstepping renforcé par une action intégrale est utilisé comme un outil pour concevoir les correcteurs d'altitude, d'attitude et de position.

Par la suite, la conception des lois de commande, réelles (U_1, U_2, U_3 et U_4) et virtuelles (u_x et u_y) par Backstepping intégral sont données en détails (Fan et al. 2017):

A. La commande de l'altitude Z:

Pour la conception d'une loi de commande pour le sous-système d'altitude (U_1). Nous obtenons au début, l'erreur de l'état x_1 , sa dérivée et son intégral:

$$\begin{aligned} e_1 &= x_{1d} - x_1 \\ \dot{e}_1 &= \dot{x}_{1d} - x_2 \\ \xi_1 &= \int_0^t e_1(\tau) d\tau \end{aligned} \quad (IV.4)$$

On définit la fonction de Lyapunov et sa dérivée:

$$V(e_1) = \frac{1}{2} e_1^2 \quad (IV.5)$$

$$\dot{V}(e_1) = e_1(\dot{x}_{1d} - x_2) \quad (IV.6)$$

Vu qu'il n'y a pas une commande directe dans (IV.6), nous considérons x_2 comme une commande virtuelle. Afin de rendre $\dot{V}(e_1)$ semi-défini négative, nous déterminons la commande virtuelle désirée comme suit:

$$x_{2d} = \dot{x}_{1d} + c_1 e_1 + \lambda_1 \xi_1 \quad (IV.7)$$

Où, c_1 et λ_1 sont positives.

L'objectif est que x_2 suit x_{2d} et nous appelons la déviation entre x_{2d} et x_2 l'état d'erreur e_2 :

$$e_2 = x_{2d} - x_2 \quad (IV.8)$$

La commande virtuelle x_2 et la dérivée de l'état d'erreur e_2 sont comme ceci:

$$x_2 = \dot{x}_{1d} + c_1 e_1 + \lambda_1 \xi_1 - e_2 \quad (IV.9)$$

$$\dot{e}_2 = \ddot{x}_{1d} + c_1 \dot{e}_1 + \lambda_1 e_1 - \frac{1}{m} (\cos x_7 \cos x_9) U_1 + g \quad (\text{IV.10})$$

Puisque la technique de commande appliquée est le Backstepping Intégral, on étend la fonction de Lyapunov par ξ_I :

$$V_1 = \frac{1}{2} e_1^2 + \frac{1}{2} \lambda_1 \xi_1^2 \quad (\text{IV.11})$$

Selon (IV.4) et (IV.9), la dérivée est définie comme suit:

$$\dot{V}_1 = e_1 \dot{e}_1 + \xi_1 \dot{\xi}_1 = -c_1^2 e_1 + e_1 e_2 \quad (\text{IV.12})$$

La dérivée de la fonction de Lyapunov \dot{V}_1 doit être semi-défini négative, e_2 doit être limité et que nous devons incorporer dans la fonction de Lyapunov:

$$V_2 = \frac{1}{2} e_1^2 + \frac{1}{2} \lambda_1 \xi_1^2 + \frac{1}{2} e_2^2 \quad (\text{IV.13})$$

Selon (IV.11) et (IV.12) la dérivée de la fonction de Lyapunov est comme suit:

$$\dot{V}_2 = -c_1^2 e_1 + e_1 e_2 + e_2 (\ddot{x}_{1d} + c_1 \dot{e}_1 + \lambda_1 e_1 - \frac{1}{m} (\cos x_7 \cos x_9) U_1 + g) \quad (\text{IV.14})$$

Afin de rendre \dot{V}_2 semi-défini négative, nous pouvons mettre la loi de commande U_1 comme suit:

$$U_1 = \frac{m}{(\cos x_7 \cos x_9)} (\ddot{x}_{1d} + (1 - c_1^2 + \lambda_1) e_1 - c_1 \lambda_1 \xi_1 + (c_1 + c_2) e_2 + g) \quad (\text{IV.15})$$

Où, c_1 , c_2 et λ_1 sont positives.

B. La commande de la position X :

Pour la conception d'une loi de commande pour le sous-système de x (u_x). Nous obtenons au début, l'erreur de l'état x_3 , sa dérivée et son intégral:

$$\begin{aligned} e_3 &= x_{3d} - x_3 \\ \dot{e}_3 &= \dot{x}_{3d} - \dot{x}_4 \\ \xi_2 &= \int_0^t e_3(\tau) d\tau \end{aligned} \quad (\text{IV.16})$$

On définit la fonction de Lyapunov et sa dérivée:

$$V(e_3) = \frac{1}{2} e_3^2 \quad (\text{IV.17})$$

$$\dot{V}(e_3) = e_3 (\dot{x}_{3d} - \dot{x}_4) \quad (\text{IV.18})$$

Vu qu'il n'y a pas une commande directe dans (IV.18), nous considérons x_4 comme une commande virtuelle, afin de rendre $\dot{V}(e_3)$ semi-défini négative et nous déterminons la commande virtuelle désirée comme suit:

$$x_{4d} = \dot{x}_{3d} + c_3 e_3 + \lambda_2 \xi_2 \quad (\text{IV.19})$$

Où, c_3 et λ_2 sont positives.

L'objectif est que x_4 suit x_{4d} et nous appelons la déviation entre x_{4d} et x_4 l'état d'erreur e_4 :

$$e_4 = x_{4d} - x_4 \quad (\text{IV.20})$$

La commande virtuelle x_4 et la dérivée de l'état d'erreur e_4 sont comme ceci:

$$x_4 = \dot{x}_{3d} + c_3 e_3 + \lambda_2 \xi_2 - e_4 \quad (\text{IV.21})$$

$$\dot{e}_4 = \ddot{x}_{3d} + c_3 \dot{e}_3 + \lambda_2 e_3 - \frac{1}{m} u_x U_1 \quad (\text{IV.22})$$

Puisque la technique de commande appliquée est le Backstepping Intégral, on étend la fonction de Lyapunov par ξ_2 :

$$V_3 = \frac{1}{2} e_3^2 + \frac{1}{2} \lambda_2 \xi_2^2 \quad (\text{IV.23})$$

Selon (IV.16) et (IV.21), la dérivée est définie comme suit:

$$\dot{V}_3 = e_3 \dot{e}_3 + \xi_2 \dot{\xi}_2 = -c_3^2 e_3 + e_3 e_4 \quad (\text{IV.24})$$

La dérivée de la fonction de Lyapunov \dot{V}_3 doit être semi-défini négative, e_4 doit être limité, que devons incorporer dans la fonction de Lyapunov:

$$V_4 = \frac{1}{2} e_3^2 + \frac{1}{2} \lambda_2 \xi_2^2 + \frac{1}{2} e_4^2 \quad (\text{IV.25})$$

Selon (IV.23) et (IV.24) la dérivée de la fonction de Lyapunov est comme suit:

$$\dot{V}_4 = -c_3^2 e_3 + e_3 e_4 + e_4 (\ddot{x}_{3d} + c_3 \dot{e}_3 + \lambda_2 e_3 - \frac{1}{m} u_x U_1) \quad (\text{IV.26})$$

Afin de rendre \dot{V}_4 semi-défini négative, on peut mettre la loi de commande u_x comme suit:

$$u_x = \frac{m}{U_1} (\ddot{x}_{3d} + (1 - c_3^2 + \lambda_2) e_3 + (c_3 + c_4) e_4 - c_3 \lambda_2 \xi_2) \quad (\text{IV.27})$$

Où, c_3 , c_4 et λ_2 sont positives.

C. La commande de la position Y:

Pour la conception d'une loi de commande pour le sous-système de y (u_y). Nous obtenons au début, l'erreur de l'état x_5 , sa dérivée et son intégral:

$$\begin{aligned} e_5 &= x_{5d} - x_5 \\ \dot{e}_5 &= \dot{x}_{5d} - \dot{x}_5 \\ \xi_3 &= \int_0^t e_5(\tau) d\tau \end{aligned} \quad (IV.28)$$

On définit la fonction de Lyapunov et sa dérivée:

$$V(e_5) = \frac{1}{2} e_5^2 \quad (IV.29)$$

$$\dot{V}(e_5) = e_5(\dot{x}_{5d} - \dot{x}_5) \quad (IV.30)$$

Vu qu'il n'y a pas une commande directe dans (IV.30), nous considérons x_6 comme une commande virtuelle, afin de rendre $\dot{V}(e_5)$ semi-défini négative et nous déterminons la commande virtuelle désirée comme suit:

$$x_{6d} = \dot{x}_{5d} + c_5 e_5 + \lambda_3 \xi_3 \quad (IV.31)$$

Où, c_5 et λ_3 sont positives.

L'objectif est que x_6 suit x_{6d} et nous appelons la déviation entre x_{6d} et x_6 l'état d'erreur e_6 :

$$e_6 = x_{6d} - x_6 \quad (IV.32)$$

La commande virtuelle x_6 et la dérivée de l'état d'erreur e_6 sont comme ceci:

$$x_6 = \dot{x}_{5d} + c_5 e_5 + \lambda_3 \xi_3 - e_6 \quad (IV.33)$$

$$\dot{e}_6 = \ddot{x}_{5d} + c_5 \dot{e}_5 + \lambda_3 \dot{e}_5 - \frac{1}{m} u_y U_1 \quad (IV.34)$$

Puisque la technique de commande appliquée est le Backstepping Intégral, on étend la fonction de Lyapunov par ξ_3 :

$$V_5 = \frac{1}{2} e_5^2 + \frac{1}{2} \lambda_3 \xi_3^2 \quad (IV.35)$$

Selon (IV.28) et (IV.33), la dérivée est définie comme suit:

$$\dot{V}_5 = e_5 \dot{e}_5 + \xi_3 \dot{\xi}_3 = -c_5^2 e_5^2 + e_5 e_6 \quad (IV.36)$$

La dérivée de la fonction de Lyapunov \dot{V}_5 doit être semi-défini négative, e_6 doit être limité et que nous devons incorporer dans la fonction de Lyapunov:

$$V_6 = \frac{1}{2}e_5^2 + \frac{1}{2}\lambda_3\xi_3^2 + \frac{1}{2}e_6^2 \quad (\text{IV.37})$$

Selon (IV.35) et (IV.36) la dérivée de la fonction de Lyapunov est comme suit:

$$\dot{V}_6 = -c_5^2e_5 + e_5e_6 + e_6(\ddot{x}_{5d} + c_5\dot{e}_5 + \lambda_3e_5 - \frac{1}{m}u_yU_1) \quad (\text{IV.38})$$

Afin de rendre \dot{V}_6 semi-défini négative, on peut mettre la loi de commande u_y comme suit:

$$u_y = \frac{m}{U_1}(\ddot{x}_{5d} + (1 - c_5^2 + \lambda_3)e_5 + (c_5 + c_6)e_6 - c_5\lambda_3\xi_3) \quad (\text{IV.39})$$

Où, c_5 , c_6 et λ_3 sont positives.

D. La commande du roulis φ :

Pour la conception d'une loi de commande pour le sous-système de roulis (U_2). Nous obtenons au début, l'erreur de l'état x_7 , sa dérivée et son intégral:

$$\begin{aligned} e_7 &= x_{7d} - x_7 \\ \dot{e}_7 &= \dot{x}_{7d} - x_8 \\ \xi_4 &= \int_0^t e_7(\tau) d\tau \end{aligned} \quad (\text{IV.40})$$

On définit la fonction de Lyapunov et sa dérivée:

$$V(e_7) = \frac{1}{2}e_7^2 \quad (\text{IV.41})$$

$$\dot{V}(e_7) = e_7(\dot{x}_{7d} - x_8) \quad (\text{IV.42})$$

Vu qu'il n'y a pas une commande directe dans (IV.42), nous considérons x_8 comme une commande virtuelle, afin de rendre $\dot{V}(e_7)$ semi-défini négative et nous déterminons la commande virtuelle désirée comme suit:

$$x_{8d} = \dot{x}_{7d} + c_7e_7 + \lambda_4\xi_4 \quad (\text{IV.43})$$

Où, c_7 et λ_4 sont positives.

L'objectif est que x_8 suit x_{8d} , nous appelons la déviation entre x_{8d} et x_8 l'état d'erreur e_8 :

$$e_8 = x_{8d} - x_8 \quad (\text{IV.44})$$

La commande virtuelle x_8 et la dérivée de l'état d'erreur e_8 sont comme ceci:

$$x_8 = \dot{x}_{7d} + c_7 e_7 + \lambda_4 \xi_4 - e_8 \quad (\text{IV.45})$$

$$\dot{e}_8 = \ddot{x}_{7d} + c_7 \dot{e}_7 + \lambda_4 e_7 - \frac{(I_y - I_z)}{I_x} x_{10} x_{12} + \frac{J_r \Omega_r}{I_x} x_{10} - \frac{l}{I_x} U_2 \quad (\text{IV.46})$$

Puisque la technique de commande appliquée est le Backstepping Intégral, on étend la fonction de Lyapunov par ξ_4 :

$$V_7 = \frac{1}{2} e_7^2 + \frac{1}{2} \lambda_4 \xi_4^2 \quad (\text{IV.47})$$

Selon (IV.40) et (IV.45), la dérivée est définie comme suit:

$$\dot{V}_7 = e_7 \dot{e}_7 + \xi_4 \dot{\xi}_4 = -c_7^2 e_7 + e_7 e_8 \quad (\text{IV.48})$$

La dérivée de la fonction de Lyapunov \dot{V}_7 doit être semi-défini négative, e_8 doit être limité que nous devons incorporer dans la fonction de Lyapunov:

$$V_8 = \frac{1}{2} e_7^2 + \frac{1}{2} \lambda_4 \xi_4^2 + \frac{1}{2} e_8^2 \quad (\text{IV.49})$$

Selon (IV.47) et (IV.48) la dérivée de la fonction de Lyapunov est comme suit:

$$\dot{V}_8 = -c_7^2 e_7 + e_7 e_8 + e_8 (\ddot{x}_{7d} + c_7 \dot{e}_7 + \lambda_4 e_7 - \frac{(I_y - I_z)}{I_x} x_{10} x_{12} + \frac{J_r \Omega_r}{I_x} x_{10} - \frac{l}{I_x} U_2) \quad (\text{IV.50})$$

Afin de rendre \dot{V}_8 semi-défini négative, on peut mettre la loi de commande U_2 comme suit:

$$U_2 = \frac{I_x}{l} (\ddot{x}_{7d} + (1 - c_7^2 + \lambda_4) e_7 + (c_7 + c_8) e_8 - c_7 \lambda_4 \xi_4 - \frac{(I_y - I_z)}{I_x} x_{10} x_{12} + \frac{J_r \Omega_r}{I_x} x_{10}) \quad (\text{IV.51})$$

Où, c_7 , c_8 et λ_4 sont positives.

E. La commande du tangage θ :

Pour la conception d'une loi de commande pour le sous-système de tangage (U_3). Nous obtenons au début, l'erreur de l'état x_9 , sa dérivée et son intégral:

$$\begin{aligned} e_9 &= x_{9d} - x_9 \\ \dot{e}_9 &= \dot{x}_{9d} - x_{10} \\ \xi_5 &= \int_0^t e_9(\tau) d\tau \end{aligned} \quad (\text{IV.52})$$

On définit la fonction de Lyapunov et sa dérivée:

$$V(e_9) = \frac{1}{2} e_9^2 \quad (IV.53)$$

$$\dot{V}(e_9) = e_9(\dot{x}_{9d} - x_{10}) \quad (IV.54)$$

Vu qu'il n'y a pas une commande directe dans (IV.54), nous considérons x_{10} comme une commande virtuelle, afin de rendre $\dot{V}(e_9)$ semi-défini négative et nous déterminons la commande virtuelle désirée comme suit:

$$x_{10d} = \dot{x}_{9d} + c_9 e_9 + \lambda_5 \xi_5 \quad (IV.55)$$

Où, c_9 et λ_5 sont positives.

L'objectif est que x_{10} suit x_{10d} , nous appelons la déviation entre x_{10d} et x_{10} l'état d'erreur e_{10} :

$$e_{10} = x_{10d} - x_{10} \quad (IV.56)$$

La commande virtuelle x_{10} et la dérivée de l'état d'erreur e_{10} sont comme ceci:

$$x_{10} = \dot{x}_{9d} + c_9 e_9 + \lambda_5 \xi_5 - e_{10} \quad (IV.57)$$

$$\dot{e}_{10} = \ddot{x}_{9d} + c_9 \dot{e}_9 + \lambda_5 \dot{e}_9 - \frac{(I_z - I_x)}{I_y} x_8 x_{12} - \frac{I_r \Omega_r}{I_y} x_8 - \frac{l}{I_y} U_3 \quad (IV.58)$$

Puisque la technique de commande appliquée est le Backstepping Intégral, on étend la fonction de Lyapunov par ξ_5 :

$$V_9 = \frac{1}{2} e_9^2 + \frac{1}{2} \lambda_5 \xi_5^2 \quad (IV.59)$$

Selon (IV.52) et (IV.57), la dérivée est définie comme suit:

$$\dot{V}_9 = e_9 \dot{e}_9 + \xi_5 \dot{\xi}_5 = -c_9^2 e_9 + e_9 e_{10} \quad (IV.60)$$

La dérivée de la fonction de Lyapunov \dot{V}_9 doit être semi-défini négative, e_{10} doit être limité que nous devons incorporer dans la fonction de Lyapunov:

$$V_{10} = \frac{1}{2} e_9^2 + \frac{1}{2} \lambda_5 \xi_5^2 + \frac{1}{2} e_{10}^2 \quad (IV.61)$$

Selon (IV.59) et (IV.60) la dérivée de la fonction de Lyapunov est comme suit:

$$\dot{V}_{10} = -c_9^2 e_9 + e_9 e_{10} + e_{10} (\ddot{x}_{9d} + c_9 \dot{e}_9 + \lambda_5 \dot{e}_9 - \frac{(I_z - I_x)}{I_y} x_8 x_{12} - \frac{I_r \Omega_r}{I_y} x_8 - \frac{l}{I_y} U_3) \quad (IV.62)$$

Afin de rendre \dot{V}_{10} semi-défini négative, on peut mettre la loi de commande U_3 comme suit:

$$U_3 = \frac{I_y}{I} (\ddot{x}_{9d} + (1 - c_9^2 + \lambda_5)e_9 + (c_9 + c_{10})e_{10} - c_9\lambda_5\xi_5 - \frac{(I_z - I_x)}{I_y} x_8 x_{12} - \frac{I_r \Omega_r}{I_y} x_8) \quad (\text{IV.63})$$

Où, c_9 , c_{10} et λ_5 sont positives.

F. La commande du lacet ψ :

Pour la conception d'une loi de commande pour le sous-système de lacet (U_4). Nous obtenons au début, l'erreur de l'état x_{11} , sa dérivée et son intégral:

$$\begin{aligned} e_{11} &= x_{11d} - x_{11} \\ \dot{e}_{11} &= \dot{x}_{11d} - x_{12} \\ \xi_6 &= \int_0^t e_{11}(\tau) d\tau \end{aligned} \quad (\text{IV.64})$$

On définit la fonction de Lyapunov et sa dérivée:

$$V(e_{11}) = \frac{1}{2} e_{11}^2 \quad (\text{IV.65})$$

$$\dot{V}(e_{11}) = e_{11}(\dot{x}_{11d} - x_{12}) \quad (\text{IV.66})$$

Vu qu'il n'y a pas une commande directe dans (IV.66), nous considérons x_{12} comme une commande virtuelle, afin de rendre $\dot{V}(e_{11})$ semi-défini négative et nous déterminons la commande virtuelle désirée comme suit:

$$x_{12d} = \dot{x}_{11d} + c_{11}e_{11} + \lambda_6\xi_6 \quad (\text{IV.67})$$

Où, c_{11} et λ_6 sont positives.

L'objectif est que x_{12} suit x_{12d} , nous appelons la déviation entre x_{12d} et x_{12} l'état d'erreur e_{12} :

$$e_{12} = x_{12d} - x_{12} \quad (\text{IV.68})$$

La commande virtuelle x_{12} et la dérivée de l'état d'erreur e_{12} sont comme ceci:

$$x_{12} = \dot{x}_{11d} + c_{11}e_{11} + \lambda_6\xi_6 - e_{12} \quad (\text{IV.69})$$

$$\dot{e}_{12} = \ddot{x}_{11d} + c_{11}\dot{e}_{11} + \lambda_6e_{11} - \frac{(I_x - I_y)}{I_z} x_8 x_{10} - \frac{1}{I_z} U_4 \quad (\text{IV.70})$$

Puisque la technique de commande appliquée est le Backstepping Intégral, on étend la fonction de Lyapunov par ξ_6 :

$$V_{11} = \frac{1}{2}e_{11}^2 + \frac{1}{2}\lambda_6\xi_6^2 \quad (\text{IV.71})$$

Selon (IV.64) et (IV.69), la dérivée est définie comme suit:

$$\dot{V}_{11} = e_{11}\dot{e}_{11} + \xi_6\dot{\xi}_6 = -c_{11}^2e_{11} + e_{11}e_{12} \quad (\text{IV.72})$$

La dérivée de la fonction de Lyapunov \dot{V}_{11} doit être semi-défini négative, e_{12} doit être limité que nous devons incorporer dans la fonction de Lyapunov:

$$V_{12} = \frac{1}{2}e_{11}^2 + \frac{1}{2}\lambda_6\xi_6^2 + \frac{1}{2}e_{12}^2 \quad (\text{IV.73})$$

Selon (IV.71) et (IV.72) la dérivée de la fonction de Lyapunov est comme suit:

$$\dot{V}_{12} = -c_{11}^2e_{11} + e_{11}e_{12} + e_{12}(\ddot{x}_{11d} + c_{11}\dot{e}_{11} + \lambda_6e_{11} - \frac{(I_x - I_y)}{I_z}x_8x_{10} - \frac{1}{I_z}U_4) \quad (\text{IV.74})$$

Afin de rendre \dot{V}_{12} semi-défini négative, on peut mettre la loi de commande U_4 comme suit:

$$U_4 = I_z(\ddot{x}_{11d} + (1 - c_{11}^2 + \lambda_6)e_{11} + (c_{11} + c_{12})e_{12} - c_{11}\lambda_6\xi_6 - \frac{(I_x - I_y)}{I_z}x_8x_{10}) \quad (\text{IV.75})$$

Où, c_{11} , c_{12} et λ_6 sont positives.

Nous avons sélectionné les paramètres du correcteur Backstepping intégral suivant les valeurs représentées dans le tableau ci-dessous:

Tableau IV.1 Les paramètres du correcteur Backstepping intégral

Paramètre	Valeur	Paramètre	Valeur
c_1	5	c_{10}	20
c_2	5	c_{11}	10.5
c_3	12	c_{12}	20
c_4	0.5	λ_1	0.3
c_5	12	λ_2	0.1
c_6	0.5	λ_3	0.1
c_7	10.5	λ_4	1
c_8	20	λ_5	1
c_9	10.5	λ_6	1

IV.4 Commande par Mode Glissant:

Dans les systèmes de commande, la commande en mode glissant (*Sliding Mode Control: SMC*) est une méthode de commande non-linéaire qui modifie la dynamique d'un système non-linéaire en

appliquant un signal de commande discontinu qui oblige le système à "glisser". La loi de commande par retour d'état n'est pas une fonction continue du temps. Au lieu de cela, il peut basculer d'une structure continue à une autre en fonction de la position actuelle dans l'espace d'état. Par conséquent, la commande en mode glissant est une méthode de commande à structure variable. Les structures de commande multiples sont conçues pour que les trajectoires se déplacent toujours vers une région adjacente avec une structure de commande différente de sorte que la trajectoire finale n'existera pas entièrement dans une structure de commande. Au lieu de cela, il glissera le long des limites des structures de commande. Le mouvement du système lorsqu'il glisse le long de ces limites est appelé un mode de glissement et le lieu géométrique constitué des limites est appelé la surface glissante (Zinober, 1990). Dans le contexte de la théorie de la commande moderne, tout système à structure variable, comme un système sous SMC, peut être considéré comme un cas particulier de système hybride dynamique, car le système traverse un espace d'états continu mais passe également par différents modes de commande discrets.

Un nouveau type de commande par mode glissant, appelé mode de glissement terminal (*Terminal Sliding Mode: TSM*), a été proposé par (Venkataraman et Gulati, 1992). La TSM est une approche de commande non-linéaire robuste.

L'idée principale de la commande par mode glissant terminal découle du travail fondamental sur les attracteurs terminaux, et est évoquée par le concept d'attracteurs terminaux qui garantissent une convergence des états dans le temps fini. Tandis qu'en mode de glissement normal, une stabilité asymptotique est promise pour conduire à la convergence des états vers l'origine. Mais cette convergence ne peut être garantie qu'en un temps infini. Dans le TSM, un terme non-linéaire est introduit dans la conception de la surface de glissement afin que le collecteur soit formulé comme un attracteur. Une fois la surface de glissement interceptée, la trajectoire est attirée dans la variété et converge vers l'origine à la suite d'une règle de puissance.

Il existe certaines variantes du TSM, notamment: TSM non-singulier, (Feng et al., 2002) et Fast TSM, (Yu et Man, 2002). Chaque commande générée est composée de deux termes:

$$u_i = u_{ieq} + u_{sw} \quad (IV.76)$$

Où u_{sw} est la partie de discontinuité (*switch*) $u_{sw}=k_i \text{signe}(surface)$. Par la suite, la conception des lois de commande, réelles (U_1, U_2, U_3 et U_4) et virtuelles (u_x et u_y) par *Fast Terminal Sliding Mode Control* est donnée en détails dans (Din et al. 2018):

A. La commande de l'altitude Z:

Pour la conception d'une loi de commande pour le sous-système d'altitude (U_1). Nous obtenons au début, l'erreur de l'état x_1 et sa dérivée:

$$\begin{aligned}\zeta_1 &= x_{1d} - x_1 \\ \dot{\zeta}_1 &= \dot{x}_{1d} - x_2\end{aligned}\quad (IV.77)$$

On définit la surface de glissement du correcteur FTSMC (*Fast Terminal Sliding Mode Controller*):

$$\sigma_1 = \dot{\zeta}_1 + \alpha_1 \zeta_1 + \beta_1 \zeta_1^{\frac{m_1}{n_1}} \quad (IV.78)$$

On pose la fonction candidate de Lyapunov et sa dérivée comme suit:

$$V_1 = \frac{1}{2} \sigma_1^2 \quad (IV.79)$$

$$\dot{V}_1 = \sigma_1 \dot{\sigma}_1 \quad (IV.80)$$

De (IV.78), la dérivée de la surface du glissement est:

$$\dot{\sigma}_1 = \ddot{\zeta}_1 + \alpha_1 \dot{\zeta}_1 + \beta_1 \frac{m_1}{n_1} \dot{\zeta}_1 \zeta_1^{\frac{m_1-n_1}{n_1}} \quad (IV.81)$$

De (IV.80) et (IV.81), la dérivée de la fonction de Lyapunov est:

$$\dot{V}_1 = \sigma_1 \left(\ddot{x}_{1d} - \frac{\cos x_7 \cos x_9}{m} U_1 - g + \alpha_1 \dot{\zeta}_1 + \beta_1 \frac{m_1}{n_1} \dot{\zeta}_1 \zeta_1^{\frac{m_1-n_1}{n_1}} \right) \quad (IV.82)$$

Afin de rendre \dot{V}_1 semi-défini négative, on peut mettre la loi de commande U_1 comme suit:

$$U_1 = \frac{m}{\cos x_7 \cos x_9} \left[\ddot{x}_{1d} + g + \alpha_1 \dot{\zeta}_1 + \beta_1 \frac{m_1}{n_1} \dot{\zeta}_1 \zeta_1^{\frac{m_1-n_1}{n_1}} + k_1 \text{signe}(\sigma_1) \right] \quad (IV.83)$$

Où, $\alpha_1, \beta_1, k_1, m_1$ et n_1 sont positives avec m_1 et n_1 sont des nombres naturels et $m_1 > n_1$.

B. La commande de la position X:

Pour la conception d'une loi de commande pour le sous-système de x (u_x). Nous obtenons au début, l'erreur de l'état x_3 et sa dérivée:

$$\begin{aligned}\zeta_2 &= x_{3d} - x_3 \\ \dot{\zeta}_2 &= \dot{x}_{3d} - x_4\end{aligned}\quad (IV.84)$$

On définit la surface de glissement du correcteur FTSMC (*Fast Terminal Sliding Mode Controller*):

$$\sigma_2 = \dot{\zeta}_2 + \alpha_2 \zeta_2 + \beta_2 \zeta_2^{\frac{m_2}{n_2}} \quad (\text{IV.85})$$

On pose la fonction candidate de Lyapunov et sa dérivée comme suit:

$$V_2 = \frac{1}{2} \sigma_2^2 \quad (\text{IV.86})$$

$$\dot{V}_2 = \sigma_2 \dot{\sigma}_2 \quad (\text{IV.87})$$

De (IV.85), la dérivée de la surface du glissement est:

$$\dot{\sigma}_2 = \ddot{\zeta}_2 + \alpha_2 \dot{\zeta}_2 + \beta_2 \frac{m_2}{n_2} \dot{\zeta}_2 \zeta_2^{\frac{m_2-n_2}{n_2}} \quad (\text{IV.88})$$

De (IV.87) et (IV.88), la dérivée de la fonction de Lyapunov est:

$$\dot{V}_2 = \sigma_2 \left(\ddot{x}_{3d} - \frac{1}{m} U_1 u_x + \alpha_2 \dot{\zeta}_2 + \beta_2 \frac{m_2}{n_2} \dot{\zeta}_2 \zeta_2^{\frac{m_2-n_2}{n_2}} \right) \quad (\text{IV.89})$$

Afin de rendre \dot{V}_2 semi-défini négative, on peut mettre la loi de commande u_x comme suit:

$$u_x = \frac{m}{U_1} \left[\ddot{x}_{3d} + \alpha_2 \dot{\zeta}_2 + \beta_2 \frac{m_2}{n_2} \dot{\zeta}_2 \zeta_2^{\frac{m_2-n_2}{n_2}} + k_2 \text{signe}(\sigma_2) \right] \quad (\text{IV.90})$$

Où, $\alpha_2, \beta_2, k_2, m_2$ et n_2 sont positives avec m_2 et n_2 sont des nombres naturels et $m_2 > n_2$.

C. La commande de la position Y:

Pour la conception d'une loi de commande pour le sous-système de y (u_y). Nous obtenons au début, l'erreur de l'état x_5 et sa dérivée:

$$\begin{aligned} \zeta_3 &= x_{5d} - x_5 \\ \dot{\zeta}_3 &= \dot{x}_{5d} - \dot{x}_6 \end{aligned} \quad (\text{IV.91})$$

On définit la surface de glissement du correcteur FTSMC (*Fast Terminal Sliding Mode Controller*):

$$\sigma_3 = \dot{\zeta}_3 + \alpha_3 \zeta_3 + \beta_3 \zeta_3^{\frac{m_3}{n_3}} \quad (\text{IV.92})$$

On pose la fonction candidate de Lyapunov et sa dérivée comme suit:

$$V_3 = \frac{1}{2} \sigma_3^2 \quad (\text{IV.93})$$

$$\dot{V}_3 = \sigma_3 \dot{\sigma}_3 \quad (IV.94)$$

De (IV.92), la dérivée de la surface du glissement est:

$$\dot{\sigma}_3 = \ddot{\zeta}_3 + \alpha_3 \dot{\zeta}_3 + \beta_3 \frac{m_3}{n_3} \dot{\zeta}_3 \zeta_3^{\frac{m_3-n_3}{n_3}} \quad (IV.95)$$

De (IV.94) et (IV.95), la dérivée de la fonction de Lyapunov est:

$$\dot{V}_3 = \sigma_3 \left(\ddot{x}_{5d} - \frac{1}{m} U_1 u_y + \alpha_3 \dot{\zeta}_3 + \beta_3 \frac{m_3}{n_3} \dot{\zeta}_3 \zeta_3^{\frac{m_3-n_3}{n_3}} \right) \quad (IV.96)$$

Afin de rendre \dot{V}_3 semi-défini négative, on peut mettre la loi de commande u_y comme suit:

$$u_y = \frac{m}{U_1} \left[\ddot{x}_{5d} + \alpha_3 \dot{\zeta}_3 + \beta_3 \frac{m_3}{n_3} \dot{\zeta}_3 \zeta_3^{\frac{m_3-n_3}{n_3}} + k_3 \text{signe}(\sigma_3) \right] \quad (IV.97)$$

Où, $\alpha_3, \beta_3, k_3, m_3$ et n_3 sont positives avec m_3 et n_3 sont des nombres naturels et $m_3 > n_3$.

D. La commande du roulis φ :

Pour la conception d'une loi de commande pour le sous-système de roulis (U_2). Nous obtenons au début, l'erreur de l'état x_7 et sa dérivée:

$$\begin{aligned} \zeta_4 &= x_{7d} - x_7 \\ \dot{\zeta}_4 &= \dot{x}_{7d} - x_8 \end{aligned} \quad (IV.98)$$

On définit la surface de glissement du correcteur FTSMC (*Fast Terminal Sliding Mode Controller*):

$$\sigma_4 = \zeta_4 + \alpha_4 \dot{\zeta}_4 + \beta_4 \zeta_4^{\frac{m_4}{n_4}} \quad (IV.99)$$

On pose la fonction candidate de Lyapunov et sa dérivée comme suit:

$$V_4 = \frac{1}{2} \sigma_4^2 \quad (IV.100)$$

$$\dot{V}_4 = \sigma_4 \dot{\sigma}_4 \quad (IV.101)$$

De (IV.99), la dérivée de la surface de glissement est:

$$\dot{\sigma}_4 = \ddot{\zeta}_4 + \alpha_4 \dot{\zeta}_4 + \beta_4 \frac{m_4}{n_4} \dot{\zeta}_4 \zeta_4^{\frac{m_4-n_4}{n_4}} \quad (IV.102)$$

De (IV.101) et (IV.102), la dérivée de la fonction de Lyapunov est:

$$\dot{V}_4 = \sigma_4 \left(\ddot{x}_{7d} - \left(\frac{I_y - I_z}{I_x} \right) x_{10} x_{12} + \frac{J_r \Omega_r}{I_x} x_{10} - \frac{l}{I_x} U_2 + \alpha_4 \dot{\zeta}_4 + \beta_4 \frac{m_4}{n_4} \dot{\zeta}_4 \zeta_4^{\frac{m_4 - n_4}{n_4}} \right) \quad (\text{IV.103})$$

Afin de rendre \dot{V}_4 semi-défini négative, on peut mettre la loi de commande U_2 comme suit:

$$U_2 = \frac{l}{I_x} \left[\ddot{x}_{7d} - \left(\frac{I_y - I_z}{I_x} \right) x_{10} x_{12} + \frac{J_r \Omega_r}{I_x} x_{10} + \alpha_4 \dot{\zeta}_4 + \beta_4 \frac{m_4}{n_4} \dot{\zeta}_4 \zeta_4^{\frac{m_4 - n_4}{n_4}} + k_4 \text{signe}(\sigma_4) \right] \quad (\text{IV.104})$$

Où, $\alpha_4, \beta_4, k_4, m_4$ et n_4 sont positives avec m_4 et n_4 sont des nombres naturels et $m_4 > n_4$.

E. La commande du tangage θ :

Pour la conception d'une loi de commande pour le sous-système de tangage (U_3). Nous obtenons au début, l'erreur de l'état x_9 et sa dérivée:

$$\begin{aligned} \zeta_5 &= x_{9d} - x_9 \\ \dot{\zeta}_5 &= \dot{x}_{9d} - \dot{x}_{10} \end{aligned} \quad (\text{IV.105})$$

On définit la surface de glissement du correcteur FTSMC (*Fast Terminal Sliding Mode Controller*):

$$\sigma_5 = \dot{\zeta}_5 + \alpha_5 \zeta_5 + \beta_5 \zeta_5^{\frac{m_5}{n_5}} \quad (\text{IV.106})$$

On pose la fonction candidate de Lyapunov et sa dérivée comme suit:

$$V_5 = \frac{1}{2} \sigma_5^2 \quad (\text{IV.107})$$

$$\dot{V}_5 = \sigma_5 \dot{\sigma}_5 \quad (\text{IV.108})$$

De (IV.106), la dérivée de la surface de glissement est:

$$\dot{\sigma}_5 = \ddot{\zeta}_5 + \alpha_5 \dot{\zeta}_5 + \beta_5 \frac{m_5}{n_5} \dot{\zeta}_5 \zeta_5^{\frac{m_5 - n_5}{n_5}} \quad (\text{IV.109})$$

De (IV.108) et (IV.109), la dérivée de la fonction de Lyapunov est:

$$\dot{V}_5 = \sigma_5 \left(\ddot{x}_{9d} - \left(\frac{I_z - I_x}{I_y} \right) x_8 x_{12} - \frac{J_r \Omega_r}{I_y} x_8 - \frac{l}{I_y} U_3 + \alpha_5 \dot{\zeta}_5 + \beta_5 \frac{m_5}{n_5} \dot{\zeta}_5 \zeta_5^{\frac{m_5 - n_5}{n_5}} \right) \quad (\text{IV.110})$$

Afin de rendre \dot{V}_5 semi-défini négative, on peut mettre la loi de commande U_3 comme suit:

$$U_3 = \frac{l}{I_y} \left[\ddot{x}_{9d} - \left(\frac{I_z - I_x}{I_y} \right) x_8 x_{12} - \frac{J_r \Omega_r}{I_y} x_8 + \alpha_5 \dot{\zeta}_5 + \beta_5 \frac{m_5}{n_5} \dot{\zeta}_5 \zeta_5^{\frac{m_5 - n_5}{n_5}} + k_5 \text{signe}(\sigma_5) \right] \quad (\text{IV.111})$$

Où, $\alpha_5, \beta_5, k_5, m_5$ et n_5 sont positives avec m_5 et n_5 sont des nombres naturels et $m_5 > n_5$.

F. La commande du lacet ψ :

Pour la conception d'une loi de commande pour le sous-système de lacet (U_4). Nous obtenons au début, l'erreur de l'état x_{11} et sa dérivée:

$$\begin{aligned}\zeta_6 &= x_{11d} - x_{11} \\ \dot{\zeta}_6 &= \dot{x}_{11d} - \dot{x}_{11}\end{aligned}\quad (IV.112)$$

On définit la surface de glissement du correcteur FTSMC (*Fast Terminal Sliding Mode Controller*):

$$\sigma_6 = \dot{\zeta}_6 + \alpha_6 \zeta_6 + \beta_6 \zeta_6^{\frac{m_6}{n_6}} \quad (IV.113)$$

On pose la fonction candidate de Lyapunov et sa dérivée comme suit:

$$V_6 = \frac{1}{2} \sigma_6^2 \quad (IV.114)$$

$$\dot{V}_6 = \sigma_6 \dot{\sigma}_6 \quad (IV.115)$$

De (IV.113), la dérivée de la surface de glissement est:

$$\dot{\sigma}_6 = \ddot{\zeta}_6 + \alpha_6 \dot{\zeta}_6 + \beta_6 \frac{m_6}{n_6} \dot{\zeta}_6 \zeta_6^{\frac{m_6-n_6}{n_6}} \quad (IV.116)$$

De (IV.115) et (IV.116), la dérivée de la fonction de Lyapunov est:

$$\dot{V}_6 = \sigma_6 \left(\ddot{x}_{11d} - \left(\frac{I_x - I_y}{I_z} \right) x_8 x_{10} - \frac{1}{I_z} U_4 + \alpha_6 \dot{\zeta}_6 + \beta_6 \frac{m_6}{n_6} \dot{\zeta}_6 \zeta_6^{\frac{m_6-n_6}{n_6}} \right) \quad (IV.117)$$

Afin de rendre \dot{V}_6 semi-défini négative, on peut mettre la loi de commande U_4 comme suit:

$$U_4 = I_z \left[\ddot{x}_{11d} - \left(\frac{I_x - I_y}{I_z} \right) x_8 x_{10} + \alpha_6 \dot{\zeta}_6 + \beta_6 \frac{m_6}{n_6} \dot{\zeta}_6 \zeta_6^{\frac{m_6-n_6}{n_6}} + k_6 \text{signe}(\sigma_6) \right] \quad (IV.118)$$

Où, $\alpha_6, \beta_6, k_6, m_6$ et n_6 sont positives avec m_6 et n_6 sont des nombres naturels et $m_6 > n_6$.

Les paramètres du correcteur Mode Glissant sont; $\alpha_{1..6} = 2, \beta_{1..6} = 5, k_{1..6} = 50, m_{1..6} = 6$ et $n_{1..6} = 2$.

IV.5 Résultats de Simulation:

Nous effectuons une simulation de suivi d'une trajectoire comme suit: le quadrotor commence de la position initiale $(x_0, y_0, z_0)=(0, 0, 0)$ et monte à $2m$ dans l'axe z . À l'instant $10s$, le quadrotor fait une trajectoire en spirale de vitesse $10cm/s$ d'altitude et tourne autour de lui-même pendant 15 seconds;

$$x_d = 4 \sin\left(\frac{2\pi}{15} t\right), y_d = 4 \cos\left(\frac{2\pi}{15} t\right), z_d = 0.1 t, \psi_d = \pi \sin\left(\frac{2\pi}{15} t\right).$$

Afin de tester la robustesse de ce correcteur, on suppose qu'il y a une masse additionnelle ($500 g$) et des incertitudes sur les valeurs des inerties à partir de l'instant $40s$. En plus, nous avons appliqué des perturbations comme des échelons sur les entrées; $10N$ à partir de l'instant $45s$ sur U_1 , $1N$ à partir de l'instant $50s$ sur U_2 , $1N$ à partir de l'instant $55 s$ sur U_3 et $0.1 N.m$ à partir de l'instant $60s$ sur U_4 dans la simulation du Backstepping. La Figure IV.2 présente les réponses des coordonnées de translation (x , y et z), la Figure IV.3 quant à elle présente les réponses des coordonnées de rotation (φ , θ et ψ), la Figure IV.4 présente la vue 3D du suivi de trajectoires, et finalement la Figure IV.5 présente les forces et couples appliqués (U_1 , U_2 , U_3 et U_4).

Après l'augmentation de la masse (à partir de l'instant $40s$), sur la Figure IV.2 et la Figure IV.4, nous remarquons qu'il y a des variations sur le comportement pour la simulation du correcteur par Mode Glissant, à l'opposé de la simulation du correcteur Backstepping. Nous remarquons aussi dans la Figure IV.5, une augmentation sur la force poussée (U_1) de $5.58N$ à $9.87N$ à cause de l'augmentation de la masse.

Concernant les perturbations des entrées sur la simulation du Backstepping, nous remarquons des variations dans les réponses à cause des perturbations qui sont rejetées rapidement. Ce qui n'est pas possible pour la simulation du Mode Glissant avec ces perturbations dures. Nous concluons que la technique la plus appropriée pour la robustesse face aux perturbations et aux incertitudes est la correction par Backstepping avec l'action intégrale.

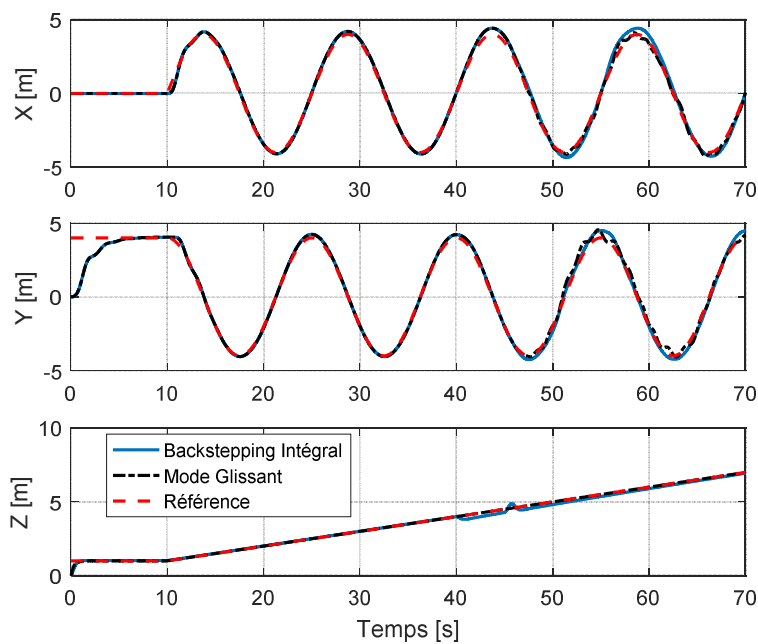


Figure IV.2 Les réponses des coordonnées de translation

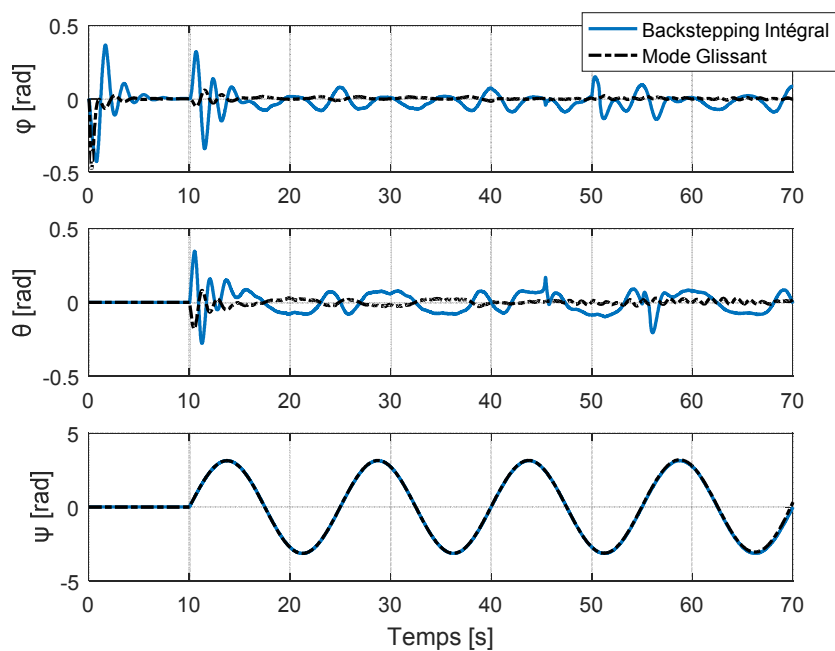


Figure IV.3 Les réponses des coordonnées de rotation

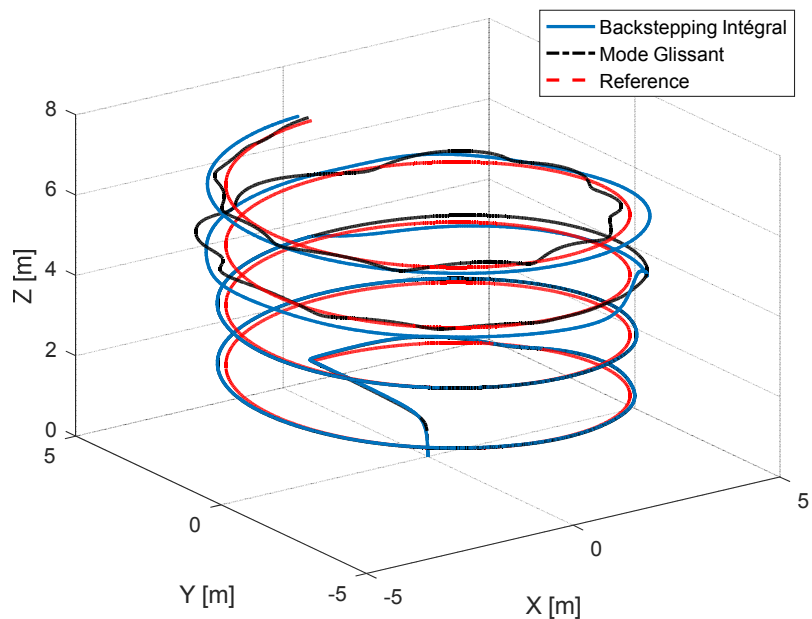


Figure IV.4 Le suivi de la trajectoire en 3D

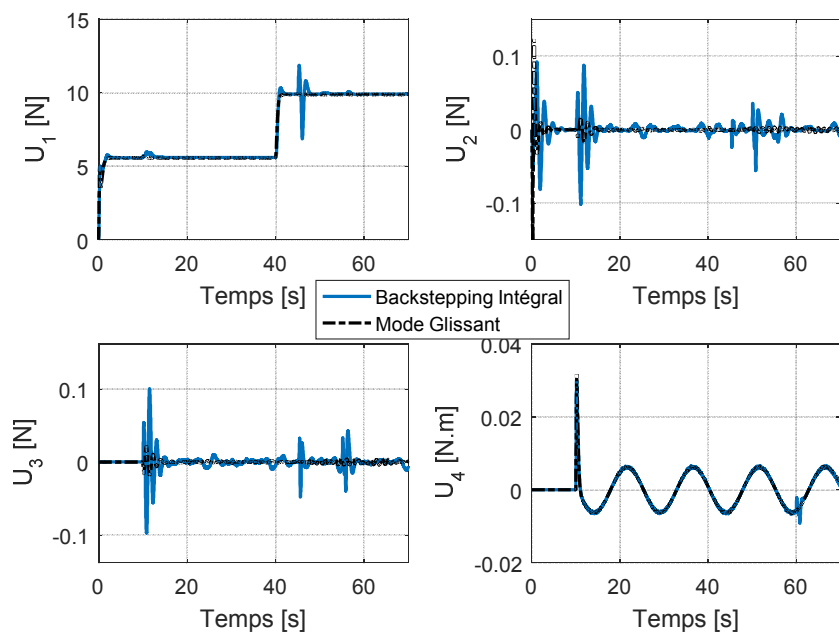


Figure IV.5 Les commandes appliquées

IV.6 Conclusion:

Dans ce chapitre, nous avons présenté la représentation d'état du quadrotor, qui est considéré comme étant un système complexe sous-actionné. Par la suite, la structure de commande en cascade a été conçue afin de découpler les degrés de liberté contrôlés (z , φ , θ et ψ) et les degrés de liberté sous-actionnés (x et y). La commande du quadrotor par les méthodes non-linéaires complexes a été réalisée à travers l'utilisation de deux techniques; Le Backstepping Intégral et le Mode Glissant de type *Fast Terminal Sliding Mode Control* (FTSMC). Aussi, afin de tester la robustesse de notre système, des perturbations ont été appliquées, et les résultats de simulation ont montré clairement que le Backstepping avec l'action d'intégral est robuste face à ces perturbations et incertitudes. L'inconvénient de ces méthodes est que dans le concevoir des correcteurs, elles ne considèrent pas les perturbations et les incertitudes réellement posés. Pour cela, la technique de commande robuste d'espace de Hardy (H_∞) a été créée pendant trois décades, et ici, la robustesse est garantie avec telle norme donnée. Par la suite, nous allons étudier et développer un correcteur de type H_∞ à la théorie des systèmes linéaires variés dans le temps (LPV).

Chapitre V:

Commande Robuste du Quadrotor Basée sur la Technique LPV

V.1 Introduction:

Un système linéaire à paramètres variés (*Linear Parameter Vaying; LPV*) est un système à une structure linéaire avec des matrices non-stationnaires. Les systèmes LPV ont attiré l'attention des chercheurs pendant ces deux dernières décades, spécialement du domaine aérospatial et de la micro-électronique, ce qui est dû à leurs caractéristiques telles que; garantir la stabilité avec robustesse sans réduire le comportement de système non-linéaire. Dans ce chapitre nous allons exploiter les techniques LPV robustes pour les appliquer sur des systèmes non-linéaires sous-actionnés. Ce qui nécessite la représentation d'un système non-linéaire comme un système LPV où les non-linéarités, les états et/ou les entrées sont considérés comme des paramètres variés. Au début, nous allons exposer les différentes représentations des systèmes LPV, suivi des différentes structures de commande. Une discussion détaillée sera présentée sur la représentation utilisée dans ce travail et qui est la représentation rationnelle, pour exposer à la fin de ce chapitre, une application sur le suivi de la trajectoire du quadrotor, ainsi que la stabilisation du pendule inversé sur chariot.

V.2 Représentation et Commande des Systèmes LPV:

V.2.1 Représentation des Systèmes LPV:

L'objectif de cette section est de montrer les différentes structures utilisées pour présenter les systèmes LPV.

V.2.1.1 Systèmes LPV Génériques:

Cette formulation des systèmes LPV est une formulation directe et la plus naturelle puisque le système LPV est directement considéré en sa forme primitive par (Shamma, 1988):

$$\dot{x}(t) = A(\rho)x(t) \tag{V.1}$$

Où $x(t) \in \mathbb{R}^n$ est le vecteur des états du système et ρ est le vecteur des paramètres variés dans le temps.

Ni transformation ni prétraitement sont appliqués à la structure du système. La seule hypothèse sous-jacente est que la matrice $A(\rho)$ soit bornée. L'analyse de stabilité ou la synthèse de commande sont directement effectuées. Cette formulation convient mieux aux systèmes dépendants de paramètres polynomiaux mais peut être utilisée avec tout type de systèmes LPV. Nous pouvons de recréer la matrice $A(\rho)$ de (V.1) comme suit:

$$A(\rho) = A_0 + \sum A_i \rho^{\alpha_i} \tag{V.2}$$

Avec $\alpha_i = [\alpha_i^1 \ \alpha_i^2 \ \dots \ \alpha_i^N]$ et $\rho^{\alpha_i} = [\rho^{\alpha_i^1} \ \rho^{\alpha_i^2} \ \dots \ \rho^{\alpha_i^N}]$. Cependant, des formulations plus appropriées pour de tels systèmes vont être présentées dans les sections suivantes.

V.2.1.2 Systèmes LPV Polytopiques:

La structure polytopique s'offre un moyen pratique pour représenter et analyser les systèmes LPV et incertains; comme c'est le cas des exemples présentés dans (Apkarian et al., 1995; Geromel et Colaneri, 2006; Chesi et al., 2009). Contrairement aux systèmes LPV génériques pour lesquels nous ne supposons aucune dépendance particulière vis-à-vis des paramètres, les systèmes polytopiques sont explicitement représentés comme une combinaison convexe de systèmes LTI. Cette propriété structurelle peut être exploitée pour obtenir des résultats de stabilité facilement vérifiables à l'aide des techniques d'optimisation convexes.

On considère ce système:

$$\begin{cases} \dot{x}(t) = A(\rho(t))x(t) + B_1(\rho(t))w(t) + B_2(\rho(t))u(t) \\ z(t) = C_1(\rho(t))x(t) + D_{11}(\rho(t))w(t) + D_{12}(\rho(t))u(t) \\ y(t) = C_2(\rho(t))x(t) + D_{21}(\rho(t))w(t) + D_{22}(\rho(t))u(t) \end{cases} \quad (\text{V.3})$$

Où $x(t) \in \mathbb{R}^n$ est le vecteur des états, $w(t) \in \mathbb{R}^q$ est le vecteur des perturbations, $u(t) \in \mathbb{R}^m$ sont les entrées de commande, $z(t) \in \mathbb{R}^p$ sont les sorties commandées, et $y(t) \in \mathbb{R}^{n_y}$ sont les sorties mesurées.

On suppose que le vecteur des paramètres variés dans le temps $\rho(t) \in \mathbb{R}^N$ appartient à l'unité simplexe \mathcal{U} où:

$$\mathcal{U} = \{\lambda \in \mathbb{R}^N : \sum_{i=1}^N \lambda_i = 1, \lambda_i \geq 0, i = 1, \dots, N\} \quad (\text{V.4})$$

En particulier, aucune hypothèse n'est faite sur les paramètres variés dans le temps.

Chaque matrice du système appartient à un polytope. On note $M(\rho(t))$ pour $(A(\rho(t)), B_1(\rho(t)), B_2(\rho(t)), C_1(\rho(t)), C_2(\rho(t)), D_{11}(\rho(t)), D_{12}(\rho(t)), D_{21}(\rho(t)), D_{22}(\rho(t)))$:

$$\mathcal{P} = \{(M(\rho(t)) : \exists \rho \in \mathcal{U}, M(\rho(t)) = \sum_{i=1}^N \rho_i M_i\} \quad (\text{V.5})$$

Tout système LPV (V.3) peut être représenté, jusqu'à un certain degré de précision, par un système LPV polytopique (exemple V.1). Les systèmes qui dépendent linéairement de paramètres prenant des valeurs dans un polyèdre convexe, tel qu'un boîtier, peuvent être représentés exactement comme des systèmes polytopiques (exemple V.2).

Exemple V.1: soit le système LPV

$$\dot{x}(t) = [\rho_1(t)A_1 + \rho_2(t)A_2]x(t) \quad (\text{V.6})$$

Avec $\rho(t) \in [-1, 1] \times [-1, 1]$ admet une représentation polytopique équivalente puisque les paramètres prennent des valeurs dans un polytope convexe. L'ensemble correspondant des valeurs est donné par:

$$E = \{(\rho_1, \rho_2): \rho_1 \in [-1, 1], \rho_2 \in [-1, 1]\} \quad (\text{V.7})$$

Par conséquent, nous pouvons considérer l'ensemble couvrant les sommets de la Figure V.1

$$E \subset E_g := \text{co} \left\{ \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\} \quad (\text{V.8})$$

Le boîtier $[-1, 1]^2$ peut être paramétré en termes de variables dans le simplexe à 4 unités, comme indiqué ci-dessous:

$$\dot{x}(t) = [f_1(\lambda(t))A_1 + f_2(\lambda(t))A_2]x(t) \quad (\text{V.9})$$

Où $\lambda(t) \in \mathbb{R}^4$ et:

$$\begin{aligned} f_1(\lambda(t)) &= (\lambda_2 + \lambda_4) - (\lambda_1 + \lambda_3) \\ f_2(\lambda(t)) &= (\lambda_3 + \lambda_4) - (\lambda_1 + \lambda_2) \end{aligned} \quad (\text{V.10})$$

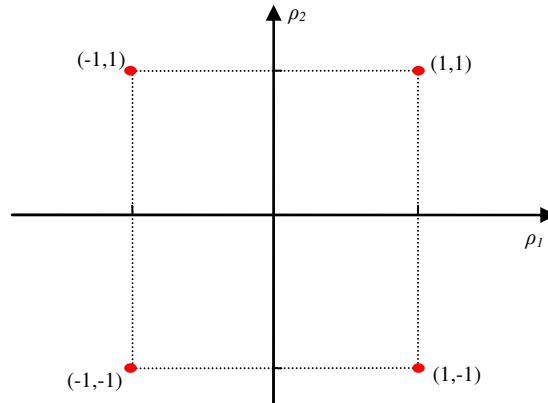


Figure V.1 Plan du polytope de l'exemple V.1

Exemple V.2: soit le système LPV

$$\dot{x}(t) = [\rho(t)A_1 + \rho^2(t)A_2]x(t) \quad (\text{V.11})$$

Avec $\rho(t) \in [-1, 1]$, nous considérons les paramètres ρ et ρ^2 comme deux paramètres distincts. L'ensemble couvrant est donné par la suite (les sommets de la Figure V.2):

$$E \subset E_g := \text{co} \left\{ \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\} \quad (\text{V.12})$$

Le boîtier $[-1, 1] \times [0, 1]$ peut être paramétré en termes de variables dans le simplexe à 4 unités, comme indiqué ci-dessous:

$$\dot{x}(t) = [f_1(\lambda(t))A_1 + f_2(\lambda(t))A_2]x(t) \quad (\text{V.13})$$

Où $\lambda(t) \in \mathbb{R}^4$ et:

$$\begin{aligned} f_1(\lambda(t)) &= (\lambda_2 + \lambda_4) - (\lambda_1 + \lambda_3) \\ f_2(\lambda(t)) &= \lambda_3 + \lambda_4 \end{aligned} \quad (\text{V.14})$$

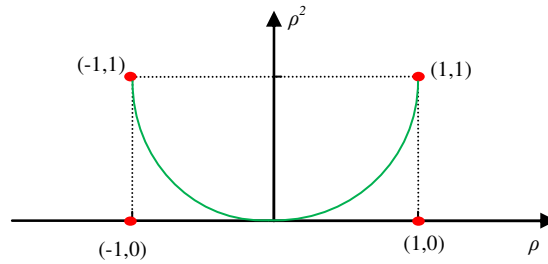


Figure V.2 Plan du polytope de l'exemple V.2

Depuis $E \subset E_g$, la représentation polytopique (V.13) et (V.14) n'est pas équivalente au système exacte (V.11). Tous les points en E_g (boîtier) situés en dehors de la parabole E (ligne verte) sont très nombreux, ça peut affecter sur la précision de la représentation polytopique (Briat, 2015). Le système peut être stable sur E , mais il ne l'est pas sur E_g .

V.2.1.3 Systèmes LPV de Forme LFT:

Les systèmes LPV sous forme LFT sont des systèmes LPV exprimés sous forme d'interconnexions de deux sous-systèmes, comme indiqué sur la Figure V.3. L'acronyme LFT signifie *Linear Fractional Transformation*. La procédure de reformulation est utilisée pour exprimer les systèmes LPV et incertains sous forme d'interconnexions à deux blocs. Cette formulation des systèmes LPV a suscité l'intérêt de (Packard, 1994) et l'approche a donné lieu à de nombreux travaux comme présenté dans (Apkarian et Gahinet, 1995; Apkarian et Adams, 1998; Scherer, 2001).

L'idée derrière la forme LFT est de recréer un système complexe comme une interconnexion de partie simple et une partie compliquée. La partie simple doit avoir des propriétés simples; la linéarité et l'invariance dans le temps. La partie compliquée contient les paramètres variantes, les non-linéarités, les retards, etc...

Nous pouvons représenter un système LPV sous la forme LFT comme ci-dessous:

$$\begin{aligned}
 \dot{x} &= A x(t) + B_\rho w_\rho(t) + B_1 w(t) + B_2 u(t) \\
 z_\rho &= C_\rho x(t) + D_{\rho\rho} w_\rho(t) + D_{\rho 1} w(t) + D_{\rho 2} u(t) \\
 z &= C_1 x(t) + D_{1\rho} w_\rho(t) + D_{11} w(t) + D_{12} u(t) \\
 y &= C_2 x(t) + D_{2\rho} w_\rho(t) + D_{21} w(t) + D_{22} u(t) \\
 w_\rho &= \Theta z_\rho
 \end{aligned}
 \tag{V.15}$$

Où $z_\rho(t)$ et $w_\rho(t) \in \mathbb{R}^r$ sont les entrées et sorties du bloque des paramètres variés dans le temps

$\Theta = \text{diag}(\rho_1 I_{r1}, \rho_2 I_{r2} \dots \rho_k I_{rk})$.

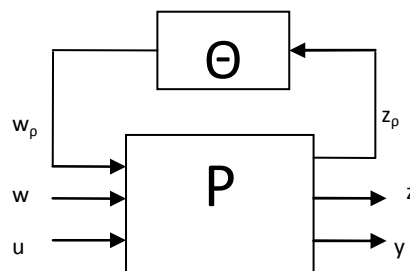


Figure V.3 Représentation LFT d'un système LPV

V.2.2 Commande des Systèmes LPV:

Les outils d'analyse des systèmes LPV sont directement hérités d'une analyse robuste et d'une théorie de commande robuste et l'intégralité de la théorie LPV n'est révélée que lorsque la conception du correcteur est l'objectif principal. Le principe de la commande robuste dans le cadre LPV est que les paramètres sont supposés connus ou mesurables. Les paramètres peuvent être utilisés dans la loi de commande par séquençement de gains; *gain scheduling*, (Figure V.4).

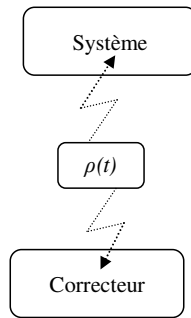


Figure V.4 Structure de commande par séquençement de gains

La difficulté majeure était l'absence de théorie générale permettant d'analyser la stabilité des systèmes LPV et de concevoir des lois de commande par séquençement de gains LPV (Briat, 2015).

V.2.2.1 Retour d'État:

La loi de commande la plus simple qui est conçue est le retour d'état à séquençement de gains qui s'écrit:

$$u(t) = K(\rho(t))x(t) \quad (\text{V.16})$$

Il s'agit d'une extension immédiate du retour d'état LTI au réglage LPV; la matrice de commande $K(\rho(t))$ est désormais fonction des paramètres. Ce correcteur est le plus facile à concevoir, mais il nécessite la connaissance de l'état complet du système à implémenter.

V.2.2.2 Retour de Sortie:

Une loi de commande simple à séquençement de gains est la loi de commande de retour de sortie qui prend la forme:

$$u(t) = K(\rho(t))y(t) \quad (\text{V.17})$$

Cette classe des correcteurs est très facile à implémenter car l'entrée de commande est calculée directement à partir de la sortie mesurée. La difficulté principale est celle d'obtenir des conditions favorables à la conception de telles lois de commande (Syrmos et al., 1997; Yuebang et al., 2012). Pourtant, il existe certaines méthodes pour les concevoir dans le cadre robuste dont elles peuvent être étendus au cas LPV (Prempain et Postlethwaite, 2005; Henrion et Lasserre, 2006).

V.2.2.3 Retour Dynamique:

Une classe de correcteurs très importante est la classe des correcteurs à retour dynamique:

$$\begin{aligned} \dot{x}_K(t) &= A_K(\rho(t))x(t) + B_K(\rho(t))y(t) \\ u(t) &= C_K(\rho(t))x_K(t) + D_K(\rho(t))y(t) \end{aligned} \quad (\text{V.18})$$

Où x_K est le vecteur des états du correcteur, A_K , B_K , C_K et D_K sont les matrices dynamiques du correcteur. Cette classe des correcteurs a été étudiée dans (Packard, 1994; Apkarian et Gahinet, 1995; Scherer, 2001; Scherer et Köse, 2012). L'avantage de cette classe est qu'elle permet de générer la commande basée sur les observateurs.

Nous allons nous concentrer ici sur le retour dynamique avec la représentation LFT. Sur la base de la structure LFT des systèmes LPV représentée dans la section V.2.1.3, le correcteur LPV-LFT à retour dynamique (Figure V.5) est donné par:

$$\begin{aligned} \dot{x}_K &= A_K x_K + B_{K1} y + B_{K\rho} w_{K\rho} \\ u &= C_{K1} x_K + D_{K11} y + D_{K1\rho} w_{K\rho} \\ z_{K\rho} &= C_{K\rho} x_K + D_{K\rho 1} y + D_{K\rho\rho} w_{K\rho} \\ w_{K\rho} &= \Theta z_{K\rho} \end{aligned} \quad (\text{V.19})$$

Où $x_K(t) \in \mathbb{R}^k$ est le vecteur des états du correcteur, $u(t) \in \mathbb{R}^m$ sont les entrées de commande, $y(t) \in \mathbb{R}^{n_y}$ sont les sorties mesurées, $z_\rho(t)$ et $w_\rho(t) \in \mathbb{R}^r$ sont les entrées et sorties du bloc des paramètres variés dans le temps $\Theta = \text{diag}(\rho_1 I_{r1}, \rho_2 I_{r2} \dots \rho_k I_{rk})$. Ce bloc est le même que celui du système à commander.

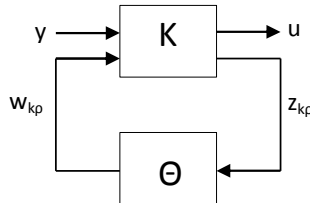


Figure V.5 Représentation d'un correcteur LPV sous forme LFT

V.3 Stabilité des Systèmes LPV:

V.3.1 Stabilité des Systèmes LPV Génériques et Polytopiques:

Nous allons considérer dans cette section un système LPV générique en boucle fermée:

$$\dot{x} = A(\rho)x(t) \quad (\text{V.20})$$

V.3.1.1 Stabilité des Systèmes LPV Génériques:

Nous considérons la fonction quadratique de Lyapunov:

$$V(x, \rho) = x^T P(\rho) x, \quad P(\rho) > 0 \quad (\text{V.21})$$

$V(x, \rho)$ est la fonction de Lyapunov pour le système (V.20), qui est appelée la fonction de Lyapunov dépendante des paramètres.

En accord avec le théorème de Lyapunov, la dérivée (V.22) doit être semi-définie négative:

$$\dot{V}(x, \rho) = x^T \left(A(\rho)^T P(\rho) + P(\rho) A(\rho) + \sum_{i=1}^N \dot{\rho}_i \frac{\partial P(\rho)}{\partial \rho_i} \right) x \quad (\text{V.22})$$

Le système LPV (V.20) est stable et robuste s'il existe une matrice différentielle $P(\rho)$ telle que la condition:

$$A(\rho)^T P(\rho) + P(\rho) A(\rho) + \sum_{i=1}^N \dot{\rho}_i \frac{\partial P(\rho)}{\partial \rho_i} < 0 \quad (\text{V.23})$$

Notant que la dépendance de l'inégalité ci-dessus est affine pour ρ_i et que ρ et $\dot{\rho}$ sont indépendantes.

V.3.1.2 Stabilité des Systèmes LPV Polytopiques:

En considérant la boucle fermée d'un système LPV sous forme polytopique, tel que:

$$\begin{aligned} \dot{x} &= \sum_{i=1}^N \rho_i A_i x(t) + \sum_{i=1}^N \rho_i B_i w(t) \\ z &= \sum_{i=1}^N \rho_i C_i x(t) + \sum_{i=1}^N \rho_i D_i w(t) \end{aligned} \quad (\text{V.24})$$

Où $x(t) \in \mathbb{R}^n$ est le vecteur des états, $w(t) \in \mathbb{R}^q$ est le vecteur des perturbations et $z(t) \in \mathbb{R}^p$ sont les sorties commandées.

La stabilité robuste de ce système est de minimiser le gain L_2 (norme) entre les sorties commandées et les perturbations:

$$\frac{\|z\|_2}{\|w\|_2} < \gamma \quad (\text{V.25})$$

Puisque un système LPV polytopique est un groupe de systèmes LTI, alors afin d'assurer la stabilité d'un système LPV polytopique, il faut assurer la stabilité de chaque $i^{\text{ème}}$ système LTI (Gahinet et Apkarian, 1994). Le système (V.24) est stable avec robustesse et assure (V.25), s'il y a une matrice X qui assure l'inégalité suivante:

$$\begin{pmatrix} A_i^T X + X A_i & X B_i & C_i^T \\ B_i^T X & -\gamma I_q & D_i^T \\ C_i & D_i & -\gamma I_p \end{pmatrix} < 0 \quad (\text{V.26})$$

V.3.2 Stabilité des Systèmes LPV de Forme LFT:

Dans notre cas d'étude, nous allons nous baser sur la représentation LFT des systèmes LPV et sur une commande par le théorème du *petit gain scalaire*.

La stabilité des systèmes LFT est toujours un sujet de recherche actif (Feyel, 2015; Liu et al., 2017; Acosta et Bottura, 2017; Behzad et al., 2018). En effet, les systèmes LFT constituent un moyen unifié de modéliser les systèmes LPV avec tout type de dépendance de paramètres: affine, polynomiale et rationnelle. En réécrivant les systèmes LPV sous la forme LFT, le système initial est divisé en deux sous-systèmes interconnectés; un système invariant (système LTI) et un autre est variable dans le temps (bloc des paramètres variées). La stabilité des systèmes LPV est ensuite déterminée à l'aide des résultats obtenus sur les systèmes interconnectés en stabilité.

La technique LPV sous forme LFT est utilisée pour représenter le système non-linéaire. Par la suite, la représentation LPV-LFT est donnée avec pris en compte de la conception de performance H_∞ , les conditions d'existence d'un correcteur à séquençement de gains H_∞ robuste s'effectue par la résolution d'un groupe de contraintes LMIs, comme présenté dans (Apkarian et Gahinet, 1995; Hjartarson et al., 2015), et puis la synthèse du correcteur qui est un système LPV sous forme LFT aussi.

V.3.2.1 Représentation LFT d'un Système LPV:

Un système est appelé LPV-LFT si nous pouvons reformuler (V.20) comme ci-dessous:

$$\begin{aligned}
 \dot{x} &= A x + B_\rho w_\rho + B_1 w + B_2 u \\
 z_\rho &= C_\rho x + D_{\rho\rho} w_\rho + D_{\rho 1} w + D_{\rho 2} u \\
 z &= C_1 x + D_{1\rho} w_\rho + D_{11} w + D_{12} u \\
 y &= C_2 x + D_{2\rho} w_\rho + D_{21} w + D_{22} u \\
 w_\rho &= \Theta z_\rho
 \end{aligned} \tag{V.27}$$

Où $x(t) \in \mathbb{R}^n$ est le vecteur des états, $w(t) \in \mathbb{R}^q$ est le vecteur des perturbations, $u(t) \in \mathbb{R}^m$ sont les entrées de commande, $z(t) \in \mathbb{R}^p$ sont les sorties commandées, $y(t) \in \mathbb{R}^{n_y}$ sont les sorties mesurées, $z_\rho(t)$ et $w_\rho(t) \in \mathbb{R}^r$ sont les entrées et sorties du bloc des paramètres variés dans le temps $\Theta = \text{diag}(\rho_1 I_{r_1}, \rho_2 I_{r_2} \dots \rho_k I_{r_k})$.

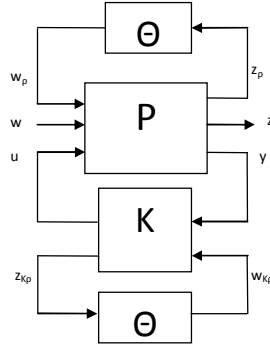


Figure V.6 La boucle fermée système-correcteur d'un système LPV sous forme LFT

La structure LPV-LFT de la boucle système-correcteur est présentée dans la Figure V.6. On note que dans cette approche de commande, le correcteur à séquençement de gains a la même structure, la même complexité du système, et aussi la même récurrence de chaque paramètre varié dans le temps. L'expression suivante est la représentation d'état du correcteur robuste à séquençement de gains.

$$\begin{aligned}
 \dot{x}_K &= A_K x_K + B_{K1} y + B_{K\rho} w_{K\rho} \\
 u &= C_{K1} x_K + D_{K11} y + D_{K1\rho} w_{K\rho} \\
 z_{K\rho} &= C_{K\rho} x_K + D_{K\rho1} y + D_{K\rho\rho} w_{K\rho} \\
 w_{K\rho} &= \Theta z_{K\rho}
 \end{aligned} \tag{V.28}$$

Où $x_K(t) \in \mathbb{R}^k$ est le vecteur des états du correcteur, $u(t) \in \mathbb{R}^m$ sont les entrées de commande, $y(t) \in \mathbb{R}^{n_y}$ sont les sorties mesurées, $z_\rho(t)$ et $w_\rho(t) \in \mathbb{R}^r$ sont les entrées et sorties du bloc des paramètres variés dans le temps $\Theta = \text{diag}(\rho_1 I_{r1}, \rho_2 I_{r2} \dots \rho_k I_{rk})$.

La synthèse du correcteur dans ce chapitre est basé sur les travaux de (Apkarian et Gahinet, 1995; Hjartarson et al., 2015), où il y a deux étapes; premièrement, un groupe des contraintes LMIs sont résolus pour optimiser la performance H_∞ de la boucle fermée et générer les matrices associés R , S , L_3 et J_3 . Deuxièmement, le correcteur (V.28) est calculé à partir de ces matrices.

V.3.2.2 Analyse de Stabilité Sous LMIs Selon le Théorème du Petit Gain Scalaire:

Théorème: S'il y a une matrice scalaire L où $F_l(P_a, K)$ est le système en boucle fermée qui vérifie:

$$\left\| \begin{pmatrix} L^{1/2} & 0 \\ 0 & I \end{pmatrix} \cdot F_l(P_a, K) \cdot \begin{pmatrix} L^{-1/2} & 0 \\ 0 & I \end{pmatrix} \right\|_\infty \leq \gamma \tag{V.29}$$

Avec: $\|\Theta\|_\infty \leq \frac{1}{\gamma}$

Alors, il y a un correcteur à séquençement de gains robuste $K(\rho) = \text{LFT}(K, \Theta)$.

$F_l(P_a, K)$: la transformation (LFT) inférieure entre le processus et le correcteur. Elle présente le système à boucle fermée.

P_a : le processus augmenté.

γ : le critère de performance.

Essentiellement, les formulations LMIs sont générées pour les systèmes LTI (Gahinet et Apkarian, 1994) et sont étendus aux systèmes LPV (Apkarian et Gahinet, 1995). On remplace ici le problème LPV original par un problème H_∞ . On pose N_R et N_S des matrices arbitraires basées sur les dimensions de $(B_2^T, D_{\rho 2}^T, D_{12}^T, 0)$ et $(C_2, D_{2\rho}, D_{21}, 0)$ respectivement. Il y a un correcteur LPV à séquencement de gains s'il y a des matrices R, S, L_3 et J_3 qui vérifient:

$$N_R^T \begin{pmatrix} AR + RA^T & RC_1^T & B_1 \\ C_1 R & -\gamma \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} & D_{11} \\ B_1^T & D_{11}^T & -\gamma \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} \end{pmatrix} N_R < 0 \quad (\text{V.30})$$

$$N_S^T \begin{pmatrix} A^T S + SA & SB_1 & C_1^T \\ B_1^T S & -\gamma \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} & D_{11}^T \\ C_1 & D_{11} & -\gamma \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} \end{pmatrix} N_S < 0 \quad (\text{V.31})$$

$$\begin{pmatrix} R & I \\ I & S \end{pmatrix} \geq 0 \quad (\text{V.32})$$

$$\begin{pmatrix} L_3 & I \\ I & J_3 \end{pmatrix} \geq 0 \quad (\text{V.33})$$

$$LJ = I, L \leftarrow \begin{pmatrix} L_1 & L_2 \\ L_2^T & L_3 \end{pmatrix}, J \leftarrow \begin{pmatrix} J_1 & J_2 \\ J_2^T & J_3 \end{pmatrix} \quad (\text{V.34})$$

B_1, C_1 et D_{11} présentées dans (V.30) et (V.31) sont augmentées par les entrées et les sorties du bloc des paramètres Θ :

$$B_1 \leftarrow (B_\rho \quad B_1), C_1 \leftarrow \begin{pmatrix} C_\rho \\ C_1 \end{pmatrix}, D_{11} \leftarrow \begin{pmatrix} D_{\rho\rho} & D_{\rho 1} \\ D_{1\rho} & D_{11} \end{pmatrix}$$

Le théorème est démontré dans (Gahinet et Apkarian, 1994) pour les systèmes LTI et adopté pour les systèmes LPV (Apkarian et Gahinet, 1995).

V.3.2.3 Synthèse du Correcteur LPV-LFT:

On suppose que la représentation d'état du correcteur (V.28) est présentée comme un système LTI nominal comme ci-dessous:

$$\Omega = \begin{pmatrix} A_K & B_{K1} & B_{K\rho} \\ C_{K1} & D_{K11} & D_{K1\rho} \\ C_{K\rho} & D_{K\rho 1} & D_{K\rho\rho} \end{pmatrix} \in \mathbb{R}^{(k+m+r \times k+ny+r)} \quad (\text{V.35})$$

On prend en considération la notation suivante:

$$\begin{aligned} A_0 &= \begin{pmatrix} A & 0 \\ 0 & 0_{k \times k} \end{pmatrix}, B_0 = \begin{pmatrix} 0 & B_\rho & B_1 \\ 0_{k \times r} & 0 & 0 \end{pmatrix}, \\ C_0 &= \begin{pmatrix} 0 & 0_{r \times k} \\ C_\rho & 0 \\ C_1 & 0 \end{pmatrix}, \mathcal{B} = \begin{pmatrix} 0 & B_2 & 0 \\ I_k & 0 & 0_{k \times r} \end{pmatrix}, \\ \mathcal{C} &= \begin{pmatrix} 0 & I_k \\ C_2 & 0 \\ 0 & 0_{r \times k} \end{pmatrix}, \mathcal{D}_{11} = \begin{pmatrix} 0_{r \times r} & 0 & 0 \\ 0 & D_{\rho\rho} & D_{\rho 1} \\ 0 & D_{1\rho} & D_{11} \end{pmatrix}, \\ \mathcal{D}_{12} &= \begin{pmatrix} 0_{r \times k} & 0 & I_r \\ 0 & D_{\rho 2} & 0 \\ 0 & D_{12} & 0 \end{pmatrix}, \mathcal{D}_{21} = \begin{pmatrix} 0_{k \times r} & 0 & 0 \\ 0 & D_{2\rho} & D_{21} \\ I_r & 0 & 0 \end{pmatrix} \end{aligned} \quad (\text{V.36})$$

On suppose les solutions R , S , L_3 et J_3 données des LMIs dans (V.30)-(V.33), le calcul de Ω est comme suit (Algorithme 5.1, (Apkarian et Gahinet, 1995)):

Algorithme 5.1: Résoudre le correcteur LPV-LFT

1: Calculer $M, N \in \mathbb{R}^{(n \times k)}$ où $MN^T = I - RS$ (V.37)

2: Calculer X_{cl} la solution unique de: $X_{cl} \begin{pmatrix} I & R \\ 0 & M^T \end{pmatrix} = \begin{pmatrix} S & I \\ N^T & 0 \end{pmatrix}$ (V.38)

3: Calculer L_1 et L_2 où: $L = \begin{pmatrix} L_1 & L_2 \\ L_2^T & L_3 \end{pmatrix} > 0$ (V.39)

4: $\mathcal{L} \leftarrow \begin{pmatrix} L & 0 \\ 0 & I_p \end{pmatrix}, J \leftarrow \mathcal{L}^{-1}$ (V.40)

5: $\Psi \leftarrow \begin{pmatrix} A_0^T X_{cl} + X_{cl} A_0 & X_{cl} B_0 & C_0^T \\ B_0^T X_{cl} & -\gamma \mathcal{L} & \mathcal{D}_{11}^T \\ C_0 & \mathcal{D}_{11} & -\gamma J \end{pmatrix}$ (V.41)

6: $P \leftarrow (\mathcal{B}^T, 0, \mathcal{D}_{12}^T), Q \leftarrow (\mathcal{C}, \mathcal{D}_{21}, 0)$ (V.42)

7: Résoudre Ω de LMI suivante: $\Psi + \begin{pmatrix} X_{cl} & 0 \\ 0 & I \end{pmatrix} P^T \Omega Q + Q^T \Omega^T P \begin{pmatrix} X_{cl} & 0 \\ 0 & I \end{pmatrix} < 0$ (V.43)

V.4 Stabilisation LPV Robuste du Quadrotor:

Le modèle mathématique du quadrotor a été présenté dans le Chapitre III, comme suit:

$$\begin{cases} \ddot{x} = \frac{1}{m} (c\varphi s\theta c\psi + s\varphi s\psi)U_z \\ \ddot{y} = \frac{1}{m} (c\varphi s\theta s\psi - s\varphi c\psi)U_z \\ \ddot{z} = \frac{1}{m} (c\varphi c\theta)U_z - g \\ \ddot{\varphi} = \frac{(I_y - I_z)}{I_x} \dot{\theta}\dot{\psi} - \frac{J_r \Omega_r}{I_x} \dot{\theta} + \frac{l}{I_x} U_\varphi \\ \ddot{\theta} = \frac{(I_z - I_x)}{I_y} \dot{\varphi}\dot{\psi} + \frac{J_r \Omega_r}{I_y} \dot{\varphi} + \frac{l}{I_y} U_\theta \\ \ddot{\psi} = \frac{(I_x - I_y)}{I_z} \dot{\varphi}\dot{\theta} + \frac{1}{I_z} U_\psi \end{cases} \quad (\text{V.44})$$

Comme il a été déjà mentionné précédemment, le quadrotor est un système sous-actionné, ainsi, quand le sous-système de rotation a été stabilisé avec robustesse, le suivi de trajectoire (translation) sera très facile à contrôler avec une structure en cascade.

Dans cette section, la commande LPV-LFT est utilisé pour stabiliser le sous système de rotation (attitude) avec robustesse, on va représenter le sous-système de rotation où son modèle a trois degrés de liberté; roulis, tangage, et lacet et six états; les trois angles $[\varphi \theta \psi]$ et leurs dérivés $[\dot{\varphi} \dot{\theta} \dot{\psi}]$, et les trois entrées $[U_\varphi U_\theta U_\psi]$:

$$\begin{cases} \ddot{\varphi} = \frac{I_y - I_z}{I_x} \dot{\theta}\dot{\psi} - \frac{J_r \Omega_r}{I_x} \dot{\theta} + \frac{l}{I_x} U_\varphi \\ \ddot{\theta} = \frac{I_z - I_x}{I_y} \dot{\varphi}\dot{\psi} + \frac{J_r \Omega_r}{I_y} \dot{\varphi} + \frac{l}{I_y} U_\theta \\ \ddot{\psi} = \frac{I_x - I_y}{I_z} \dot{\varphi}\dot{\theta} + \frac{1}{I_z} U_\psi \end{cases} \quad (\text{V.45})$$

$$\text{Avec:} \quad \Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4 \quad (\text{V.46})$$

Et:

$$\begin{cases} U_\varphi = b(\omega_4^2 - \omega_2^2) \\ U_\theta = b(\omega_3^2 - \omega_1^2) \\ U_\psi = d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{cases} \quad (\text{V.47})$$

Les paramètres du quadrotor ont été présentés dans le Tableau III.1.

Puisque le modèle sélectionné est symétrique tant que $(I_x = I_y)$, l'équation différentielle du lacet peut être linéarisé et va être donné telle que $(\ddot{\psi} = \frac{1}{I_z} U_\psi)$. Nous avons donc deux sous-systèmes; le sous-système non-linéaire, roulis-tangage (φ, θ) et le sous-système linéaire, lacet (ψ) .

Nous n'avons pas besoin d'une approche de commande complexe pour le sous-système de lacet, donc nous allons le commander par un correcteur H_∞ linéaire. Par la suite, la représentation du sous-système roulis-tangage sera comme un système quasi-LPV.

V.4.1 Sélection des Paramètres:

La représentation LPV-LFT d'un système non-linéaire n'est pas unique; la sélection des paramètres a influencé sur la complexité de synthèse du correcteur, en regardant le modèle (V.45), puisque le but est de découpler le lacet du sous-système roulis-tangage nous avons choisi ($\rho_1 = \dot{\psi}$), et ($\rho_2 = \Omega_r$) comme des paramètres variés dans le temps, et puis nous avons créé le modèle LPV-LFT (Figure V.7) du sous-système roulis-tangage et nous obtenons la représentation d'état comme suit:

$$\begin{bmatrix} \dot{x} \\ z_\rho \\ y \end{bmatrix} = \begin{bmatrix} \mathcal{A} & \mathcal{B}_\rho & \mathcal{B}_1 \\ \mathcal{C}_\rho & \mathcal{D}_{\rho\rho} & \mathcal{D}_{\rho 1} \\ \mathcal{C}_1 & \mathcal{D}_{1\rho} & \mathcal{D}_{11} \end{bmatrix} \cdot \begin{bmatrix} x \\ w_\rho \\ u \end{bmatrix} \quad (\text{V.48})$$

$$w_\rho = \Theta z_\rho$$

$$\text{où: } \mathcal{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathcal{B}_\rho = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{I_y - I_z}{I_x} & -\frac{J_r}{I_x} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{I_z - I_x}{I_y} & \frac{J_r}{I_y} \end{bmatrix},$$

$$\mathcal{B}_1 = \begin{bmatrix} 0 & 0 \\ \frac{l}{I_x} & 0 \\ 0 & 0 \\ 0 & \frac{l}{I_y} \end{bmatrix}, \mathcal{C}_\rho = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \mathcal{C}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$\mathcal{D}_{\rho\rho} = 0_{4 \times 4}, \quad \mathcal{D}_{\rho 1} = 0_{4 \times 2}, \quad \mathcal{D}_{1\rho} = 0_{2 \times 4}, \quad \mathcal{D}_{11} = 0_{2 \times 2}$$

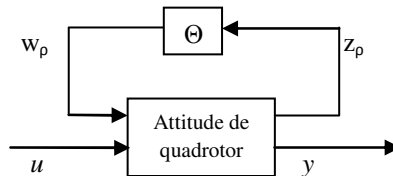


Figure V.7 Schéma LPV-LFT d'attitude du quadrotor

Nous avons quatre états $[\varphi \dot{\varphi} \theta \dot{\theta}]$, deux entrées (U_φ, U_θ) et deux paramètres variés (ρ_1, ρ_2) avec deux récurrences pour chacun, c-à-d $\Theta = \text{diag}(\rho_1, \rho_2, \rho_1, \rho_2)$. Selon le lacet, il est un système LTI simple avec deux états et une seule entrée. Maintenant, nous avons la représentation LPV-LFT

d'attitude du quadrotor, mais nous n'avons pas généré encore la boucle fermée. Par la suite, la boucle fermée va être obtenue et alors la synthèse du correcteur H_∞ à séquençement de gains est faite.

V.4.2 Synthèse du Correcteur:

Après la modélisation du système, il est nécessaire d'établir l'interconnexion entre le processus et les différentes entrées extérieures; les signaux de références, perturbations et les bruits. En plus, les fonctions de pondération qui présentent les performances désirées sont établies, par sensibilité mixte, par les signaux des erreurs et les entrées (les signaux de commande).

De plus nous avons interconnecté le quadrotor avec la dynamique des actionneurs. Le quadrotor est commandé par les vitesses angulaires des rotors ($\omega_1 \dots \omega_4$), mais pour simplifier la compréhension du processus, les vitesses des rotors sont couplés pour avoir la commande de roulis, la commande de tangage et la commande de lacet (V.47). La dynamique du rotor obtenue de (Bouabdallah, 2007) est présentée par la fonction de transfert:

$$D(s) = \frac{\omega_i}{\omega_{di}} = \frac{0.936}{0.178s+1} \quad (\text{V.49})$$

Où ω_i et ω_{di} sont la vitesse angulaire actuelle et désirée du $i^{\text{ème}}$ rotor, respectivement. L'entrée désirée (U_d) est générée par le correcteur et l'entrée actuelle (U_a) est appliquée directement au modèle du processus après le passage de (U_d) par les actionneurs, en le combinant avec (V.47) et (V.49):

$$U_a = D^2(s).U_d \quad (\text{V.50})$$

L'équation (V.50) présente la dynamique des actionneurs pour chaque entrée (U_ϕ , U_θ , U_ψ); commandes roulis, du tangage, et du lacet. Selon les entrées extérieures, nous considérons trois types; les angles désirés, les perturbations des entrées et les bruits des capteurs (Figure V.8).

Puisque nous avons considéré la division du modèle d'attitude en deux différents sous-systèmes; un LTI (sous-système de lacet) et un LPV (sous-système de roulis-tangage), le correcteur de chaque sous-système a la même nature que le système à commander.

V.4.2.1 Le Correcteur H_∞ LTI:

La synthèse du correcteur H_∞ LTI pour le sous-système de lacet est réalisée par la résolution des équations de Riccati et loop shaping (Balas et al., 2009).

Les fonctions de pondération sont, $W_{e\psi} = \frac{s+1}{s+0.2}$ pour le critère d'erreur et $W_{u\psi} = 10^{-2} \frac{s^2+15s+1}{5s^2+5s+1}$ pour le critère de commande. Ces fonctions présentent le comportement désiré de la boucle fermée, dans notre cas; une bande passante limitée à 1 rad/s et l'erreur permanente admissible est égal à 0.2. Le gain d'haute fréquence est limité à 10^{-2} par la bande passante du correcteur (7.5 rad/s) (Zhou et Doyle, 1998). Nous avons un correcteur H_∞ LTI avec un critère de performance $\gamma_\psi=1.0008$.

V.4.2.2 Le Correcteur H_∞ LPV:

Le correcteur LPV est conçu par le théorème de petit gain scalaire et l'optimisation convexe par LMI (Apkarian et Gahinet, 1995). L'avantage de l'algorithme de synthèse d'un correcteur LPV-LFT est qu'il n'y a pas de besoin de considérer les bandes des dérivées de les paramètres variés, le critère de performance obtenu, γ , est la norme L_2 quand il n'y a pas des limites imposées à la vitesse de changement de valeur de paramètre par le temps. Le tableau V.1 présente les intervalles des paramètres de sous-système de roulis-tangage.

Tableau V.1 Les intervalles des paramètres variés

Paramètre	description	intervalle
ρ_1	$\dot{\psi}$	$[-2, 2] \text{ rad.s}^{-1}$
ρ_2	Ω_r	$[-300, 300] \text{ rad.s}^{-1}$

Les fonctions de pondération sont obtenues de même que celles du sous-système de lacet, une fonction d'erreur $W_{e\varphi,\theta} = \frac{s+1}{s+0.2} * I_2$ et l'autre de la commande; $W_{u\varphi,\theta} = 10^{-2} \frac{s^2+25s+1}{5s^2+5s+1} * I_2$, où I_2 est la matrice identité de dimension 2. Parce que le sous-système du roulis-tangage est un système MIMO, nous avons présenté ses fonctions de pondération W_e et W_u comme des matrices diagonales.

Ces fonctions présentent le comportement désiré de la boucle fermée, dans notre cas; une bande passante limitée à 1 rad/s et l'erreur permanente admissible est égal à 0.2. Le gain d'haute fréquence est limité à 10^{-2} par la bande passante du correcteur (12.5 rad/s) (Zhou et Doyle, 1998).

L'algorithme de synthèse du correcteur est détaillé en sections V.3.2.2 et V.3.2.3. Nous avons un correcteur H_∞ LPV avec un critère de performance $\gamma_{\varphi,\theta}=2.89$. L'algorithme de commande des systèmes LPV-LFT (Hjartarson et al., 2015) n'impose pas le taux de variation des paramètres variés dans le temps et aussi, il est juste une limite supérieure sur la norme L_2 , donc, si les perturbations et les bruits ont une norme L_2 limité par 1, la norme L_2 du système en boucle fermée ne dépasse pas 2.89 pour toute trajectoire des paramètres $(\dot{\psi}, \Omega_r)$ comme mentionné dans Tableau V.1.

Afin de combiner les deux différents correcteurs, nous supposons que le correcteur LPV-LFT (pour le sous-système de roulis-tangage) a une représentation comme (V.35). On appelle ce correcteur

$$K_{LPV} = \begin{bmatrix} A_{K1} & B_{K1} & B_{K\rho} \\ C_{K1} & D_{K11} & D_{K1\rho} \\ C_{K\rho} & D_{K\rho1} & D_{K\rho\rho} \end{bmatrix} \text{ qui a une structure LFT inférieure comme présenté à la Figure V.5.}$$

On appelle le correcteur LTI (pour le sous-système de lacet) $K_{LTI} = \begin{bmatrix} A_{K2} & B_{K2} \\ C_{K2} & D_{K2} \end{bmatrix}$, le correcteur combiné pour le système complet est montré par ses matrices;

$$A_K = \begin{bmatrix} A_{K1} & \mathbb{0} \\ \mathbb{0} & A_{K2} \end{bmatrix}, B_K = \begin{bmatrix} B_{K1} & \mathbb{0} & B_{K\rho} \\ \mathbb{0} & B_{K2} & \mathbb{0} \end{bmatrix}, C_K = \begin{bmatrix} C_{K1} & \mathbb{0} \\ \mathbb{0} & C_{K2} \\ C_{K\rho} & \mathbb{0} \end{bmatrix}, D_K = \begin{bmatrix} D_{K11} & \mathbb{0} & D_{K1\rho} \\ \mathbb{0} & D_{K2} & \mathbb{0} \\ D_{K\rho1} & \mathbb{0} & D_{K\rho\rho} \end{bmatrix} \quad (\text{V.51})$$

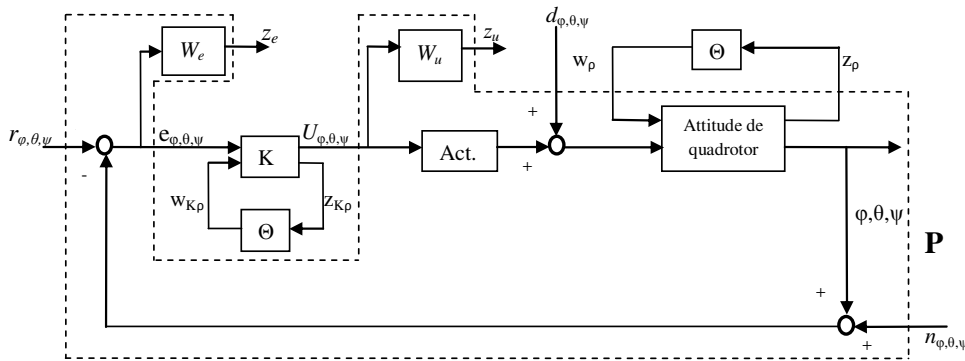


Figure V.8 La boucle fermée système-correcteur d'attitude du quadrotor

V.5 Suivi de Trajectoire et Simulation:

V.5.1 Suivi de Trajectoire:

Comme nous avons mentionné précédemment, le modèle du quadrotor est compliqué car il s'agit d'un système sous-actionné; où les mouvements translationnelles sur les axes x et y n'ont pas d'entrées directes, et il n'y a pas de sens à appliquer une stratégie de commande directe. Par conséquent, nous devons diviser le modèle du quadrotor en deux blocs en cascade. Etant donné que la commande du sous-système de rotation (attitude) a été établie par LPV, nous analysons le suivi de la trajectoire par Backstepping intégral comme nous l'avons fait dans la section IV.3.

Les états: $x_1 = z, x_2 = \dot{z}, x_3 = x, x_4 = \dot{x}, x_5 = y, x_6 = \dot{y}$.

La représentation d'état du sous-système de translation:

$$\dot{X} = \begin{cases} x_2 \\ \frac{1}{m}(c\varphi c\theta)U_z - g \\ x_4 \\ \frac{1}{m}u_x U_z \\ x_6 \\ \frac{1}{m}u_y U_z \end{cases} \quad (\text{V.52})$$

Où, les commandes virtuelles:

$$\begin{aligned} u_x &= (c\varphi s\theta c\psi + s\varphi s\psi) \\ u_y &= (c\varphi s\theta s\psi - s\varphi c\psi) \end{aligned} \quad (\text{V.53})$$

Les lois de commande de translation sont les mêmes que celles de la section IV.3:

$$U_z = \frac{m}{(\cos\varphi \cos\theta)} (\ddot{x}_{1d} + (1 - c_1^2 + \lambda_1)e_1 - c_1\lambda_1\xi_1 + (c_1 + c_2)e_2 + g) \quad (\text{V.54})$$

$$u_x = \frac{m}{U_z} (\ddot{x}_{3d} + (1 - c_3^2 + \lambda_2)e_3 + (c_3 + c_4)e_4 - c_3\lambda_2\xi_2) \quad (\text{V.55})$$

$$u_y = \frac{m}{U_z} (\ddot{x}_{5d} + (1 - c_5^2 + \lambda_3)e_5 + (c_5 + c_6)e_6 - c_5\lambda_3\xi_3) \quad (\text{V.56})$$

Avec: $e_1 = x_{1d} - x_1$, $\xi_1 = \int_0^t e_1(\tau) d\tau$, $e_2 = x_{2d} - x_2$, $e_3 = x_{3d} - x_3$, $\xi_2 = \int_0^t e_3(\tau) d\tau$, $e_4 = x_{4d} - x_4$, $e_5 = x_{5d} - x_5$, $\xi_3 = \int_0^t e_5(\tau) d\tau$, $e_6 = x_{6d} - x_6$.

Les commandes virtuelles (u_x, u_y) sont utilisées pour générer les angles désirés:

$$\begin{cases} \varphi_d = \sin^{-1}(u_x \sin \psi - u_y \cos \psi) \\ \theta_d = \sin^{-1}\left(\frac{u_x \cos \psi + u_y \sin \psi}{\cos \varphi}\right) \end{cases} \quad (\text{V.57})$$

La structure de commande générale est présentée dans la Figure V.9 où; la stabilisation d'attitude est établie par LPV H_∞ avec robustesse. Par contre le suivi de trajectoire (translation) est établi par Backstepping intégral (Hasseni et Abdou, 2018).

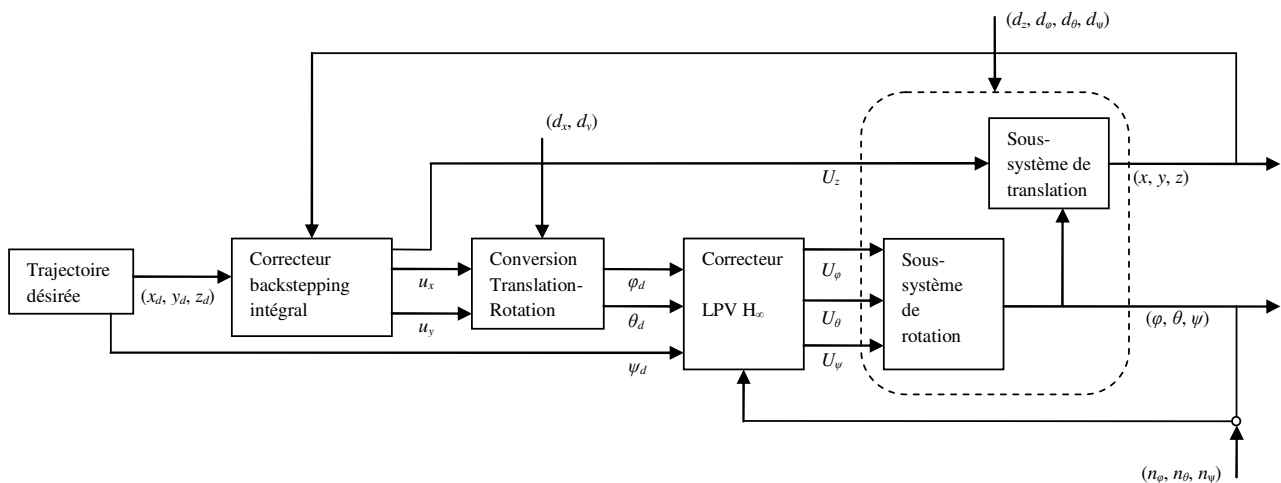


Figure V.9 La structure de commande du quadrotor

V.5.2 Résultats de Simulation:

La trajectoire que nous allons simuler est comm suit; en premier mode, le quadrotor monte à 2 m d'altitude et en deuxième mode, il fait une trajectoire comme un carré de 10 m de coté à la fois sur x et y , et en même temps, il tourne sur lui-même (ψ) pendant 50 secondes.

Les perturbations sont supposées être des impulsions d'amplitude $d_\phi=0.1\text{ N}$ dans l'instant 14 s , $d_\theta=0.1\text{ N}$ dans l'instant 16 s , et $d_\psi=0.03\text{ Nm}$ dans l'instant 17 s . Tant les bruits appliqués sont des signaux aléatoires permanents dans un intervalle de $[-0.5\ 0.5]$ rad. De plus, pour montrer la robustesse face aux incertitudes paramétriques, nous verrons dans les prochains figures; Figure V.10 - Figure V.13 la simulation du suivi de trajectoire avec trois valeurs d'inerties différentes sur l'axe x , y , z et l'inertie du rotor (I_x , I_y , I_z , et J_r); paramètres nominaux, et $\pm 20\%$. La Figure V.14 présente les signaux de commande avec différents paramètres d'inertie.

C'est clair que nous avons une stabilité robuste face aux perturbations, bruits et les incertitudes paramétriques. Malgré qu'il y a des bruits de mesure de $[-0.5\ 0.5]$ rad, les variations dans les réponses des angles (Figure V.11) sont très petits et ils n'ont pas d'effet sur les réponses de translation (Figure V.10). En plus, les perturbations d_ϕ , d_θ , d_ψ sont rejetés rapidement, et toutes les trajectoires sont presque identiques.

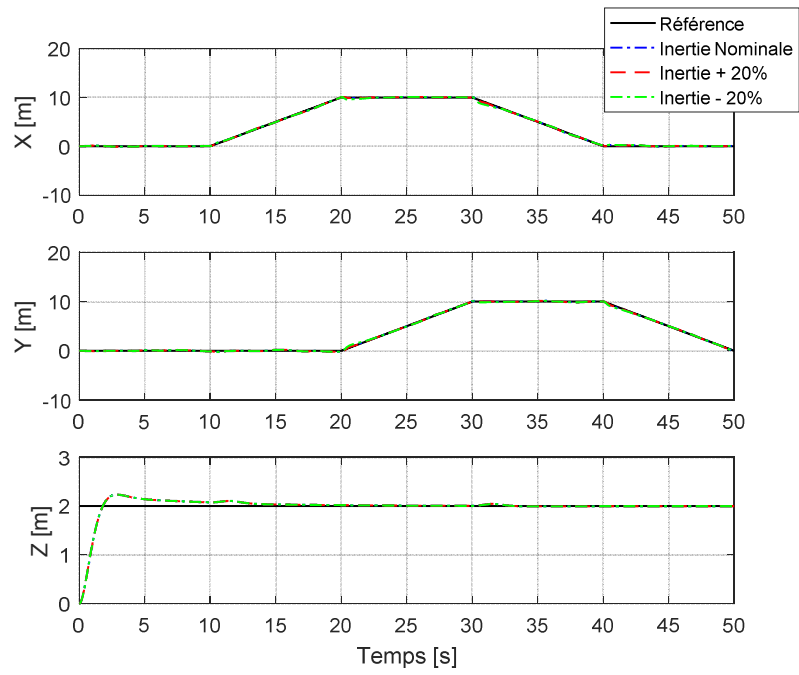


Figure V.10 Les réponses des coordonnées de translation avec des incertitudes sur les inerties

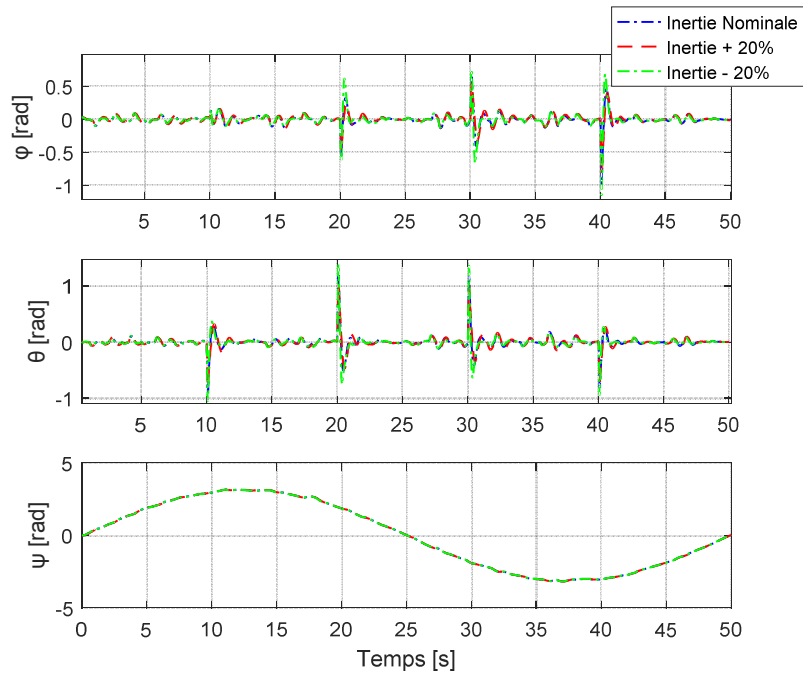


Figure V.11 Les réponses des coordonnées d'orientation avec des incertitudes sur les inerties

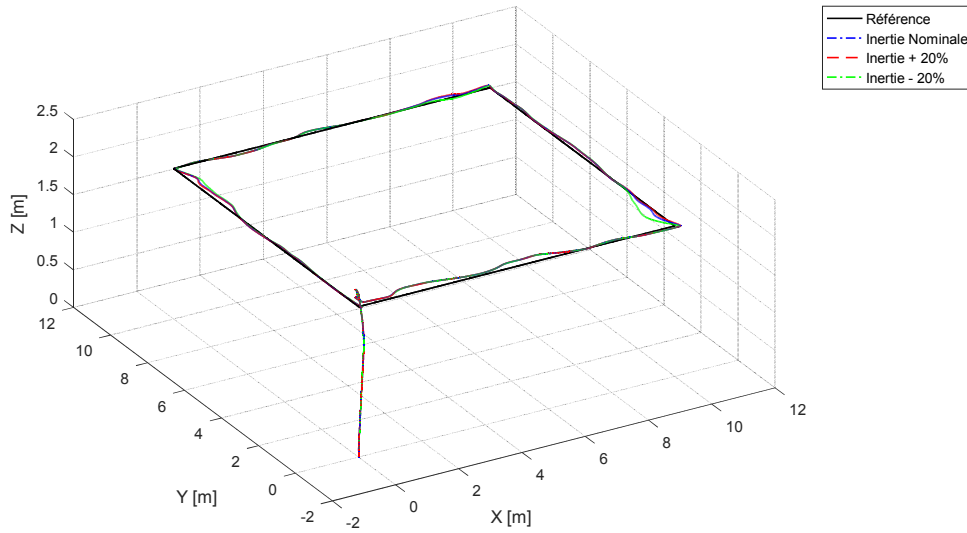


Figure V.12 La trajectoire en 3D

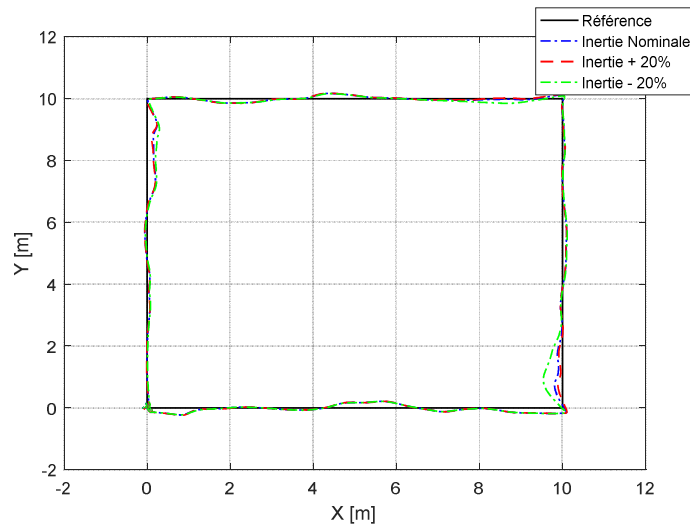


Figure V.13 Une vue horizontale de la trajectoire

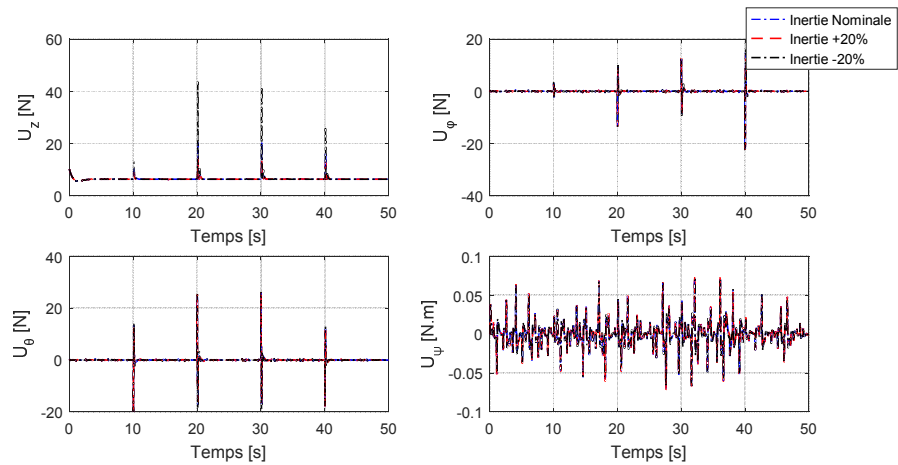


Figure V.14 Les signaux de commande

Afin de montrer l'effet de l'action d'intégral, nous avons appliqué des perturbations comme des échelons de $0.5 N$ sur U_z à partir de l'instant $20 s$, sur u_x à partir de l'instant $16 s$ et sur u_y à partir de l'instant $14 s$. Nous allons tester le suivi de la même trajectoire de la simulation précédente par la stratégie hybride (Backstepping intégrale / LPV H_∞) et celle sans l'intégral (Backstepping / LPV H_∞). La Figure V.15 et La Figure V.16 présentent le suivi de trajectoire en 3D et les réponses des coordonnées de translation.

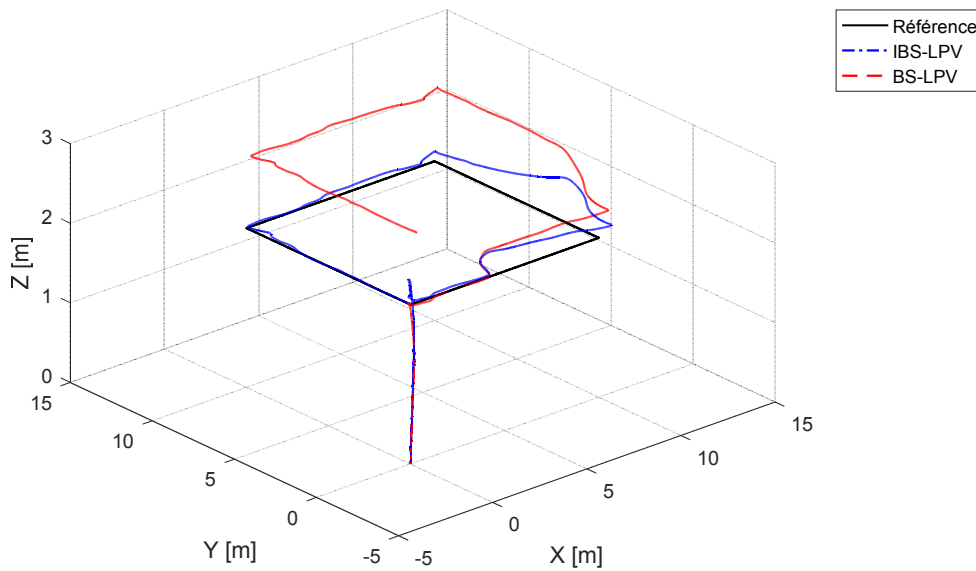


Figure V.15 La trajectoire en 3D du IBS-LPV et BS-LPV

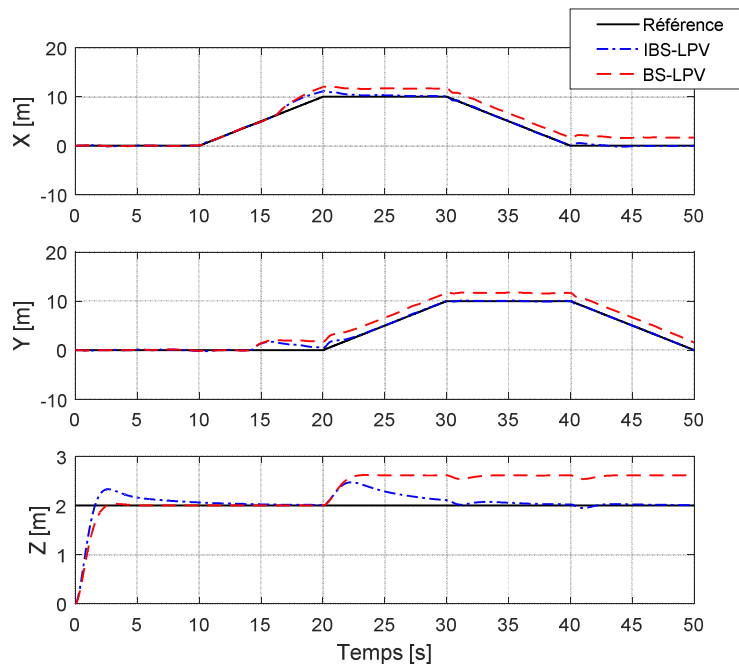


Figure V.16 Les réponses de coordonnées de translation du IBS-LPV et BS-LPV

Nous pouvons observer que le correcteur Backstepping integral / LPV (IBS-LPV) est mieux que le Backstepping / LPV (BS-LPV) en terme de performance sur le suivi de la trajectoire. La réponse du système avec IBS-LPV montre que les perturbations sont rejetées, ce qui n'est pas le cas pour le BS-LPV, où il y a des erreurs permanentes notés tel que; $0.5 m$ sur l'altitude (z), $2 m$ sur la coordonné x et $2 m$ sur la coordonné y .

V.6 Commande d'un Système Sous-Actionné; Pendule Inversé:

Dans cette section, nous allons commander un véhicule qui s'appelle *Personal Pendulum Vehicle* (Véhicule Pendule Personnel) (Fiacchini et al., 2006) par la même approche vue précédemment, LPV-LFT robuste. Cette section est une amélioration par rapport au (Hasseni et Abdou, 2017), cette amélioration se concentre sur deux points; éliminer le contrainte de sous-actionné et l'optimisation les fonctions de pondération. Ce véhicule (Figure V.17) est un pendule inversé avec deux roues, où l'utilisateur contrôle le mouvement d'avancer ou de reculer en s'inclinant.



Figure V.17 Vue générale du véhicule pendule personnel

Le modèle du pendule inversé sur chariot est un benchmark pour plusieurs applications comme le segway (Boubaker, 2012), c'est un véhicule avec deux degrés de liberté; rotation (θ) et translation (x), mais il a une seule entrée qui est la force linéaire (F) (Figure V.18).

Le modèle dynamique de ce véhicule est dérivé de la formulation d'Euler-Lagrange, en considérant le modèle utilisé dans (Fiacchini et al., 2006; Raffo et al., 2007):

$$\begin{bmatrix} M + m & Ml \cos(\theta) \\ Ml \cos \theta & M l^2 \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} b & -Ml \sin(\theta)\dot{\theta} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ -Mgl \sin \theta \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix} \quad (\text{V.58})$$

Avec; m est la masse du chariot, M est la masse du pendule qui présente dans notre cas l'utilisateur, l est la longueur du bras du pendule, g est la constante de gravité, b est le coefficient de force de friction, et F est la force appliquée. Leurs valeurs sont présentées dans le Tableau V.2.

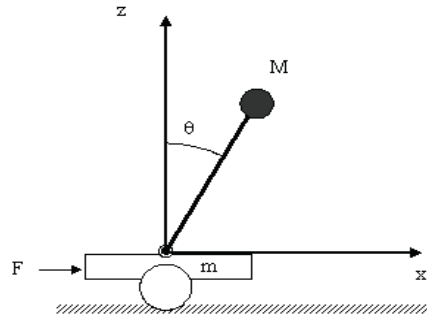


Figure V.18 Schéma géométrique du pendule inversé sur chariot

Tableau V.2 Les valeurs nominales du pendule inversé sur chariot

Paramètre	Description	Valeur	Unité
m	La masse du chariot	35	Kg
M	La masse du penule	70	Kg
l	Longueur du bras	1	m
g	Constante de gravité	9.8	m.s^{-2}
b	Coefficient de friction	40	N.s.m^{-1}

V.6.1 Représentation LFT- LPV du Pendule Inversé:

À partir de (V.58) nous pouvons extraire ces deux équations:

$$\begin{cases} (M + m)\ddot{x} + Ml \cos(\theta) \ddot{\theta} + b\dot{x} - Ml \sin(\theta)\dot{\theta}^2 = F \\ Ml \cos(\theta) \ddot{x} + Ml^2 \ddot{\theta} - Mgl \sin(\theta) = 0 \end{cases} \quad (\text{V.59})$$

Le terme $(Ml \sin(\theta)\dot{\theta}^2)$ rend l'application de la technique LPV-LFT difficile. Ainsi, en changeant le signal généré par le correcteur, le nouveau modèle est donné par:

$$\begin{cases} (M + m)\ddot{x} + Ml \cos(\theta) \ddot{\theta} + b\dot{x} = u \\ Ml \cos(\theta) \ddot{x} + Ml^2 \ddot{\theta} - Mgl \sin(\theta) = 0 \end{cases} \quad (\text{V.60})$$

Où:

$$u = F + Ml \sin(\theta)\dot{\theta}^2 \quad (\text{V.61})$$

En prenant en considération les non-linéarités comme des paramètres variés, et par des petits changements pour rendre le système commandable et observable, la représentation LPV-LFT est:

$$\begin{bmatrix} \dot{x} \\ z_\rho \\ y \end{bmatrix} = \begin{bmatrix} \mathcal{A} & \mathcal{B}_\rho & \mathcal{B}_1 \\ \mathcal{C}_\rho & \mathcal{D}_{\rho\rho} & \mathcal{D}_{\rho 1} \\ \mathcal{C}_1 & \mathcal{D}_{1\rho} & \mathcal{D}_{11} \end{bmatrix} \cdot \begin{bmatrix} x \\ w_\rho \\ u \end{bmatrix} \quad (\text{V.62})$$

$$w_\rho = \theta z_\rho$$

Avec x est le vecteur des états $x = [\theta \quad \dot{\theta} \quad \ddot{x}]^T$.

Les matrices sont alors:

$$\begin{aligned} \mathcal{A} &= \begin{bmatrix} 0 & 1 & 0 \\ \frac{g}{l} & 0 & \frac{b}{(M+m)l} \\ 0 & 0 & \frac{-b}{(M+m)} \end{bmatrix}, \mathcal{B}_\rho = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{g}{l} & \frac{1}{(M+m)l} & 0 & \frac{M}{(M+m)} \\ 0 & 0 & \frac{Ml}{(M+m)} & 0 \end{bmatrix}, \\ \mathcal{B}_1 &= \begin{bmatrix} 0 \\ -1 \\ \frac{1}{(M+m)} \end{bmatrix}, \mathcal{C}_\rho = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & \frac{b}{(M+m)l} \\ \frac{g}{l} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathcal{C}_1 = [1 \quad 0 \quad 0], \\ \mathcal{D}_{\rho\rho} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{g}{l} & \frac{1}{(M+m)l} & 0 & \frac{M}{(M+m)} \\ 0 & 0 & 1 & 0 \end{bmatrix}, \mathcal{D}_{\rho 1} = \begin{bmatrix} 0 \\ -1 \\ -1 \\ \frac{1}{(M+m)} \end{bmatrix}, \\ \mathcal{D}_{1\rho} &= [0 \quad 0 \quad 0 \quad 0], \mathcal{D}_{11} = 0, \theta = \begin{bmatrix} \rho_1 & 0 & 0 & 0 \\ 0 & \rho_2 & 0 & 0 \\ 0 & 0 & \rho_3 & 0 \\ 0 & 0 & 0 & \rho_4 \end{bmatrix} \end{aligned} \quad (\text{V.63})$$

Tous les paramètres $\rho_1 \dots \rho_4$ sont reliés à l'angle θ , théoriquement $-90^\circ \leq \theta \leq 90^\circ$ mais pour éviter la non-commandabilité, on réduit l'intervalle à $-80^\circ \leq \theta \leq 80^\circ$. Le Tableau V.3 présente l'intervalle de chaque paramètre.

Tableau V.3 Les intervalles des paramètres variés

Paramètre	Description	Valeur min.	Valeur max.
ρ_1	$\left(\frac{\sin \theta}{\theta}\right) - 1$	-0.3	0
ρ_2	$\cos \theta - 1$	-0.83	0
ρ_3	$\cos \theta$	0.17	1
ρ_4	$\cos^2 \theta$	0.03	1

V.6.2 Sélection des Paramètres des Fonctions de Pondération:

Afin d'obtenir la boucle fermée, nous interconnectons le système avec les entrées externes; la perturbation (d) et le bruit de mesure (n). Le signal de référence est toujours zéro dans notre cas

parce que l'objectif est de stabiliser le véhicule à l'angle nul. En plus de ces entrées externes, nous plaçons aussi les fonctions de pondération; à l'erreur W_e et à la commande W_u . Comme dernière étape, nous avons obtenu la force appliquée (F) à partir de la commande générée (u) de (V.61):

$$F = u - Ml \sin(\theta)\dot{\theta}^2 \quad (V.64)$$

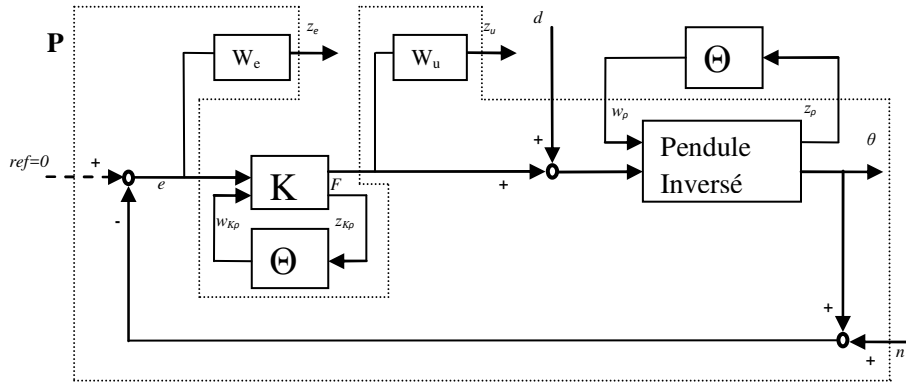


Figure V.19 L'interconnexion en boucle fermée du système-correcteur

Parfois, la sélection des paramètres des fonctions de pondération est difficile, surtout dans la commande soit disant LPV H_∞ où elle présente la norme L_2 de l'espace de temps et les fonctions de pondération sont des présentations fréquentielles de la performance (Do et al., 2011; Vu et al., 2017). Dans cette section, nous essayons d'obtenir les paramètres des fonctions de pondération optimaux par les algorithmes génétiques.

Les fonctions de pondération présentées dans la Figure V.19 sont considérées comme des filtres de premier ordre:

$$W_e = k_e \frac{\tau_{e1}s+1}{\tau_{e2}s+1} \quad (V.65)$$

$$W_u = k_u \frac{\tau_{u1}s+1}{\tau_{u2}s+1} \quad (V.66)$$

L'algorithme d'optimisation par GA est le même utilisé dans le chapitre III. Nous présentons les paramètres à optimiser (les solutions) dans un vecteur:

$$sol. = [k_e \tau_{e1} \tau_{e2} k_u \tau_{u1} \tau_{u2}] \quad (V.67)$$

Le coût choisi dans ce cas est multi-objectif, où nous avons pris deux critères temporaires, qui sont le temps de réponse (TR) et le premier dépassement (D) avec les coefficients 0.3 et 0.7 respectivement:

$$Fob = 0.3 TR + 0.7 D \quad (V.68)$$

La synthèse du correcteur est faite par la même technique qu'est faite dans le cas du quadrotor, le théorème de petit gain scalaire (Hjartarson et al., 2015).

Comme résultat, les fonctions de pondération optimales sont comme suit:

$$W_e = 3.286 \frac{4.015s+1}{6.603s+1} \quad (\text{V.69})$$

$$W_u = 0.673 \frac{6.362s+1}{0.242s+1} \quad (\text{V.70})$$

V.6.3 Stabilité Étendue de 2DOF:

Un système sous-actionné est un système mécanique qui a moins d'entrées que de degrés de liberté, comme c'est le cas du pendule inversé, qui a deux degrés de liberté (θ, x) et une seule entrée (F) . Dans la section précédente, nous avons considéré le pendule inversé comme un système mono-variable, la stabilisation de l'état de vitesse se fait très lentement. À l'inverse du quadrotor, nous ne pouvons pas appliquer la commande en cascade parce que le modèle est très complexe, donc, nous devons considérer la contrainte du sous-actionné imposée.

Il y a plusieurs classifications, l'une des plus communes est le *Control Flow Diagram* (CFD), qui est basée sur la relation entre les degrés de liberté. Il y a trois structures; chaîne, arbre, et sommet-isolé avec plus de détails dans (Seto and Baillieul, 1994; Choukchou-Braham et al., 2014a).

Le pendule inversé est sous la structure arbre, qui peut être divisée en deux sous-classes (A1, et A2). Nous pouvons convertir un système (A1) à une structure chaîne et changer les états (Rudra et al., 2017), contrairement à (A2), où ce n'est pas possible d'appliquer le changement. Dans (Choukchou-Braham et al., 2014b), les auteurs ont utilisé une méthode où les degrés de liberté étaient commandées simultanément sans changer le correcteur, ce que nous utilisons ici. Nous incluons un terme lié à (θ) et un terme lié à (x) . Pour le pendule inversé sur chariot, la vitesse linéaire (\dot{x}) doit être nulle en même temps que (θ) , et nous notons l'état \dot{x} par v . Le seul terme qui contient information sur (θ) est la sortie (Figure V.19), alors, nous changeons la boucle fermée par l'ajout d'un terme proportionnel sur v ($k_v v$), le gain k_v présente le poids de la vitesse par rapport à l'angle (θ) (Figure V.20). Ainsi, nous avons choisi $k_v=0.02$ et la sortie sera $(\theta + 0.02 v)$.

Remarque: nous n'avons pas conçu un autre correcteur, mais équiper le correcteur développé dans la section précédente.

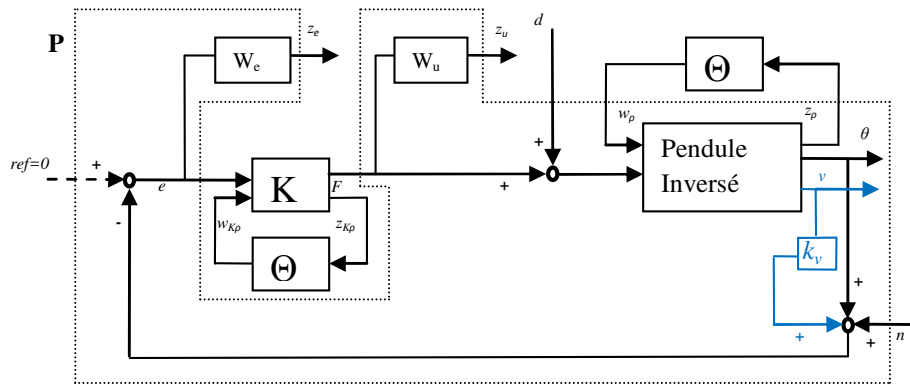


Figure V.20 La boucle fermée du système-correcteur sous-actionné

V.6.4 Résultats de simulation:

Dans notre système la masse du pendule M présente la masse de l'utilisateur. Nous allons appliquer les incertitudes sur la masse en prenant 70 Kg comme la masse nominale et aussi deux masses différentes (40 Kg et 100 Kg), en outre, nous appliquons une perturbation échelon de 200 N sur l'entrée à partir de l'instant 5 s .

La Figure V.21 présente les réponses de l'angle (θ), la vitesse linéaire (V) et la force appliquée (F) pour ces conditions. Nous pouvons remarquer que l'angle et la vitesse sont stabilisés et atteindront le point d'équilibre (zéro) dans le même instant, et les deux degrés de liberté sont stabilisés.

Nous observons, aussi, que le correcteur est robuste face aux incertitudes et perturbations; Le premier dépassement de l'angle est entre -20° et -30° pour $M=100 \text{ Kg}$ et $M=40 \text{ Kg}$ respectivement.

Nous notons aussi que la vitesse maximale est de 8 m/s , 9 m/s et 10 m/s pour $M=100 \text{ Kg}$, $M=70 \text{ Kg}$, et $M=40 \text{ Kg}$ respectivement.

La simulation précédente est réalisée avec 60° pris comme angle initial. Sur la Figure V.22 la simulation est réalisée avec des angles initiaux différents (60° , 30° , -30° et -60°) et la masse nominale (70 Kg). L'angle et la vitesse sont stabilisés et atteindront le point d'équilibre (zéro) dans le même instant, et le correcteur est robuste quelque soit l'angle initial.

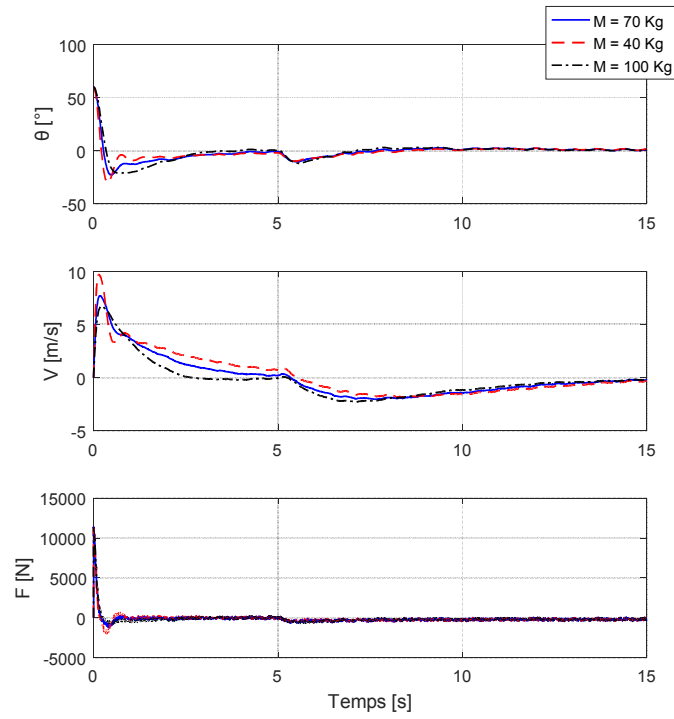


Figure V.21 Les réponses de l'angle (θ), la vitesse (V) et la force (F) avec des incertitudes sur la masse M

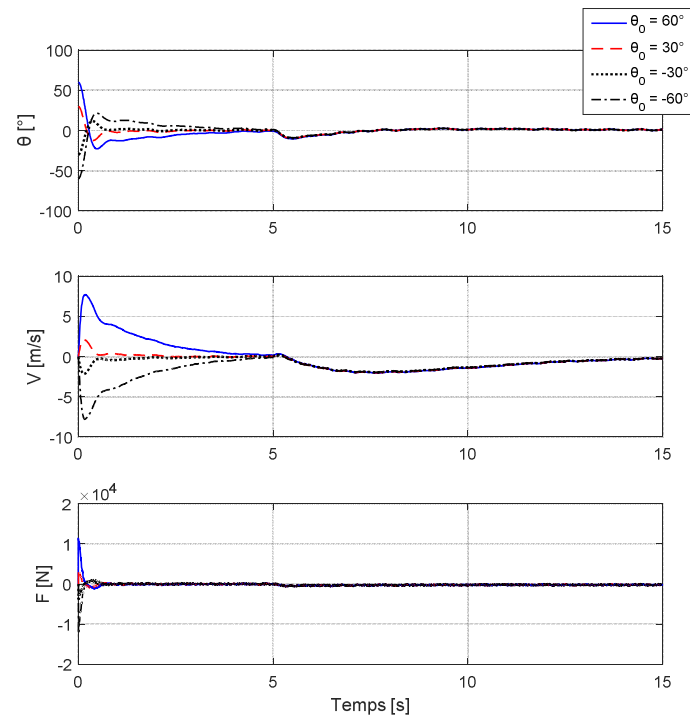


Figure V.22 Les réponses de l'angle (θ), la vitesse (V) et la force (F) avec différents angles initiaux

V.7 Conclusion:

Nous avons commandé, dans ce chapitre, des systèmes non-linéaires par la technique LPV-LFT, où le système non-linéaire a été représenté comme un système LPV rationnel. L'avantage de ce dernier est que nous n'avons pas réduit la performance du système non-linéaire par la linéarisation, mais plutôt nous avons représenté le système non-linéaire complet par une représentation LPV, cela le système est appelé quasi-LPV. Par la suite, nous avons développé une technique robuste basée sur le petit gain scalaire, qui garantit la robustesse de la stabilité face aux entrées externes; les perturbations et les bruits, ainsi que les incertitudes paramétriques. Cette technique a été appliquée sur la stabilité d'attitude du quadrotor, malgré la complexité de son modèle, nous avons pu appliquer la commande en cascade, où le suivi de la trajectoire a été fait par Backstepping intégral. La classe des systèmes sous-actionnés qui est approprié au quadrotor est la structure chaîne où nous avons commandé des degrés de liberté qui ont des commandes directes et en suite nous avons commandé les degrés qui n'ont pas des commandes directes par considérer certains signaux de références des degrés internes comme des commandes virtuelles. Cette technique n'a pas pu être appliquée sur le pendule inversé parce qu'il est de classe arbre A2 (Choukchou-Braham et al., 2014a). Nous avons neutralisé la contrainte de sous-actionné par stabiliser les deux degrés de liberté simultanément. Nous avons sélectionné les paramètres des fonctions de pondération de pendule inversé par les algorithmes génétiques. La technique de commande LPV robuste est réalisée avec succès, où la robustesse est garantie face aux entrées externes et aux incertitudes paramétriques. Dans le chapitre suivant, nous allons étudier et développer le correcteur robuste H_∞ par une autre méthode, qui est les inégalités non-linéaires de Hamilton Jacobi Bellman Isaacs et elle est appelée H_∞ non-linéaire, afin de développer les lois de commande des DOF actionnés avec leur couplage avec les DOF sous-actionnés.

Chapitre VI:

Commande Robuste du Quadrotor par la Technique H_∞ Non-Linéaire

VI.1 Introduction:

Dans ce chapitre, nous allons commander le quadrotor par la commande robuste H_∞ non linéaire, puisque la commande est générée à partir d'une résolution des équations optimales non linéaires. La commande est appropriée aux systèmes sous-actionnés. À l'inverse des techniques montrées dans les trois chapitres précédents où nous avons appliqué une structure en cascade, dans ce chapitre l'objectif est de commander les degrés de liberté contrôlés (actionnés) par leur couplage dynamique avec les degrés de liberté sous-actionnés. La commande par couple calculé a été créée premièrement pour les robots manipulateurs, cependant nous l'exploitons ici pour le quadrotor. La commande par couple calculé nécessite la connaissance du modèle presque complet. Pour cela, nous allons au début, présenter la commande par couple calculé, puis, nous allons formuler le modèle, presque, complet du quadrotor en utilisant la méthode Euler-Lagrange. En suite, nous allons établir les lois de commande par H_∞ non linéaire, et à la fin une simulation du suivi de la trajectoire va présenter afin de tester la robustesse.

VI.2 Commande par Couple Calculé:

La commande par couple calculé est appelée aussi commande dynamique car elle utilise le modèle dynamique du robot. Elle nécessite la connaissance du modèle le plus complet possible, et la bonne identification de ses paramètres.

Pour bien comprendre le problème de commande par couple calculé, nous allons premièrement faire appel au modèle dynamique de forme général d'un robot avec n degrés de liberté:

$$\Gamma = M(q, \dot{q})\ddot{q} + C(q, \dot{q})\dot{q} + G(q) \quad (\text{VI.1})$$

Avec:

Γ : le vecteur des couples/forces (entrées de commandes).

M : matrice d'inertie définie positive ($n \times n$).

C : matrice de Coriolis ($n \times n$).

G : le vecteur des couples de gravité ($n \times 1$).

q : le vecteur des positions des degrés de liberté.

\dot{q} : le vecteur des vitesses des degrés de liberté.

\ddot{q} : le vecteur des accélérations des degrés de liberté.

On suppose que les positions et les vitesses sont mesurées sans bruits. On pose \hat{M} , \hat{C} , et \hat{G} les estimations de M , C , et G respectivement. Par conséquent, de l'équation (VI.1), si nous avons défini une loi de commande Γ tel que (Chung et al., 2008):

$$\Gamma = \widehat{M}(q, \dot{q})y + \widehat{C}(q, \dot{q})\dot{q} + \widehat{G}(q) \quad (\text{VI.2})$$

Après le remplacement de (VI.2) dans (VI.1), on déduit que dans le cas idéal de modélisation parfaite et avec une absence des perturbations, le problème se réduit à une commande de n double intégrateurs où; $\ddot{q} = y$.

On pose \ddot{q}^d , \dot{q}^d , et q^d comme étant l'accélération désirée, la vitesse désirée et la position désirée du degré de liberté q respectivement et leurs erreurs sont:

$$\ddot{e} = \ddot{q} - \ddot{q}^d \quad (\text{VI.3})$$

$$\dot{e} = \dot{q} - \dot{q}^d \quad (\text{VI.4})$$

$$e = q - q^d \quad (\text{VI.5})$$

Pour suivre la trajectoire désirée, on définit y :

$$y = \ddot{q}^d - u \quad (\text{VI.6})$$

La dynamique de l'erreur est:

$$\ddot{e} = -u \quad (\text{VI.7})$$

On sélectionne la commande de la boucle externe u comme suit:

$$u = k_d \dot{e} + k_p e + k_i \int e \quad (\text{VI.8})$$

Le vecteur des couples est donné par:

$$\Gamma = \widehat{M}(q, \dot{q})[\ddot{q}^d - k_d \dot{e} - k_p e - k_i \int e] + \widehat{C}(q, \dot{q})\dot{q} + \widehat{G}(q) \quad (\text{VI.9})$$

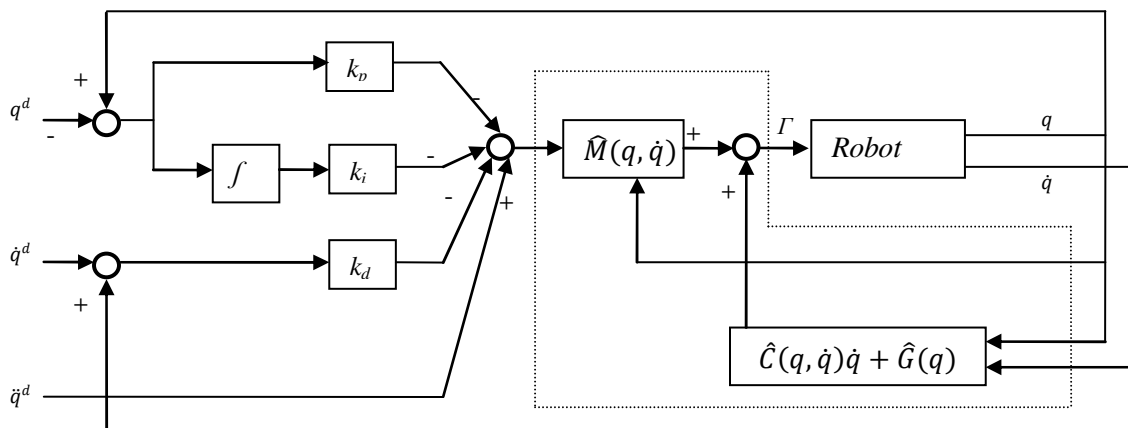


Figure VI.1 Schéma de commande par couple calculé

VI.3 Modélisation Complète du Quadrotor:

Le modèle dynamique du système est obtenu, en supposant que le quadrotor est un véhicule rigide, dont son mouvement dans l'espace est réalisé par une force principale (poussée) et trois couples. Cependant, ce type de véhicule est un système de vol de structure légère et, par conséquent, les effets de gyroscopes résultant du corps et des quatre hélices doivent être inclus dans le modèle dynamique. En outre, le quadrotor est un système mécanique sous-actionné avec six degrés de liberté et seulement quatre entrées de commande. En raison des complexités présentées, certaines hypothèses sont formulées à des objectifs de modélisation (Carrillo et al., 2013).

Les effets de moment causés par le corps rigide sur la dynamique de translation sont négligés, de même que l'effet de sol. La structure du quadrotor est supposée symétrique, ce qui donne une matrice d'inertie diagonale.

Le mouvement du quadrotor résulte des modifications de la vitesse des rotors. Pour obtenir un mouvement en avant, la vitesse du rotor arrière doit être augmentée et, simultanément, la vitesse du rotor avant doit être diminuée. Le déplacement latéral est effectué avec la même procédure mais en utilisant les moteurs droit et gauche. Le mouvement de lacet est obtenu à partir de la différence de contre couple entre chaque paire d'hélices, c'est-à-dire en accélérant les deux rotors tournant dans le sens inverse des aiguilles d'une montre, et inversement.

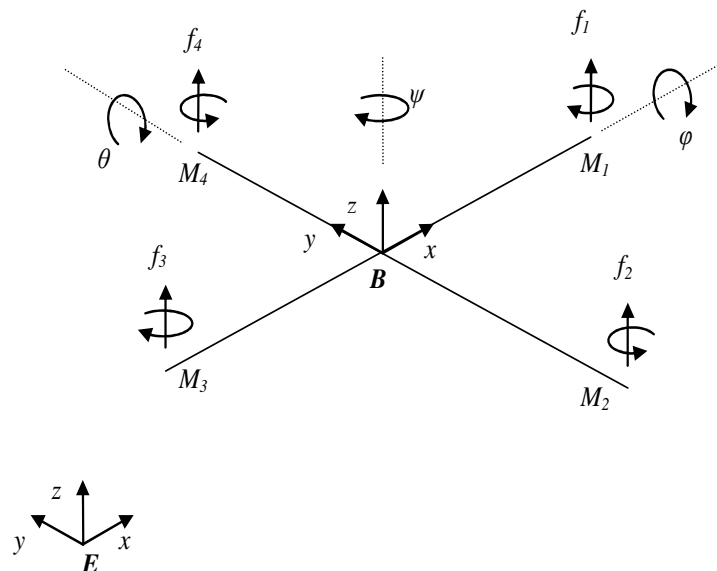


Figure VI.2 Schéma général de la structure du quadrotor

VI.3.1 La Cinématique du Quadrotor:

Le quadrotor en tant que corps rigide est caractérisé par un repère qui est lié à lui (B : body), et le repère inertiel (E : earth) est considéré comme fixe par rapport à la terre.

L'orientation entre les repères est donnée par la matrice de rotation (R_B^E), construite par les matrices de transformation des 3 angles (Figure VI.3): selon x (roulis: φ), selon y (tangage: θ), et selon z (lacet: ψ)

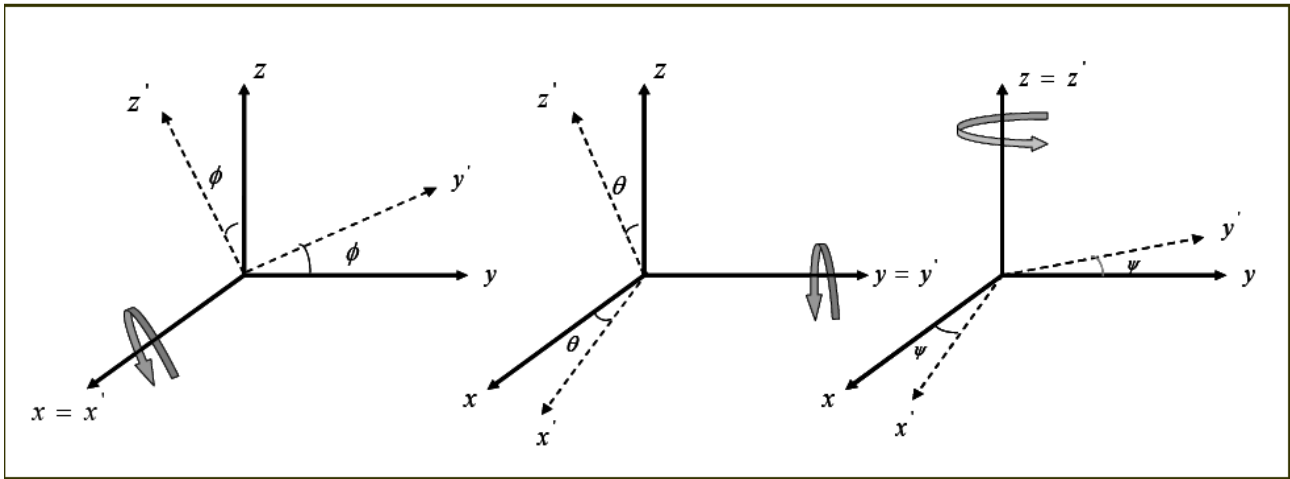


Figure VI.3 Les trois rotations d'Euler roulis φ , tangage θ et lacet ψ

Les matrices de rotation:

$$R_B^{v2}(x, \varphi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi \\ 0 & \sin \varphi & \cos \varphi \end{pmatrix} R_{v2}^{v1}(y, \theta) = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix}$$

$$R_{v1}^E(z, \psi) = \begin{pmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$v1$: l'axe sur y , $v2$: l'axe sur x où $v1$ est orthogonal avec $v2$.

$$R_B^E = R_{v1}^E(z, \psi) \cdot R_{v2}^{v1}(y, \theta) \cdot R_B^{v2}(x, \varphi)$$

$$= \begin{pmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi \\ 0 & \sin \varphi & \cos \varphi \end{pmatrix}$$

$$R_B^E = \begin{pmatrix} \cos\theta\cos\psi & \sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi \\ \cos\theta\sin\psi & \cos\varphi\cos\psi + \sin\varphi\sin\theta\sin\psi & \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi \\ -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{pmatrix} \quad (\text{VI.10})$$

Les vitesses angulaires:

$\Omega = \begin{pmatrix} p \\ q \\ r \end{pmatrix}$: Les vitesses angulaires dans le repère B .

$\dot{\eta} = \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}$: les vitesses angulaires dans E .

$$\Omega = \begin{pmatrix} p \\ q \\ r \end{pmatrix} = R_{v_2}^B(\varphi) \cdot \begin{pmatrix} \dot{\varphi} \\ 0 \\ 0 \end{pmatrix} + R_{v_2}^B(\varphi) \cdot R_{v_1}^{v_2}(\dot{\theta}) \cdot \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + R_{v_2}^B(\varphi) \cdot R_{v_1}^{v_2}(\theta) \cdot R_E^{v_1}(\dot{\psi}) \cdot \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix}$$

$\dot{\varphi}, \dot{\theta}$ et $\dot{\psi}$ très petits alors: $R_{v_2}^B(\varphi) = R_{v_1}^{v_2}(\dot{\theta}) = R_E^{v_1}(\dot{\psi}) = I$

$$\begin{aligned} \Omega = \begin{pmatrix} p \\ q \\ r \end{pmatrix} &= \begin{pmatrix} \dot{\varphi} \\ 0 \\ 0 \end{pmatrix} + R_{v_2}^B(\varphi) \cdot \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + R_{v_2}^B(\varphi) \cdot R_{v_1}^{v_2}(\theta) \cdot \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} \\ &= \begin{pmatrix} \dot{\varphi} \\ 0 \\ 0 \end{pmatrix} + (R_B^{v_2}(\varphi))^{-1} \cdot \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + (R_B^{v_2}(\varphi))^{-1} \cdot (R_{v_2}^{v_1}(\gamma, \theta))^{-1} \cdot \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} \\ &= \begin{pmatrix} \dot{\varphi} \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{pmatrix} \cdot \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} \end{aligned}$$

$$\Omega = \begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} \dot{\varphi} - \dot{\psi}\sin\theta \\ \dot{\theta}\cos\varphi + \dot{\psi}\sin\varphi\cos\theta \\ \dot{\psi}\cos\varphi\cos\theta - \dot{\theta}\sin\varphi \end{pmatrix} = W_\eta \dot{\eta} \quad (\text{VI.11})$$

Où:

$$W_\eta = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\varphi & \sin\varphi\cos\theta \\ 0 & -\sin\varphi & \cos\varphi\cos\theta \end{pmatrix} \quad (\text{VI.12})$$

Le vecteur $\xi = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ présente la position du centre de masse du quadrotor dans le repère inertiel E ,

Le vecteur $v = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}$ présente la vitesse du quadrotor dans le repère inertiel E .

Et V présente la vitesse du quadrotor dans le repère fixé dans le corps B .

Les équations cinématiques du mouvement que ce soit rotationnel ou translationnel sont obtenues par la matrice de rotation R_B^E , et la cinématique translationnelle peut être écrite comme:

$$v = R_B^E V \quad (\text{VI.13})$$

$$\begin{aligned} \dot{\eta} &= W_\eta^{-1} \Omega \\ \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} &= \begin{pmatrix} 1 & \sin\phi \tan\theta & \cos\phi \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \end{aligned} \quad (\text{VI.14})$$

VI.3.2 Les Équations d'Euler-Lagrange:

Les équations de mouvement du quadrotor peuvent être exprimées par la formule d'Euler-Lagrange basé sur le concept d'énergie cinétique et potentielle.

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} &= \Gamma_i \\ L &= E_c - E_p \end{aligned} \quad (\text{VI.15})$$

Où L est le lagrangien, E_c est l'énergie cinétique totale, E_p est l'énergie potentielle, q_i est la coordonnée généralisée et Γ_i sont les forces/couples généralisées données par les forces/couples non-conservées.

Les coordonnées généralisées pour un corps rigide tournant dans l'espace tridimensionnel (Carrillo et al., 2013) peuvent être écrit comme:

$$q = [x \ y \ z \ \phi \ \theta \ \psi]^T \quad (\text{VI.16})$$

L'expression langrangienne du quadrotor est donnée par:

$$L = E_{c_{Trans}} + E_{c_{Rot}} - E_p \quad (\text{VI.17})$$

Où, $E_{c_{Trans}}$ est l'énergie cinétique translationnelle, $E_{c_{Rot}}$ est l'énergie cinétique rotationnelle, et E_p est l'énergie potentielle.

Tout d'abord, le terme d'énergie de translation est développé à la connaissance de chaque vitesse de coordonnée généralisée. La vitesse linéaire est donnée par $v = \dot{\xi}$ et la vitesse quadratique est $\dot{\xi}^2 = \dot{\xi}^T \dot{\xi} = (\dot{x}^2 + \dot{y}^2 + \dot{z}^2)$. Alors, l'énergie cinétique de translation peut s'écrire comme suit:

$$E_{c_{Trans}} = \frac{m}{2} \dot{\xi}^T \dot{\xi} \quad (VI.18)$$

À partir de l'hypothèse que le quadrotor est supposé symétrique, la matrice d'inertie devient diagonale:

$$J = \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix} \quad (VI.19)$$

Soit $E_{c_{Rot}}$ l'énergie cinétique rotationnelle dans le repère B , qui est donnée par:

$$\begin{aligned} E_{c_{Rot}} &= \frac{1}{2} \Omega^T J \Omega \\ E_{c_{Rot}} &= \frac{1}{2} I_x p^2 + \frac{1}{2} I_y q^2 + \frac{1}{2} I_z r^2 \\ E_{c_{Rot}} &= \frac{1}{2} I_x (\dot{\phi} - \dot{\psi} \sin \theta)^2 + \frac{1}{2} I_y (\dot{\theta} \cos \varphi + \dot{\psi} \sin \varphi \cos \theta)^2 + \frac{1}{2} I_z (\dot{\psi} \cos \varphi \cos \theta - \dot{\theta} \sin \varphi)^2 \end{aligned} \quad (VI.20)$$

À partir de la définition de W_η depuis (VI.11) et (VI.12), nous posons la matrice $M(\eta)$ comme ceci:

$$M(\eta) = W_\eta^T J W_\eta \quad (VI.21)$$

Alors, l'énergie cinétique de rotation peut être réécrite à partir de la fonction des coordonnées généralisées η comme ceci:

$$E_{c_{Rot}} = \frac{1}{2} \dot{\eta}^T M(\eta) \dot{\eta} \quad (VI.22)$$

L'énergie potentielle E_p est donnée par:

$$E_p = mgz \quad (VI.23)$$

L'équation de mouvement complète est obtenue de l'expression lagrangienne (VI.15), telle que:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \begin{bmatrix} F_\xi \\ \tau_\eta \end{bmatrix} \quad (VI.24)$$

Où, τ_η représente les couples de roulis, et tangage et de lacet:

$$\tau_\eta = \begin{pmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} \quad (VI.25)$$

La force F_ξ est la force appliquée au quadrotor en raison de l'entrée de commande principale F_z dans l'axe z , avec:

$$F_B = \begin{pmatrix} 0 \\ 0 \\ F_z \end{pmatrix} \quad (\text{VI.26})$$

$$F_\xi = R_B^E F_B \quad (\text{VI.27})$$

Les équations d'Euler-Lagrange peuvent être divisées en dynamiques de translation et de rotation, nous posons alors $e_z = [0 \ 0 \ 1]^T$ étant les équations de mouvement de translation ξ :

$$m\ddot{\xi} + mge_z = F_\xi \quad (\text{VI.28})$$

Les équations d'Euler-Lagrange pour les coordonnées de rotation η , écrit sous forme générale (Carrillo et al., 2013):

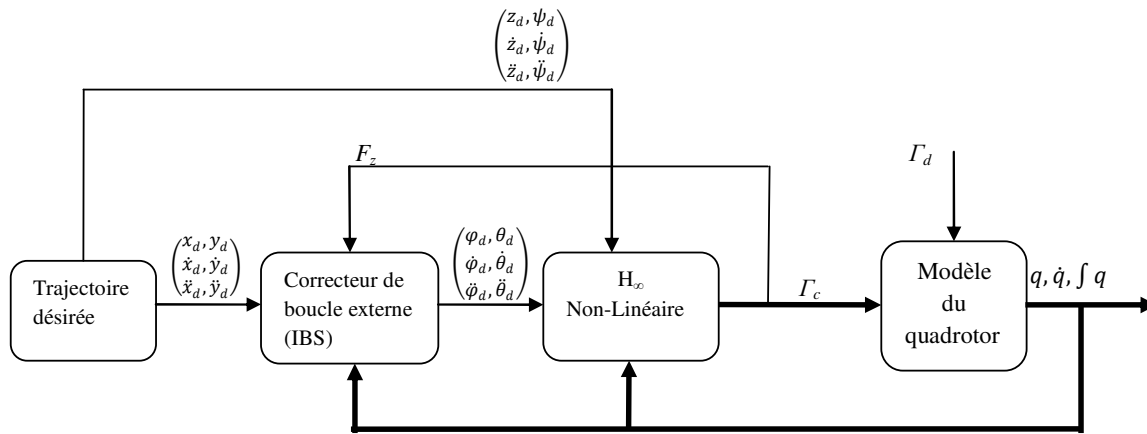
$$M(\eta)\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} = \tau_\eta \quad (\text{VI.29})$$

Alors, le modèle mathématique décrivant le mouvement de rotation du quadrotor obtenu à partir de la formule d'Euler-Lagrange est donnée par:

$$\ddot{\eta} = M(\eta)^{-1}(\tau_\eta - C(\eta, \dot{\eta})\dot{\eta}) \quad (\text{VI.30})$$

VI.4 Commande H_∞ Non-Linéaire du Quadrotor:

Parce que le quadrotor est un robot sous-actionné, la méthode utilisée dans ce cas est la commande H_∞ non linéaire appropriée aux systèmes sous-actionnés où nous développons les commandes réelles (les commandes des degrés de liberté actionnés) par leur couplage avec les degrés de liberté sous-actionnés. Les commandes virtuelles (les commandes des degrés de liberté sous-actionnés) seront développées par Backstepping intégral. La structure de commande est présentée par la figure suivante:


 Figure VI.4 La structure de commande H_∞ non-linéaire utilisée

VI.4.1 Transformation de Modèle pour le Robot:

Après la modélisation, nous allons mettre le modèle du quadrotor sous sa forme générale. Premièrement, nous allons indiquer les degrés de liberté contrôlés (*controlled*) par $q_c = [z \ \varphi \ \theta \ \psi]^T$, et les degrés de liberté sous-actionnés (*underactuated*) par $q_u = [x \ y]^T$.

Le vecteur des forces/couples appliqués sont la force de poussée (F_z) et les trois couples des angles $\Gamma_c = [F_z \ \tau_\varphi \ \tau_\theta \ \tau_\psi]^T$, et bien sûr les forces/couples sous-actionnés sont nulles $\Gamma_u = [0 \ 0]^T$.

On note que Γ_{ud} et Γ_{cd} indiquent les perturbations. Le modèle complet du quadrotor est donné par:

$$\begin{bmatrix} \Gamma_u \\ \Gamma_c \end{bmatrix} + \begin{bmatrix} \Gamma_{ud} \\ \Gamma_{cd} \end{bmatrix} = \begin{bmatrix} \bar{M}_{uu}(q) & \bar{M}_{uc}(q) \\ \bar{M}_{cu}(q) & \bar{M}_{cc}(q) \end{bmatrix} \begin{bmatrix} \ddot{q}_u \\ \ddot{q}_c \end{bmatrix} + \begin{bmatrix} \bar{C}_{uu}(q, \dot{q}) & \bar{C}_{uc}(q, \dot{q}) \\ \bar{C}_{cu}(q, \dot{q}) & \bar{C}_{cc}(q, \dot{q}) \end{bmatrix} \begin{bmatrix} \dot{q}_u \\ \dot{q}_c \end{bmatrix} + \begin{bmatrix} \bar{G}_u(q) \\ \bar{G}_c(q) \end{bmatrix} \quad (\text{VI.31})$$

En raison que nous allons de commander le quadrotor par robustesse, nous considérons les incertitudes paramétriques des matrices dynamiques, où chaque matrice \bar{A} peut s'écrire:

$$\bar{A} = A + \Delta A \quad (\text{VI.32})$$

Où:

$$M_{uu}(q) = \begin{bmatrix} mC\theta C\psi & mC\theta S\psi \\ m(S\varphi S\theta C\psi - C\varphi S\psi) & m(S\varphi S\theta S\psi + C\varphi C\psi) \end{bmatrix}$$

$$M_{uc}(q) = \begin{bmatrix} -mS\theta & 0_{1 \times 3} \\ mS\varphi C\theta & 0_{1 \times 3} \end{bmatrix}$$

$$M_{cu}(q) = \begin{bmatrix} m(C\varphi S\theta C\psi + S\varphi S\psi) & m(C\varphi S\theta S\psi - S\varphi C\psi) \\ 0_{3 \times 1} & 0_{3 \times 1} \end{bmatrix}$$

$$M_{cc}(q) = \begin{bmatrix} mC\varphi C\theta & 0_{1 \times 3} \\ 0_{3 \times 1} & M(\eta) \end{bmatrix}$$

$$M(\eta) = \begin{bmatrix} I_x & 0 & -I_x S\theta \\ 0 & I_y C^2\varphi + I_z S^2\varphi & (I_y - I_z)C\varphi S\varphi C\theta \\ -I_x S\theta & (I_y - I_z)C\varphi S\varphi C\theta & I_x S^2\theta + I_y S^2\varphi C^2\theta + I_z C^2\varphi C^2\theta \end{bmatrix}$$

$$C_{uu}(q, \dot{q}) = 0_{2 \times 2}$$

$$C_{uc}(q, \dot{q}) = 0_{2 \times 4}$$

$$C_{cu}(q, \dot{q}) = 0_{4 \times 2}$$

$$C_{cc}(q, \dot{q}) = \begin{bmatrix} 0 & 0_{1 \times 3} \\ 0_{3 \times 1} & C(\eta, \dot{\eta}) \end{bmatrix}$$

$$C(\eta, \dot{\eta}) = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$$

$$c_{11} = 0$$

$$c_{12} = (I_y - I_z)(\dot{\theta}C\varphi S\varphi + \dot{\psi}S^2\varphi C\theta) + (I_z - I_y)\dot{\psi}C^2\varphi C\theta - I_x\dot{\psi}C\theta$$

$$c_{13} = (I_z - I_y)\dot{\psi}C\varphi S\varphi C^2\theta$$

$$c_{21} = (I_z - I_y)(\dot{\theta}C\varphi S\varphi + \dot{\psi}S^2\varphi C\theta) + (I_y - I_z)\dot{\psi}C^2\varphi C\theta + I_x\dot{\psi}C\theta$$

$$c_{22} = (I_z - I_y)\dot{\psi}C\varphi S\varphi$$

$$c_{23} = -I_x\dot{\psi}C\theta S\theta + I_y\dot{\psi}S^2\varphi C\theta S\theta + I_z\dot{\psi}C^2\varphi S\theta C\theta$$

$$c_{31} = (I_y - I_z)\dot{\psi}C^2\theta C\varphi S\varphi - I_x\dot{\theta}C\theta$$

$$c_{32} = (I_z - I_y)(\dot{\theta}C\varphi S\varphi S\theta + \dot{\psi}S^2\varphi C\theta) + (I_y - I_z)\dot{\psi}C^2\varphi C\theta + I_x\dot{\psi}C\theta S\theta - I_y\dot{\psi}S^2\varphi C\theta S\theta - I_z\dot{\psi}C^2\varphi C\theta S\theta$$

$$c_{33} = (I_y - I_z)\dot{\psi}C\varphi S\varphi C^2\theta - I_y\dot{\theta}S^2\varphi C\theta S\theta - I_z\dot{\theta}C^2\varphi C\theta S\theta + I_x\dot{\theta}C\theta S\theta$$

$$G_u(q) = \begin{bmatrix} -mgS\theta \\ mgS\varphi C\theta \end{bmatrix}$$

$$G_c(q) = \begin{bmatrix} mgC\varphi C\theta \\ 0_{3 \times 1} \end{bmatrix}$$

Les matrices des incertitudes sont:

$$\begin{aligned} \delta_u &= \Gamma_{ud} - (\Delta M_{uu}(q)\ddot{q}_u + \Delta M_{uc}(q)\ddot{q}_c + \Delta C_{uu}(q, \dot{q})\dot{q}_u + \Delta C_{uc}(q, \dot{q})\dot{q}_c + \Delta G_u(q)) \\ \delta_c &= \Gamma_{cd} - (\Delta M_{cu}(q)\ddot{q}_u + \Delta M_{cc}(q)\ddot{q}_c + \Delta C_{cu}(q, \dot{q})\dot{q}_u + \Delta C_{cc}(q, \dot{q})\dot{q}_c + \Delta G_c(q)) \end{aligned} \quad (\text{VI.33})$$

Avec prise en compte de la partition des degrés de liberté; contrôlés et sous-actionnés, et de la deuxième ligne de (VI.31):

$$\Gamma_c + \delta_c = M_{cu}(q)\ddot{q}_u + M_{cc}(q)\ddot{q}_c + C_{cu}(q, \dot{q})\dot{q}_u + C_{cc}(q, \dot{q})\dot{q}_c + G_c(q) \quad (\text{VI.34})$$

L'accélération des degrés de liberté contrôlés peut être isolée (Siqueira et al., 2011):

$$\ddot{q}_c = -M_{cc}^{-1}(q)(C_{cc}(q, \dot{q})\dot{q}_c + G_c(q) - \bar{\Gamma}_c - \delta_c) \quad (\text{VI.35})$$

Avec:

$$\bar{\Gamma}_c = \Gamma_c - M_{cu}(q)\ddot{q}_u - C_{cu}(q, \dot{q})\dot{q}_u \quad (\text{VI.36})$$

En définissant le vecteur des erreurs des degrés de libertés contrôlés comme suit:

$$x_c = \begin{bmatrix} \ddot{\tilde{q}}_c \\ \dot{\tilde{q}}_c \\ \int \tilde{q}_c dt \end{bmatrix} = \begin{bmatrix} \dot{q}_c - \dot{q}_c^d \\ q_c - q_c^d \\ \int (q_c - q_c^d) dt \end{bmatrix} \quad (\text{VI.37})$$

L'équation (VI.35) peut être réécrite sous forme de représentation d'état:

$$\dot{x}_c = \bar{f}(x_c, t) + \bar{g}(x_c, t)\bar{\Gamma}_c + \bar{k}(x_c, t)\delta_c + g_0(q, \dot{q}, \dot{q}_c^d, \ddot{q}_c^d) \quad (\text{VI.38})$$

Où:

$$\bar{f}(x_c, t) = \begin{bmatrix} -M_{cc}^{-1}(q)C_{cc}(q, \dot{q}) & \mathbb{0} & \mathbb{0} \\ \mathbb{1} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & \mathbb{1} & \mathbb{0} \end{bmatrix} x_c$$

$$\bar{g}(x_c, t) = \bar{k}(x_c, t) = \begin{bmatrix} M_{cc}^{-1}(q) \\ \mathbb{0} \\ \mathbb{0} \end{bmatrix}$$

$$g_0(q, \dot{q}, \dot{q}_c^d, \ddot{q}_c^d) = \begin{bmatrix} -M_{cc}^{-1}(q)(M_{cc}(q)\ddot{q}_c^d + C_{cc}(q, \dot{q})\dot{q}_c^d + G_c(q)) \\ \mathbb{0} \\ \mathbb{0} \end{bmatrix}$$

Où q_c^d , \dot{q}_c^d et $\ddot{q}_c^d \in \mathbb{R}^4$ sont la trajectoire désirée et la vitesse et l'accélération correspondantes, respectivement. $\mathbb{1}$ est la matrice identité et $\mathbb{0}$ la matrice de zéro. Notez que le terme intégral a été inclus dans le vecteur d'erreur, celui-ci permettra d'obtenir une erreur nulle d'état stable lorsque des perturbations agissent sur le système (Raffo et al., 2011; 2015).

Le correcteur H_∞ non-linéaire est utilisé dans la boucle interne et prend en compte le comportement dynamique global de quadrotor.

Lors de l'étape précédente de synthèse de la loi de commande, la transformation d'état suivante est utilisée:

$$z = T_0 x_c = \begin{bmatrix} T_1 & T_2 & T_3 \\ \mathbb{0} & \mathbb{1} & \mathbb{1} \\ \mathbb{0} & \mathbb{0} & \mathbb{1} \end{bmatrix} \begin{bmatrix} \dot{\tilde{q}}_c \\ \tilde{q}_c \\ \int \tilde{q}_c dt \end{bmatrix} \quad (\text{VI.39})$$

Avec T_1 , T_2 et T_3 sont des matrices positives constantes carrées.

$$\begin{aligned} z_1 &= T_1 \dot{\tilde{q}}_c + T_2 \tilde{q}_c + T_3 \int \tilde{q}_c dt \\ z_2 &= \tilde{q}_c + \int \tilde{q}_c dt \\ z_3 &= \int \tilde{q}_c dt \end{aligned} \quad (\text{VI.40})$$

Pour obtenir la représentation d'état, nous avons:

$$\begin{aligned} \dot{z}_1 &= T_1 \ddot{\tilde{q}}_c + T_2 \dot{\tilde{q}}_c + T_3 \tilde{q}_c \\ \dot{z}_2 &= \dot{\tilde{q}}_c + \tilde{q}_c \\ \dot{z}_3 &= \tilde{q}_c \end{aligned} \quad (\text{VI.41})$$

De (VI.40) et z_2 on a:

$$\dot{z}_3 = z_2 - z_3 \quad (\text{VI.42})$$

De (VI.40) et z_1 on a:

$$\ddot{\tilde{q}}_c = T_1^{-1} z_1 - T_1^{-1} T_2 (z_2 - z_3) - T_1^{-1} T_3 z_3 \quad (\text{VI.43})$$

Et alors:

$$\dot{z}_2 = T_1^{-1} z_1 + (\mathbb{1} - T_1^{-1} T_2) z_2 + (T_1^{-1} (T_2 - T_3) - \mathbb{1}) z_3 \quad (\text{VI.44})$$

De (VI.35) on a:

$$\ddot{\tilde{q}}_c = -M_{cc}^{-1}(q) (C_{cc}(q, \dot{q}) (\dot{\tilde{q}}_c + \dot{q}_c^d) + G_c(q)) - \ddot{q}_c^d + M_{cc}^{-1}(q) \bar{\Gamma}_c + M_{cc}^{-1}(q) T_1 \delta_c \quad (\text{VI.45})$$

Et alors:

$$\begin{aligned} \dot{z}_1 &= -M_{cc}^{-1}(q) C_{cc}(q, \dot{q}) T_1 \dot{\tilde{q}}_c + M_{cc}^{-1}(q) C_{cc}(q, \dot{q}) \left(T_2 (\dot{\tilde{q}}_c - \dot{q}_c^d) + T_3 \left(\int \tilde{q}_c dt - \int \tilde{q}_c dt \right) \right) - T_1 \ddot{q}_c^d \\ &\quad - M_{cc}^{-1}(q) T_1 \left(C_{cc}(q, \dot{q}) \dot{q}_c^d + G_c(q) \right) + M_{cc}^{-1}(q) T_1 \bar{\Gamma}_c + T_2 \dot{\tilde{q}}_c + T_3 \tilde{q}_c + M_{cc}^{-1}(q) T_1 \delta_c \\ \dot{z}_1 &= -M_{cc}^{-1}(q) C_{cc}(q, \dot{q}) z_1 + M_{cc}^{-1}(q) T_1 (\bar{\Gamma}_c - F(x_e)) + M_{cc}^{-1}(q) T_1 \delta_c \end{aligned} \quad (\text{VI.46})$$

Où:

$$F(x_e) = M_{cc}(q) (\ddot{q}_c^d - T_1^{-1} T_2 \dot{\tilde{q}}_c - T_1^{-1} T_3 \tilde{q}_c) + C_{cc}(q, \dot{q}) (\dot{q}_c^d - T_1^{-1} T_2 \dot{\tilde{q}}_c - T_1^{-1} T_3 \int \tilde{q}_c dt) + G_c(q) \quad (\text{VI.47})$$

Avec $x_e = (\tilde{q}_c, \dot{\tilde{q}}_c, \int \tilde{q}_c dt, \dot{q}_c^d, \ddot{q}_c^d)$.

La représentation d'état (VI.38) peut être écrite sous la forme:

$$\dot{x}_c = f(x_c, t) + g(x_c, t) T_1 (\bar{\Gamma}_c - F(x_e)) + k(x_c, t) w \quad (\text{VI.48})$$

Avec:

$$f(x_c) = T_0^{-1} \begin{bmatrix} -M_{cc}^{-1}(q)C_{cc}(q, \dot{q}) & \mathbb{0} & \mathbb{0} \\ T_1^{-1} & \mathbb{1} - T_1^{-1}T_2 & T_1^{-1}(T_2 - T_3) - \mathbb{1} \\ \mathbb{0} & \mathbb{1} & -\mathbb{1} \end{bmatrix} T_0 x_c$$

$$g(x_c) = h(x_c) = T_0^{-1} \begin{bmatrix} M_{cc}^{-1}(q) \\ \mathbb{0} \\ \mathbb{0} \end{bmatrix}$$

Où $w = T_1 \delta_c$, le vecteur des perturbations inclus les entrées externes et les perturbations.

En considérant la commande:

$$u = T_1(\bar{\Gamma}_c - F(x_c)) \quad (\text{VI.49})$$

Par conséquent, la commande u est utilisée dans la représentation d'état pour appliquer les résultats théoriques de l' H_∞ non linéaire présentés dans (Van der Schaft, 2000; Ortega et al., 2005), la représentation d'état est alors:

$$\dot{x}_c = f(x_c, t) + g(x_c, t)u + k(x_c, t)w \quad (\text{VI.50})$$

VI.4.2 H_∞ Non-Linéaire:

La performance du système commandé peut être dérivée de la variable du critère suivant $\xi \in \mathbb{R}^8$:

$$\xi = W \begin{bmatrix} h(x_c) \\ u \end{bmatrix} \quad (\text{VI.51})$$

Où $h(x_c) \in \mathbb{R}^4$ représente une fonction du vecteur des états à commander, et $W \in \mathbb{R}^{8 \times 8}$ est la matrice de pondération. En prenant en compte la représentation d'état (VI.50) et la définition de la variable du critère, et si les états x_c sont supposés être valables pour la mesure, alors le problème de la commande H_∞ non-linéaire peut être posé comme suit (Van der Schaft, 2000):

Trouver la plus petite valeur $\gamma^ > 0$ telle que; pour tout $\gamma \geq \gamma^*$, il existe une commande u , tel que le gain L_2 des signaux de perturbations w à la variable $\xi = W[h^T(x_c) \quad u^T]^T$ est inférieur ou égal à un niveau d'atténuation donné γ :*

$$\int_0^T \|\xi\|_2^2 dt \leq \gamma^2 \int_0^T \|w\|_2^2 dt \quad (\text{VI.52})$$

Le terme intégral sur le coté gauche de l'inégalité (VI.52) peut être écrit comme:

$$\|\xi\|_2^2 = \xi^T \xi = [h(x_c)^T \quad u^T] W^T W \begin{bmatrix} h(x_c) \\ u \end{bmatrix} \quad (\text{VI.53})$$

La matrice symétrique définie positive $W^T W$ est écrite comme:

$$W^T W = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \quad (\text{VI.54})$$

Les matrices Q et R sont symétriques défini-positives, le fait que $W^T W > 0$ garantie que $Q - SR^{-1}S^T > \mathbb{0}$. En prenant en compte la définition du vecteur d'erreur x_c , les structures suivantes sont considérées pour les matrices Q et S dans (VI.54).

$$Q = \begin{bmatrix} Q_1 & Q_{12} & Q_{13} \\ Q_{12} & Q_2 & Q_{23} \\ Q_{13} & Q_{23} & Q_3 \end{bmatrix}, S = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix} \quad (\text{VI.55})$$

Sous ces hypothèses, une commande optimale à retour d'état u^* est calculée si une solution $V(x_c, t)$, avec $x_{c0}=0$ et $V(x_{c0}, t)=0$, est trouvée pour l'équation HJBI (Hamilton-Jacobi-Bellman-Isaacs) suivante (Van der Schaft, 2000):

$$\begin{aligned} \frac{\partial V}{\partial t} + \frac{\partial^T V}{\partial x_c} f(x_c, t) + \frac{1}{2} \frac{\partial^T V}{\partial x_c} \left[\frac{1}{\gamma^2} k(x_c, t) k^T(x_c, t) - g(x_c, t) R^{-1} g^T(x_c, t) \right] \frac{\partial V}{\partial x_c} - \\ \frac{\partial^T V}{\partial x_c} g(x_c, t) R^{-1} S^T h(x_c) + \frac{1}{2} h^T(x_c) (Q - SR^{-1}S^T) h(x_c) = 0 \end{aligned} \quad (\text{VI.56})$$

Pour chaque $\gamma > \sqrt{\sigma_{\max}(R)} \geq 0$, où σ_{\max} symbolise la valeur singulière maximale. La commande à retour d'état optimale est dérivée comme suit (Feng et Postlethwaite, 1994):

$$u^* = -R^{-1} \left(S^T h(x_c) + g^T(x_c, t) \frac{\partial V}{\partial x_c} \right) \quad (\text{VI.57})$$

Comme indiqué précédemment, la solution de l'équation HJBI dépend du choix de la variable du critère, ζ , et en particulier du choix de la fonction $h(x_c)$. Dans notre cas, nous considérons que cette fonction est égale au vecteur d'erreur, c'est-à-dire $h(x_c)=x_c$. Une fois cette fonction sélectionnée, le calcul de la commande u^* nécessitera de trouver la solution V de l'équation HJBI (Van der Schaft, 2000). Les détails de la manière dont cette solution est réalisée peuvent être trouvés dans (Ortega et al., 2005). Le théorème suivant aidera à faire cela.

Théorème: soit la fonction scalaire V :

$$V = \frac{1}{2} x_c^T T_0^T \begin{bmatrix} M_{cc}(q) & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & Y & X - Y \\ \mathbb{0} & X - Y & Z + Y \end{bmatrix} T_0 x_c \quad (\text{VI.58})$$

Où X, Y , et $Z \in \mathbb{R}^{4 \times 4}$ sont des matrices constantes, symétriques et défini positive tel que $Z - XY^{-1}X + 2X > \mathbb{0}$, la matrice T_0 est défini dans (VI.39). Soit la matrice T :

$$T = [T_1 \quad T_2 \quad T_3] \quad (\text{VI.59})$$

Si ces matrices vérifient l'équation suivante:

$$\begin{bmatrix} \mathbb{0} & Y & X \\ Y & 2X & Z + 2X \\ X & Z + 2X & \mathbb{0} \end{bmatrix} + Q + \frac{1}{\gamma^2} T^T T - (S^T + T)^T R^{-1} (S^T + T) = \mathbb{0} \quad (\text{VI.60})$$

Alors la fonction V constitue une solution de l'équation de HJBI.

La preuve de ce théorème est trouvée dans (Ortega et al., 2005).

Une fois la matrice T est calculée par résolution de certaines équations de Riccati (Ortega et al., 2005), en remplaçant V dans (VI.57), la commande optimale u^* est:

$$u^* = -R^{-1}(S^T + T)x_c \quad (\text{VI.61})$$

VI.4.3 Calcul des Forces/Couples:

Maintenant, nous allons calculer les forces/couples appliquées sur le quadrotor. À partir de (VI.47) et (VI.49), nous pouvons extraire le vecteur des forces/couples \bar{F}_c :

$$\bar{F}_c = F(x_e) + T_1^{-1}u \quad (\text{VI.62})$$

Après quelques modifications, nous avons:

$$\begin{aligned} \bar{F}_c &= M_{cc}(q)\ddot{q}_c^d + M_{cc}(q)\ddot{\tilde{q}}_c - M_{cc}(q)\ddot{\tilde{q}}_c - M_{cc}(q)T_1^{-1}T_2\ddot{\tilde{q}}_c - M_{cc}(q)T_1^{-1}T_3\ddot{\tilde{q}}_c + C_{cc}(q, \dot{q})\dot{q}_c^d \\ &\quad + C_{cc}(q, \dot{q})\dot{\tilde{q}}_c - C_{cc}(q, \dot{q})\dot{\tilde{q}}_c - C_{cc}(q, \dot{q})T_1^{-1}T_2\dot{\tilde{q}}_c - C_{cc}(q, \dot{q})T_1^{-1}T_3 \int \tilde{q}_c dt + G_c(q) \\ &\quad + T_1^{-1}u \\ \bar{F}_c &= M_{cc}(q)\ddot{q}_c + C_{cc}(q, \dot{q})\dot{q}_c + G_c(q) - M_{cc}(q)T_1^{-1}(T_1\ddot{\tilde{q}}_c + T_2\ddot{\tilde{q}}_c + T_3\ddot{\tilde{q}}_c) - C_{cc}(q, \dot{q})T_1^{-1}(T_1\dot{\tilde{q}}_c \\ &\quad + T_2\dot{\tilde{q}}_c + T_3 \int \tilde{q}_c dt) + T_1^{-1}u \end{aligned}$$

Le vecteur des forces/couples appliquées est:

$$\bar{F}_c = M_{cc}(q)\ddot{q}_c + C_{cc}(q, \dot{q})\dot{q}_c + G_c(q) - T_1^{-1}(M_{cc}(q)T\dot{x}_c + C_{cc}(q, \dot{q})Tx_c) + T_1^{-1}u \quad (\text{VI.63})$$

Cette loi de commande est divisée en trois parties différentes; la première consiste en les trois premiers termes de cette équation, conçus pour contrer les effets de la dynamique du système. La deuxième partie comprend des termes comprenant le vecteur d'erreur x_c et ses dérivées. Ces deux termes de la loi de commande permettent un suivi parfait, ce qui signifie qu'ils représentent l'effort

de la commande essentielle nécessaire à la réalisation. Enfin, la troisième partie comprend le vecteur u , qui représente l'effort de la commande supplémentaire nécessaire au rejet des perturbations. Nous l'appliquons dans (VI.35):

$$M_{cc}(q)T\dot{x}_c + C_{cc}(q, \dot{q})Tx_c = u + w \quad (\text{VI.64})$$

Où $w = T_1\delta_c$. Afin de calculer les forces/couples par la méthode *couple calculée*, nous allons extraire la commande d'accélération \ddot{q}_c :

$$\ddot{q}_c = \ddot{q}_c^d - K_D\dot{\tilde{q}}_c - K_P\tilde{q}_c - K_I \int \tilde{q}_c dt \quad (\text{VI.65})$$

Avec:

$$\begin{aligned} K_D &= T_1^{-1}(T_2 + M_{cc}^{-1}(q)C_{cc}(q, \dot{q})T_1 + M_{cc}^{-1}(q)R^{-1}(S_1 + T_1)) \\ K_P &= T_1^{-1}(T_3 + M_{cc}^{-1}(q)C_{cc}(q, \dot{q})T_2 + M_{cc}^{-1}(q)R^{-1}(S_2 + T_2)) \\ K_I &= T_1^{-1}(M_{cc}^{-1}(q)C_{cc}(q, \dot{q})T_3 + M_{cc}^{-1}(q)R^{-1}(S_3 + T_3)) \end{aligned} \quad (\text{VI.66})$$

Un cas particulier est obtenu quand les composants de la matrice de pondération $W^T W$ vérifient:

$$Q = \begin{bmatrix} \omega_1^2 \mathbb{1} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & \omega_2^2 \mathbb{1} & \mathbb{0} \\ \mathbb{0} & \mathbb{0} & \omega_3^2 \mathbb{1} \end{bmatrix}, S = \begin{bmatrix} \mathbb{0} \\ \mathbb{0} \\ \mathbb{0} \end{bmatrix}, R = \omega_u^2 \mathbb{1} \quad (\text{VI.67})$$

Les solutions des équations de Riccati selon (Ortega et al., 2005):

$$T_1 = Q_1^{1/2} = \omega_1 \mathbb{1}, T_3 = Q_3^{1/2} = \omega_3 \mathbb{1}, T_2 = \sqrt{\omega_2^2 + 2\omega_1\omega_3} \mathbb{1} \quad (\text{VI.68})$$

Et les gains sont:

$$\begin{aligned} K_D &= \frac{\sqrt{\omega_2^2 + 2\omega_1\omega_3}}{\omega_1} \mathbb{1} + M_{cc}^{-1}(q) \left(C_{cc}(q, \dot{q}) + \frac{1}{\omega_u^2} \mathbb{1} \right) \\ K_P &= \frac{\omega_3}{\omega_1} \mathbb{1} + \frac{\sqrt{\omega_2^2 + 2\omega_1\omega_3}}{\omega_1} M_{cc}^{-1}(q) \left(C_{cc}(q, \dot{q}) + \frac{1}{\omega_u^2} \mathbb{1} \right) \\ K_I &= \frac{\omega_3}{\omega_1} M_{cc}^{-1}(q) \left(C_{cc}(q, \dot{q}) + \frac{1}{\omega_u^2} \mathbb{1} \right) \end{aligned} \quad (\text{VI.69})$$

Finalement, les commandes (forces/couples) appliquées au quadrotor peuvent être obtenues depuis (VI.36), nous avons $\Gamma_c = \bar{\Gamma}_c + M_{cu}(q)\ddot{q}_u + C_{cu}(q, \dot{q})\dot{q}_u$ et alors:

$$\Gamma_c = M_{cc}(q)\ddot{q}_c + C_{cc}(q, \dot{q})\dot{q}_c + G_c(q) + M_{cu}(q)\ddot{q}_u + C_{cu}(q, \dot{q})\dot{q}_u \quad (\text{VI.70})$$

VI.4.4 Résultats de Simulation:

Afin de tester la stratégie de commande, une simulation a été effectuée. Le quadrotor exécute un suivi d'une trajectoire. Les performances obtenues par cette stratégie ont été vérifiées en considérant un modèle plus précis qui imite un quadrotor. Afin de tester la robustesse de la stratégie vis-à-vis des incertitudes paramétriques, une fraction de $\pm 40\%$ a été considérée dans l'incertitude des éléments suivants; l'inertie selon x , l'inertie selon y , l'inertie selon z et la masse. En plus, des perturbations affectant tous les degrés de liberté ont été appliquées à différents instants pour vérifier le rejet des perturbations de la stratégie de commande utilisée.

Les valeurs nominales du quadrotor sont présentées dans le Tableau VI.1:

Tableau VI.1 Les valeurs nominales du quadrotor utilisé

Paramètre	Description	Valeur	Unité
m	Masse	0.74	Kg
l	Longueur du bras	0.21	m
I_x	Inertie de rotation selon l'axe x	$4 \cdot 10^{-3}$	Kg.m^2
I_y	Inertie de rotation selon l'axe y	$4 \cdot 10^{-3}$	Kg.m^2
I_z	Inertie de rotation selon l'axe z	$8.4 \cdot 10^{-3}$	Kg.m^2
g	Constante de gravité	9.8	m.s^{-2}

Initialement, le quadrotor commence de la position initiale $(x_0, y_0, z_0)=(4, 0, 0)$ et puis monte à 1 m dans l'axe z , à l'instant 10 s le quadrotor fait une trajectoire en spirale:

$$x_d = 4 \cos\left(\frac{t}{2}\right) \text{ m}, y_d = 4 \sin\left(\frac{t}{2}\right) \text{ m}, z_d = 1 + \left(\frac{t}{5}\right) \text{ m}, \psi_d = \frac{\pi}{3} \text{ rad.}$$

Les perturbations appliquées sont de forme échelon, en tant que couples et forces tels que:

$A_x=1 \text{ N}$ à l'instant $t=30 \text{ s}$; $A_y=1 \text{ N}$ à l'instant $t=40 \text{ s}$; $A_z=1 \text{ N}$ à l'instant $t=20 \text{ s}$; $A_\phi=1 \text{ N.m}$ à l'instant $t=25 \text{ s}$; $A_\theta=1 \text{ N.m}$ à l'instant $t=35 \text{ s}$; et $A_\psi=1 \text{ N.m}$ à l'instant $t=15 \text{ s}$.

Les paramètres du correcteur Backstepping intégral sont; $c_3=c_5=1.5$; $c_4=c_6=1$; $\lambda_2=\lambda_3=0.3$.

Nous remarquons que les paramètres du correcteur H_∞ non-linéaire sont basés sur le choix des gains de pondération. Les gains étaient réglés avec ces valeurs; $\omega_1=0.1$, $\omega_2=5$, $\omega_3=9$, $\omega_u=0.5$.

Dans cette simulation, il y a trois tests; le premier est réalisé avec les valeurs nominales des paramètres. Dans le deuxième test, premièrement on suppose que les paramètres étaient nominaux, puis ils seront nominaux à $+40\%$ à partir de l'instant 20 s . Dans le troisième test, premièrement

nous supposons que les paramètres étaient nominaux, puis ils seront nominaux à -40% à partir de l'instant 20 s.

La Figure VI.5 présente la simulation du suivi de la trajectoire en 3D, alors que la Figure VI.6 présente les réponses des degrés de liberté de translation (x, y, z) et la Figure VI.7 présente les réponses de degrés de liberté de rotation (φ, θ, ψ).

Les figures de Figure VI.5 à Figure VI.7, indiquent une excellente performance de suivi de référence même si les perturbations externes aux forces et couples sont considérés. Ces résultats illustrent le comportement robuste fourni par la structure de commande utilisée.

L'évolution temporelle de la position de translation est illustrée à la Figure VI.6. Cette action intégrale permet d'obtenir une erreur nulle dans le suivi de la trajectoire lorsque des perturbations affectent le mouvement x et y du quadrotor.

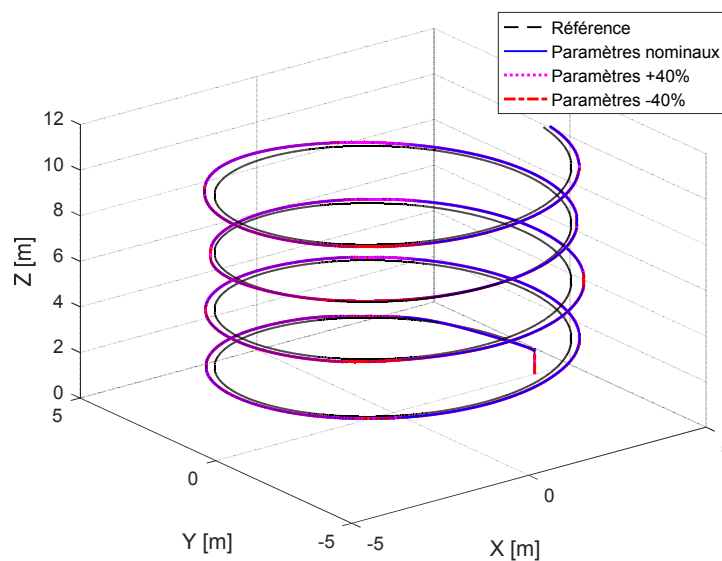


Figure VI.5 Suivi de la trajectoire en 3D (1^{ère} trajectoire)

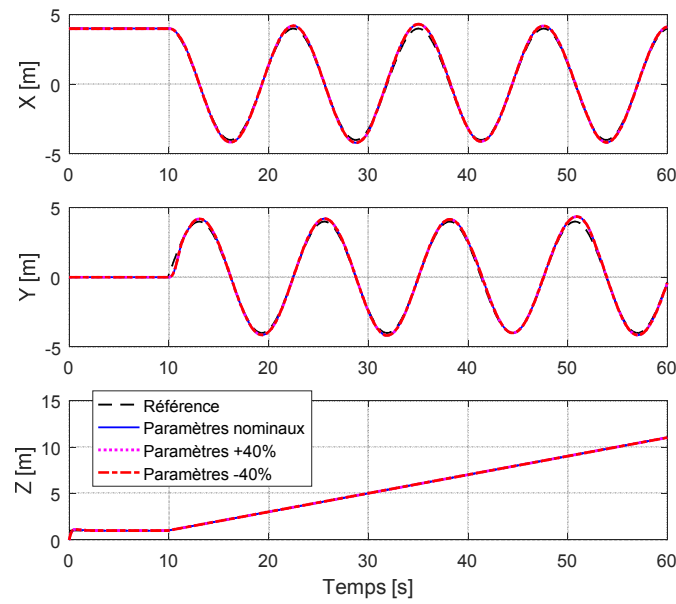


Figure VI.6 Réponses des coordonnées de la position (1^{ère} trajectoire)

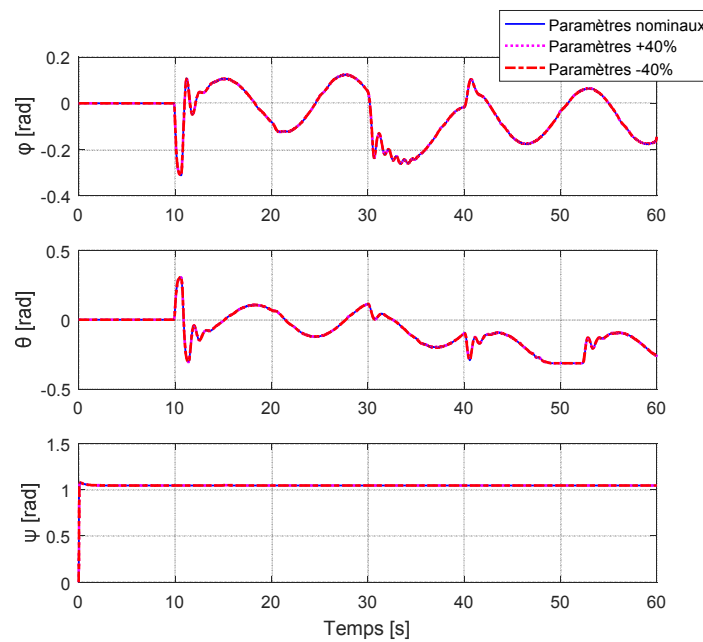


Figure VI.7 Réponses des coordonnées de rotation (1^{ère} trajectoire)

VI.5 Commande H_∞ Non-Linéaire Adaptative:

De manière générale, trois classes de stratégies de commande sont disponibles dans la littérature pour les robots. Ils sont classés en fonction du niveau de connaissance du concepteur sur le modèle dynamique du robot (Siqueira et al., 2011). Les stratégies de la première classe considèrent que le modèle mathématique et les valeurs des paramètres cinématiques et dynamiques sont bien connus et

disponibles pour le correcteur. Les incertitudes paramétriques sont traitées comme des perturbations agissant sur le système à éliminer par un correcteur robuste. Le correcteur présenté jusqu'ici dans ce chapitre appartient à cette classe. La deuxième classe de stratégies de commande considère que les expressions analytiques du modèle dynamique du robot sont connues, mais que les valeurs des paramètres utilisées dans la conception du correcteur sont imprécises. Dans ce cas, une loi de commande adaptative peut être utilisée pour estimer des paramètres incertains. La propriété de paramètre linéaire a été largement utilisée pour résoudre ce problème (Chen et al., 1997; Monfared et al., 2018); cela indique que le modèle dynamique des robots peut être exprimé comme le produit d'une matrice de régression, et d'un vecteur contenant les paramètres incertains. Ces paramètres sont mis à jour en ligne par une loi de commande basée sur les erreurs. Les correcteurs de la troisième classe considèrent que le modèle dynamique du robot et les valeurs de ses paramètres sont inconnus. Dans ce cas, des stratégies de commande telles que les réseaux de neurones, la logique floue et les algorithmes génétiques ont été utilisés pour estimer le modèle dynamique du robot (Siqueira et Terra, 2009).

Dans cette section, nous supposons que le modèle nominal du quadrotor est connu et nous estimons uniquement la partie incertaine de la dynamique grâce à des approches basées sur le paramétrage linéaire et les réseaux de neurones.

La structure de commande utilisée est présentée par la figure suivante:

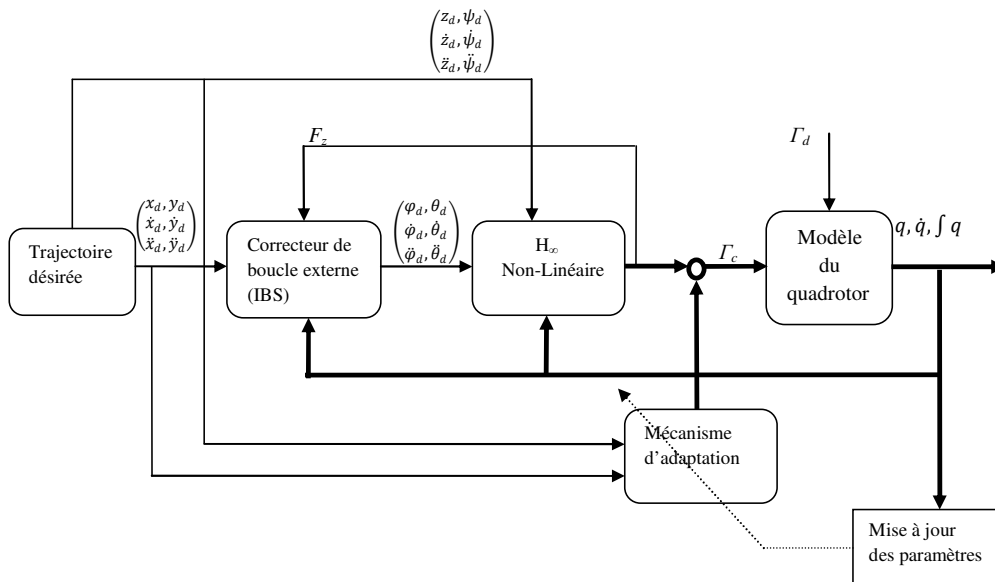


Figure VI.8 La structure de commande adaptative utilisée

VI.5.1 Adaptation par Paramétrage Linéaire:

Une des stratégies de commande adaptative les plus utilisées pour les robots prend l'avantage de la propriété de paramétrage linéaire du modèle dynamique du robot:

Propriété: On considère les matrices dynamiques d'un robot, $M(q)$, $C(q, \dot{q})$, $F(\dot{q})$, et $G(q)$ et un vecteur des degrés de liberté $q \in \mathbb{R}^n$. Il est toujours possible de trouver une matrice à dimensions $n \times p$ des fonctions connues, $A(q, \dot{q}, \ddot{q}, \int q)$ et un vecteur à dimension p avec des composants dépendant des paramètres du robot λ tel que:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(\dot{q}) + G(q) = A(q, \dot{q}, \ddot{q}, \int q)\lambda \quad (\text{VI.71})$$

On note que les deux expressions de forces/couples (VI.62) et (VI.70) sont équivalents. Afin de développer une commande adaptative, on appelle l'expression (VI.62):

$$\bar{\Gamma}_c = F(x_e) + T_1^{-1}u \quad (\text{VI.72})$$

De l'expression (VI.36) qui présente la relation entre $\bar{\Gamma}_c$ et Γ_c on a:

$$\Gamma_c = F(x_e) + M_{cu}(q)\ddot{q}_u + C_{cu}(q, \dot{q})\dot{q}_u + T_1^{-1}u \quad (\text{VI.73})$$

On pose un nouveau terme $E(x_e)$:

$$E(x_e) = F(x_e) + M_{cu}(q)\ddot{q}_u + C_{cu}(q, \dot{q})\dot{q}_u \quad (\text{VI.74})$$

De l'expression (VI.74) et (VI.36) qui présente l'expression de $F(x_e)$, on a:

$$E(x_e) = M_{cc}(q)(\ddot{q}_c^d - T_1^{-1}T_2\dot{\tilde{q}}_c - T_1^{-1}T_3\tilde{q}_c) + C_{cc}(q, \dot{q})(\dot{q}_c^d - T_1^{-1}T_2\tilde{q}_c - T_1^{-1}T_3 \int \tilde{q}_c dt) + G_c(q) + M_{cu}(q)\ddot{q}_u + C_{cu}(q, \dot{q})\dot{q}_u \quad (\text{VI.75})$$

On peut réécrire le vecteur des forces/couples:

$$\Gamma_c = E(x_e) + T_1^{-1}u \quad (\text{VI.76})$$

Avec $x_e = (\tilde{q}_c, \dot{\tilde{q}}_c, \int \tilde{q}_c dt, \dot{q}_c^d, \ddot{q}_c^d, \dot{q}_u, \ddot{q}_u)$. Les couples appliqués résultants sont donnés par (VI.76). Ces couples sont essentiellement les mêmes que ceux présentés dans l'équation (VI.70) pour la commande H_∞ non-linéaire. La différence ici est que, on utilise les matrices dynamiques complètes au lieu des matrices dynamiques nominales. On peut considérer le terme basé sur le modèle $E(x_e)$ comme composé des termes nominaux et incertains définis par:

$$E(x_e) = E_0(x_e) + \Delta E(x_e) \quad (\text{VI.77})$$

Où:

$$E_0(x_e) = M_{cc}(q)(\ddot{q}_c^d - T_1^{-1}T_2\dot{\tilde{q}}_c - T_1^{-1}T_3\tilde{q}_c) + C_{cc}(q, \dot{q})(\dot{q}_c^d - T_1^{-1}T_2\dot{\tilde{q}}_c - T_1^{-1}T_3 \int \tilde{q}_c dt) + G_c(q) + M_{cu}(q)\ddot{q}_u + C_{cu}(q, \dot{q})\dot{q}_u \quad (\text{VI.78})$$

$$\Delta E(x_e) = \Delta M_{cc}(q)(\ddot{q}_c^d - T_1^{-1}T_2\dot{\tilde{q}}_c - T_1^{-1}T_3\tilde{q}_c) + \Delta C_{cc}(q, \dot{q})(\dot{q}_c^d - T_1^{-1}T_2\dot{\tilde{q}}_c - T_1^{-1}T_3 \int \tilde{q}_c dt) + \Delta G_c(q) + \Delta M_{cu}(q)\ddot{q}_u + \Delta C_{cu}(q, \dot{q})\dot{q}_u \quad (\text{VI.79})$$

La conception du correcteur présenté dans la Section VI.4 considère les termes incertains comme faisant partie de la perturbation w . Ici, l'hypothèse de paramétrage linéaire de $\Delta E(x_e)$ est utilisée pour concevoir une loi de commande adaptative pour apprendre le comportement de ce terme incertain. Selon la propriété précédente, $\Delta E(x_e)$ peut être exprimé comme:

$$\Delta E(x_e) = A \hat{\lambda} \quad (\text{VI.80})$$

Le correcteur adaptatif décrit, vise à estimer la partie incertaine $\Delta E(x_e)$ et à satisfaire une performance de H_∞ désirée. Nous pouvons énoncer le problème de la commande adaptative H_∞ non-linéaire tel que: étant donné un niveau d'atténuation γ trouver un correcteur adaptatif.

Nous considérons la formulation ci-dessous:

$$\dot{\hat{\lambda}} = \alpha(x_c, t), \quad \Gamma_c = E_0(x_e) + A\hat{\lambda} + T_1^{-1}u \quad (\text{VI.81})$$

Et que le système en boucle fermée satisfasse la performance suivante:

$$\int_0^T \|\xi\|_2^2 dt \leq \tilde{\lambda}^T \Lambda \tilde{\lambda} + \gamma^2 \int_0^T \|w\|_2^2 dt \quad (\text{VI.82})$$

Où $\tilde{\lambda} = \lambda - \hat{\lambda}$ dénote l'erreur d'estimation des paramètres et $\hat{\lambda}$ est le vecteur des valeurs estimés.

La solution de ce problème de commande non-linéaire peut être trouvée en suivant la dérivation basée sur la théorie présentée à la Section VI.4 et l'approche de développer le correcteur adaptatif proposée dans (Chen et al., 1997). Pour ce but, nous considérons la fonction de Lyapunov suivante:

$$V = \frac{1}{2} x_c^T P x_c + \frac{1}{2} \tilde{\lambda}^T \Lambda \tilde{\lambda} \quad (\text{VI.83})$$

Où P est la matrice semi-positive symétrique suivante:

$$P = T_0^T \begin{bmatrix} M_{cc}(q) & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & Y & X - Y \\ \mathbb{0} & X - Y & Z + Y \end{bmatrix} T_0 \quad (\text{VI.84})$$

Et Λ est une matrice symétrique positive.

Basé sur la solution présentée dans la Section VI.4 et (VI.61), la loi de commande adaptative suivante est une solution au problème de la commande adaptative H_∞ non-linéaire:

$$\begin{aligned}\hat{\lambda} &= -\Lambda^{-1}A^T T_1(S^T + T)x_c \\ \Gamma_c &= E_0(x_e) + A\hat{\lambda} - T_1^{-1}R^{-1}(S^T + T)x_c\end{aligned}\tag{VI.85}$$

En revenant sur le modèle du quadrotor, le terme $\Delta E(x_e) = A\hat{\lambda}$ présente la partie du modèle dynamique des incertitudes. La matrice A est comme suit:

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{16} \\ A_{21} & A_{22} & \cdots & A_{26} \\ A_{31} & A_{32} & \cdots & A_{36} \\ A_{41} & A_{42} & \cdots & A_{46} \end{bmatrix}\tag{VI.86}$$

Tableau VI.2 Les coefficients de la matrice d'adaptation A

Paramètre	Les termes des coefficients
A_{11}	$(c\varphi c\theta(a_{11} + 1) + (c\varphi s\theta c\psi + s\varphi s\psi)a_{31} + (c\varphi s\theta s\psi - s\varphi c\psi)a_{32})$
A_{12}	0
A_{13}	0
A_{14}	0
A_{15}	0
A_{16}	0
A_{21}	0
A_{22}	$a_{12} - s\theta a_{14}$
A_{23}	$(-\dot{\psi}c^2\varphi c\theta + \dot{\theta}c\varphi s\varphi + \dot{\psi}s^2\varphi c\theta)a_{23} + (-\dot{\psi}c\varphi s\varphi c^2\theta)a_{24}$
A_{24}	$(-\dot{\theta}c\varphi s\varphi - \dot{\psi}s^2\varphi c\theta + \dot{\psi}c^2\varphi c\theta)a_{23} + (\dot{\psi}c\varphi s\varphi c^2\theta)a_{24}$
A_{25}	$-\dot{\psi}c\theta a_{23}$
A_{26}	$\dot{\psi}c\theta a_{23}$
A_{31}	0
A_{32}	$-\dot{\psi}s\theta c\theta a_{24}$
A_{33}	$(c^2\varphi)a_{13} + (c\varphi s\varphi c\theta)a_{14} + (\dot{\psi}c^2\varphi c\theta - \dot{\theta}c\varphi s\varphi - \dot{\psi}s^2\varphi c\theta)a_{22} + (-\dot{\varphi}c\varphi s\varphi)a_{23} + (\dot{\psi}s^2\varphi c\theta s\theta)a_{24}$
A_{34}	$(s^2\varphi)a_{13} - (c\varphi s\varphi c\theta)a_{14} + (\dot{\theta}c\varphi s\varphi + \dot{\psi}s^2\varphi c\theta - \dot{\psi}c^2\varphi c\theta)a_{22} + (\dot{\varphi}c\varphi s\varphi)a_{23} + (\dot{\psi}c^2\varphi s\theta c\theta)a_{24}$
A_{35}	$\dot{\psi}c\theta a_{22}$

A_{36}	$-\dot{\psi}c\theta a_{22}$
A_{41}	0
A_{42}	$(-s\theta)a_{12} + (s^2\varphi)a_{14} + (\dot{\psi}c\theta s\theta)a_{23} + (\dot{\theta}c\theta s\theta)a_{24}$
A_{43}	$(c\varphi s\varphi c\theta)a_{13} + (s^2\varphi c^2\theta)a_{14} + (\dot{\psi}c^2\theta s\varphi c\varphi)a_{22} + (-\dot{\theta}c\varphi s\varphi s\theta - \dot{\varphi}s^2\varphi c\theta + \dot{\varphi}c^2\varphi c\theta - \dot{\psi}s^2\varphi s\theta c\theta)a_{23} + (\dot{\varphi}c\varphi s\varphi c^2\theta - \dot{\theta}s^2\varphi c\theta s\theta)a_{24}$
A_{44}	$(-c\varphi s\varphi c\theta)a_{13} + (c^2\varphi c^2\theta)a_{14} + (-\dot{\psi}c^2\theta s\varphi c\varphi)a_{22} + (\dot{\theta}c\varphi s\varphi s\theta + \dot{\varphi}s^2\varphi c\theta - \dot{\varphi}c^2\varphi c\theta - \dot{\psi}c^2\varphi s\theta c\theta)a_{23} + (-\dot{\varphi}c\varphi s\varphi c^2\theta - \dot{\theta}c^2\varphi c\theta s\theta)a_{24}$
A_{45}	$-\dot{\theta}c\theta a_{22}$
A_{46}	$\dot{\theta}c\theta a_{22}$
a_{11}	$\ddot{z}^d - T_1^{-1}T_2\dot{\tilde{z}} - T_1^{-1}T_3\tilde{z}$
a_{12}	$\ddot{\varphi}^d - T_1^{-1}T_2\dot{\tilde{\varphi}} - T_1^{-1}T_3\tilde{\varphi}$
a_{13}	$\ddot{\theta}^d - T_1^{-1}T_2\dot{\tilde{\theta}} - T_1^{-1}T_3\tilde{\theta}$
a_{14}	$\ddot{\psi}^d - T_1^{-1}T_2\dot{\tilde{\psi}} - T_1^{-1}T_3\tilde{\psi}$
a_{21}	$\dot{z}^d - T_1^{-1}T_2\tilde{z} - T_1^{-1}T_3 \int \tilde{z}$
a_{22}	$\dot{\varphi}^d - T_1^{-1}T_2\tilde{\varphi} - T_1^{-1}T_3 \int \tilde{\varphi}$
a_{23}	$\dot{\theta}^d - T_1^{-1}T_2\tilde{\theta} - T_1^{-1}T_3 \int \tilde{\theta}$
a_{24}	$\dot{\psi}^d - T_1^{-1}T_2\tilde{\psi} - T_1^{-1}T_3 \int \tilde{\psi}$
a_{31}	\ddot{x}
a_{32}	\ddot{y}

Le vecteur des paramètres incertains est comme suit:

$$\hat{\lambda} = [\lambda_1 \lambda_2 \lambda_3 \lambda_4 \lambda_5 \lambda_6]^T \quad (\text{VI.87})$$

Où:

$$\lambda_1 = \Delta m, \lambda_2 = \Delta I_x, \lambda_3 = \Delta I_y, \lambda_4 = \Delta I_z, \lambda_5 = \Delta I_y \Delta I_z, \lambda_6 = \Delta I_x \Delta I_z$$

On garde les paramètres du correcteur comme ils sont mentionnés dans la Section VI.4. Avec $\Lambda = 10 \mathbb{1}_6$ où $\mathbb{1}$ est la matrice identité.

VI.5.2 Adaptation par Réseaux de Neurones:

Lorsque le modèle mathématique du robot n'est connu que de manière artificielle, par exemple lorsque les composants de friction ne peuvent pas être entièrement modélisés, des réseaux de neurones peuvent être utilisés pour estimer les éléments inconnus du modèle. Un avantage des

stratégies basées sur les réseaux de neurones est qu'elles n'ont pas besoin d'informations sur le modèle analytique du système. Dans cette section, un réseau de neurones $\Delta E(x_e, B)$; où B est un vecteur contenant les paramètres du réseau (les poids), est utilisé pour approximer le terme incertain $\Delta E(x_e)$ dans (VI.77). Il s'agit d'une approche alternative au paramétrage linéaire présenté précédemment.

Nous définissons n réseaux de neurones $\Delta E_k(x_e, \beta_k)$; $k=1 \dots n$ composé de neurones non-linéaires dans chaque couche cachée et de neurones linéaires dans les couches d'entrée et de sortie, avec des paramètres ajustables β_k dans les couches de sortie (Chang, 2000). Les réseaux de neurones à sortie unique se présentent sous la forme:

$$\Delta E_k(x_e, \beta_k) = \sum_{i=1}^p TH(\sum_{j=1}^{ik} w_{ij}^k x_{ej} + b_i^k) \beta_{ki} = \xi_k^T \beta_k \quad (\text{VI.88})$$

Où:

$$\xi_k = \begin{bmatrix} \xi_{k1} \\ \vdots \\ \xi_{kp} \end{bmatrix} = \begin{bmatrix} TH(\sum_{j=1}^{ik} w_{1j}^k x_{ej} + b_1^k) \\ \vdots \\ TH(\sum_{j=1}^{ik} w_{pj}^k x_{ej} + b_p^k) \end{bmatrix}, \beta_k = \begin{bmatrix} \beta_{k1} \\ \vdots \\ \beta_{kp} \end{bmatrix}$$

Et p est le nombre de neurones dans la couche cachée, ik est le nombre de neurones dans la couche d'entrée, x_e est le vecteur des entrées. Les poids w_{ij}^k et le biais b_i^k sont constants et spécifiés par l'utilisateur. La fonction $TH(\cdot)$ est la fonction non-linéaire *tangente hyperbolique*:

$$TH(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (\text{VI.89})$$

Nous allons spécifier les nombres des entrées, les poids, et le nombre des neurones de la couche des entrées de notre système (le quadrotor) ultérieurement. Les réseaux de neurones complets sont présentés comme suit:

$$\Delta E(x_e, B) = \begin{bmatrix} \Delta E_1(x_e, \beta_1) \\ \Delta E_2(x_e, \beta_2) \\ \vdots \\ \Delta E_n(x_e, \beta_n) \end{bmatrix} = \begin{bmatrix} \xi_1^T \beta_1 \\ \xi_2^T \beta_2 \\ \vdots \\ \xi_n^T \beta_n \end{bmatrix} = \begin{bmatrix} \xi_1^T & 0 & \dots & 0 \\ 0 & \xi_2^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \xi_n^T \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix} = \Xi B \quad (\text{VI.90})$$

Nous pouvons énoncer le problème de la commande adaptative H_∞ non-linéaire basée sur les réseaux de neurones par: étant donné un niveau d'atténuation γ trouver un correcteur adaptatif.

Nous considérons la formulation ci-dessous:

$$\dot{B} = \alpha(x_c, t), \Gamma_c = E_0(x_e) + \Xi B + T_1^{-1} u \quad (\text{VI.91})$$

Et que le système en boucle fermée satisfasse la performance suivante:

$$\int_0^T \|\xi\|_2^2 dt \leq \tilde{B}^T N \tilde{B} + \gamma^2 \int_0^T \|w\|_2^2 dt \quad (\text{VI.92})$$

Nous considérons les résultats de la Section VI.5.1, pour la fonction de Lyapunov:

$$V = \frac{1}{2} x_c^T P x_c + \frac{1}{2} B^T N B \quad (\text{VI.93})$$

Où N est une matrice symétrique positive.

En se basant sur la solution présentée dans la Section VI.5.1, la loi de commande adaptative suivante est une solution au problème de la commande adaptative H_∞ non-linéaire basée sur les réseaux de neurones:

$$\begin{aligned} \dot{B} &= -N^{-1} E^T T_1 (S^T + T) x_c \\ \Gamma_c &= E_0(x_e) + \tilde{E} B - T_1^{-1} R^{-1} (S^T + T) x_c \end{aligned} \quad (\text{VI.94})$$

Dans le cas du quadrotor, nous avons 4 entrées (F_z , τ_ϕ , τ_θ et τ_ψ) alors nous avons besoin de 4 réseaux, où la sortie de chaque réseau présente l'action adaptative. Les spécifications des réseaux sont:

$$E = \begin{bmatrix} \xi_1^T & 0 & 0 & 0 \\ 0 & \xi_2^T & 0 & 0 \\ 0 & 0 & \xi_3^T & 0 \\ 0 & 0 & 0 & \xi_4^T \end{bmatrix} \quad (\text{VI.95})$$

Le réseau 1: le terme adaptatif de la force F_z ; la sortie $\Delta E_1(x_e, B)$, selon la dynamique E dans (VI.75); les 9 entrées sont $x_e = [\ddot{z}^d, \dot{z}^d, \dot{z}, z^d, z, \int z^d, \int z, \ddot{x}, \dot{y}]$. $\xi_1^T = [\xi_{11} \dots \xi_{17}]$ est calculé comme présenté dans (VI.88) où les poids constants $w_i^1 = [-1, -1, 1, -1, 1, -1, 1, 1, 1]$ de chaque neurone de la couche cachée pour les entrées x_e respectivement (-1 pour le valeur désiré et 1 pour le valeur actuel). Nous posons le vecteur des biais; $b_1^T = [-1.5, -1, -0.5, 0, 0.5, 1, 1.5]$.

Le réseau 2: le terme adaptatif du couple τ_ϕ ; la sortie $\Delta E_2(x_e, B)$, selon la dynamique E dans (VI.75); les 7 entrées sont $x_e = [\ddot{\phi}^d, \dot{\phi}^d, \dot{\phi}, \phi^d, \phi, \int \phi^d, \int \phi]$. $\xi_2^T = [\xi_{21} \dots \xi_{27}]$ est calculé comme présenté dans (VI.88) où les poids constants $w_i^2 = [-1, -1, 1, -1, 1, -1, 1]$ de chaque neurone de la couche cachée pour les entrées x_e respectivement (-1 pour le valeur désiré et 1 pour le valeur actuel). Nous posons le vecteur des biais; $b_2^T = [-1.5, -1, -0.5, 0, 0.5, 1, 1.5]$.

Le réseau 3: le terme adaptatif du couple τ_θ ; la sortie $\Delta E_3(x_e, B)$, selon la dynamique E dans (VI.75); les 7 entrées sont $x_e = [\ddot{\theta}^d, \dot{\theta}^d, \dot{\theta}, \theta^d, \theta, \int \theta^d, \int \theta]$. $\xi_3^T = [\xi_{31} \dots \xi_{37}]$ est calculé comme

présenté dans (VI.88) où les poids constants $w_i^3 = [-1, -1, 1, -1, 1, -1, 1]$ de chaque neurone de la couche cachée pour les entrées x_e respectivement (-1 pour le valeur désiré et 1 pour le valeur actuel). Nous posons le vecteur des biais; $b_3^T = [-1.5, -1, -0.5, 0, 0.5, 1, 1.5]$.

Le réseau 4: le terme adaptatif du couple τ_ψ ; la sortie $\Delta E_4(x_e, B)$, selon la dynamique E dans (VI.75); les 7 entrées sont $x_e = [\ddot{\psi}^d, \dot{\psi}^d, \psi, \dot{\psi}^d, \psi, \int \psi^d, \int \psi]$. $\xi_4^T = [\xi_{41} \dots \xi_{47}]$ est calculé comme présenté dans (VI.88) où les poids constants $w_i^4 = [-1, -1, 1, -1, 1, -1, 1]$ de chaque neurone de la couche cachée pour les entrées x_e respectivement (-1 pour le valeur désiré et 1 pour le valeur actuel). Nous posons le vecteur des biais; $b_4^T = [-1.5, -1, -0.5, 0, 0.5, 1, 1.5]$.

Les réseaux sont spécialisés et les paramètres ajustables des réseaux sont les poids entre la couche cachée et celle de la sortie; B . Parce que nous avons 4 réseaux et 7 neurones de la couche cachée de chaque réseau, la matrice d'adaptation B est calculée dans (VI.94):

$$B = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \end{bmatrix} \quad (VI.96)$$

Où:

$$\begin{aligned} \beta_1 &= [\beta_{11} \ \beta_{12} \ \beta_{13} \ \beta_{14} \ \beta_{15} \ \beta_{16} \ \beta_{17}] \\ \beta_2 &= [\beta_{21} \ \beta_{22} \ \beta_{23} \ \beta_{24} \ \beta_{25} \ \beta_{26} \ \beta_{27}] \\ \beta_3 &= [\beta_{31} \ \beta_{32} \ \beta_{33} \ \beta_{34} \ \beta_{35} \ \beta_{36} \ \beta_{37}] \\ \beta_4 &= [\beta_{41} \ \beta_{42} \ \beta_{43} \ \beta_{44} \ \beta_{45} \ \beta_{46} \ \beta_{47}] \end{aligned} \quad (VI.97)$$

Les quatre figures suivantes présentent les schémas des réseaux.

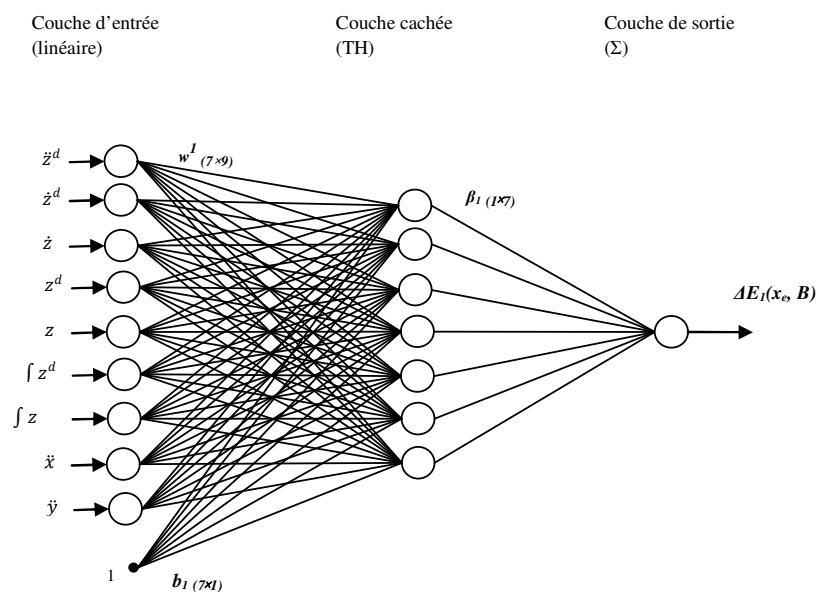


Figure VI.9 Schéma du réseau de neurone pour l'adaptation de $F_z, \Delta E_1(x_e, B)$

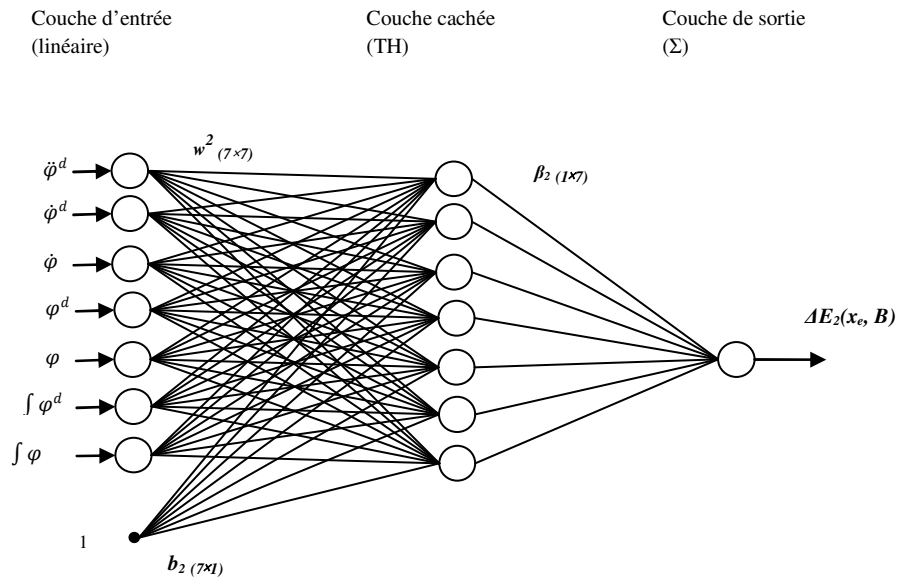


Figure VI.10 Schéma du réseau de neurone pour l'adaptation de τ_φ , $\Delta E_2(x_e, B)$

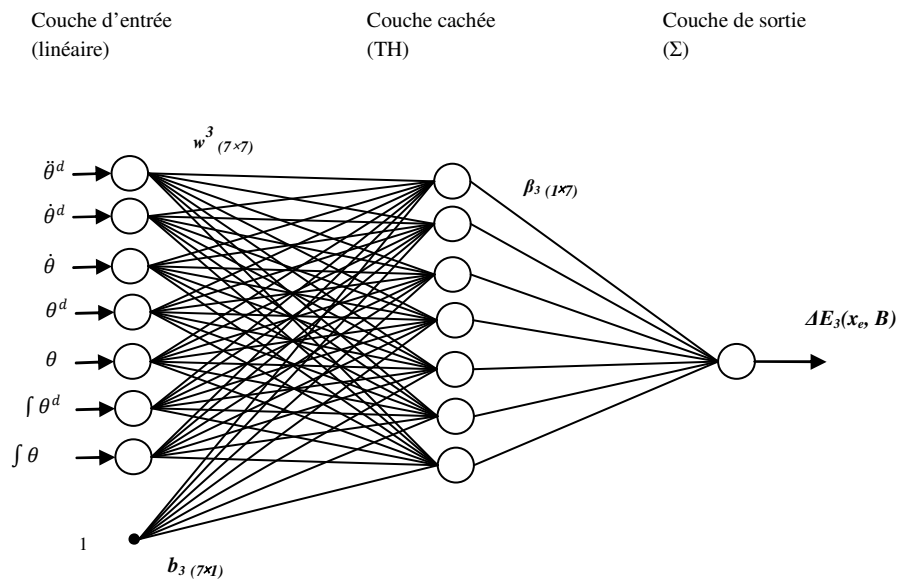


Figure VI.11 Schéma du réseau de neurone pour l'adaptation de τ_θ , $\Delta E_3(x_e, B)$

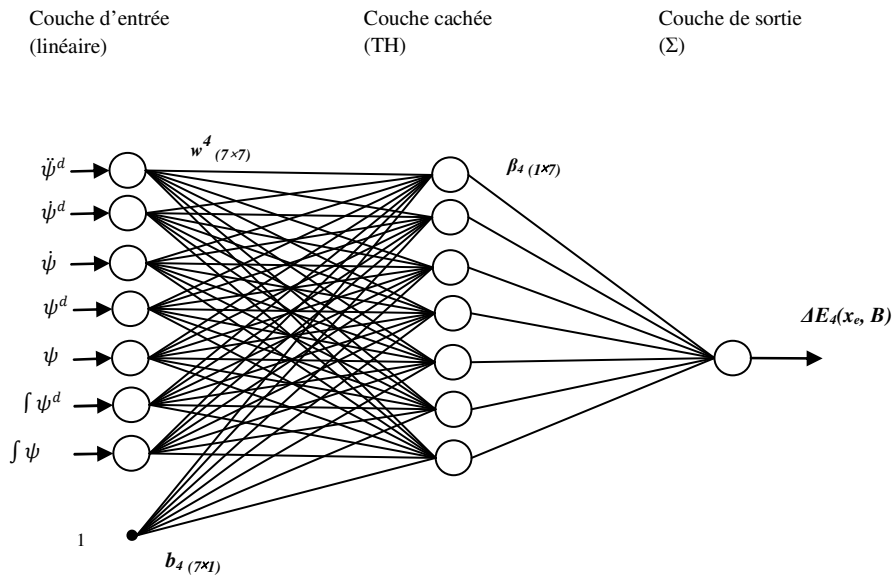


Figure VI.12 Schéma du réseau de neurone pour l'adaptation de τ_ψ , $\Delta E_d(x_e, B)$

Nous gardons les paramètres du correcteur comme il est mentionné dans la Section VI.4 avec $N=6 \mathbb{1}_{28}$ où $\mathbb{1}$ est la matrice identité.

VI.5.3 Résultats de Simulation:

Initialement, le quadrotor commence de la position initiale $(x_0, y_0, z_0) = (0.5, 0, 0)$ et puis monte à 1 m dans l'axe z , à l'instant 10 s le quadrotor fait une trajectoire comme suit:

$$x_d = 0.5 \cos\left(\frac{\pi}{20} t\right), y_d = 0.5 \sin\left(\frac{\pi}{20} t\right), z_d = 3 - \left(2 \cos\left(\frac{\pi}{20} t\right)\right), \psi_d = \frac{\pi}{3} \text{ rad.}$$

Les perturbations appliquées sont de forme échelon, en tant que couples et forces tels que:

$A_x=1 \text{ N}$ à l'instant $t=30 \text{ s}$; $A_y=1 \text{ N}$ à l'instant $t=40 \text{ s}$; $A_z=1 \text{ N}$ à l'instant $t=20 \text{ s}$; $A_\varphi=0.1 \text{ N.m}$ à l'instant $t=25 \text{ s}$; $A_\theta=0.1 \text{ N.m}$ à l'instant $t=35 \text{ s}$; et $A_\psi=0.01 \text{ N.m}$ à l'instant $t=15 \text{ s}$.

Les paramètres des correcteurs H_∞ non-linéaire et Backstepping intégral sont les mêmes dans la section VI.4.

Dans ce test, nous supposons qu'il y a des incertitudes paramétriques tels que: $\Delta I_x = 0.7 I_x \sin\left(\frac{2\pi}{5} t\right)$, $\Delta I_y = 0.7 I_y \cos\left(\frac{2\pi}{5} t\right)$, $\Delta I_z = 0.7 I_z \sin\left(\frac{2\pi}{5} t\right)$. Concernant la masse, nous supposons que la trajectoire a commencé avec la masse nominale du quadrotor, à partir de l'instant 35 s, une masse additionnelle ($\Delta m = 3 \text{ kg}$) a été ajoutée.

Ces conditions de la simulation sont appliquées pour trois tests de simulation; la commande H_∞ non-linéaire qui est développée dans la Section VI.4 (*IBS- H_∞ NL*), la commande H_∞ non-linéaire adaptative qui est développée dans la Section VI.5.1 (*IBS- H_∞ NL Adapt.*) et la commande H_∞ non-

linéaire adaptative par les réseaux de neurones qui est développée dans la Section VI.5.2 (*IBS- H_∞ NL Adapt.-RN*).

La Figure VI.13 présente la simulation du suivi de la trajectoire en 3D, alors que la Figure VI.14 présente les réponses des degrés de liberté de translation (x , y , z), la Figure VI.15 présente les réponses des degrés de liberté de rotation (φ , θ , ψ), et la Figure VI.16 présente les commandes (la force F_z et les couples; τ_φ , τ_θ et τ_ψ). Nous remarquons que les trois approches indiquent une performance excellente de suivi de référence même si les perturbations externes aux forces et couples sont considérées. Nous remarquons une petite variation à l'instant 35 s (l'instant d'ajout une masse) dans la courbe du correcteur sans adaptation (*IBS- H_∞ NL*), nous pouvons conclure alors que les deux correcteurs adaptatifs présentés dans cette section sont plus efficaces et robustes vis-à-vis des incertitudes paramétriques.

Afin d'évaluer les trois correcteurs, le Tableau VI.3 présente leur évaluation selon le critère intégral d'erreur carré (*ISE: Integral Square Error*) de toutes les variables (x , y , z , φ , θ et ψ). Le Tableau VI.4 présente l'évaluation des trois correcteurs selon le critère intégral d'effort carré (*ISEF: Integral Square Effort*) de toutes les entrées (F_z , τ_φ , τ_θ et τ_ψ).

Évidemment, à partir des deux tableaux, les correcteurs adaptatifs donnent le meilleur suivi de trajectoire et des efforts minimaux.

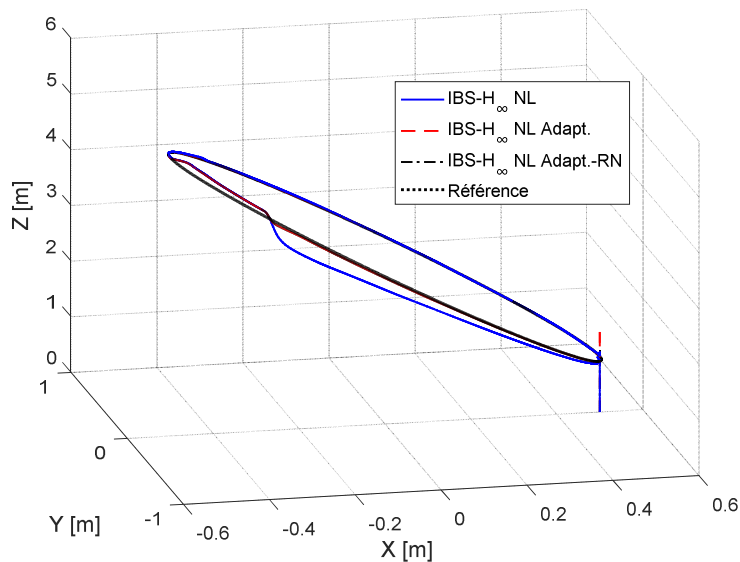


Figure VI.13 Suivi de la trajectoire en 3D (2^{ème} trajectoire)

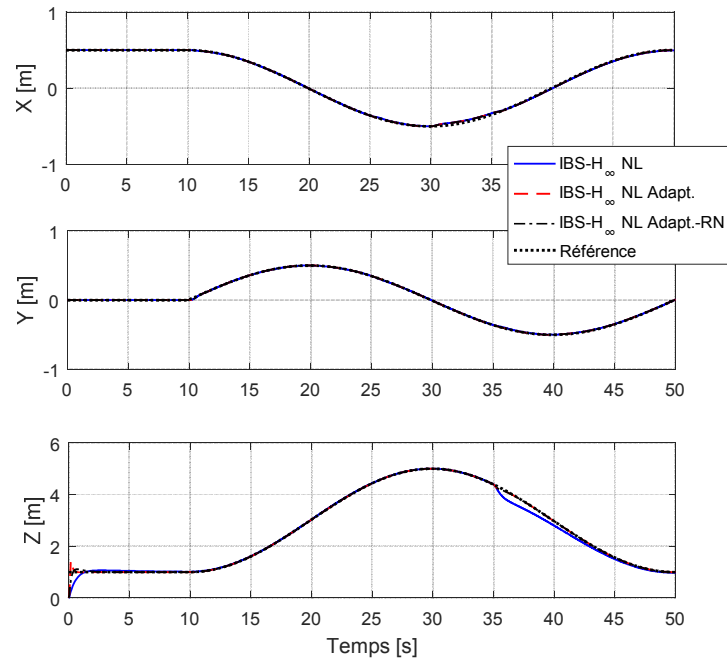


Figure VI.14 Réponses des coordonnées de la position (2^{ème} trajectoire)

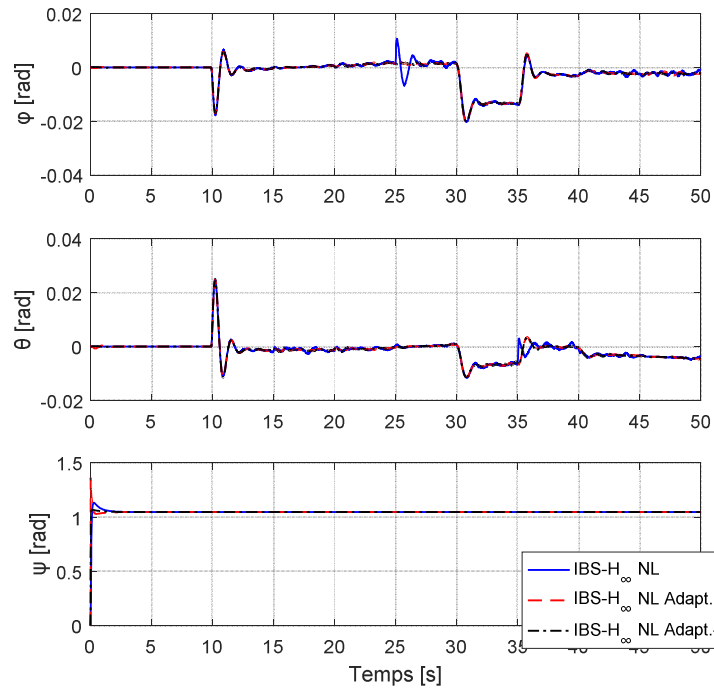


Figure VI.15 Réponses des coordonnées de rotation (2^{ème} trajectoire)

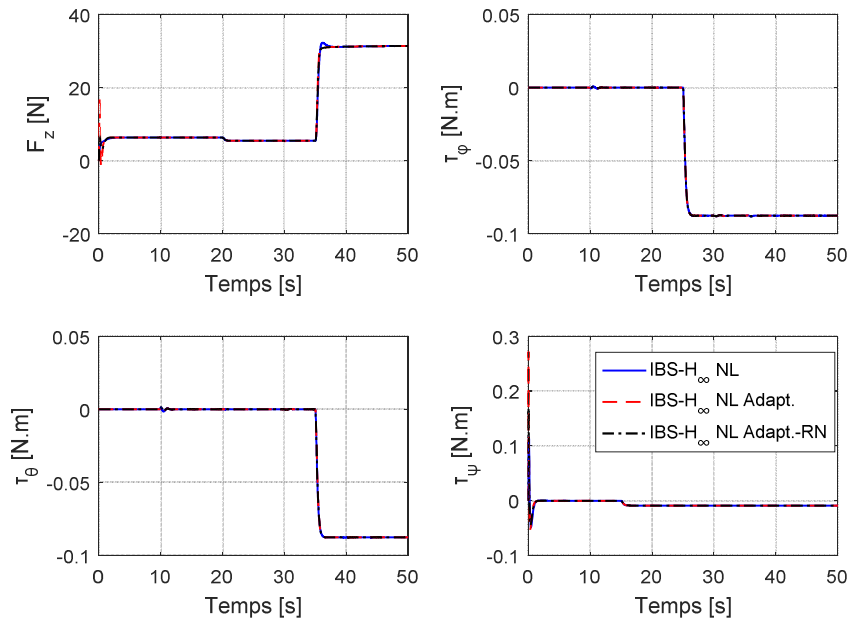

Figure VI.16 Les entrées de commande (2^{ème} trajectoire)

Tableau VI.3 Comparaison de l'ISE entre les trois correcteurs

	<i>IBS-H_∞ NL</i>	<i>IBS-H_∞ NL Adapt.</i>	<i>IBS-H_∞ NL Adapt.-RN</i>
x	$3.7 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$3.6 \cdot 10^{-3}$
y	$4.63 \cdot 10^{-4}$	$2.69 \cdot 10^{-4}$	$3.27 \cdot 10^{-4}$
z	0.584	0.085	0.053
φ	$2.36 \cdot 10^{-5}$	$3.02 \cdot 10^{-8}$	$2 \cdot 10^{-7}$
θ	$2.1 \cdot 10^{-5}$	$2.88 \cdot 10^{-8}$	$2.13 \cdot 10^{-7}$
ψ	0.059	0.022	0.014

Tableau VI.4 Comparaison de l'ISEF entre les trois correcteurs

	<i>IBS-H_∞ NL</i>	<i>IBS-H_∞ NL Adapt.</i>	<i>IBS-H_∞ NL Adapt.-RN</i>
F_z	16864	15908	17479
τ_φ	0.202	0.192	0.218
τ_θ	0.122	0.115	0.128
τ_ψ	0.0036	0.0055	0.0042

VI.6 Conclusion:

Dans ce chapitre, nous avons effectué une commande du quadrotor qui est un système non-linéaire et sous-actionné par la technique de commande par couple calculé (*CTC: Computed Torque Control*). Le calcul des forces/couples par le couple calculé est fortement basé sur la connaissance du modèle. Après l'obtention de ce dernier, une commande robuste non-linéaire a été établie par la technique H_∞ non-linéaire. La robustesse a été renforcée par l'action adaptative en considérons deux méthodes; paramétrage linéaire et par les réseaux de neurones. Les résultats présentent un excellent suivi de trajectoire avec robustesse vis-à-vis des perturbations et des incertitudes paramétriques.

On peut resumer les différences entre les approches développées dans cette thèse par la comparaison présenté dans Tableau VI.5. Les correcteurs sont ; Proportionel Intégral Dérivé de chapitre 3 (**PID**), BackStepping Intégral (**IBS**) et Mode Glissant de type Fast Terminal (**FTSMC**) de chapitre 4, le correcteur H_∞ Linéaire à paramètres variés (**LPV**) de chapitre 5, et H_∞ non-linéaire adaptatif (**H_∞ NL**) de chapitre 6.

Tableau VI.5 Comparaison entre les différents correcteurs

Correc.	Rejection des perturbations	Dépendamment au modèle	Complexité	Robustesse	Remarque
PID	+	Sans modèle	Simple	-	Ordre bas
IBS	+	Basé sur le modèle	Complexe	+	Non-linéaire
FTSMC	+	Sans modèle	Complexe	+	Non-linéaire
LPV	++	Sans modèle	Complexe	++	Ordre élevé, robuste
H_∞ NL	++	Basé sur le modèle	Complexe	++	Ordre élevé, robuste, adaptatif

Conclusion Générale:

Le développement des véhicules aériens autonomes UAV dans les domaines aussi bien militaires que civils, a suscité pendant plusieurs décades un intérêt particulier pour les chercheurs du domaine de l'aérodynamique et ceci du fait que ses systèmes sont très utiles lorsqu'il s'agit d'exécuter des tâches dans des environnements dangereux et inaccessibles. Plusieurs aspects de ce domaine ont été développés en littérature parmi lesquels on trouve la commande des UAV.

Dans ce travail, nous avons développé des stratégies de commande robustes pour résoudre le problème de suivi de trajectoire pour les UAV, en particulier le quadrotor. Ce dernier est un système mécanique sous-actionné, du fait qu'il possède six degrés de liberté et seulement quatre entrées. Les stratégies utilisées dans ce travail ont été conçues en tenant compte, des perturbations externes persistantes affectant le système, de la dynamique-non modélisée et aussi des incertitudes structurelles et paramétriques. Par conséquent, des structures de commande en cascade ont également été développées en tenant compte de cette fonctionnalité pour obtenir les performances souhaitées.

Après l'élaboration d'un modèle dynamique simplifié du système quadrotor, les travaux réalisés en considérant ce type de structure ont été présentés aux troisième et quatrième chapitres, où différentes techniques ont été utilisées pour commander les mouvements de rotation et de translation. De nombreux correcteurs linéaires et non-linéaires classiques ont été conçus afin d'atteindre le suivi de trajectoire avec un test de robustesse contre les perturbations externes et les incertitudes paramétriques.

La théorie avancée de la commande robuste, H_∞ , a été utilisée au cinquième chapitre, où la structure en cascade a également été appliquée. La théorie H_∞ étendue pour les techniques LPV a été appliquée pour stabiliser le sous-système rotationnel, avec un modèle présenté initialement comme un système quasi-LPV avec une représentation LFT. Ce dernier offre l'avantage de représenter le modèle non-linéaire exact d'origine sans la nécessité de le linéariser à des sous-systèmes LTI. Le correcteur du sous-système de translation est développé par la technique Backstepping intégral qui donne un bon suivi de trajectoire et une robustesse contre les incertitudes, les perturbations et les bruits.

Dans le sixième chapitre, la commande par couple calculé, qui est développée principalement pour les systèmes mécaniques, nécessitant le modèle complet a été exploitée dans ce travail, en se basant sur le modèle presque complet et par conséquent la conception du correcteur a été effectuée. Les correcteurs conçus pour les systèmes mécaniques sont communément basés sur des modèles

nominaux, mais ces modèles impliquent des incertitudes structurelles et paramétriques. En premier lieu, un correcteur H_∞ non-linéaire approprié aux systèmes sous-actionnés a été développé avec considération soit les perturbations, soit les incertitudes comme des entrées externes.

Cette hypothèse n'est pas très réaliste, et pour améliorer la robustesse du correcteur et neutraliser les erreurs de modélisation, la solution était la génération un signal de commande supplémentaire adaptatif en estimant les incertitudes. Cette technique d'adaptation a été effectuée à partir de deux méthodes; le paramétrage linéaire et aussi en introduisant de l'intelligence artificielle à travers l'application des réseaux de neurones. Cette technique de commande robuste adaptative a donné un niveau de robustesse mieux que les techniques classiques en ce qui concerne les perturbations et surtout les incertitudes.

Perspectives:

À partir des résultats atteints à travers les travaux réalisés dans cette thèse, plusieurs perspectives peuvent être mentionnées:

- De nombreux systèmes aérodynamiques ont des entrées bornées, il serait ainsi intéressant de développer des lois de commande en présence de saturations sur les entrées (Weston et Postlethwaite, 2000; Li et al., 2017; Liu et al., 2019).
- Renforcer la mesure par des observateurs; si le quadrotor est équipé d'un capteur de position GPS, l'erreur de mesure peut être dans l'intervalle $\pm 2 m$ et l'information est acquise au moins chaque seconde. Cependant, la dynamique de ce type de véhicules peut être de l'ordre des millisecondes. Entre chaque deux échantillons de GPS, les états doivent être estimés, et après la mesure est reçu, il doit mettre à jour.
- Il est aussi envisageable d'appliquer de telles stratégies de commande sur les systèmes physiques et robotiques, par exemple; bras manipulateur (Firouzeh et al., 2015), main-effecteur d'un robot (Prattichizzo et al., 2012) et robot marcheur (Takase et al., 2013).

Bibliographie:

Abdou, L., et F. Soltani, « OS-CFAR and CMLD threshold optimization in distributed systems using evolutionary strategies », *Signal, Image and Video Processing*. 2(2):155-167, 2008.

Acampora, G., H. Ishibuchi, et A. Vitiello, « A comparison of multi-objective evolutionary algorithms for the ontology meta-matching problem », *IEEE Congress on Evolutionary Computation*. Pékin, Chine, 06 Juillet 2014.

Acosta, J.A.P., et C.P. Bottura, « Discretization of linear parameter varying systems in the LFT representation with parameter dependent sampling rates », *Proceedings Computer Science and Electronic Engineering*. 139-144, Colchester, UK, 2017.

Alejo, D., J.A. Cobano, G. Heredia, et A. Ollero, « Collision-free 4D trajectory planning in unmanned aerial vehicles for assembly and structure construction », *Journal of Intelligent & Robotic Systems*. 73(1-4): 783-795, 2013.

Allibert, G., D. Abeywardena, M. Bangura, et R. Mahony, « Estimating body-fixed frame velocity and attitude from inertial measurements for a quadrotor vehicle », *Proceedings IEEE Conference on Control Applications*. 978-983, Juan Les Antibes, France, 2014.

Alsakini, S.R., A.T. Alobaidi, et A.J. Sultan, « Turbo generator system identification using genetic algorithm », *Journal of Engineering and Sustainable Development*. 20(6): 12-31, 2016.

Apkarian, P., P. Gahinet, et G. Becker, « Self-scheduled control of linear parameter varying systems: a design example », *Automatica*. 31(9): 1251-1261, 1995.

Apkarian, P., et P. Gahinet, « A convex characterization of gain-scheduled H_∞ controller », *IEEE Transactions on Automatic Control*. 40(5): 853-864, 1995.

Apkarian, P., et R.J. Adams, « Advanced gain-scheduling techniques for uncertain systems », *IEEE Transactions on Control Systems Technology*. 6(1): 21-32, 1998.

Apkarian, P., et H.D. Tuan, « Parameterized LMIs in control theory », *Proceedings IEEE Confernece on Decision and Control*. 152-157, Tampa, Florida, USA, 1998.

Apkarian, P., et D. Noll, « Nonsmooth H_∞ synthesis », *IEEE Transactions on Automatic Control*. 51(1): 71-86, 2006.

Balas, G., R. Chiang, A. Packard, et M.G. Safonov, *Robust Control Toolbox™ the user's guide*. Mathworks. Natick, USA, 2009.

Ball, J., C. Foias, J.W. Helton, et A. Tannenbaum, « On a local nonlinear commutant lifting theorem », *Journal of Mathematics*. 36(3): 693-709, 1987.

Bangura, M., et R. Mahony, « Real-time model predictive control for quadrotors », *IFAC Proceedings Volumes*, 47(3): 11773-11780, 2014.

Bangura, M., R. Mahony, H. Lim, et H.J. Kim, « An open-source implementation of a unit quaternion based attitude and trajectory tracking for quadrotors », *Proceedings Australian Conference on Robotics and Automation*, Melbourne, Australie, 2014.

Bangura, M., R. Naldi, et R. Mahony, « Aerodynamics of rotor blades for quadrotors », <https://arxiv.org/abs/1601.00733>.

Bangura, M., et R. Mahony, « Thrust control for multirotor aerial vehicles », *IEEE Transactions on Robotics*. 33(2): 390-405, 2017.

Bangura, M., *Aerodynamics and Control of Quadrotors*. Thèse de doctorat, The Australian National University, Australie, 2017.

- Başar, T., et P. Bernhard, *H_∞ Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Boston, MA, USA, 2008.
- Beard, R.W., G.N. Saridis, et J.T. Wen, « Galerkin approximations of the generalized hamilton-jacobi-bellman equation », *Automatica*. 33(12): 2159-2177, 1997.
- Beard, R.W., et T.W. McLain, « Successive Galerkin approximation algorithms for nonlinear optimal and robust control », *International Journal of Control*. 71(5): 717-743, 1998.
- Beard, R.W., G. Saridis, et J.T. Wen, « Approximate solutions to the time-invariant Hamilton-jacobi-bellman equation », *Journal of Optimization Theory and Applications*. 96(3): 589-626, 1998.
- Behzad, H., A. Casavola, F. Tedesco, et M.A. Sadrina, « Use of LPV-LFT unknown input observers for the design of fault tolerant sensor reconciliation schemes », *Proceedings Iranian Conference on Electrical Engineering*. 899-905, Mashhad, Iran, 2018.
- Blondel, V., et J.N. Tsitsiklis, « NP-Hardness of some linear control problems », *SIAM Journal on Control and Optimization*. 35(6): 2118-2127, 1997.
- Borges, R.A., et P.L.D. Peres, « H_∞ LPV filtering for linear systems with arbitrarily time-varying parameters in polytopic domains », *Proceedings IEEE Conference on Decision and Control*. San Diego, Californie, USA, 2006.
- Bouabdallah, S., *Design and Control of Quadrotors with Application to Autonomous Flying*. Thèse de doctorat, EPFL, Lausanne, Suisse, 2007.
- Boubaker, O., « The inverted pendulum: a fundamental benchmark in control theory and robotics », *Proceedings International Conference on Education and e-Learning Innovations*. Sousse, Tunisie, 2012.
- Brescianini, D., M. Hehn, et R. D'Andrea, *Nonlinear Quadcopter Attitude Control*. Rapport technique, Department of Mechanical and Process Engineering, ETHZ, Zürich, Suisse, 2013a.
- Brescianini, D., M. Hehn, et R. D'Andrea, « Quadcopter pole acrobatics », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 3472-3479, Tokyo, Japon, 2013b.
- Briat, C., O. Sename, et J.F. Lafay, « H_∞ delay-scheduled control of linear systems with time-varying delays », *IEEE Transactions on Automatic Control*. 42(8): 2255-2260, 2009.
- Briat, C., *Linear Parameter-Varying and Time-Delay Systems: Analysis, Observation, filtering & Control*. Springer-Verlag. Berlin Heidelberg, Allemagne, 2015.
- Bristeau, P.J., P. Martin, E. Salaün, et N. Petit, « The role of propeller aerodynamics in the model of a quadrotor UAV », *Proceedings European Control Conference*. 683-688, Budapest, Hongrie, 2009.
- Bruzelius, F., S. Pettersson, et C. Breitholtz, « Linear parameter-varying descriptions of nonlinear systems », *Proceedings American Control Conference*. 1374-1379, Boston, MA, USA, 2004.
- Burri, M., M. Dätwiler, M.W. Achtelik, et R. Siegwart, « Robust state estimation for micro aerial vehicles based on system dynamics », *Proceedings IEEE International Conference on Robotics and Automation*. 5278-5283, Seattle, WA, USA, 2015.
- Burri, M., M. Bloesch, Z. Taylor, R. Siegwart, et J. Nieto, « A framework for maximum likelihood parameter identification applied on MAVs », *Journal of Field Robotics*. 35(1): 5-22, 2017.
- Carrillo, L., A. Lopez, R. Lozano, et C. Pégard, *Quad Rotorcraft Control*. Springer-Verlag. Londres, UK, 2013.
- Cerone, V., D. Piga, D. Reguto, et R. Tóth, « Fixed order LPV controller design for LPV models in input-output form », *Proceedings IEEE Conference on Decision and Control*. 6297-6302, Hawaii, USA, 2012.

- Chang, Y.C., «Neural network-based H_∞ tracking control for robotic systems », *IEEE Proceedings of Control Theory Applications*. 147(3): 303-311, 2000.
- Chaturvedi, N., A. Sanyal, et N.H. McClamroch, « Rigid-body attitude control », *IEEE Control Systems*. 31(3): 30-51, 2011.
- Chen, B.S., T.S. Lee, et J.H. Feng, « A nonlinear H_∞ control design in robotic systems under parameter perturbation and external disturbance », *International Journal of Control*. 59(2): 439-461, 1994.
- Chen, B.S., Y.C. Chang, et T.C. Lee, « Adaptive control in robotic systems with H_∞ tracking performance », *Automatica*. 33(2): 227-234, 1997.
- Chen, Y.Y., B.S. Chen, et C.S. Tseng, « Adaptive fuzzy mixed H_2/H_∞ lateral control of nonlinear missile systems », *Proceedings IEEE International Conference on Fuzzy Systems*. 512-516, Saint Louis, Missouri, USA, 2003.
- Chen, C., M. Li, J. Sui, K. Wie, et Q. Pei, « A genetic algorithm-optimized fuzzy logic controller to avoid rear-end collisions », *Journal of Advanced Transportation*. 50(8): 1735-1753, 2016.
- Chesi, G., A. Garulli, A. Tesi, et A. Vicino, *Homogeneous Polynomial Forms for Robustness Analysis of Uncertain Systems*. Springer. Berlin, Allemagne, 2009.
- Chircop, K., et D. Zammit-Mangion, « On ε -constraint based methods for the generation of Pareto frontiers », *Journal of Mechanics Engineering and Automation*. 3: 279-289, 2013.
- Choukchou-Braham, A., B. Cherki, M. Djemaï, et K. Busawon, « Classification of Underactuated Mechanical Systems », *Analysis and Control of Underactuated Mechanical Systems*, A. Choukchou-Braham, B. Cherki, M. Djemaï, et K. Busawon (eds.), 35-54. Springer. London, UK, 2014a.
- Choukchou-Braham, A., B. Cherki, M. Djemaï, et K. Busawon, « Control Design Schemes for Underactuated Mechanical Systems », *Analysis and Control of Underactuated Mechanical Systems*, A. Choukchou-Braham, B. Cherki, M. Djemaï, et K. Busawon (eds.), 55-91. Springer. London, UK, 2014b.
- Chung, W., L.C. Fu, et S.H. Hsu, « Motion Control », *Springer Handbook of Robotics*, B. Siciliano, et O. Khatib (eds.), 133-159. Springer-Verlag. Berlin Heidelberg, Allemagne, 2008.
- Corti, A., et M. Lovera, « Attitude regulation for spacecraft with magnetic actuators: an LPV approach », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.), 339-355. Springer. New York, USA, 2012.
- Daher-Adegas, F., C. Solth, et J. Stoustrup, « Structured linear parameter varying control of wind turbines », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.), 303-337. Springer. New York, USA, 2012.
- Das, J., G. Cross, C. Qu, A. Makineni, P. Tokekar, Y. Mulgaonkar, et V. Kumar, « Devices, systems, and methods for automated monitoring enabling precision agriculture », *Proceedings IEEE International Conference on Automation Science and Engineering*. 462-469, Gothenburg, Suède, 2015.
- Dasgupta, D., et D.R. McGregor, « Nonstationary function optimization using the structured genetic algorithm », *Parallel Problem Solving from Nature*. 2: 145-154, 1992.
- Deb, K., et D. Deb, « Analysing mutation schemes for real-parameter genetic algorithms », *International Journal of Artificial Intelligence and Soft Computing*. 14(1):1-28, 2014.
- Debert, M., G. Colin, G. Bloch, et Y. Chamaillard, « An observer looks at the cell temperature in automotive battery packs », *Control Engineering Practice*. 21(8): 1035-1042, 2013.
- De-Crousaz, C., F. Farshidian, M. Neunert, et J. Buchli, « Unified motion control for dynamic quadrotor maneuvers demonstrated on slung load and rotor failure tasks », *Proceedings International Conference on Robotics and Automation*. 2223-2229, Seattle, WA, USA, 2015.

- Din, S.U., Q. Khan, F.U. Rehman, et R. Akmeliawanti, « A comparative experimental study of robust sliding mode control strategies for underactuated systems », *IEEE Access*. 6: 1927-1939, 2018.
- Do, A.L., O. Sename, L. Dugard, et B. Soualmi, « Multi-objective optimization by genetic algorithm in H_∞ /LPV control of semi-active suspension », *Proceedings IFAC World Congress*. 7162-7167, Milan, Italie, 2011.
- Do, A.L., O. Sename, et L. Dugard, « LPV modeling and control of semi-active dampers in automotive systems », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.), 381-411. Springer. New York, USA, 2012.
- Doyle, J., K. Glover, P. Khargonekar, et B. Francis, « State-space solutions to standard H_2 and H_∞ control problem », *IEEE Transactions on Automatic Control*. 34(8): 831-847, 1989.
- Dub, M., et A. Stefek, « Using PSO method for system identification », *Mechatronics 2013: Recent Technological and Scientific Advances*, T. Březina, et R. Jabłoński (eds.), 143-150. Springer. Suisse, 2014.
- Fan, Y., Y. Cao, et T. Li, « Adaptive integral backstepping control for trajectory tracking of a quadrotor », *Proceedings International Conference on Information, Cybernetics and Computational Social Systems*. 619-624, Dalian, Chine, 2017.
- Fässler, M., *Quadrotor Control for Accurate Agile Flight*. Thèse de doctorat, Université de Zurich, Suisse, 2018.
- Feng, W., et I. Postlethwaite, « Robust non-linear H_∞ /adaptive control of robot manipulator motion », *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*. 208(4):221-230, 1994.
- Feng, Y., X. Yu, et Z. Man, « Non-singular terminal sliding mode control of rigid manipulators », *Automatica*. 8(12): 2159-2167, 2002.
- Ferrin, J., R. Leishman, R. Beard, et T. McLain, « Differential flatness based control of a rotorcraft for aggressive maneuvers », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2688-2693, San Francisco, CA, USA, 2011.
- Feyel, P., « Evolutionary fixed-structure LPV/LFT controller synthesis », *IFAC PapersOnLine*. 48(25): 114-119, 2015.
- Fiacchini, M., A. Viguria, R. Cano, A. Prieto, F.R. Rubio, J. Aracil, et C. Canudas-de-Wit, « Design and experimentation of a personal penulum vehicle », *Proceedings Portuguese Conference on Automatic Control*. Lisbonne, Portugal, 2006.
- Firouzeh, A., S.S.M. Salehian, A. Billard, et J. Paik, « An under actuated robotic arm with adjustable stiffness shape memory polymer joints », *Proceedings International Conference on Robotics and Automation*. 2536-2543, Seattle, WA, USA, 2015.
- Forrai, A., T. Ueda, et T. Yumura, « Electromagnetic actuator control: a linear parameter-varying (LPV) approach », *IEEE Transactions on Industrial Electronics*. 554(3): 1430-1441, 2007.
- Franchi, A., et A. Mallet, « Adaptive closed-loop speed control of BLDC motors with applications to multi-rotor aerial vehicles », *Proceedings IEEE International Conference on Robotics and Automation*. 5203-5208, Singapore, 2017.
- Fresk, E., et G. Nikolakopoulos, « Full quaternion based attitude control for a quadrotor », *Proceedings European Control Conference*. 3864-3869, Zürich, Suisse, 2013.
- Fu, M., « Pole placement via static output feedback is NP-hard », *IEEE Transactions on Automatic Control*. 49(5): 855-857, 2004.
- Gahinet, P., et P. Apkarian, « A linear matrix inequality approach to H_∞ control », *International Journal of Robust and Nonlinear Control*. 4(4): 421-448, 1994.

- Gauthier, C., O. Sename, L. Dugard, et G. Meisssonier, « An LFT approach to H_∞ control design for diesel engine common rail injection system », *Oil & Gas Science and Technology – Revue IFP*. 62(4): 513-522, 2007.
- Geromel, J.C., et P. Colaneri, « Robust stability of time-varying polytopic systems », *Systems & Control Letters*. 55: 81-85, 2006.
- Gill, R., et R. D’Andrea, « Propeller thrust and drag in forward flight », *Proceedings IEEE Conference on Control Technology and Applications*. 73-79, Hawai’i, USA, 2017.
- Grabe, V., M. Riedel, H.H. Bühlhoff, P.R. Giordano, et A. Franchi, « The TeleKyb framework for a modular and extendible ROS-based quadrotor control », *Proceedings European Conference on Mobile Robots*. 19-25, Barcelone, Espagne, 2013.
- Hamer, M., M. Waibel, et R. D’Andrea, « Knowledge transfer for high-performance quadcopter maneuvers », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1714-1719, Tokyo, Japon, 2013.
- Hansen, N., D.V. Arnold, et A. Auger, « Evolution strategies », *Springer Handbook of Computational Intelligence*, J. Kacprzyk, et W. Pedrycs (eds.), 871-898. Springer. Berlin, Allemagne, 2015.
- Hardt, M., J.W. Helton, et K. Kreutz-Delgado, « Numerical solution of nonlinear H_2 and H_∞ control problems with application to jet engine compressors », *IEEE Transactions on Control Systems Technology*. 8(1): 98-111, 2000.
- Hashemi, S.M., et H. Werner, « Observed-based LPV control of a nonlinear PDE », *Proceedings IEEE Conference on Decision and Control*. 2010-2015, Orlando, Florida, USA, 2011.
- Hasseni, S., et L. Abdou, « Robust LPV control applied to a personal pendulum vehicle », *Proceedings International Conference on Sciences and Techniques of Automatic control and computer engineering*. 6-11, Monastir, Tunisie, 2017.
- Hasseni, S., et L. Abdou, « Integral Backstepping/LFT-LPV H_∞ control for the trajectory tracking of a quadcopter », *Proceedings International Conference on Systems and Control*. 348-353, Valence, Espagne, 2018.
- Hasseni, S., L. Abdou, et H. Glida, « Parameters tuning of a quadrotor PID controllers by using nature-inspired algorithms », *Evolutionary Intelligence*. 2019. <https://doi.org/10.1007/s12065-019-00312-8>.
- He, B., et M. Yang, « Robust LPV control of diesel auxiliary power unit for series hybrid electric vehicles », *IEEE Transactions on Power Electronics*. 21(3): 791-798, 2006.
- He, Y.Q., et J.D. Han, « Acceleration feedback enhanced H_∞ disturbance attenuation control for a class of nonlinear underactuated vehicle systems », *Acta Automatica Sinica*. 34(5): 558-564, 2008.
- Heemels, W.P.M.H., J. Daafouz, et G. Millerioux, « Observer-based control of discrete-time LPV systems with uncertain parameters », *IEEE Transactions on Automatic Control*. 55(9): 2130-2135, 2010.
- Hehn, M., et R. D’Andrea, « A flying inverted pendulum », *Proceedings IEEE International Conference on Robotics and Automation*. 736-770, Shanghai, Chine, 2011.
- Hehn, M., R. Ritz, et R. D’Andrea, « Performance benchmarking of quadrotor systems using time-optimal control », *Autonomous Robots*, 33(1-2): 69-88, 2012.
- Hehn, M., et R. D’Andrea, « An iterative learning scheme for high performance, periodic quadcopter trajectories », *Proceedings European Control Conference*. 1799-1804, Zürich, Suisse, 2013.
- Hehn, M., et R. D’Andrea, « A frequency domain iterative learning algorithm for high-performance, periodic quadcopter maneuvers », *Mechatronics*. 24(8): 954-965, 2014.
- Hehn, M., et R. D’Andrea, « Real-time trajectory generation for quadcopters », *IEEE Transactions on Robotics*. 31(4): 877-892, 2015.

- Henrion, D., et J.B. Lasserre, « Convergent relaxations of polynomial matrix inequalities and static output feedback », *IEEE Transactions on Automatic Control*. 51(2): 192-202, 2006.
- Hjartarson, A., P. Seiler, et A. Packard, « LPV Tools: A Toolbox for modeling, analysis and synthesis of parameter varying control systems », *IFAC PapersOnLine*. 48(26): 136-145, 2015.
- Hoffmann, G.M., H. Huang, S.L. Waslander, et C.J. Tomlin, « Quadrotor helicopter flight dynamics and control: Theory and experiment », *Proceedings AIAA Guidance, Navigation and Control Conference and Exhibit*. Hilton Head, South Carolina, USA, 2007.
- Hughes, H.D., et F. Wu, « LPV H_∞ control for flexible hypersonic vehicle », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Sherer (eds.), 413-444. Springer. New York, USA, 2012.
- Iwasaki, T., et S. Hara, « Well-posedness of feedback systems: insight into exact robustness analysis and approximate computations », *IEEE Transactions on Automatic Control*. 43(5): 619-630, 1998.
- Jaen-Cuellar, A.Y., R.D.J. Romero-Troncoso, L. Morales-Valazquez, et R.A. Osornio-Rios, « PID-controller tuning optimization with genetic algorithms in servo systems », *International Journal of Advanced Robotic Systems*. 10: 324-337, 2013.
- Jain, T., et M.J. Nigam, « Optimization of PD-PI controller using swarm intelligence », *International Journal of Computational Cognition*. 6(4):55-59, 2008.
- Jimenez-Cano, A.E., J. Martin, G. Heredia, A. Ollero, et R. Cano, « Control of an aerial robot with multi-link arm for assembly tasks », *Proceedings IEEE International Conference on Robotics and Automation*. 4916-4921, Karlsruhe, Allemagne, 2013.
- Jogindar-Singh, K.S.M., I. Elamvazuthi, K.Z.K. Shaari, et N. Perumal, « Development of PID controller tuning tool based on cuckoo search algorithms », *Proceedings IEEE International Symposium in Robotics and Manufacturing Automation*. Kuala Lumpur, Malisie, 2017.
- Johansson, R., « Quadratic optimization of motion coordination and control », *IEEE Transactions on Automatic Control*. 35(11): 1197-1208, 1990.
- Jung, M., et K. Glover, « Calibratable linear parameter-varying control of a turbocharged diesel engine », *IEEE Transactions on Control Systems Technology*. 14(1): 45-62, 2006.
- Kai, J.M., G. Allibert, M.D. Hua, et T. Hamel, « Nonlinear feedback control of quadrotors exploiting first-order drag effects », *IFAC PapersOnLine*. 50(1): 8189-8195, 2017.
- Kamel, M., K. Alexis, M. Achtelik, et R. Siegwart, « Fast nonlinear model predictive control for multicopter attitude tracking on $SO(3)$ », *Proceedings IEEE International Conference on Control Applications*. 1160-1166, Sydney, NSW, Australie, 2015.
- Kawai, Y., et K. Uchiyama, « Design of frequency shaped LQR considering dynamic characteristics of the actuator », *Proceedings International Conference on Unmanned Aircraft Systems*. 1235-1239, Arlington, VA, USA, 2016.
- Khalil, H.K., *Nonlinear Systems*. 2^{ème} édition. Prentice-Hall. Upper Saddle River, NJ, USA, 1996.
- Klempka, R., et B. Filipowicz, « Comparison of using the genetic algorithm and cuckoo search for multicriteria optimisation with limitation », *Turkish Journal of Electrical Engineering and Computer Sciences*. 25(2):1300-1310, 2017.
- Kominek, A., H. Werner, M. Garwon, et M. Schultalbers, « Identification of low-complexity LPV input-output models for control of a turbocharged combustion engine », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.). 445-460, New York, USA, 2012.

- Kondak, K., K. Krieger, A. Albu-Schaeffer, M. Schwarzbach, M. Laiacker, I. Maza, A. Rodriguez-Castano, et A. Ollero, « Closed-loop behavior of an autonomous helicopter equipped with a robotic arm for aerial manipulation tasks », *International Journal of Advanced Robotic Systems*. 10(2): 145-152, 2013.
- Kumar, S., et R. Gil, « Path following for quadrotors », *Proceedings IEEE Conference on Control Technology and Applications*. 2075-2081, Hawai'i, USA, 2017.
- Kwiatkowski, A., et H. Werner, « LPV control of a 2-DOF robot using parameter reduction », *Proceedings IEEE Conference on Decision and Control*. 3369-3374, Seville, Espagne, 2005.
- Kwiatkowski, A., J.P. Blath, H. Werner, A. Ali, et M. Schultalbers, « Linear parameter varying PID controller design for charge control of a spark-ignited engine », *Control Engineering Practice*. 17(11): 1307-1317, 2009.
- Lawrence, D.A., et W.J. Rugh, « Gain scheduling dynamic linear controllers for a nonlinear plant », *Automatica*. 31(3): 381-390, 1995.
- Lee, T., M. Leok, et N.H. McClamroch, « Geometric tracking control of a quadrotor UAV on SE(3) », *Proceedings IEEE Conference on Decision and Control*, 5420-5425, Atlanta, USA, 2010.
- Leishman, R.C., J.C. Macdonald, R.W. Beard, et T.W. McLain, « Quadrotors and accelerometers: State estimation with an improved dynamic model », *IEEE Control Systems*. 34(1): 28-41, 2014.
- Leith, D.J., et W.E. Leithead, « Survey of gain-scheduling analysis and design », *International Journal of Control*. 73(11): 1001-1025, 2000.
- Li, Q., J. Qian, Z. Zhu, X. Bao, M.K. Helwa, et A.P. Schoellig, « Deep neural networks for improved, impromptu trajectory of quadrotors », *Proceedings IEEE International Conference on Robotics and Automation*. 5183-5189, Singapore, 2017.
- Li, M., C. Liu, T. Li, et C.L.P. Chen, « New second order sliding mode control design for course-keeping control of ship with input saturation », *Proceedings International Conference on Advanced Mechatronic Systems*. 462-466, Xiamen, Chine, 2017.
- Liu, Q., V. Vittal, et N. Elia, « Expansion of system operating range by an interpolated LPV FACTS controller using multiple Lyapunov functions », *IEEE Transactions on Power Systems*. 21(3): 1311-1320, 2006.
- Liu, Q., H.S. Abbas, J. Mohammadpour, et H. Werner, « Distributed controller design for LPV/LFT distributed systems in input-output form », *IFAC PapersOnLine*. 50(1): 11409-11414, 2017.
- Liu, B., M. Hou, et Y. Li, « Field-of-view and impact angle constrained guidance law for missiles with time-varying velocities », *IEEE Access*. 7: 61717-61727, 2019.
- López-Martínez, M., M.G. Ortega, et F.R. Rubio, « Nonlinear L_2 control of a laboratory helicopter with variable speed rotors », *Automatica*. 43(4): 655-661, 2007.
- Lu, B., F. Wu, et S. Kim, « Switching LPV control of an F-16 aircraft via controller state reset », *IEEE Transactions on Control Systems Technology*. 14(2): 267-277, 2006.
- Lupashin, S., et R. D'Andrea, « Adaptive fast open-loop maneuvers for quadrocopters », *Autonomous Robots*. 33(1-2): 89-102, 2012.
- Lupashin, S., M. Hehn, M.W. Mueller, A.P. Schoellig, M. Sherback, et R. D'Andrea, « A platform for aerial robotics research and demonstration: the flying machine arena », *Mechatronics*. 24(1): 41-54, 2014.
- Luspay, T., T. Péni, et B. Kulcsár, « Constrained freeway traffic control via linear parameter varying paradigms », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.), 461-482. Springer. New York, USA, 2012.

- Mahony, R., V. Kumar, et P. Corke, « Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor », *IEEE Robotics & Automation Magazine*. 19(3): 20-32, 2012.
- Martin, P., et E. Salaün, « The true role of accelerometer feedback in quadrotor control », *Proceedings IEEE International Conference on Robotics and Automation*. 1623-1639, Anchorage, AK, USA, 2010.
- Mehendale, C.S., et K.M. Grigoriadis, « A new approach to LPV gain-scheduling design and implementation », *Proceedings IEEE conference on Decision and Control*. 2942-2947, Atlantis, Paradise Island, Bahamas, 2004.
- Mellinger, D., et V. Kumar, « Minimum snap trajectory generation and control for quadrotors », *Proceedings IEEE International Conference on Robotics and Automation*. 2520-2525, Shanghai, Chine, 2011.
- Merino, L., F. Caballero, J.R. Martínez-de-Dios, I. Maza, et A. Ollero, « An unmanned aircraft system for automatic forest fire monitoring and measurement », *Journal of Intelligent & Robotic Systems*. 65(1-4): 533-548, 2011.
- Michael, N., D. Mellinger, Q. Lindsey, et V. Kumar, « The GRASP multiple micro-UAV testbed », *IEEE Robotics & Automation Magazine*. 17(3): 56-65, 2010.
- Michieletto, G., M. Ryll, et A. Franchi, « Control of statically hoverable multi-rotor aerial vehicles and application to rotor-failure robustness for hexarotors », *Proceedings IEEE International Conference on Robotics and Automation*. 2747-2752, Singapore, 2017.
- Mohammadpour, J., et C.W. Scherer, *Control of Linear Parameter Varying Systems with Applications*. Springer. New York, USA, 2012.
- Molugaram, K., et G. Shanker-Rao, « Curve fitting », *Statistical Techniques for Transportation Engineering*, K. Molugaram, et G. Shanker-Rao (eds.), 281-292. Elsevier. Indie, 2017.
- Monfared, S.B., A. Kalhor, et M.A. Atashgah, « Robust adaptive H_∞ control for attitude control of a quadrotor », *Proceedings International Conference on Robotics and Mechatronics*. 560-565, Tehran, Iran, 2018.
- Monteiro, J.C., F. Lizarralde, et L. Hsu, « Optimal control allocation of quadrotor UAVs subject to actuator constraints », *Proceedings American Control Conference*. 500-505, Boston, MA, USA, 2016.
- Mueller, M.W., et R. D'Andrea, « A model predictive controller for quadcopter state interception », *Proceedings European Control Conference*. 1383-1389, Zürich, Suisse, 2013.
- Mueller, M.W., et R. D'Andrea, « Relaxed hover solutions for multicopters: application to algorithmic redundancy and novel vehicles », *The International Journal of Robotics Research*. 35(8): 873-889, 2015.
- Mulgaonkar, Y., G. Cross, et V. Kumar, « Design of small, safe and robust quadrotor swarms », *Proceedings International Conference on Robotics and Automation*. 2208-2215, Seattle, WA, USA, 2015.
- Murphy, R.R., *Disaster Robotics*. MIT Press. Cambridge, MA, USA, 2014.
- Nazarahari, M., E. Khanmirza, et S. Doostie, « Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm », *Expert Systems with Applications*. 115: 106-120, 2019.
- Olfati-Saber, R., *Nonlinear Control of Underactuated Mechanical Systems with Application to Robotics and Aerospace Vehicles*. Thèse de doctorat, MIT, Boston, MA, USA, 2001.
- Oliveira, R.C.L.F., V.F. Montagner, P.L.D. Peres, et P.A. Bliman, « LMI relaxations for H_∞ control of time-varying polytopic systems by means of parameter-dependent quadratically stabilizing gains », *IFAC Proceedings Volume*. 40(20): 614-619, 2007.
- Omari, S., M.D. Hua, G. Ducard, et T. Hamel, « Nonlinear control of VTOL UAVs incorporating flapping dynamics », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2419-2425, Tokyo, Japon, 2013.

- Ortega, M.G., M. Vargas, C. Vivas, et F.R. Rubio, « Robustness improvement of a nonlinear H_∞ controller for robot manipulators via saturation functions », *Journal of Robotic Systems*. 22(8):421-437, 2005.
- Ouaarab, A., B. Ahiod, et X.S. Yang, « Discrete cuckoo search algorithm for the travelling salesman problem », *Neural Computing and Applications*. 24(7-8):1659-1669, 2014.
- Özaslan, T., G. Loianno, J. Keller, C.J. Taylor, V. Kumar, J.M. Wozencraft, et T. Hood, « Autonomous navigation and mapping for inspection of penstocks and tunnels with UAVs », *IEEE Robotics and Automation Letters*. 2(3):1740-1747, 2017.
- Packard, A., « Gain scheduling via linear fractional transformations », *Systems & Control Letters*. 22(2): 79-92, 1994.
- Pasqualetti, F., A. Franchi, et F. Bullo, « On cooperative patrolling: optimal trajectories, complexity analysis, and approximation algorithms », *IEEE Transactions on Robotics*. 28(3): 592-606, 2012.
- Perez, D., I. Maza, F. Caballero, D. Scarlatti, E. Casado, et A. Ollero, « A ground control station for a multi-UAV surveillance system », *Journal of Intelligent & Robotic Systems*, 69(1-4): 119-130, 2012.
- Pfifer, H., et P. Seiler, « Robustness analysis of linear parameter varying systems using integral quadratic constraints », *International Journal of Robust and Nonlinear Control*. 25(15): 2843-2864, 2015.
- Postlethwaite, I., et A. Bartoszewicz, « Application of nonlinear H_∞ control to the tetrabot robot manipulator », *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*. 212(6): 459-465, 1998.
- Pounds, P., R. Mahony, et P. Corke, « Modelling and control of a quad-rotor robot », *Proceedings Australian Conference on Robotics and Automation*. Auckland, Nouvelle-Zélande, 2006.
- Poussout-Vassal, C., C. SPelta, O. Sename, S.M. Savaresi, et L. Dugard, « Survey and performance evaluation on some automotive semi-active suspension control methods – a comparative study on a single-corner model », *Annual Reviews in Control*. 36(1): 148-160, 2012.
- Powers, C., D. Mellinger, A. Kushleyev, B. Kothmann, et V. Kumar, « Influence of aerodynamics and proximity effects in quadrotor flight », *Proceedings International Symposium on Experimental Robotics*. 289-302, Québec, Canada, 2012.
- Prattichizzo, D., M. Malvezzi, M. Gabiccini, et A. Bicchi, « On the manipulability ellipsoids of underactuated robotic hands with compliance », *Robotics and Autonomous Systems*. 60: 337-346, 2012.
- Prempain, E., et I. Postlethwaite, « Static H_∞ loop shaping control of a fly-by-wire helicopter », *Automatica*. 41(9): 1517-1528, 2005.
- Price, K., R.M. Storn, et J.A. Lampinen, *Differential Evolution: A Survey of the State-of-the-Art*. Springer-Verlag. Berlin, Allemagne, 2005.
- Raffo, G.V., M.G. Ortega, et F.R. Rubio, « MPC with nonlinear H_∞ control for path tracking of a quad-rotor helicopter », *IFAC Proceedings Volumes*. 41(2): 8564-8569, 2008.
- Raffo, G.V., M.G. Ortega, et F.R. Rubio, « Path tracking of a UAV via an underactuated H_∞ control strategy », *European Journal of Control*. 17(2):194-213, 2011.
- Raffo, G.V., M.G. Ortega, et F.R. Rubio, « Robust nonlinear control for path tracking of a quad-rotor helicopter », *Asian Journal of Control*. 17(1):142-156, 2015.
- Ranjitham, G., et K.R. Shankar-Kumar, « Large scale multiple-input multiple-output (LS-MIMO) detection using genetic cat swarm optimization », *International Journal of Advanced Engineering Technology*. 7(2): 536-541, 2016.

- Reyes-Valeria, E., R. Enriquez-Caldera, S. Camacho-Lara, et J. Guichard, « LQR control for a quadrotor using unit quaternions: modeling and simulation », *Proceedings International Conference on Electronics, Communications and Computing*. 172-178, Cholula, Mexique, 2013.
- Reyhanoglu, M., A.J. van der Schaft, N. McClamroch, et I. Kolmanovsky, « Dynamics and control of a class of underactuated mechanical systems », *IEEE Transactions on Automatic Control*. 44(9): 1663-1671, 1999.
- Rhimian, M.A., et M.S. Tavazoei, « Improving integral square error performance with implementable fractional-order PI controllers », *Optimal Control Applications and Methods*. 35(3):303-323, 2014.
- Ritz, R., M. Hehn, S. Lupashin, et R. D'Andrea, « Quadrocopter performance benchmarking using optimal control », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5179-5186, San Francisco, CA, USA, 2011.
- Robert, D., O. Sename, et D. Simon, « Synthesis of a sampling period dependent controller using LPV approach », *IFAC Proceedings Volumes*. 39(9): 226-231, 2006.
- Rodrigues, M., M. Sahnoun, D. Theilliol, et J.C. Ponsart, « Sensor fault detection and isolation filter for polytopic LPV systems: A winding machine application », *Journal of Process Control*. 23(6): 805-816, 2013.
- Rout, B., B.B. Pati, et S. Panda, « Modified SCA algorithm for SSSC damping controller design in power system », *ECTI Transactions on Electrical Engineering, Electronics and Communications*. 16(1):46-63, 2018.
- Rudra, S., R.K. Barai, et M. Maitra, « Block Backstepping Control of the Underactuated Mechanical Systems », *Block Backstepping Design of Nonlinear State Feedback Control Law for Underactuated Mechanical Systems*, S. Rudra, R.K. Barai, et M. Maitra (eds.), 31-52. Springer-Verlag. Singapore, 2017.
- Rugh, W.J., et J.S. Shamma, « Research on gain scheduling », *Automatica*. 36(10): 1401-1425, 2000.
- Ryll, M., G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, et A. Franchi, « 6D physical interaction with a fully actuated aerial robot », *Proceedings IEEE International Conference on Robotics and Automation*. 5190-5195, Singapore, 2017.
- Sage, H., M. de Mathelin, et E. Ostertag, « Robust control of robot manipulators: a survey », *International Journal of Control*. 72(16): 1498-1522, 1999.
- Scherer, C.W., P. Gahinet, et M. Chilali, « Multiobjective output-feedback control via LMI optimization », *IEEE Transactions on Automatic Control*. 42(7): 896-911, 1997.
- Scherer, C.W., « LPV control and full block multipliers », *Automatica*. 37(3): 361-375, 2001.
- Scherer, C.W., et I.E. Köse, « Gain-scheduled control synthesis using dynamic D-scales », *IEEE Transactions on Automatic Control*. 57(9): 2219-2234, 2012.
- Schoellig, A.P., C. Wiltsche, et R. D'Andrea, « Feed-forward parameter identification for precise periodic quadrocopter motions », *Proceedings IEEE American Control Conference*. 4313-4318, Montréal, QC, Canada, 2012a.
- Schoellig, A.P., F.L. Mueller, et R. D'Andrea, « Optimization-based iterative learning for precise quadrocopter trajectory tracking », *Autonomous Robots*. 33(1-2): 103-127, 2012b.
- Schulz, M., F. Augugliaro, R. Ritz, et R. D'Andrea, « High-speed, steady flight with a quadrocopter in a confined environment using a tether », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1279-1284, Hamburg, Allemagne, 2015.
- Seiler, P., G. Balas, et A. Packard, « Linear parameter-varying control for the X-53 active aeroelastic wing », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.), 483-512. Springer. New York, USA, 2012.

- Sename, O., P. Gáspár, et J. Bokor, *Robust Control and Linear Parameter Varying Approaches – Application to Vehicle Dynamics*. Springer. Berlin, Allemagne, 2013.
- Seto, D., et J. Baillieul, «Control problem in super-articulated mechanical systems », *IEEE Transactions on Automatic Control*. 39(14): 2442-2453, 1994.
- Shahemabadi, A.R., S.B.M. Noor, et F.S. Taip, « Analytical formulation of the integral square error for linear stable feedback control system », *Proceedings IEEE International Conference on Control System, Computing and Engineering*. Penang, Malaisie, 29 Novembre 2013.
- Shamma, J.S., *Analysis and design of gain-scheduled control systems*. Thèse de doctorat, MIT, Boston, USA, 1998.
- Shamma, J.S., et M. Athans, « Guaranteed properties of gain scheduled control control of linear parameter-varying plants », *Automatica*. 27(3): 559-564, 1991.
- Shamma, J.S., et M. Athans, « Gain scheduling: potential hazards and possible remedies », *IEEE Control Systems Magazine*. 12(3): 101-107, 1992.
- Shin, J.Y., « Analysis of linear parameter varying system models based on reachable sets », *Proceedings American Control Conference*. 35-40, Anchorage, AK, USA, 2002.
- Siqueira, A.A.G., et M.H. Terra, « Nonlinear H_∞ control for underactuated manipulators with robustness tests », *Revista de Controle & Automação*. 15(3): 339-350, 2004a.
- Siqueira, A.A.G., et M.H. Terra, « Nonlinear and Markovian H_∞ controls of underactuated manipulators », *IEEE Transactions on Control Systems Technology*. 12(6): 811-826, 2004b.
- Siqueira, A.A.G., M.H. Terra, et B.C.O. Maciel, « Nonlinear mixed H_2/H_∞ control applied to manipulators via actuation redundancy », *Control Engineering Practice*. 14(4): 327-335, 2006.
- Siqueira, A.A.G., et M.H. Terra, « Mixed model-based/neural network H_∞ impedance control of constrained manipulators », *Proceedings International Conference on Control and Automation*. 1901-1906, Christchurch, Nouvelle-Zelande, 2009.
- Siqueira, A.A.G., M.H. Terra, et M. Bergerman, *Robust Control of Robots*. Springer-Verlag. London, UK, 2011.
- Sivalingam, R., S. Chinnamuthu, et S.S. Dash, « A hybrid stochastic fractal search and local unimodal sampling based multistage PDF plus (1+PI) controller for automatic generation control of power systems », *Journal of the Franklin Institute*. 354(12):4762-4783, 2017.
- Spong, M.W., « Partial feedback linearization of underactuated mechanical systems », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 314-321, Munich, Allemagne, 1994.
- Spong, M.W., « Underactuated mechanical systems », *Control Problem in Robotics and Automation*, B. Siciliano, et K.P. Valavanis (eds.), 135-150. Springer. Berlin Heidelberg, Allemagne, 1998.
- Spong, M.W., S. Hutchinson, et M. Vidyasagar, *Robot Modeling and Control*. John Wiley & Sons. New York, USA, 2006.
- Storn, R.M., et K. Price, « Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces », *Journal of Global Optimization*. 11(4):341-359, 1997.
- Suresh, M., et R. Meenakumari, « Software tools for analyzing the integration of various renewable energy systems », *International Journal of Industrial Engineering*. 2(6): 135-140, 2018.
- Svacha, J., K. Mohta, et V. Kumar, « Improving quadrotor trajectory tracking by compensating for aerodynamic effects », *Proceedings International Conference on Unmanned Aircraft Systems*. 860-866, Miami, FL, USA, 2017.

- Syrmos, V.L., C.T. Abdallah, P. Dorato, et K. Grigoriadis, « Static output feedback –a survey », *Automatica*. 33(2): 125-137, 1997.
- Szabó, Z., P. Gáspár, et J. Bokor, « Design of integrated vehicle chassis control based on LPV methods », *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour, et C.W. Scherer (eds.), 513-534. Springer. New York, USA, 2012.
- Takase, N., J. Botzheim, et N. Kubota, « Robot edutainment on walking motion of multi-legged robot », *Proceedings International Conference on Robot, Vision and Signal Processing*. 229-233, Kitakyushu, Japon, 2013.
- Tamilchelvan, P., et M. Arounassalame, « Robust control design for wind energy conversion system using Genetic Algorithm (GA) tuned fractional order controller », *Proceedings International Conference on Circuit, Power and Computing Technologies*. Nagercoil, Inde, 2016.
- Tognon, M., A. Testa, E. Rossi, et A. Franchi, « Takeoff and landing on slopes via Inclined hovering with a tethered aerial robot », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1702-1707, Daejeon, Korea, 2016.
- Tóth, R., *Modeling and identification of linear parameter-varying systems an orthonormal basis function approach*. Thèse de doctorat, Technische Universiteit Delft, Pays-bas, 2008.
- Tóth, R., *Modeling and Identification of Linear Parameter-Varying Systems*. Springer. Berlin Heidelberg, Allemagne, 2010.
- Toussaint, G., T. Başar, et F. Bullo, « H_∞ -optimal control techniques for nonlinear underactuated systems », *Proceedings IEEE Confernece on Decision and Control*. 2078-2083, Sydney, Australie, 2000.
- Valdez, F., J.C. Vazquez, et F. Gaxiola, « Fuzzy dynamic parameter adaptation in ACO and PSO for designing fuzzy controllers: the cases of water level and temperature control », *Advances in Fuzzy Systems*. 2018.
- Van der Schaft, A.J., « L_2 -gain analysis of nonlinear systems and nonlinear stat feedback control », *IEEE Transactions on Automatic Control*. 37(6): 770-784, 1992.
- Van der Schaft, A.J., *L_2 Gain and Passivity Techniques in Nonlinear Control*. Springer-Verlag. New York, USA, 2000.
- Venkataraman, S.T., et S. Gulati, « Control of nonlinear systems using terminal sliding modes », *Proceedings American Control Conference*. 891-893, Chicago, IL, USA, 1992.
- Vu, V.T., O. Sename, L. Dugard, et P. Gaspar, « Multi objective H_∞ active-roll bar control for heavy vehicles », *IFAC PapersOnLine*. 50(1): 13802-13807, 2017.
- Wang, B., K.A. Ghamry, et Y. Zhang, « Trajectory tracking and attitude control of an unmanned quadrotor helicopter considering actuator dynamics », *Proceedings Chinese Control Conference*. 10798-10800, Chengdu, Chine, 2016.
- Wei, X., et L. del Re, « Gain scheduled H_∞ control for air path systems of diesel engines using LPV techniques », *IEEE Transactions on Control Systems Technologies*. 15(3): 406-415, 2007.
- Weston, P.F., et I. Postlethwaite, « Linear conditioning for systems containing suturing actuators », *Automatica*. 36(9): 1347-1354, 2000.
- Wichlund, K.Y., O.J. Sordalen, et O. Egeland, « Control properties of underactuated vehicles », *Proceedings IEEE International Conference on Robotics and Automation*. 2009-2014, Nagoya, Japon, 1995.
- White, B.A., L. Bruyere, et A. Tsourdos, « Missile autopilot design using quasi-LPV polynomial eigenstructure assignment », *IEEE Transactions on Aerospace Electronic System*. 43(4): 1470-1483, 2007.
- Wright, A., *Genetic Algorithms for Real Parameter Optimization*. Morgan Kaufmann. San Mateo, USA, 1991.

- Yang, C.D., et H.Y. Chen, « Three-dimensional nonlinear H_∞ guidance law », *International Journal of Robust and Nonlinear Control*. 11(2): 109-129, 2001.
- Yang, X.S., et S. Deb, « Cuckoo search via Lévy flights », *World Congress on Nature and Biologically Inspired Computing*. Coimbatore, Inde, 09 Décembre 2009.
- Yang, X.S., et S. Deb, « Multiobjective cuckoo search for design optimization », *Computers and Operations Research*. 40(6):1616-1624, 2013.
- Yang, X.S., *Nature-Inspired Optimization Algorithms*. Elsevier. Londres, UK, 2014.
- Yazgan, H.R., F. Yener, S. Soysal, et A.E. Gür, « Comparison performances of PSO and GA to tuning PID controller for the DC motor », *Sakarya University Journal of Science*. 23(2): 162-174, 2019.
- Yu, X., et Z. Man, « Fast terminal sliding-mode control design for nonlinear dynamical systems », *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*. 49(2): 261-264, 2002.
- Yu, Y., S. Yang, M. Wang, C. Li, et Z. Li, « High performance full attitude control of a quadrotor on $SO(3)$ », *Proceedings IEEE International Conference on Robotics and Automation*. 1698-1703, Seattle, WA, USA, 2015.
- Yuebang, H., P. Hailong, et Z. Hongbo, « Structured H_∞ controller design for unmanned helicopters using H_∞ extended static output feedback control », *Proceedings Chinese Control Conference*. Hefei, Chine, 2012.
- Yüksel, B., G. Buondonno, et A. Franchi, « Differential flatness and control of protocentric aerial manipulators with any number of arms and mixed rigid/elastic-joints », *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. 561-566, Daejeon, Korea, 2016.
- Zames, G., « On the input-output stability of time-varying nonlinear feedback systems Part one: Conditions derived using concepts of loop gain, conicity, and positivity », *IEEE Transactions on Automatic Control*. 11(2): 228-238, 1966.
- Zinober, A.S.I., *Deterministic Control of Uncertain Systems*. Peter Peregrinus Press. London, UK, 1990.
- Zhang, Q., M. Yuan, et R. Song, « Robot trajectory planning method based on genetic chaos optimization algorithm », *Proceedings International Conference on Advanced Robotics*. 602-607, Hong Kong, Chine, 2017.
- Zhou, K., et J.C. Doyle, *Essentials of Robust Control*. Parentice Hall. New Jersey, USA, 1998.
- Zhou, S., M.K. Helwa, et A.P. Schoellig, « Design of deep neural networks as add-on blocks for improving impromptu trajectory tracking », <https://arxiv.org/abs/1705.10932>.