République Algérienne Démocratique et Populaire Ministère de l'Enseignement Supérieur et de la Recherche Scientifique Ecole Nationale Polytechnique





Département Automatique Laboratoire de commande de processus

## **THESE de Docteur en AUTOMATIQUE**

# Commande non linéaire robuste par forwarding d'un quadrirotor étude et application

## **OUENDI Fatima**

## Sous la direction de Mr Mohamed TADJINE Présentée et soutenue publiquement le 06/07/2022

#### **Composition du Jury**

Président : Directeur : Co-Directeur de thèse : Examinateurs : Mr Lazhari NEZLIProfesseur, ENPMr Mohamed TADJINEProfesseur, ENPMr Hachemi CHEKIREBProfesseur, ENPMr Khelifa BENMANSOURProfesseur, ESDATMr Mouloud BOUCHOUCHAMaitre de conférences/A, EMPMr Ouahid BOUCHHIDAProfesseur, université. MédéaMr Rachid ILLOULMaitre de conférences/A, ENP

ENP 2022

République Algérienne Démocratique et Populaire Ministère de l'Enseignement Supérieur et de la Recherche Scientifique Ecole Nationale Polytechnique





Département Automatique Laboratoire de commande de processus

## **THESE de Docteur en AUTOMATIQUE**

# Commande non linéaire robuste par forwarding d'un quadrirotor étude et application

## **OUENDI Fatima**

## Sous la direction de Mr Mohamed TADJINE Présentée et soutenue publiquement le 06/07/2022

#### **Composition du Jury**

Président : Directeur : Co-Directeur de thèse : Examinateurs :

Mr Lazhari NEZLI	Professeur, ENP
Mr Mohamed TADJINE	Professeur, ENP
Mr Hachemi CHEKIREB	Professeur, ENP
Mr Khelifa BENMANSOUR	Professeur, ESDAT
Mr Mouloud BOUCHOUCHA	Maitre de conférences/A, EMP
Mr Ouahid BOUCHHIDA	Professeur, université. Médéa
Mr Rachid ILLOUL	Maitre de conférences/A, ENP

ENP 2022

#### **Remerciements**

Ce travail a été réalisé au sein du Laboratoire de commande des processus LCP de l'Ecole Nationale Polytechniques d'Alger.

En tout premier lieu, je remercie le bon Dieu, tout puissant de m'avoir accordé la puissance et la volonté pour achever ce travail.

Je souhaite exprimer ma profonde gratitude et ma plus vive reconnaissance à mon directeur de thèse Monsieur TADJINE Mohamed, Professeur à l'Ecole Nationale Polytechniques pour sa pleine contribution tout au long de ce travail et pour la qualité de son encadrement scientifique. C'est avec beaucoup de chance que j'ai eu l'honneur et le plaisir de travailler sous sa direction. Mes premiers, mes plus vifs et profonds remerciements lui sont adressés.

Je tiens également à exprimer ma profonde gratitude à mon co-directeur de thèse, le professeur CHEKIREB Hachemi. Depuis le début de ma thèse il a été très présent et m'a accueillie avec gentillesse et patience à chaque fois que je venais le consulter.

Je remercie infiniment Monsieur HOUTI Smail doctorant à l'UVSQ de Versailles pour sa précieuse collaboration. Je ne peux oublier sa générosité, sa patience et ses encouragements. Je tiens à également à exprimer ma reconnaissance à Monsieur MOKHTARI Khalil pour son aide précieuse durant la période du stage. Je tiens à remercier Monsieur BENALLEGUE Abdelaziz Professeur à l'université de Versailles, de m'avoir accueillie dans son laboratoire et pour son aide.

J'exprime ma reconnaissance à Monsieur NEZLI Lazhari Professeur à l'ENP d'Alger, de m'avoir fait l'honneur d'accepter de présider le jury de cette thèse. J'exprime également mes sincères remerciements à Monsieur BENMANSOUR Khelifa, Professeur, à l'ESDAT, Monsieur BOUCHOUCHA Mouloud MCA à l'EMP d'Alger Monsieur BOUCHHIDA Ouahid, Professeur, à l'université de Médéa et Monsieur ILLOUL Rachid Maitre de conférences/A, à l'ENP d'Alger, pour l'intérêt qu'ils ont témoigné à ce travail et pour avoir accepté de faire partie du jury de cette thèse. La réalisation de cette thèse ne saurait être possible sans le soutien de mes proches. Ainsi, c'est avec un grand plaisir et reconnaissance, que je remercie ma famille pour leur présence, encouragement et patience.

Mes remerciements vont également à mes amies, et mes collègues de l'UMMTO en particulier Ouerdia CHILALI et Ouerdia MEGHERBI, ainsi que les professeurs GUERMAH Saïd, MAIDI Ahmed pour leur aide très précieuse.

A tous ceux et celles qui ont contribué de près ou de loin, par leurs conseils, leurs suggestions et par leurs encouragements, à la réalisation de ce travail.

A ma mère A mon défunt père

#### الملخص

تلعب الطائرات بدون طيار دورًا مهمًا بشكل متزايد في العديد من المجالات، لا سيما الروبوت الطائر ذو أربع مراوح. إنها منصة البحث الأكثر استخدامًا بتمثل هذه الروبوتات الجوية حاليًا الحل الأمثل لتحقيق تحليق ثابت والتحليق إلى الأمام بتتاول هذه الأطروحة مشكلة التحكم في التوجه والموضع لروبوت الطائر ذو أربع مراوح، مع الأخذ بعين الإعتبار الظواهر المختلفة هذه الأطروحة مشكلة التحكم في التوجه والموضع لروبوت الطائر ذو أربع مراوح، مع الأخذ بعين الإعتبار الظواهر المختلفة وهذه الأطروحة مشكلة التحكم في التوجه والموضع لروبوت الطائر ذو أربع مراوح، مع الأخذ بعين الإعتبار الظواهر المختلفة ولا يمكن أن تؤثر على ديناميكياته بتكمن الصعوبة في قيادة هذا النوع من المركبات في ضمان تتبع المسا ر مع رفض الاضطراب. نقدم تقنية تحكم غير خطية و هي عبارة عن التحكم PID\_Backstepping الهجين والتحكم والاموم الأمر المؤمل المؤمل الأمر وتقنية أمر إعادة التوجيه. ثانيًا، اختبرنا هذه التقنيات المختلفة في المحاكاة على PID\_Backstepping وقناية أمر إعادة التوجيه. ثانيًا، اختبرنا هذه التقنيات المختلفة في المحاكاة على PID\_Backstepping وتقنية أمر إعادة التوجيه. ثانيًا، اختبرنا هذه التقنيات المختلفة في المحاكاة على على الاستقرار ليهتين المقترحتين على وتقنية أمر إعادة التوجيه. تانيًا الخبي مراوح للمختبر في الاتجاه بيعتمد تحليل الاستقرار ليهاتين المتحكم PID\_Backstepping وتقنية أمر إعلى يوبوت الطائر ذو أربع مراوح للمختبر في الاتجاه وعادة التوجيه وكذلك عنصر التحكم والتوجيين على نظرية vaلي يعمل على استقرار الروبوت الطائر ذو أربع مراوح بعد المسار المطلوب والذي يرفض تمامًا الاضطرابات الخارجية والذي يعمل على استقرار الروبوت الطائر ذو أربع مراوح بعد المسار المطلوب والذي يرفض تمامًا الاضطرابات الخارجية والذي يعمل على استخدام التحكم في تعدكم والا والد المولية في الملوب والذي يرفض الاضلور والارجين على والذي يحمل ملي يعمل على استخدام التحكم في تعرفي وحدة التحكم ووردة التوجيه وتسمح لنا هذه التقنية بتبني حل وسط بين تعويض والداخلية. يتم الدي ير معام الحام وي تشكيل الحلقات في وحدة التحكم وورموحة الموروب والحي وولي والوقت الحقي والداخلية والداخلية. يتم معام ملي ما مع مالذوب الحابية والداخلات أو المخرجات تم الينفية في الوقعا ليروبم والمولي المومة ووحدة تحكم طيران المودية والموارابات ال

بع مراوح، PID\_Backstepping ، Lyapunov التقدمي الكلمات الرئيسية: الروبوت الطائر ذو أربع مراوح، PIXHAWK الأنظمة اللاخطية ، رفض الاضطراب

#### Abstract

Nowadays, we see a growing popularity of UAV (Unmanned Aerial Vehicles), especially quadrotors. They are the mostly used ones as a research platform. These aerial robots currently represent the most efficient solution to achieve hovering and advancing flight.

This thesis deals with attitude and position control of quadrotors taking into account the various physical phenomena, which can affect the dynamics of the drone. The difficulty in the piloting of this kind of vehicles is to ensure good performances of trajectory tracking while guaranteeing a rejection of the disturbances. We present a technique of nonlinear control by forwarding. This control allows the navigation of the quadricopter in attitude and in position by adopting the architecture in cascade. Then, we highlight a hybrid control, combining the control in position by the backstepping technique and the control in attitude by the PID regulator. The stability analysis of these two proposed controllers is based on the Lyapunov theory. Simulation results show the efficiency of the forwarding control as well as the PID\_Backstepping control which achieve the desired trajectory tracking and rejection of both external and internal disturbances. The loopshaping control is used on the PID controller, this technique allows us to adopt a

compromise between noise compensation such as modeling uncertainties and input or output disturbances. The real time implementation of the orientation control with a PID controller has been done on a LISV quadrirotor. This quadrotor is equipped with a PIXHAWK flight controller with PX4 software.

**Key words:** quadrotor, forwarding control, PID\_Backstepping control, Lyapunov, nonlinear systems, rejection of disturbance, loopshaping, robust control, Pixhawk, PX4.Loopshaping.

#### Résumé :

Les drones, spécialement les quadrirotors, jouent un rôle de plus en plus important dans beaucoup de domaines. Ils constituent la plateforme de recherche la plus utilisée. Ces robots aériens représentent actuellement la solution la plus performante pour réaliser le vol stationnaire et le vol d'avancement. Cette thèse traite le problème de la commande d'orientation et de position, en tenant compte des différents phénomènes qui peuvent influencer la dynamique du quadcopter. La difficulté dans le pilotage de ce genre de véhicules est d'assurer de bonnes performances de suivi de trajectoires, tout en garantissant un rejet de perturbation. En premier lieu, nous avons développé des correcteurs basés sur des techniques linéaires et non linéaires qui sont la commande hybride PID\_Backstepping, la commande loopshaping et la technique de commande forwarding. En deuxième lieu, nous avons testé ces différentes techniques en simulation sur Matlab-Simulink et nous avons appliqué la commande PID sur le quadrirotor de laboratoire en orientation. L'analyse de la stabilité des deux contrôleurs non linéaires proposés, la commande forwarding et la commande combinée PID\_backstepping, est basée sur la théorie de Lyapunov. Les résultats de simulation montrent l'efficacité de ces commandes en poursuite de la trajectoire désirée, et en rejet de perturbations externes et internes. La commande loopshaping est utilisée sur le contrôleur PID. Cette technique nous permet d'adopter un compromis entre la compensation du bruit, comme les incertitudes de modélisation, et les perturbations en entrée ou en sortie. L'implémentation en temps réel de la commande en orientation avec un régulateur PID a été effectuée sur un quadrirotor de LISV. Ce quadrirotor est doté d'un contrôleur de vol PIXHAWK muni du logiciel PX4.

**Mots clé :** quadrirotor, commande forwarding, PID\_Backstepping, Lyapunov, systèmes non linéaires, rejet de perturbations, remodelage de boucle, commande robuste, Pixhawk, PX4.

#### TABLE DES MATIÈRES

Inti	Introduction générale	
	Chapitre 1: Modélisation du quadrirotor et état de l'art sur sa commande	
1. I	Introduction	20
1.1	MOUVEMENTS DU QUADRIROTOR	20
	1.1.1 Mouvement vertical	22
	1.1.2 Mouvement de roulis	22
	1.1.3 Mouvement de tangage	23
	1.1.4 Mouvement de lacet	23
	1.1.5 Mouvement de translation	24
1.2	Modèle dynamique du quadrirotor	25
	1.2.1 Hypothèses simplificatrices	25
	1.2.2 Angles d'Euler	26
	1.2.3 Vitesses angulaires	27
	1.2.4 Vitesses linéaires	28
1.3	Effets physiques agissant sur le quadrirotor	28
	1.3.1 Forces	28
	1.3.2 Moments	29
	1.3.3 Développement du modèle mathématique selon Newton-Euler	31
	Equation du mouvement de translation	33
	1.3.4 Equation du mouvement de rotation	33
1.4	Relation entre forces, moments et vitesse des moteurs	34
1.5	Représentation d'état du système	35
1.6	Dynamique du moteur	37
1.7	Stratégie de commande en cascade du quadrirotor	37
1.8	Etat de l'art sur la commande des drones quadrirotors	38

Chapitre 2: Rappels théoriques sur les commandes robustes linéaires et non linéaires	
2 Introduction	
2.1 Commande Forwarding	
2.1.1 Etapes de synthèse	
2.2 Commande par backstepping	
2.2.1 Etapes de synthèse	
2.2.2 Principe de la commande backstepping46	
2.2.3 Objectifs de la commande backstepping :	
2.2.4 Avantages de la commande backstepping	
2.2.5 Domaines d'application du backstepping	
2.2.6 Exemple d'application	
2.3 Régulateur proportionnel-intégrale-dérivée PID	
2.3.1 Linéarisation du modèle autour d'un point d'équilibre	
2.4 Méthode du loopshaping	
2.4.1 Notion de modelage de boucle (loopshaping)	
2.4.2 Spécifications de performance par loopshaping	
2.4.3 Exemple de Loopshaping	
2.5 Etude de la robustesse des commandes	
2.6 Conclusion	
Chapitre 3: Application et simulation sur le modèle d'état du quadrirotor	
3 Introduction	
3.1 Synthèse des lois de commande forwarding pour le quadrirotor	
3.1.1 Résultats de simulation64	
3.1.1.1 Test de vol sans perturbations externes	
3.1.1.2 Test de vol en présence des perturbations externes	
3.2 Synthèse des lois de commande PID_Backstepping pour le quadrirotor75	
3.2.1 Commande en position par la commande Backstepping	

3.3	Commande en attitude par la technique PID	. 78
	3.3.1 Expression du modèle linéarisé	. 78
	3.3.2 Commande en attitude et altitude par PID	. 78
	3.3.3 Lois de commande par PID_Backstepping	. 79
	3.3.4 Résultats de simulation de la commande PID_backstepping	. 79
3.4	Conclusion	. 89

### Chapitre 4: Mise en œuvre expérimentale

4 Introduction	91
4.1 Présentation du banc d'essai	92
4.1.1 Contrôleur de vol pixhawk	93
4.1.2 Structure logicielle	95
4.1.3 Les moteurs sans balais (brushless motors)	96
4.1.4 Variateurs de vitesse (ESC)	96
4.1.5 Batterie lipo	97
4.1.6 Hélices	97
4.1.7 Radiocommande	98
4.2 Récupération des données du drone vers MATLAB/Simulink	. 101
4.2.1 Validation expérimentale de la commande PID en attitude	101
4.3 Conclusion	104
Conclusion générale	. 106
Références bibliographiques	.109

## LISTE DES FIGURES

Figure 1- 1: Représentation du quadrotor.	21
Figure 1- 2: Mouvements du quadrirotor.	. 22
Figure 1- 3: Illustration du mouvement vertical	23
Figure 1- 4: Illustration du mouvement de roulis	24
Figure 1- 5: Illustration du mouvement de tangage.	. 24
Figure 1- 6: Illustration du mouvement de lacet.	25
Figure 1- 7: Illustration du mouvement de translation	. 25
Figure 1- 8: Repères du quadrotor	. 27
Figure 1- 9: Paramètrisation de l'orientation du drone dans l'espace	
Figure 1- 10: Identification du sens de rotation des rotors	. 32
Figure 1- 11: Commande d'un quadrirotor en attitude et position	. 39
Figure 2-1: Loopshaping de la houcle ouverte	54
Figure 2- 2: diagramme de Bode de G(P)	56
Figure 2-3: diagrammes de Bode de $J(n)$	58
Figure 2-4: Courbe de gain de L(p).	59
Figure 2- 5: Réponses indicielles	60
Figure 2- 6: Réponses à une perturbation en échelon.	. 60
1.5 m 2 of 1.0 Foreston a sure bet are constructed and the construction of the constru	
Figure 3- 1: Résultats de poursuite dans le cas de trajectoires de référence de type échelon.	. 68
Figure 3- 2: Erreurs de poursuite	. 68
Figure 3- 3: Signaux de commande	. 69
Figure 3- 4: Rejet d'une perturbation constante appliquée à la 25ème seconde	.71
Figure 3- 5: Réponse système avec la commande forwarding en présence d'une perturbatio	n 70
	. 72
Figure 3- 6: Signaux de commande pour la perturbation aléatoire.	.73
Figure 3- 7: Réponse du système avec la commande forwarding dans le cas de variations de	es
paramétres.	.74
Figure 3- 8: Signaux de commande pour une variation parametrique.	. 75
Figure 3- 9: Erreurs poursuite par la commande forwarding dans le cas de variations des	75
parametres.	. 75
Figure 3- 10: Poursuite de trajectoires de reference sinusoidales.	. /6
Figure 3- 11: Commande d'un quadri-rotor en attitude et en position	
Figure 3-12: Poursuite des trajectoires de references de type echelon	. 83
Figure 5-15: Reponse du système dans le cas d'une commande PID_Backstepping avec	0.4
Eigene 2, 14. Dépende du quettre commendé des la conjution para Né	. 84
Figure 5- 14: Keponse du système commande dans le cas de variations parametrique	. 83
Figure 5- 15: Reponse systeme avec la commande PID_backstepping en presence d'une	07
perturbation aleatoire	. 80

Figure 4- 1: Quadrirotor DIY utilisé.	
Figure 4- 2: Banc d'essai Quad équipé du pilote automatique Pixhawk	
Figure 4- 3: Montage du pixhawk2	
Figure 4- 4: Entrées/Sorties d'un Pixhawk.	100
Figure 4- 5: Schéma de principe du rôle de pixhawk	101
Figure 4- 6: Conception d'un moteur brushless.	102
Figure 4- 7: Variateur de vitesse 30A (ESC).	103
Figure 4- 8: Batterie lipo (lithium).	103
Figure 4- 9: Hélices.	104
Figure 4- 10: Radiocommande FlySky	105
Figure 4- 11: Récepteur de la radiocommande.	106
Figure 4- 12: Schéma de câblage du drone.	106
Figure 4- 13: Liaison entre MATLAB/Simulink et le système à partir de PIXHAWK	106
Figure 414: Compilation réussie du modèle	107
Figure 4- 15: Signal de commande de tangage.	108
Figure 4- 16: Signal de commande de roulis	108
Figure 4- 17: Erreur de poursuite de tangage	108
Figure 4- 18: Poursuite de roulis.	109
Figure 4- 19: Erreur de poursuite en roulis.	109
Figure 4- 20: Erreur de poursuite en lacet	109
Figure 4- 21: Différence entre la vitesse mesurée et la vitesse désirée en lacet	109

## LISTE DES TABLEAUX

Tableau 3-1: Paramètres du quadrirotor.	. 61
Tableau 3-2: Paramètres régulateurs PID	. 79

#### Liste des symboles et unités des mesures

- E(x, y, z): Repère fixe lié à la terre.
- B(x, y, z) : Repère mobile lié au robot volant.
- E: Vecteur de position du robot volant.
- x: Position du robot volant par rapport à l'axe x (m).
- *y* : Position du robot volant par rapport à l'axe y (*m*).
- z: Position du robot volant par rapport à l'axe z(m).
- $\phi$  : angle de roulis (*rad*)
- $\theta$  : angle de tangage (*rad*)
- $\psi$  : angle de lacet (*rad*)
- P: Poids du robot volant (N).
- FT: La somme totale des forces créées par chaque rotor (N)
- F1: Force de poussée créée par le rotor 1(N)
- F2: Force de poussée créée par le rotor 2(N)
- F3: Force de poussée créée par le rotor 3(N)
- F4: Force de poussée créée par le rotor 4(N)
- $F\eta$ : Forces de translations appliquées au robot volant (N)
- $\Gamma i$ : Moments angulaires du robot volant autour de x, y, et z (N.m)
- $\tau x$ : Moment angulaire autour de x (*N*.*m*)
- $\tau y$ : Moment angulaire autour de y (*N*.*m*)
- $\tau z$ : Moment angulaire autour de z (N.m)
- $M_{ah}$ : Moment gyroscopique des hélices (N.m)
- wi: Vitesses de rotation des moteurs (rad/s)
- m: Masse du quadrotor (kg)
- g: Gravité ( $m/s^2$ )
- $j_x$ : Inertie selon l'axe x ( $kg.m^2$ )
- $j_y$ : Inertie selon l'axe y ( $kg.m^2$ )
- $j_z$ : Inertie selon l'axe z ( $kg.m^2$ )
- $j_{rz}$ : Moment d'inertie du moteur par rapport à  $z(kg.m^2)$
- b : Coefficient de poussée ( $N.s^2$ )
- d : Coefficient de trainée ( $N.s^2$ ).
- U1 : Loi de commande désirée suivant l'axe Z (N)
- U2 : Loi de commande désirée de roulis (N)
- U3: Loi de commande désirée de tangage (N)
- U4: Loi de commande désirée de lacet (N)

 $x_d$ : Position désirée suivant x (m)

- $y_d$ : Position désirée suivant y (*m*)
- $z_d$ : Position désirée suivant z (*m*)
- $\phi_d$ : Trajectoire de roulis désirée (*rad*)
- $\theta_d$ : Trajectoire de tangage désirée (*rad*)
- $\psi_d$ : Trajectoire de lacet désirée (*rad*)

#### Acronymes

- VTOL: Vertical Take-Off and Landing (décollage et atterissage vertical)
- UAV : Unmanned Aerial Vehicles (véhicule aérien sans pilote)
- GPS: Global Positioning System
- PID: Proportional Integral Derivative (proportionnel, integral, dérivé)
- LQR : Linear Quadratic Regulator
- PWM : Pulse Width Modulation (modulation de largeur d'impulsion)
- IMU : Inertial Measurement Unit (Unité de Mesure Inertielle)
- MAV : Micro Air Vehicle (véhicule aérien micro ou miniature)
- DOF: Degrees-Of-Freedom (degrés de liberté)
- RC: Radio Control
- SISO: Single Input Single Output
- DIY: Do It Yourself
- GSC: Ground Station Control
- ESC : (Electronic Speed Controller) Régulateur électronique de vitesse.

# Introduction générale

#### **Introduction générale**

Ces dernières années, l'intérêt porté par la communauté scientifique à la thématique des drones UAV (Unmanned Aerial Vehicles) n'a cessé d'augmenter [3, 4, 7, 9,10, 12] Ces robots volants étaient utilisés au début dans le domaine militaire comme avions pilotés à distance afin de remplacer l'homme dans les missions risquées telles que les attaques aériennes et la reconnaissance vidéo [30].

Récemment, le domaine d'application de ces engins volants est devenu très varié, tel que la prise de vue aérienne, l'inspection des zones agricoles, des réseaux électriques, des chemins de fer. Les quadrirotors sont utilisés également pour la recherche des rescapés dans un environnement dangereux et confiné, comme un immeuble en feu ou un bâtiment sinistré suite à un séisme.

En effet, le recours à ces robots aériens autonomes présente plusieurs avantages par rapport aux véhicules avec pilotes, principalement par leur utilisation dans les milieux urbains encombrés grâce à leur petite taille et leur mode de décollage et d'atterrissage vertical [3-4].

Il existe une grande diversité de drones qui diffèrent selon leurs tailles, leurs structures, leurs complexités ou leurs autonomies ainsi que leurs domaines d'application.

Le quadrirotor dont la commande fait l'objet de cette thèse appartient à la catégorie des drones à voilures tournantes. Ce sont des véhicules aériens à décollage et atterrissage verticaux (VTOL) propulsés par quatre moteurs. Ils sont peu coûteux et mécaniquement simples à réaliser, ce qui permet leur construction à grandes quantités [30].

Néanmoins, le quadrirotor est un système très sensible aux perturbations du vent, aux défaillances des capteurs et des actionneurs qui peuvent conduire à l'instabilité du véhicule et créer de graves problèmes et difficultés dans le contrôle de ce dernier [35, 44, 46, 69, 73, 74].

Sa dynamique est fortement non linéaire et le nombre de degrés de liberté dépasse le nombre de commandes (système sous-actionné). Par conséquent, la synthèse des lois de commande devient difficile, la majorité des modèles présentent un compromis entre les performances et le mode de vol et la complexité de la loi de commande.

En effet, les contrôleurs linéaires comme le contrôleur PID [26,28, 49, 72] et le contrôleur LQR sont conçus pour les vols quasi-stationnaires. D'autre part, les contrôleurs non linéaires comme les contrôleurs backstepping, mode glissant et neuro flou, sont difficiles à mettre en œuvre sur de petits microcontrôleurs [16-29, 35,37,].

#### Objectifs

L'objectif principal de ce travail est de concevoir et développer des commandes linéaires et non linéaires permettant au drone quadrirotor de maintenir un positionnement autonome et stable dans l'espace.

Les contrôleurs choisis sont les contrôleurs non linéaires forwarding, le contrôleur hybride qui est une commande PID\_Backstepping non linéaire robuste. Les performances de celle-ci sont comparées à une autre commande combinée qui est la commande PID-backstepping qui est une commande à structure modulaire, il est possible de décomposer la dynamique du système considéré en deux sous-systèmes : de position et d'attitude et de concevoir des lois de commande séparées pour chacun de ces sous-systèmes. Nous choisissons ainsi un contrôleur PID pour le contrôle d'attitude alors que pour le sous-système de position, nous choisissons un contrôleur backstepping.

L'analyse de la stabilité est effectuée grâce à la théorie de Lyapunov. Plusieurs tests de simulations en boucle fermée sont effectués sur le quadrirotor tout en introduisant des perturbations de différentes natures.

La première perturbation est une constante d'une valeur de 10% de la valeur de la consigne et on a trouvé que le drone parvient à l'éliminer sans trop d'efforts. La deuxième perturbation est une variation paramétrique appliquée sur les paramètres du drone, cette perturbation a été réalisée en multipliant tous les paramètres par un coefficient constant, et encore une fois nous avons constaté que ses réponses sont acceptables et la commande arrive à corriger l'erreur sans grande dépense d'énergie. La troisième perturbation est un bruit blanc aléatoire, et ce dernier est éliminé grâce à la commande forwarding et la commande PID\_backstepping. A la fin, nous avons implémenté la commande PID sur un quadrirotor travaillant avec le contrôleur PIXhawk, et les résultats observés sont assez satisfaisants. La structure appliquée est la structure en cascade qui est très utilisée dans le domaine de l'aéronautique. Cette structure nous permet de commander la dynamique de translation à travers la dynamique de rotation. Les contraintes non holonomes sont prises en considération dans la modélisation du drone.

La commande loop-shaping est une commande linéaire qui sert à augmenter les performances des régulateurs PID tout en définissant les limites de régulation en termes de performance ou en termes de rejet de bruit. Cette technique nous l'avons appliquée sur le régulateur PID afin de définir les limites de rejet de perturbations ou de bruit [18, 27,58].

#### **Organisation du manuscrit**

Ce manuscrit est décomposé en quatre chapitres :

- Dans le premier chapitre, nous présentons les modélisations cinématique et dynamique du quadrirotor et ses mouvements de base. On donne aussi la forme d'état. Un état de l'art sur les différentes techniques de commande du quadrirotor est donné dans ce chapitre.
- Dans le second chapitre, nous présentons les différentes techniques de commande utilisées : la commande forwarding, la technique backstepping et la commande linéaire PID ainsi que la commande combinée PID\_Backstepping, et la commande PID loopshaping.
- Dans le troisième chapitre, on synthétisera un contrôleur basé sur la méthode Forwarding pour commander l'UAV en position et attitude. Un deuxième contrôleur hybride PID\_Backstepping est synthétisé aussi pour commander le drone en attitude et en position. La robustesse est vérifiée en simulant le système dans un environnement exposé à des perturbations.
- Dans le quatrième chapitre, à l'aide d'un banc d'essai pour quadrotor basé sur le pilote PIXHawk, nous avons implémenté la commande PID en attitude à l'intérieur du laboratoire et nous avons validé expérimentalement les résultats théoriques proposés.

Nous terminons cette thèse par une conclusion générale qui récapitule l'ensemble des travaux réalisés et des perspectives à envisager pour la suite de ce travail.

# Chapitre 1 Modélisation du quadrirotor et état de l'art sur sa commande

#### 1. Introduction

Le quadrotor est une machine volante qui possède quatre moteurs (M1, M2, M3, M4), auxquels sont fixées quatre hélices Figure 1-1 montées symétriquement sur les extrémités, et séparées en deux groupes tournant dans deux sens opposés. L'efficacité du couple de rotation est équilibrée grâce à la structure spécifique du quadrotor et grâce aux moteurs légers et puissants du quadrotor. Ainsi, il a des accélérations plus élevées que les hélicoptères traditionnels. Le quadrotor peut accomplir des tâches telles que la manœuvre, la cartographie et la navigation. La dynamique du système pour les mouvements linéaires et les rotations angulaires du quadrotor, sont couplées. La modélisation et la simulation du quadrotor doivent être effectuées pour déterminer ses caractéristiques de vol et la conception de régulateurs robustes de position et d'attitude [4].



Figure 1-1: Représentation du quadrotor.

La simplicité de la conception mécanique qui élimine la complexité de la commande du rotor principal de l'hélicoptère classique, les drones quadrotor deviennent des drones multi rotor différents. Ils peuvent être considérés comme des prototypes de drones solides

#### 1.1 MOUVEMENTS DU QUADRIROTOR

Dans les hélicoptères classiques, quand le rotor principal tourne, il produit un couple réactif qui inciterait le corps de l'hélicoptère à tourner dans la direction opposée, si ce couple, n'est pas contrarié. Ceci est, habituellement, fait en ajoutant un rotor de queue qui produit une poussée dans une direction latérale. Cependant, ce rotor, avec son alimentation électrique associée, ne fait aucune contribution à la poussée. Par contre, dans le cas du quadrirotor, le rotor droit et le rotor gauche tournent dans le sens des aiguilles d'une montre et dans la direction opposée des rotors avant et arrière. Ceci neutralise effectivement le couple réactif non désiré et

permet au véhicule de planer sans tourner hors de la commande. D'ailleurs, différemment aux hélicoptères classiques, toute l'énergie dépensée pour contrecarrer le mouvement de rotation contribue à la force de poussée [8, 10, 11, 23].

Les mouvements de base du quadrirotor Figure 1-12 sont réalisés en variant la vitesse de chaque rotor changeant de ce fait la poussée produite. Le quadrirotor s'incline vers la direction du rotor le plus lent, qui tient compte alors de la translation le long de cet axe. Par conséquent, comme dans un hélicoptère classique, les mouvements sont couplés, signifiant que le quadrirotor ne peut pas réaliser la translation sans roulis ou tangage. Ce qui signifie qu'un changement de la vitesse d'un rotor se traduit dans un mouvement en au moins trois degrés de liberté. Par exemple, augmentant la vitesse du moteur gauche aura comme conséquence un mouvement de roulis (le quadrirotor s'incline vers le rotor le plus lent, vers la droite), un mouvement de lacet (l'équilibre entres les rotors, qui tournent dans le sens des aiguilles d'une montre et les rotors qui tournent dans le sens inverse est perturbé, créant ainsi, un mouvement de rotation horizontal), et une translation (le mouvement de roulis incline l'armature et avec lui l'orientation de la force de poussée).



Figure 1-2: Mouvements du quadrirotor.

#### **1.1.1 Mouvement vertical**

- Toute la force de portance devrait seulement être le long de l'axe z, avec une grandeur exactement opposée à la force de pesanteur Figure 1-3. Les poussées produites par chaque rotor doivent être identiques.
- Le mouvement de montée ou de descente est obtenu par la variation de la vitesse de rotation des moteurs (par conséquant la poussée produite). Si la force de portance est supérieure au poids du quadrirotor, le mouvement est ascendant. Si la force de portance est inférieure au poids du quadrirotor, le mouvement est descendant.



Figure 1-3: Illustration du mouvement vertical.

#### 1.1.2 Mouvement de roulis

La Figure 1-4 montre comment un mouvement de roulis est obtenu. Dans ce cas, on applique un couple autour de l'axe x, c'est-à-dire en appliquant une différence de poussée entre le rotor 2 et le rotor 4. Ce mouvement (rotation autour de l'axe x) est couplé avec un mouvement de translation selon l'axe y.



Figure 1-4: Illustration du mouvement de roulis.

#### 1.1.3 Mouvement de tangage

La figure 1-5 montre comment un mouvement de tangage est obtenu. Dans ce cas, on applique un couple autour de l'axe y, c'est-à-dire en appliquant une différence de poussée entre le rotor 1 et le rotor 3. Ce mouvement (rotation autour de y) est couplé avec un mouvement de translation selon l'axe x.



Figure 1- 5: Illustration du mouvement de tangage.

#### 1.1.4 Mouvement de lacet

La Figure 1-6 montre comment le mouvement de lacet est obtenu. Dans ce cas, nous voulons appliquer un couple autour de l'axe z, qui est obtenu en appliquant une différence de vitesse entre les rotors {1,3} et {2,4}. Ce mouvement n'est pas un résultat direct de la poussée

produit par les propulseurs mais par les couples réactifs produits par la rotation des rotors. La direction de la force de poussée ne se décale pas pendant le mouvement, mais l'augmentation de la force de portance dans une paire de rotors doit être égale à la diminution des autres paires pour s'assurer que toute la force de poussée demeure la même.



Figure 1- 6: Illustration du mouvement de lacet.

#### 1.1.5 Mouvement de translation

La Figure 1-7 montre comment la translation horizontale est réalisée. Dans ce cas, nous voulons appliquer une force le long de x ou de y qui est réalisée en inclinant le corps (par le tangage ou le roulis) et en augmentant toute la poussée produite pour garder la force de portance du composant de z de la poussée égale à la force de pesanteur.



Figure 1-7: Illustration du mouvement de translation

#### 1.2 Modèle dynamique du quadrirotor

Les quadrirotors sont des systèmes mécaniques complexes. Leurs mouvements sont régis par plusieurs effets que ce soit mécanique ou aérodynamiques. Le modèle du quadrotor doit prendre en considération tous les effets qui affectent son mouvement, entre autres les effets gyroscopiques.

Afin d'aboutir à un modèle représentant la dynamique du quadrirotor, nous avons admis les hypothèses qui vont suivre

#### **1.2.1** Hypothèses simplificatrices

Les hypothèses simplificatrices considérées sont [23]

- La structure du quadrirotor est supposée rigide et symétrique, ce qui induit que la matrice d'inertie sera supposée diagonale,
- Les hélices sont supposées rigides, pour pouvoir négliger l'effet de leur déformation lors de la rotation.
- Le centre de masse et l'origine du repère, lié à la structure, coïncident.
- Les forces de portance et de traînée sont proportionnelles aux carrés de la vitesse de rotation des rotors, ce qui est une approximation très proche du comportement aérodynamique.
- Deux repères sont utilisés pour décrire le mouvement du drone. Le repère I représente le repère inertiel. Il est fixe par rapport à la terre. Le B qui est le repère du corps du quadrotor. L'axe z du repère B est toujours normal au corps du quadrotor [23].

Pour décrire la transformation entre les repères, des matrices de rotation peuvent être utilisées. Nous définissons la matrice de rotation x y R pour passer du repère  $\{y\}$  au repère  $\{x\}$ . Pour élaborer le modèle mathématique du quadrirotor, nous utilisons deux repères ; un repère fixe lié à la terre  $R^b$  et un autre mobile  $R^m$ . Le passage, entre le repère mobile et le repère fixe est donné par une matrice dite matrice de transformation T qui contient l'orientation et la position du repère mobile par rapport au repère fixe.

On choisit pour les axes la convention donnée par la Figure 1-8 :



Figure 1-8: Repères du quadrotor.

# $\begin{bmatrix} R & \varepsilon \\ 0 & 1 \end{bmatrix}$ (I.1)

Avec R la matrice de rotation (décrit l'orientation de l'objet mobile) et  $\varepsilon = [x \ y \ z]^T$  est le vecteur de position. Pour déterminer les éléments de la matrice de rotation R, on utilise les angles d'Euler.

#### 1.2.2 Angles d'Euler

Au début, le repère mobile coïncide avec le repère fixe. Ensuite le repère mobile fait un mouvement de rotation autour de l'axe *x* d'un angle de roulis  $(-\pi/2 < \phi < \pi/2)$  suivi d'une rotation autour de l'axe *y* d'un angle de tangage  $(-\pi/2 < \theta < \pi/2)$ , puis suivi d'une rotation autour de l'axe *z* d'angle de lacet  $(-\pi < \psi < \pi)$ . La matrice de rotation R est donnée comme suit :

$$R(\psi, \theta, \Phi) = \text{Rot}(z, \psi) \text{Rot}(y, \theta) \text{Rot}(x, \Phi)$$
(I.2)  
Où :

$$R(\psi, \theta, \Phi) = \begin{bmatrix} C\psi & -S\psi & 0\\ S\psi & C\psi & 0\\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta\\ 0 & 1 & 0\\ -S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0\\ 0 & C\phi & -S\phi\\ 0 & S\phi & C\phi \end{bmatrix}$$
(I.3)

La transformation qui permet de passer du repère mobile au repère fixe est donnée par

$$R(\psi, \theta, \Phi) = \begin{bmatrix} C\theta C\psi & C\psi S\theta S\Phi - S\psi C\Phi & C\psi S\theta C\Phi + S\psi S\Phi \\ C\theta S\psi & S\psi S\theta S\Phi + C\psi C\theta & S\psi S\theta C\Phi - C\psi S\Phi \\ -S\theta & C\theta S\Phi & C\theta C\Phi \end{bmatrix}$$
(I.4)

Où, C et S, désignent respectivement, les fonctions trigonométriques cos et sin.



Figure 1-9: Paramètrisation de l'orientation du drone dans l'espace.

#### 1.2.3 Vitesses angulaires

Les vitesses de rotations  $\Omega 1$ ,  $\Omega 2$  et  $\Omega 3$  dans le repère fixe sont exprimées en fonction des vitesses de rotations  $\dot{\phi}$ ,  $\dot{\theta}$  et  $\dot{\psi}$  dans le repère mobile, comme suit :

$$\Omega = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + rot_x (\phi)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + (rot_y (\theta) rot_x (\phi))^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix}$$
(I.5)

En effet, la rotation en roulis a lieu lorsque les repères sont encore confondus. Puis, en ce qui concerne le tangage, le vecteur représentant la rotation doit être exprimé dans le repère fixe : il est donc multiplié par  $rot_x(\phi)^{-1}$ . De même, le vecteur représentant la rotation en lacet doit

être exprimé dans le repère fixe qui a déjà subi deux rotations. On arrive, ainsi, à l'équation suivante :

$$\Omega = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\theta} c\phi \\ -\dot{\theta} s\phi \end{bmatrix} + \begin{bmatrix} -\dot{\psi}s\theta \\ \dot{\psi}s\phi c\theta \\ \dot{\psi}s\phi c\theta \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \dot{\psi}s\theta \\ \dot{\theta} c\phi + \dot{\psi}s\phi c\theta \\ \dot{\psi}s\phi c\theta - \dot{\theta}s\phi \end{bmatrix}$$
(I.6)  
$$\Omega = \begin{bmatrix} 1 & 0 & -S\theta \\ 0 & C\Phi & S\Phi C\theta \\ 0 & -S\Phi & C\Phi C\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$
(I.7)

Quand le quadrirotor fait des petites rotations, on peut faire les approximations suivantes :  $c\phi=c\theta=c\psi=1$  et  $s\phi=s\theta=s\psi=0$ .

Dans ce cas la vitesse angulaire est donnée comme suit :

$$\Omega = \left[\dot{\phi} \ \dot{\theta} \ \dot{\psi}\right]^T \tag{I.8}$$

#### 1.2.4 Vitesses linéaires

Les vitesses linéaires  $v_x^b$ ,  $v_y^b$ , *et*  $v_z^b$  dans le repère fixe en fonction des vitesses linéaires  $v_x^m$ ,  $v_y^m$ , *et*  $v_z^m$  dans le repère mobile sont données par :

$$\mathbf{v} = \begin{bmatrix} v_x^b \\ v_y^b \\ v_z^b \end{bmatrix} = R \mathbf{x} \begin{bmatrix} v_x^m \\ v_y^m \\ v_z^m \end{bmatrix}$$
(I.9)

#### **1.3** Effets physiques agissant sur le quadrirotor

Pour notre cas, on va considérer uniquement les forces et les moments appliqués au quadrotor générés par les effets aérodynamiques, la rotation des hélices et de la précession gyroscopique. Nous détaillerons par la suite chacun des effets séparément, ensuite ces forces et moments seront appliqués au modèle du quadrotor. A la fin, nous ajoutons ces résultantes à l'équation de la dynamique.

#### 1.3.1 Forces

Les forces agissant sur le système du quadcopter sont :

#### 1.3.1.1 Poids du quadrirotor

Cette force est due à la masse de l'objet. Elle est toujours perpendiculaire à la surface de la terre. Elle est donnée dans le référentiel inertiel par :

 $P=m.g \tag{I.10}$ 

Où m est la masse totale et g est la gravité.

#### **1.3.1.2** Forces de poussée

Elles sont provoquées par la rotation des moteurs. Elles sont perpendiculaires sur le plan des hélices. Ces forces sont proportionnelles au carré de la vitesse de rotation des moteurs :

$$F_i = b\omega_i^2 \tag{I.11}$$
Avec  $i=1:\overline{4}$ 

b : est le coefficient de portance, il dépend de la forme et du nombre des pales et de la densité de l'air.

#### **1.3.1.3** Forces de traînée

La force de traînée est le couplage entre une force de pression et la force de frottement visqueux. Dans ce cas, on a deux forces de traînée agissant sur le système qui sont :

La traînée dans les hélices : elle agit sur les pales, elle est proportionnelle à la densité de l'air, à la forme des pales et au carré de la vitesse de rotation de l'hélice, elle est donnée par la relation suivante :

$$T_h = d\omega_i^2 \tag{I.12}$$

*d* est le coefficient de drag il dépend de la fabrication de l'hélice.

La traînée selon les axes (x, y, z) : elle est due au mouvement du corps du quadrirotor

$$F_t = K_{ft} \cdot v \tag{I.13}$$

Avec  $k_{ft}$  le coefficient de traînée de translation et v est la vitesse linéaire.

#### 1.3.2 Moments

Il y a plusieurs moments agissant sur le quadrirotor. Ces moments sont dûs aux forces de poussée et de traînée et aux effets gyroscopiques.

#### 1.3.2.1 Moments dûs aux forces de poussée

La rotation autour de l'axe x : Elle est due au moment créé par la différence entre les forces de portance des rotors 2 et 4, ce moment est donné par la relation suivante :

$$M_x = (F_4 - F_2) = lb(d\omega_4^2 - \omega_2^2)$$
(I.14)

Où est la longueur du bras entre le rotor et le centre de gravité du quadrirotor.

La rotation autour de l'axe y : Elle est due au moment créé par la différence entre les forces de portance des rotors 1 et 3. Ce moment est donné par la relation suivante :

$$M_y = (F_3 - F_1) = lb(d\omega_3^2 - \omega_1^2)$$
(I.15)

#### 1.3.2.2 Moments dûs aux forces de traînée

La rotation autour de l'axe z : elle est due à un couple réactif provoqué par les couples de traînée dans chaque hélice, ce moment est donné par la relation suivante :

$$M_z = b(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{I.16}$$

Le moment résultant des frottements aérodynamiques, est donné par :

$$M_a = K_{fa} \Omega^2 \tag{I.17}$$

Où  $K_{fa}$  est le coefficient des frottements aérodynamiques et  $\Omega$  est la vitesse angulaire.

#### **1.3.2.3 Effet gyroscopique**

L'effet gyroscopique se définit comme la difficulté de modifier la position ou l'orientation du plan de rotation d'une masse tournante. L'effet gyroscopique est ainsi nommé en référence au mode de fonctionnement du gyroscope, appareil de contrôle de mouvement utilisé dans l'aviation (du grec *gyro* qui signifie rotation et *scope*, observer).

Dans notre cas, il y a deux moments gyroscopiques, le premier est le moment gyroscopique des hélices, l'autre est le moment gyroscopique dû aux mouvements du quadrirotor.

Moment gyroscopique des hélices : il est donné par la relation suivante :

$$M_{gh} = \sum_{i=1}^{4} \Omega \Lambda J_r \begin{pmatrix} 0\\ 0\\ -1^{i+1}\omega_i \end{pmatrix}$$
(I.18)

Où Jr est l'inertie des rotors.

Moment gyroscopique dû aux mouvements de quadrirotor : il est donné par la relation suivante

$$M_{gm} = \Omega \wedge J\Omega \tag{I.19}$$

Avec J est l'inertie du système.

#### 1.3.3 Développement du modèle mathématique selon Newton-Euler

Après avoir présenté les équations de mouvement dans les paragraphes précédents, nous pouvons actuellement passer au modèle complet du quadrotor. La configuration du quadrotor est décrite à la Figure 1-10. Les moteurs sont numérotés dans le sens horaire avec le moteur 1 comme étant celui de l'avant de l'appareil par rapport au référentiel  $F_b$ . Les deux moteurs 1 et 3 tournent dans le sens horaire contrairement au sens des moteurs 2 et 4.



Figure 1-10: Identification du sens de rotation des rotors

En se basant sur les équations des forces appliquées au quadrirotor les équations (I.9, I.10, I.11, I.12), et les moments agissants sur le quadrirotor les équations (I.13, I.14, I.15, I.16, I.17, I.18). Pour résumer l'ensemble des équations décrivant le modèle complet du quadrotor, on utilise la formulation de Newton-Euler. Le modèle dynamique de système est à présent sous la forme suivante [5] :

$$\begin{cases}
\dot{\varepsilon} = v \\
m\ddot{\varepsilon} = Ff + Ft + Fg \\
J\dot{\Omega} = -\Omega \wedge J\Omega + M_f - M_a - M_{gh} \\
\dot{R} = RS(\Omega)
\end{cases}$$
(I.20)

Avec :

- $\mathcal{E}$  : est le vecteur de position du quadri rotor  $\mathcal{E} = [x \ y \ z]^{T}$ ,
- m : la masse totale du quadri rotor,
- $\Omega$  : la vitesse angulaire exprimée dans le repère fixe,
- R : la matrice de rotation,
- $\Lambda$  : le produit vectoriel,
- J : matrice d'inertie symétrique de dimension (3\*3), elle est donnée par :

$$\mathbf{J} = \begin{bmatrix} I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z \end{bmatrix}$$
(I.21)

S(Ω) est la matrice antisymétrique pour le vecteur de vitesses angulaires  $Ω = [Ω_1 Ω_2 Ω_3]^T$ 

$$S(\Omega) = \begin{bmatrix} 0 & -\Omega 3 & \Omega 2\\ \Omega 3 & 0 & -\Omega 1\\ -\Omega 2 & \Omega 1 & 0 \end{bmatrix}$$
(I.22)

• F<sub>f</sub> est la résultante des forces de poussée générées par les quatre rotors, elle est donnée par :

$$F_{f} = \begin{bmatrix} C\Phi C\psi S\theta + S\psi S\Phi \\ C\Phi S\theta S\psi - S\Phi C\psi \\ C\Phi C\theta \end{bmatrix} * \sum_{i=1}^{4} F_{i}$$
(I.23)  
Avec :

$$\sum_{i=1}^{4} F_{i=1} \sum_{i=1}^{4} K p \,\omega_{i}^{2} = u_{1}$$
(I.24)

u1: force de poussée « portance »,

- Kp : désigne le coefficient de portance
- $\omega_i$ : désigne la vitesse angulaire du moteur

Ft : est la résultante des forces de trainée selon les axes (x, y, z), elle est donnée par :

$$F_{t} = \begin{bmatrix} -K_{ftx} & 0 & 0\\ 0 & -K_{fty} & 0\\ 0 & 0 & -K_{ftz} \end{bmatrix} \dot{\varepsilon}$$
(I.25)

K<sub>ftx</sub> , K<sub>fty</sub> , K<sub>ftz</sub>: sont les coefficients de traînée de translation.

Fg regroupe les forces liées à la gravité, elle est donnée par :

$$F_{g} = \begin{bmatrix} 0\\0\\-mg \end{bmatrix}$$
(I.26)

M<sub>f</sub> moment provoqué par les forces de poussée et de traînée.

$$M_{f} = \begin{bmatrix} l(F4 - F2) \\ l(F3 - F1) \\ ld(\omega_{1}^{2} - \omega_{2}^{2} + \omega_{3}^{2} - \omega_{4}^{2}) \end{bmatrix}$$
(I.27)

l est la distance entre le centre de masse du quadri rotor et l'axe de rotation du rotor.

d est le coefficient de trainée.

Ma: moment résultant des frottements aérodynamiques.

$$M_{a} = \begin{bmatrix} K_{fax} & 0 & 0 \\ 0 & K_{fay} & 0 \\ 0 & 0 & K_{faz} \end{bmatrix} \Omega^{2}$$
(I.28)

 $K_{\text{fax}}$  ,  $K_{\text{fay}}$  ,  $K_{\text{faz}}$  sont les coefficients des frottements aérodynamiques.

#### Equation du mouvement de translation

On a

$$m\ddot{\varepsilon} = F_f + F_t + F_g \tag{I.29}$$

On remplace chaque force par son expression ((1.23), (1.25), (1.26)) dans (1.29) on trouve :

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} C\Phi C\Psi S\theta + S\Psi S\Phi \\ C\Phi S\theta S\Psi - S\Phi C\Psi \\ C\Phi C\theta \end{bmatrix} \sum_{i=1}^{4} Fi + \begin{bmatrix} -K_{ftx}\dot{x} \\ -K_{fty}\dot{y} \\ -K_{ftz}\dot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$
(I.30)  
Avec  $\sum_{i=1}^{4} Fi = u_1$ 

On obtient alors les équations différentielles qui définissent la dynamique de la translation :

$$\begin{cases} \ddot{x} = \frac{1}{m} \{ (C\phi C\psi S\theta + S\psi S\phi) u_1 - K_{ftx} \dot{x} \} \\ \ddot{y} = \frac{1}{m} \{ (C\phi S\theta S\psi - S\Phi C\Psi) u_1 - K_{fty} \dot{y} \} \\ \ddot{z} = \frac{1}{m} \{ (C\phi C\theta) u_1 - K_{ftz} \dot{z} \} \text{-g} \end{cases}$$
(I.31)

#### **1.3.4** Equation du mouvement de rotation

On applique le même principe de Newton pour le cas de rotation, on trouve la formule suivante :

$$\mathbf{J}\dot{\Omega} = -\Omega_{\Lambda}\mathbf{J}\Omega + M_{\mathrm{f}} - M_{\mathrm{a}} - M_{gh} \tag{I.32}$$

On remplace chaque moment par son expression dans (I.20), on obtient alors les équations différentielles définissant la dynamique de la rotation :

$$\begin{cases} \ddot{\Phi} = \frac{1}{I_x} [\dot{\theta} \dot{\psi} (I_y - I_z) - J_r \,\overline{\Omega}_r \, \dot{\theta} - K_{fax} \dot{\phi^2} + u_2] \\ \ddot{\theta} = \frac{1}{I_y} [(\dot{\phi} \dot{\psi} (I_z - I_x) + J_r \overline{\Omega}_r \dot{\phi} - K_{fay} \dot{\theta}^2 + u_3] \\ \ddot{\psi} = \frac{1}{I_z} [\dot{\theta} \dot{\phi} (I_x - I_y) - K_{faz} \dot{\psi}^2 + u_4] \end{cases}$$
(I.33)

• 
$$\Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4$$

• 
$$u_2 = l(F4 - F2)$$
 (I.34)

• 
$$u_3 = l(F3 - F1)$$

• 
$$u_4 = d(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2)$$

En résumé, le modèle dynamique complet, régissant la dynamique du quadrirotor, est :

$$\begin{cases} \ddot{x} = \frac{1}{m} [(C\varphi C\psi S\theta + S\psi S\varphi)u1 - K_{ftx}\dot{x}] \\ \ddot{y} = \frac{1}{m} [(C\varphi S\theta C\psi + S\varphi C\psi)u1 - K_{fty}\dot{y}] \\ \ddot{z} = \frac{1}{m} [(C\varphi C\theta)u1 - K_{ftz}\dot{z})] - g \\ \ddot{\Phi} = \frac{1}{I_x} [\dot{\theta}\dot{\psi}(I_y - I_z) + u_2 - J_r\overline{\Omega}_r\dot{\theta} - Kf_{ax}\dot{\phi}^2] \\ \ddot{\theta} = \frac{1}{I_y} [\dot{\psi}\dot{\phi}(I_z - I_x) + u_3 + J_r\overline{\Omega}_r\dot{\phi} - K_{fay}\dot{\theta}^2 \\ \ddot{\psi} = \frac{1}{I_z} [\dot{\theta}\dot{\phi}(I_x - I_y) + u_4 - K_{faz}\dot{\psi}^2] \end{cases}$$
(I.35)

#### **1.4** Relation entre forces, moments et vitesse des moteurs

Les variables u<sub>1</sub>, u<sub>2</sub>, u<sub>3</sub> et u<sub>4</sub> sont les entrées de commande du système qui s'écrivent en fonction des vitesses angulaires des quatre rotors comme suit :
$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -lb & 0 & lb \\ -lb & 0 & lb & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}$$
(I.36)

En inversant la matrice, on obtient la relation entre la vitesse des moteurs et les variables de commande.

Posons :

$$D = \begin{bmatrix} b & b & b & b \\ 0 & -l.b & 0 & l.b \\ -l.b & 0 & l.b & 0 \\ d & -d & d & -d \end{bmatrix}$$
(I.36)

D'où

$$\omega_{i}^{2} = D^{-1}U_{i}$$

$$\begin{bmatrix} \omega_{1}^{2} \\ \omega_{2}^{2} \\ \omega_{3}^{2} \\ \omega_{4}^{2} \end{bmatrix} = D^{-1} \begin{bmatrix} u_{1} \\ u_{2} \\ u_{3} \\ u_{4} \end{bmatrix}$$
(I.37)
$$= 1 \qquad 1 \qquad 1 \qquad 1 \qquad 1$$

$$\omega_1^2 = \frac{1}{4b}u_1 - \frac{1}{2bl}u_3 + \frac{1}{4d}u_4$$
$$\omega_2^2 = \frac{1}{4b}u_1 - \frac{1}{2bl}u_2 - \frac{1}{4d}u_4$$
$$\omega_3^2 = \frac{1}{4b}u_1 + \frac{1}{2bl}u_3 + \frac{1}{4d}u_4$$
$$\omega_4^2 = \frac{1}{4b}u_1 + \frac{1}{2bl}u_2 - \frac{1}{4d}u_4$$

# 1.5 Représentation d'état du système

On choisit le vecteur d'état suivant :

$$X = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \varphi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^{T}$$
$$X = [x_{1}, x_{2}, x_{3}, x_{4}, x_{5}, x_{6}, x_{7}, x_{8}, x_{9}, x_{10}, x_{11}, x_{12}]^{T}$$

et on considère le changement de variables suivant :

$$\begin{cases} x_1 = x ; x_2 = y ; x_3 = z ; x_4 = \dot{x}_1 ; x_5 = \dot{x}_2 ; x_6 = \dot{x}_3 \\ x_7 = \phi ; x_8 = \theta ; x_9 = \psi ; x_{10} = \dot{\phi} ; x_{11} = \dot{\theta} ; x_{12} = \dot{\psi} \end{cases}$$
(I.38)

Le modèle d'état du quadrirotor est :

$$\begin{split} \dot{x}_{1} &= \dot{x} = x_{4} \\ \dot{x}_{2} &= \dot{y} = x_{5} \\ \dot{x}_{3} &= \dot{z} = x_{6} \\ \dot{x}_{4} &= \ddot{x} = [a_{9}x_{4} + \frac{1}{m}u_{x}u_{1}] \\ \dot{x}_{5} &= \ddot{y} = [a_{10}x_{5} + \frac{1}{m}u_{y}u_{1}] \\ \dot{x}_{5} &= \ddot{y} = [a_{10}x_{5} + \frac{1}{m}u_{y}u_{1}] \\ \dot{x}_{6} &= \ddot{z} = \frac{1}{m}[\cos(x_{7})\cos(x_{8})u_{1}] + a_{11}x_{6} - g \quad (I.39) \\ \dot{x}_{7} &= \dot{\phi} = x_{10} \\ \dot{x}_{8} &= \dot{\theta} = x_{11} \\ \dot{x}_{9} &= \dot{\psi} = x_{12} \\ \dot{x}_{10} &= \ddot{\phi} = [a_{1}x_{11}x_{12} + a_{2}x_{10}^{2} + a_{3}x_{11}\Omega_{r} + b_{1}u_{2}] \\ \dot{x}_{11} &= \ddot{\theta} = [a_{4}x_{10}x_{12} + a_{5}x_{11}^{2} + a_{6}x_{10}\Omega_{r} + b_{2}u_{3}] \\ \dot{x}_{12} &= \ddot{\psi} = [a_{7}x_{11}x_{10} + a_{8}x_{12}^{2} + b_{3}u_{4}] \end{split}$$

Les coefficients  $a_i$ ;(*i*=1,11) et  $b_j$ ;(*j*=1,3) sont définis comme suit :

$$\begin{cases} a_{1} = \frac{I_{y} - I_{z}}{I_{x}}; a_{2} = \frac{-K_{fax}}{I_{x}}; a_{3} = \frac{-J_{r}}{I_{x}}; a_{4} = \frac{I_{z} - I_{x}}{I_{y}}; a_{5} = \frac{-K_{fay}}{I_{y}}; a_{6} = \frac{-J_{r}}{I_{y}} \\ a_{7} = \frac{I_{x} - I_{y}}{I_{z}}; a_{8} = \frac{-K_{faz}}{I_{z}}; a_{9} = \frac{-K_{ftx}}{m}; a_{10} = \frac{-K_{fty}}{m}; a_{11} = \frac{-K_{ftz}}{m} \\ b_{1} = \frac{1}{I_{x}}; b_{2} = \frac{1}{I_{y}}; b_{3} = \frac{1}{I_{z}} \end{cases}$$
(I.40)

Avec :

$$\begin{cases} u_x = (\cos(x_7)\sin(x_8)\cos(x_9) + \sin(x_9)\sin(x_7)) \\ u_y = \cos(x_7)\sin(x_8)\sin(x_9) - \sin(x_7)\cos(x_9) \end{cases}$$
(I.41)

 $u_x$  et  $u_y$  sont les forces de translation reçues par le contrôleur de la boucle externe La formulation du bloc de correction des angles  $\varphi_d$  et  $\theta_d$  desirés est présentée comme suit

$$\begin{cases} \varphi_d = \arcsin(\frac{u_x \sin(\Psi) - u_y \cos(\Psi)}{u_1}) \\ \theta_d = \arcsin(\frac{u_x \cos(\Psi) + u_y \sin(\Psi)}{\cos(\varphi_d) u_1}) \end{cases}$$
(I.42)

#### 1.6 Dynamique du moteur

Généralement les moteurs utilisés dans les quadri rotor sont des moteurs à courant continu, leur dynamique est donnée par les équations suivantes :

$$\begin{cases} V_i = ri + L\frac{di}{dt} + K_e \omega_i \\ K_m i = J_r \frac{d\omega_i}{dt} + C_s + K_r \omega_i^2 \end{cases}$$
(I.43)

Les différents paramètres du moteur sont définis comme suit :

- V<sub>i</sub> : tension d'entrée du moteur.
- *i* : courant d'inductance.
- $\omega_i$ : vitesse angulaire du moteur.
- *K<sub>e</sub>* : constante du couple électrique.
- $K_m$  : constante du couple mécanique.
- $K_r$  : constante du couple de charge.
- *r*, *L* : résistance et inductance du moteur.
- $J_r$ : inertie du rotor.
- $C_s$  : coefficient du frottement.

Avec :

$$\dot{\omega}_i = bV_i - B_0 - B_1 \omega_i - B_2 \omega_i^2 \quad i \in [1,4]$$

$$Avec \quad b = \frac{K_m}{J_r}; \quad B_0 = \frac{C_s}{J_r}; \quad B_1 = \frac{K_e K_m}{r J_r}; \quad B_2 = \frac{K_r}{J_r};$$

$$(I.44)$$

#### 1.7 Stratégie de commande en cascade du quadrirotor

La structure de commande la plus courante pour les quadrirotors consiste en deux boucles de commande, une pour la position, et l'autre pour l'orientation. Une telle architecture est présentée dans presque tous les systèmes quadrirotors.

La boucle de commande de position de haut niveau (extérieure) la sortie de cette boucle ,de régulation, génère l'attitude désirée qui est, ensuite, commandée par une boucle de commande

de bas niveau (intérieure) basée sur l'hypothèse que la dynamique d'attitude d'un quadrirotor est beaucoup plus rapide que sa dynamique de position.

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 [31].

Concernant notre travail, nous avons adopté une stratégie de commande basée principalement sur la décomposition du système d'origine en deux sous-systèmes. Le premier concerne la commande en position, et le deuxième concerne la commande en orientation. Le schéma de la Figure 1-11 représente la stratégie de commande adoptée [11]



Figure 1-11: Commande d'un quadrirotor en attitude et position.

Après avoir défini les valeurs désirées ( $x_d$ ,  $y_d$ ,  $z_d$ ,  $\Psi_d$ ), le contrôleur de position génère les valeurs requises ( $\varphi_d$ ,  $\theta_d$ ) pour le contrôleur de rotation, les quantités mesurées sont ensuite renvoyées aux deux contrôleurs [11].

#### **1.8** Etat de l'art sur la commande des drones quadrirotors

Puisque le quadrotor trouve ses applications dans de nombreux environnements différents avec des scénarios différents et compliqués. Les chercheurs continuent à développer des contrôleurs robustes, afin de maintenir la stabilité du système dans un environnement complexe (incluant des perturbations externes comme les rafales de vent, etc..).

La littérature propose plusieurs techniques avancées de commande linéaire et non linéaire, pour résoudre le problème de contrôle du drone dans un environnement perturbé réaliste. Par exemple, [3] étudie une commande floue adaptative robuste pour le quadrotor en présence de perturbations dues au vent. Une nouvelle commande robuste, pour l'attitude et la position d'un mini-drone, est proposée dans [68]. Dans [16], un contrôleur de suivi de trajectoire avec rejet des perturbations est présenté. Une méthode basée sur le contrôle en mode glissant du second ordre est proposée dans [44]. Dans [3], les auteurs ont abordé le problème de la commande de stabilité et de suivi de trajectoire d'un quadrotor en présence de perturbations externes et d'une erreur de modélisation.

Dans [43] l'auteur a proposé un régulateur adaptatif synergique (ASC) comme solution au problème de la conception de la loi de commande synergique standard, lorsque les paramètres et la dynamique du système sont inconnus. Dans [29], un nouveau schéma de commande d'attitude du quadrirotor en présence d'incertitudes et de perturbations, utilise la commande supertwisting modifiée MSTW, dans le but de résoudre le problème de la supertwisting classique, qui est le chattering.

Dans [23] l'auteur a proposé le réglage des paramètres des régulateurs PID d'un quadrotor à l'aide des algorithmes inspirés de la nature.

De nombreuses techniques sont développées et appliquées pour suivre les trajectoires planifiées, comme la commande prédictive (MPC) qui peut être utile pour traiter simultanément la génération et le suivi de la trajectoire. Cette méthode est avantageuse par l'exécution d'une commande par rétroaction et la génération de trajectoire lisse simultanément. Cette technique a été appliquée dans le domaine du quadrotor [25]

#### Conclusion

Dans ce chapitre, nous avons présenté une modélisation mathématique d'un drone quadrirotor à six degrés de liberté selon certaines hypothèses simplificatrices. Sa modélisation a été établie par l'usage du formalisme de Newton-Euler. Ce formalisme nous a permis d'établir le modèle dynamique du quadricopter.

L'analyse du modèle mathématique obtenu nous montre que le quadrirotor est un système instable à deux dynamiques fortement couplées qui sont la dynamique de rotation et la dynamique de translation.

# Chapitre 2

# Rappels théoriques sur les commandes robustes linéaires et non linéaires

#### 2 Introduction

La commande robuste est une branche de l'automatique qui a été développée suite à l'apparition de nouvelles exigences que les méthodes traditionnelles de régulation ne pouvaient pas régler.

En effet, pour synthétiser un régulateur, on utilise le modèle mathématique du système à commander. Si les paramètres du modèle sont variants ou incertains, et si le système est exposé à des perturbations internes et externes, les performances du système risquent de ne plus être assurées, et le système peut même devenir instable. La commande robuste est donc une technique de commande qui vise à garantir les performances et la stabilité d'un système face aux incertitudes du modèle et aux perturbations du milieu. Parmi les objectifs de la commande robuste, nous pouvons citer :

- Assurer la stabilité du système bouclé.
- Assurer les exigences du cahier des charges (temps de réponse, temps de montée, erreur statique, dépassement...).
- Assurer la robustesse : cela consiste à assurer les deux points précédents malgré les erreurs de modélisation.

Il sera nécessaire de faire un compromis entre la robustesse et les performances afin d'obtenir une loi de commande satisfaisante.

#### 2.1 Commande Forwarding

C'est une technique de stabilisation des systèmes non linéaires qui est jugée robuste, qui a émergé ces dernières années. La synthèse d'une telle commande se fait de manière systématique, elle est basée sur l'approche de Lyapunov, d'une manière récursive pour un système qui peut être écrit sous la forme triangulaire supérieure suivante [53], [54], [55] :

$$\dot{x}_1 = f_1(x_2, \dots, x_n) + g_1(x_2, \dots, x_n)u$$

#### Chapitre 2 Rappels théoriques sur les commandes robustes linéaires et non linéaires

$$\begin{cases} \dot{x}_{1} = f_{1}(x_{2}, \dots, x_{n}) + g_{1}(x_{2}, \dots, x_{n})u \\ \dot{x}_{2} = f_{2}(x_{2}, \dots, x_{n}) + g_{2}(x_{2}, \dots, x_{n})u \\ \vdots \\ \dot{x}_{n-1} = f_{n-1}(x_{n}) + g_{n-1}(x_{n})u \\ \dot{x}_{n} = u \end{cases}$$
(II.1)

Où u est l'entrée,  $x \in \mathbb{R}^n$ ,  $u \in \mathbb{R}$  et  $g = [g_1, \dots, g_{n-1}, 1]^T$ .

Ces systèmes sont dits de forme feedforward où chaque composante de l'état intervient dans celles qui les suivent dans la chaine d'intégration en partant de la commande.

Notons que la théorie de base de forwarding a les mêmes caractéristiques que la technique du backstepping. Cependant, elles se distinguent par deux points majeurs : les familles de systèmes auxquelles elles s'appliquent sont différentes et la théorie de base du backsteppinq procure généralement des lois de commandes non bornées tandis que celle du forwarding conduit à des lois de commande bornées [74, 75,76].

L'objectif de la commande forwarding est de déterminer une loi qui assure la stabilisation globale du système. Cela s'effectue en plusieurs étapes.

En premier lieu, nous essayons de stabiliser  $x_n$  à travers u. En prenant une fonction de Lyapunov  $V_n$  qui est définie positive sur R et on cherche à trouver un retour d'état u = $\alpha_n(x_n)$  pour stabiliser  $x_n$ .

L'étape suivante consiste à augmenter le système, de  $x_n$  à  $x_{n-1}$ , et remplacer la commande u par  $u = \alpha_n (x_n) + V_{n-1}$ . En prenant une fonction de Lyapunov  $V_{n-1}$  définie positive sur  $\mathbb{R}^2$ , on cherche à trouver un retour d'état  $V_{n-1} = \alpha_{n-1}$  qui permet de stabiliser le système augmenté.

On refait la même procédure jusqu'à la dernière étape où l'on calcule une commande *u* qui permet d'avoir une stabilisation globale du système.

#### 2.1.1 Etapes de synthèse

La synthèse de la loi de commande pour le système s'effectue en plusieurs étapes [53], [54], [55]

#### **<u>1. Première étape</u>**

On considère le système :

$$\dot{x}_n = u$$
 (II.2)  
Soit la fonction de Lyapunov :  $V_n = \frac{1}{2} x_n^2$ .  
Sa dérivée est donnée par :  $\dot{V}_n = x_n u$ .  
Pour rendre  $\dot{V}_n$  définie négative, on peut prendre :  
 $u = \alpha_n (x_n) = -\lambda_n Lg V_n = -\lambda_n x_n$  avec  $\lambda_n > 0$  (II.3)

Ce qui donne  $\dot{V}_n = -\lambda_n x_n^2$  qui est définie négative sur R.

## 2. Deuxième étape

Soit le système augmenté :

$$\begin{cases} \dot{x}_{n-1} = f_{n-1}(x_n) + g_{n-1}(x_n)u \\ \dot{x}_n = u \end{cases}$$
(II.4)

Pour  $u = \alpha_n (x_n)$ , le système devient :

$$\begin{cases} \dot{x}_{n-1} = \phi_{n-1}(x_n) \\ \dot{x}_n = -\lambda_n(x_n) \\ \phi_{n-1}(x_n) = f_{n-1}(x_n) - \lambda_n g_{n-1}(x_n) x_n \end{cases}$$
(II.5)

Le terme  $\phi_{n-1}$  est appelé terme d'interconnexion.

La solution temporelle du système précédent sera

$$\begin{cases} \tilde{x}_{n}(t) = x_{n}(0)e^{-\lambda_{n}t} \\ \tilde{x}_{n-1}(t) = \int_{0}^{t} (f_{n-1}(\tilde{x}_{n}) - \lambda_{n}g_{n-1}(\tilde{x}_{n})\tilde{x}_{n})dt + x_{n-1}(0) \end{cases}$$
(II.6)

La nouvelle entrée de commande pour le système (II.4) sera :

$$u = \alpha_n(x_n) + v_{n-1} \tag{II.7}$$

La fonction de Lyapunov pour le système est donnée par

$$V_{n-1} = V_n + \frac{1}{2}x_{n-1}^2 + \int_0^\infty \tilde{x}_{n-1}(t)\phi_{n-1}(\tilde{x}_n(t))dt$$

$$= V_n + \frac{1}{2}\lim_{t \to \infty} \check{x}_{n-1}^2(t)$$
(II.8)

Sa dérivée est [22] :

$$\dot{V}_{n-1} = \dot{V}_{n-1}|_{u=\alpha_n(x_n)} + (L_g V_{n-1})v_{n-1}$$
(II.9)

$$= -\lambda_n (L_g V_n)^2 + (L_g V_{n-1}) v_{n-1}$$

Donc pour rendre  $\dot{V}_{n-1}$  définie négative, on peut prendre :

$$v_{n-1} = -\lambda_{n-1}L_g V_{n-1} \quad \text{Avec } \lambda_{n-1} \ge 0 \tag{II.10}$$

Dans ce cas, on aura  $\dot{V}_{n-1} = -\lambda_n (L_g V_n)^2 - \lambda_{n-1} (L_g V_{n-1})^2$ . Cette fonction est définie négative sur  $R^2$ . La commande qui assure la stabilisation globale du système (II.4) est alors [22] :

$$u = \alpha_{n-1}(x_n, x_{n-1}) = \alpha_n(x_n) + v_{n-1}$$
(II.11)

#### <u>L'étape n</u>

Dans cette étape, on considère le système complet décrit par l'équation (II.1)

Pour  $u = \alpha_2(x_2 \dots, x_n)$  (calculée dans l'étape (n-1)), le système (II.1) devient :

$$\begin{cases} x_1 = \phi_1(x_2, \dots, x_n) \\ \dot{x}_2 = f_2(x_3, \dots, x_n) + g_2(x_3, \dots, x_n) \alpha_2(x_2, \dots, x_n) \\ \vdots \\ \dot{x}_{n-1} = f_{n-1}(x_n) + g_{n-1}(x_n)(x_2, \dots, x_n) \\ \dot{x}_n = (x_2, \dots, x_n) \end{cases}$$
(II.12)

Avec le terme d'interconnexion :

$$\phi_1(x_2, \dots, x_n) = f_1(x_2, \dots, x_n) + g_1(x_2, \dots, x_n)\alpha_2(x_2, \dots, x_n)$$
(II.13)

On note  $\tilde{x}_i(t)$ , pour  $i \in [1, n]$ , la solution temporelle du système (II.12). La nouvelle entrée de commande pour (II.1) sera :

$$u = \alpha_2(x_2, \dots x_n) + v_1$$
 (II.14)

La fonction de Lyapunov est alors

$$V_{1} = V_{2} + \frac{1}{2}x_{1}^{2} + \int_{0}^{\infty} \tilde{x}_{1}(t)\phi_{1}(\tilde{x}_{2}(t), \dots, \tilde{x}_{n}(t))dt$$

$$= V_{2} + \frac{1}{2}\lim_{t \to \infty} \check{x}_{1}^{2}(t)$$
(II.15)

Sa dérivée est :

$$\dot{V}_{1} = \dot{V}_{1}|_{u=\alpha_{2}(x_{n})} + (L_{g}V_{1})v_{1}$$

$$= -\lambda_{2}(L_{g}V_{2})^{2} + (L_{g}V_{1})v_{1}$$
(II.16)

Donc pour rendre  $\dot{V}_1$  définie négative, on peut prendre le retour d'état non linéaire suivant :

$$\nu_1 = -\lambda_1 L_g V_1 \operatorname{Avec} \lambda_1 > 0 \tag{II.17}$$

Il en résulte :

$$\dot{V}_1 = -\lambda_2 (L_g V_2)^2 - \lambda_1 (L_g V_1)^2$$
 (II.18)

Cette fonction est définie négative donc la stabilité globale du système (II.1) est assurée. La commande finale est [54] :

$$u = \alpha_1(x_1, \dots x_n) = \alpha_2(x_2, \dots x_n) + v_1$$
(II.19)

# 2.2 Commande par backstepping

La commande par Backstepping est une loi de commande pour les systèmes non linéaires basée sur le théorème de Lyapunov. Elle s'applique à des systèmes de la forme cascade triangulaire. Chaque état du système commande l'état précédent et s'appelle Commande Virtuelle, jusqu'au dernier état qui est commandé par la commande réelle. L'objectif de cette technique est de calculer en plusieurs étapes une loi de commande qui assure la stabilité globale du système.

Le backstepping représente une méthode récursive qui permet de construire une loi de commande qui garantit, en tout temps, la stabilité du système. L'écriture des états sous forme paramétrique pure met en évidence les sous-systèmes. Pour chacune de ces parties, il faut trouver, à l'aide d'une fonction de Lyapunov, une commande qui permet de stabiliser ce sous-système.

Pour ce faire, l'état suivant est considéré comme la nouvelle entrée de commande (commande virtuelle). L'ordre du sous-système est ensuite augmenté et le développement précédent est recommencé. A la fin, une loi de commande est obtenue [59].

#### 2.2.1 Etapes de synthèse

Le Backstepping n'est en fait que la construction de la fonction de Lyapunov ainsi que la commande étape par étape, et cela pour un système qui peut être écrit sous la forme, dite cascade, suivante :

$$\begin{cases} \dot{x}_{1} = x_{2} + \Phi_{1}(x_{1}) \\ \dot{x}_{2} = x_{3} + \Phi_{2}(x_{1}, x_{2}) \\ \vdots \\ \dot{x}_{n-1} = x_{n} + \Phi_{n-1}(x_{1}, x_{2}, \dots, x_{n-1}) \\ \dot{x}_{n} = u + \Phi_{n}(x_{1}, x_{2}, \dots, x_{n}) \\ y = x_{1} \end{cases}$$
(II.20)

Où  $x \in \mathbb{R}^n$  et  $u \in \mathbb{R}$ 

#### 2.2.2 Principe de la commande backstepping

La commande backstepping est basée sur la décomposition du système entier de la commande, qui est généralement multi variable et d'ordre élevé, en une cascade de soussystèmes de commande du premier ordre. Pour chaque sous-système, une loi de commande dite virtuelle est calculée. La détermination des lois de commande qui découle de cette approche est basée sur l'emploi des fonctions de Lyaponuv soit définie positive et sa dérivée soit toujours négative.

#### 2.2.3 Objectifs de la commande backstepping :

- Mettre en place des lois de commande qui stabilisent les différentes étapes du système en cascade,
- Association de la passivité au backstepping, nous donnons plus d'ouverture à certaines conceptions notamment en robotique,
- Assurer l'obtention de lois de commande très performantes,
- Calculer en plusieurs étapes une commande qui garantit la stabilité globale du système.

#### 2.2.4 Avantages de la commande backstepping

- Ne conduit pas à l'annulation des non linéarités utiles et permet de poursuivre des objectifs de stabilisation ou de poursuite, plutôt que des objectifs de linéarisation.
- Présente l'avantage d'être robuste vis-à-vis des variations paramétriques de la machine ainsi qu'une bonne poursuite des références.
- Impose des propriétés désirées de stabilité en fixant, dès le début de la conception, puis en calculant les termes à ajouter à cette fonction d'une manière récursive.
- Garder les propriétés du système initial dans la loi de commande.

#### 2.2.5 Domaines d'application du backstepping

C'est grâce aux travaux de KOKOTOVIC [54 - 55], un modèle mathématique est fourni par la conception de loi de contrôle pour différents systèmes non linéaires en utilisant cette technique.

Le backstepping représente une méthode récursive qui permet de construire une loi de commande qui garantit, en tout temps, la stabilité du système. L'écriture des états sous forme paramétrique pure met en évidence les sous-systèmes. Pour chacune de ces parties, il faut

trouver à l'aide d'une fonction de Lyapunov, une commande qui permet de stabiliser ce soussystème.

#### 2.2.6 Exemple d'application

$$\begin{cases} \dot{x}_1 = f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 = f_2(x_1, x_2) + g_2(x_1, x_2)x_3 \\ \dot{x}_3 = f_3(x_1, x_2, x_3) + g_3(x_1, x_2, x_3)u \end{cases}$$
(II. 21)

Où  $g_i$  et  $f_i$  sont des fonctions non-linéaires connues, tel que :

$$F_i(0) = 0G_i(x) \neq 0 \forall (x_1, x_2, x_3) \in \mathbb{R}^3$$
(II. 22)

On désire imposer à la sortie  $y = x_1$  le signal de référence  $y_r$ 

Où  $y_r$ ,  $\dot{y}_r$ ,  $\ddot{y}_r$  et  $\ddot{y}_r$  sont supposées connues et uniformément bornées. Le système étant du troisième ordre.

La conception de la loi de commande s'effectue en trois étapes :

#### • Etape 1 :

On considère d'abord le premier sous-système :

$$\dot{x}_1 = f_1(x_1) + g_1(x_1)x_2$$
 (II.23)

La variable d'état  $x_2$  est traitée comme une commande et l'on définit la première valeur désirée :

$$x_1^d = \alpha_0 = y_r \tag{II.24}$$

La première variable d'erreur se définit par :

$$e_1 = x_1 - \alpha_0 \tag{II.25}$$

Sa dérivée est donnée comme suit :

$$\dot{e}_1 = \dot{x}_1 - \dot{\alpha}_0 = f_1 + g_1 x_2 - \dot{\alpha}_0 \tag{II.26}$$

Pour un tel système la fonction quadratique :

$$V_1(e_1) = \frac{1}{2}e_1^2 \tag{II.27}$$

Sa dérivée est donnée par :

$$\dot{V}_1 = e_1 \dot{e}_1 = e_1 [f_1 + g_1 x_2 - \dot{\alpha}_0]$$
 (II. 28)

Un choix judicieux de  $x_2$  qui rend  $\dot{V}_1$  négative et assure la stabilité de l'origine du soussystème décrit par (II.24) est :

$$x_2^d = \alpha_1 = \frac{1}{g_1} \left[ -k_1 e_1 - f_1 + \dot{\alpha}_1 \right]$$
(II. 29)

 $k_1 > 0$  est un paramètre de conception.

Par conséquence, l'équation (II. 28) devient :

$$\dot{V}_1 = -k_1 e_1^2 \le 0 \tag{II.30}$$

#### • Etape 2 :

On considère, dans ce cas, les deux premiers sous-systèmes :

$$\begin{cases} \dot{x}_1 = f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 = f_2(x_1, x_2) + g_2(x_1, x_2)x_3 \end{cases}$$
(II. 31)

Et on définit la nouvelle variable d'erreur :

$$e_2 = x_2 - \alpha_1 \tag{II.32}$$

La dynamique des erreurs  $(e_1, e_2)$  s'écrit comme suit :

$$\begin{cases} \dot{e}_1 = f_1 + g_1(e_2 + \alpha_1) - \dot{\alpha}_0 \\ \dot{e}_2 = f_2 + g_2 x_3 - \dot{\alpha}_1 \end{cases}$$
(II. 33)

Pour lequel on choisit comme fonction de Lyapunov [31] :

$$V_2 = (e_1, e_2) = V_1 + \frac{1}{2}e_2^2$$
 (II. 34)

Cette fonction de Lyapunov a pour dérivée :

$$\dot{V}_{1}(e_{1},e_{2}) = \dot{V}_{1} + e_{2}\dot{e}_{2}$$

$$= e_{1}[f_{1} + g_{1}(e_{2} + \alpha_{1}) - \dot{\alpha}_{0}] + e_{2}[f_{2} + g_{2}x_{3} - \dot{\alpha}_{1}]$$

$$= -k_{1}e_{1}^{2} + e_{2}[f_{2} + g_{1}e_{1} + g_{2}x_{3} - \dot{\alpha}_{1}]$$
(II. 35)

On prend  $x_3$  comme une deuxième commande virtuelle dont la valeur est donnée par :

$$(x_3)_d = \alpha_2 = \frac{1}{g_2} [\dot{\alpha}_1 - g_1 e_1 - f_2 - k_2 e_2]$$
(II. 36)

Où  $k_2 > 0$ , et  $\dot{\alpha}_1$  est calculé analytiquement

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_1}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_1}{\partial \dot{y}_r} \ddot{y}_r$$
(II. 37)

Un tel choix permet de réduire la dérivée à :

$$\dot{V}_2 \le -k_1 e_1^2 - k_2 e_2^2 \le 0 \tag{II.38}$$

• Etape 3 :

Le système (II.21) est maintenant considéré dans sa globalité.

La nouvelle variable d'erreur est :

$$e_3 = x_3 - \alpha_2$$
 (II. 39)

Ce qui permet d'écrire la dynamique des erreurs  $(e_1, e_2 et e_3)$ :

$$\begin{cases} \dot{e}_1 = f_1 + g_1(e_2 + \alpha_1) - \dot{\alpha}_0 \\ \dot{e}_2 = f_2 + g_2(e_3 + \alpha_2) - \dot{\alpha}_1 \\ \dot{e}_3 = f_3 + g_3 U - \dot{\alpha}_3 \end{cases}$$
(II. 40)

On prend comme fonction de Lyapunov

$$V_3(e_1, e_2, e_3) = V_2 + \frac{1}{2}e_3^2$$
 (II. 41)

Sa dérivée s'écrit sous la forme :

$$\dot{V}_3(e_1, e_2, e_3) = \dot{V}_2 + e_3 \dot{e}_3 = -k_1 e_1^2 - k_2 e_2^2 + e_3 [f_3 + g_2 e_2 + g_3 U - \dot{\alpha}_2]$$
(II. 42)

Le choix approprié de la vraie commande U est par :

$$U = \frac{1}{g_3} [\dot{a}_2 - f_3 - g_2 e_2 - k_3 e_3]$$
(II. 43)

Où  $k_3 > 0$ , *et*  $\dot{\alpha}_2$  est calculé analytiquement comme suit :

$$\dot{\alpha}_2 = \frac{\partial \alpha_2}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_2}{\partial x_2} \dot{x}_2 + \frac{\partial \alpha_2}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_2}{\partial \dot{y}_r} \ddot{y}_r + \frac{\partial \alpha_2}{\partial \ddot{y}_r} \ddot{y}_r$$
(II. 44)

Avec ce choix, on trouve :

$$\dot{V}_3(e_1, e_2, e_3) \le -k_1 e_1^2 - k_2 e_2^2 - k_3 e_3^2 \le 0$$
 (II. 45)

D'où la stabilité en boucle fermée du système (II.21) et la réduction à zéro de l'erreur de poursuite $(y - y_r)$ , par conséquent les deux principaux objectifs, la stabilité et la poursuite, sont atteints.

#### 2.3 Régulateur proportionnel-intégrale-dérivée PID

La technique de commande par PID (Proportionnel, Intégral, Dérivé) est certainement l'approche la plus connue et la plus simple à implémenter sur un système. Elle facilite la compréhension du rôle physique de chacun des termes de la commande, ce qui permet donc de régler les gains. Cette technique de commande a été testée avec succès sur plusieurs plateformes expérimentales comme les hélicoptères, les quadrotors et les drones à rotors coaxiaux.

Dans le domaine de l'industrie, la plupart des régulateurs utilisés sont des régulateurs PID. Les raisons du succès du PID sont :

• Structure simple.

• Hautes performances dans plusieurs processus.

Dans la robotique, les techniques PID représentent les bases du contrôle.

La loi de commande du PID est donnée comme suit

 $(t) = K_p \times (t) + K_i \int \varepsilon(t) dt + K_d \varepsilon(t) dt$ 

Où *K*<sub>d</sub> est la constante de dérivation.

Le premier terme  $k_p$  est proportionnel à l'erreur. Il définit la bande passante proportionnelle. À l'intérieur de cet intervalle, la sortie doit être proportionnelle à l'erreur, tandis qu'à l'extérieur la sortie doit être au minimum ou au maximum. Le deuxième terme intégrateur varie selon l'intégrale de l'erreur. Bien que ce composant augmente le dépassement et le temps de réglage, il a une propriété unique, c'est qu'il élimine l'erreur statique. Le troisième terme K<sub>d</sub> varie selon la dérivée de l'erreur. Ce terme aide à réduire les dépassements et le temps de réglage [49].

#### 2.3.1 Linéarisation du modèle autour d'un point d'équilibre

Pour notre système, défini par le vecteur d'état (II-31) et le modèle d'état (II-34), nous aurons un système de la forme :

$$\dot{x}(t) = f(x) + g(x)u(t)$$
 (II-46)

#### Où

La linéarisation de notre système est réalisée par le calcul des jacobiens autour du point d'équilibre défini par :

$$x_e = [0 \ 0 \ 0 \ 0 \ constante \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{II-47}$$

Avec le nouveau vecteur d'état :

$$\checkmark x_{L}^{T} = [Z \dot{Z} \phi \dot{\phi} \theta \dot{\theta} \psi \dot{\psi}] \text{ et } x_{L}^{T} = [x_{1} \ x_{2} \ x_{3} \ x_{4} \ x_{5} \ x_{6} \ x_{7} \ x_{8}]$$

Nous aurons :

#### **Chapitre 2** Rappels théoriques sur les commandes robustes linéaires et non linéaires

A =	$ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\$	$     1 \\     0 \\    $	0 0 0 0 0 0 0	0 0 1 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 1 0 0 0	0 0 0 0 0 0 0	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ \end{array}$	(II-48)	et	B =	$\begin{bmatrix} 0 \\ B_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	0 0 0 B <sub>2</sub> 0 0 0 0 0	0 0 0 0 B <sub>3</sub> 0 0	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ B_4 \end{bmatrix}$	(II-49)
	Ave	ec :														

 $B_1 = \frac{1}{m}$  et  $B_2 = b_1$  et  $B_3 = b_2$  et  $B_4 = b_3$ .

D'où le modèle linéaire suivant :

Comme nous pouvons constater, le modèle (II-50) est sous forme d'une chaîne d'intégrateurs, et peut être décomposé en quatre sous-systèmes SISO comme suit :

• Chaîne d'altitude (z) :

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ B_1 \end{bmatrix} \mathbf{u}_1$$
 (II-51)

• Chaîne de roulis (φ):

$$\begin{bmatrix} \dot{\mathbf{x}}_3\\ \dot{\mathbf{x}}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1\\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_3\\ \mathbf{x}_4 \end{bmatrix} + \begin{bmatrix} 0\\ B_2 \end{bmatrix} \mathbf{u}_2$$
 (II-52)

• Chaîne de tangage  $(\theta)$  :

$$\begin{bmatrix} \dot{\mathbf{x}}_5\\ \dot{\mathbf{x}}_6 \end{bmatrix} = \begin{bmatrix} 0 & 1\\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_5\\ \mathbf{x}_6 \end{bmatrix} + \begin{bmatrix} 0\\ B_3 \end{bmatrix} \mathbf{u}_3$$
 (II-53)

• Chaîne de lacet  $(\Psi)$  :

$$\begin{bmatrix} \dot{\mathbf{x}}_7 \\ \dot{\mathbf{x}}_8 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_7 \\ \mathbf{x}_8 \end{bmatrix} + \begin{bmatrix} 0 \\ B_4 \end{bmatrix} \mathbf{u}_4$$
 (II-54)

## 2.4 Méthode du loopshaping

Le domaine fréquentiel est un outil puissant pour l'analyse et la synthèse des correcteurs. Les performances désirées (rejet de perturbation, rapidité, etc.) peuvent être assurées en imposant une certaine forme pour le lieu fréquentiel du système en boucle fermée. Parmi les techniques les plus répandues, on trouve les méthodes de modelage ou de mise en forme de boucle (loopshaping) [58, 61, 18]

On appelle spécification loopshaping le fait de spécifier la boucle ouverte de l'asservissement à travers les diverses spécifications désirées en boucle fermée car il est plus facile de travailler sur un unique transfert que sur une multitude de transferts.

#### 2.4.1 Notion de modelage de boucle (loopshaping)

Le choix des pulsations spécifiques  $\omega_b$  et  $\omega_h$  dépend des applications spécifiques et des connaissances que l'on possède sur les perturbations et les bruits réels affectant le système. C'est une étape importante dans l'étape de modélisation. Les spécifications fréquentielles précédentes reviennent à donner une forme particulière sur l'ensemble des pulsations à la représentation fréquentielle des fonctions de sensibilité ou à la fonction de transfert en boucle ouverte L(p).

Si l'on souhaite travailler uniquement sur cette dernière, la traduction des exigences de performance vues précédemment reviennent à modeler la courbe de la boucle ouverte sur différentes plages de fréquences. Réaliser un compensateur remplissant de telles exigences est appelé faire la synthèse d'un correcteur par loopshaping. La courbe de gain dans le plan de Bode de la boucle ouverte  $L(j\omega) = K(j\omega)G(j\omega)$  prend classiquement la forme donnée par la Figure 2-1.



Figure 2-1: Loopshaping de la boucle ouverte

#### Chapitre 2 Rappels théoriques sur les commandes robustes linéaires et non linéaires

#### 2.4.2 Spécifications de performance par loopshaping

Les spécifications de performance par loopshaping sont élaborées sur le gain de boucle, la fonction de transfert en boucle ouverte  $L(j\omega) = K(j\omega)G(j\omega)$ .

- 1) spécification du nombre d'intégrateurs dans L(p) par la règle des intégrateurs,
- spécification de la pente de la courbe de gain en boucle ouverte |L(jω)| sur certaines plages de fréquence :
  - pente de -20 dB/décade autour de  $\omega_{co}$  pour satisfaire la condition de roll-off,
  - pente supérieure à −20 dB/décade aux hautes fréquences (ω ≥ ω<sub>h</sub>) pour le filtrage des bruits.
  - pente aux basses fréquences ( $\omega \le \omega_b$ ) définie par le modèle de la perturbation b(p).
- 3) spécification d'une pulsation de coupure  $\omega_{c0}$  pour répondre aux impératifs de stabilité en termes de marge de phase et de gain.

L'intérêt de spécifier les performances par loopshaping provient principalement de la simplicité de l'expression de la boucle ouverte comme fonction du compensateur K(p) puisque :

$$L(p) = K(p) G(p)$$
(II.55)

La fonction de transfert du compensateur est déduite par la méthode d'inversion du modèle.

$$K(p) = L(p)G^{-1}(p)$$
 (II.56)

Cette méthode est généralement bien adaptée pour les problèmes simples.

#### 2.4.3 Exemple de Loopshaping

Le modèle du système est donné par sa fonction de transfert en boucle ouverte pour un asservissement à retour unitaire.

$$G(p) = \frac{3(-2p+1)}{(5p+1)(10p+1)}$$
(II.57)

Ce modèle possède les caractéristiques suivantes :

- Il est à non minimum de phase du fait de la présence d'un zéro instable p=0.5.
- Il est stable en boucle ouverte puisque ses pôles sont  $p_1=-1/5$  et  $p_2 = -1/10$ correspondant aux pulsations de cassure caractéristiques  $\omega_{p1} = 0.2$  rad/s et  $\omega_{p2} = 0.1$ rad/s.

Le diagramme de Bode de la courbe de gain et de phase de cette fonction de transfert permet de calculer une marge de phase  $M_{\varphi}$ = 48.5 deg, une marge de gain  $K_g$  = 2.5 et des pulsations de coupure  $\omega_{-180^\circ}$  = 0.41 rad/s,  $\omega_{co}$  = 0.2 rad/s.

# Bode Diagrams Gm=7.9588 dB (at 0.41231 rad/sec), Pm=48.448 deg. (at 0.20372 rad/sec) 10 0 -10 Phase (deg); Magnitude (dB) -20 -30 -50 -100 -150 -200 -250 10<sup>0</sup> 10<sup>-1</sup> Frequency (rad/sec)

Figure 2- 2: diagramme de Bode de G(P).

On souhaite corriger ce système afin qu'il vérifie les spécifications suivantes.

- Le système corrigé ne doit pas présenter d'erreur en régime permanent à une réponse en échelon de position.
- Le modèle de la perturbation étant un premier ordre donné par :

$$w(p) = G_b(p)a(p) = \frac{1}{p+1}a(p)$$
(II.58)

On désire que la sortie revienne à zéro le plus rapidement possible si  $a(p) = \frac{1}{n}$ .

• Le bruit de mesure intervenant au-delà de 0.5 Hz doit être filtré.

Les spécifications du loop-shaping sont remplies de la manière suivante :

#### Chapitre 2 Rappels théoriques sur les commandes robustes linéaires et non linéaires

#### **Première spécification :**

La règle des intégrateurs indique qu'il est nécessaire d'ajouter une intégration dans la boucle ouverte conduisant à écrire :

$$L(p) = \frac{3(-2p+1)}{p(5p+1)(10p+1)}$$
(II.59)

#### **Deuxième spécification :**

La condition de roll-off est vérifiée du fait de l'intégration. Le bruit de mesure intervient à partir de la pulsation  $\omega_h = 3$  rad/s. Cela implique l'ajout éventuel d'un pôle dont la pulsation de cassure est supérieure à cette pulsation. La pente aux basses fréquences est de -20 dB/dec du fait de l'intégration. Cela conduit à simplifier les deux pôles de G(p) qui sont susceptibles de fournir trop de roll-off sur la mauvaise plage de pulsations du fait de leur pulsation de cassure respective. L'analyse de la perturbation b(p) conduit à écrire :

$$|G_b(j\omega)| = \frac{1}{\sqrt{1+\omega^2}} \tag{II.60}$$

Pour $\omega = 0.1 rad/s$ ,  $|G_b(j\omega)| \sim 0 dB$ . Cela permet de choisir  $\omega_b = 0.1 rad/s$ .

A ce stade, on a donc une fonction de transfert en boucle ouverte :

$$L(p) = \frac{K(-2p+1)}{p(\alpha p+1)(\beta p+1)}$$
(II.61)

On choisit  $\alpha = 2$  afin d'avoir une pulsation de coupure  $\omega_{p\alpha} = 0.5$  rad/s et  $\beta = 0.33$ Afin d'avoir  $\omega_{p\beta} = 3$ rad et un roll-off de 2 au-delà, on obtient :

$$L(p) = \frac{-2p+1}{p(2p+1)(0.33p+1)}$$
(II.62)

Donc on fait le tracé dans le plan de Bode pour obtenir les marges de phase et de gain :

$$M_{\phi}$$
= -52 deg,  $K_{g}$  = -0.44

Et des pulsations de coupure  $\omega_{-180^\circ} = 0.43$  rad/s,  $\omega_{co} = 0.95$  rad/s.



Figure 2- 3: diagrammes de Bode de L(p).

Ce système est visiblement instable en boucle fermée.

#### Troisième spécification :

A l'aide d'un gain K que l'on introduit dans la boucle ouverte, la courbe de gain est déplacée verticalement afin de déplacer la pulsation de coupure  $\omega_{co}$  vers la gauche. La courbe de gain est descendue de -15 dB conduisant au calcul de K = 0.15 et à une pulsation de coupure  $\omega_{co} = 0.2$  rad/s. On obtient :

$$L(p) = \frac{0.15(-2p+1)}{p(2p+1)(0.33p+1)}$$
(II.63)

Les diagrammes de Bode d'une telle boucle ouverte conduit à calculer une marge de phase et une marge de gain  $M_{\phi} = 54$  deg, Kg = 2.92 et des pulsations de coupure :

 $\omega_{-180o}=0.44~rad/s,\,\omega_{co}=0.15~rad/s$ 

#### Chapitre 2 Rappels théoriques sur les commandes robustes linéaires et non linéaires



Figure 2- 4: Courbe de gain de L(p).

Par la méthode d'inversion de modèle, on obtient un compensateur suivant :

$$K(p) = \frac{0.15(10p+1)(5p+1)}{p(2p+1)(0.33p+1)}$$
(II.64)

Les figures suivantes sont les réponses temporelles à une entrée en échelon de position unitaire r(p) = 1/p pour le système bouclé non corrigé (pour lequel il existe une erreur de position) et le système bouclé corrigé par K(p).



Figure 2- 5: Réponses indicielles.

La Figure 2-6 représente les sorties du système quand a(p) = 1/p pour le système non corrigé et le système corrigé.



Figure 2-6: Réponses à une perturbation en échelon.

# 2.5 Etude de la robustesse des commandes

Suposons que

 $\dot{x} = u + \Delta(x)$  avec  $x \in \mathcal{R}$ 

Ou  $\Delta(x)$  est une erreur de modélisation inconnue mais bornée.

 $|\Delta(x)| \leq \overline{\Delta}|$  avec  $\overline{\Delta}$  connu

Soit

 $V(x) = \frac{1}{2}x^2$  une fonction définie positive sur R

$$\dot{V} = x\dot{x} = x[u + \Delta(x)]$$

Soit  $u = -\lambda_1 x - \lambda_2 \gamma(x)$  où  $\gamma(x)$  est une fonction non linéaire verifiant :

 $x \gamma(x) \ge |x|$  par exemple  $(\gamma(x) = sign(x))$  $\dot{V} = -\lambda_1 x^2 + x[\Delta(x) - \lambda_2 \gamma(x)]$ 

Le terme 2 est négatif pour  $\lambda_2 \ge \overline{\Delta}$ Donc  $\dot{V} = -\lambda_1 x^2$  fonction définie négative sur R

# 2.6 Conclusion

Ce chapitre est consacré à la présentation de la théorie de chacune des techniques de commande utilisées, suivie d'un exemple d'application qui permet de mieux comprendre leur principe de synthèse. L'application de ces différentes techniques fera l'objet du prochain chapitre.

# Chapitre 3

# Application et simulation sur le modèle d'état du quadrirotor

## **3** Introduction

Dans ce chapitre, nous allons concevoir deux lois de commande basées sur les techniques forwarding et la technique combinée PID\_Backstepping. La commande forwarding est une méthode de commande robuste qui permet de prendre en compte les perturbations sur le système ainsi que les erreurs commises lors de la modélisation qui sont aussi appelées incertitudes ou variations paramétriques [48].

Le modèle dynamique du quadrirotor est donné par l'équation (I.35). Les paramètres du quadrirotor sont donnés par le Tableau 3-1:

Designation	Paramètre	Valeur	Unité
Masse	М	0.650	Kg
Moment d'inertie du quadrirotor par rapport à x	$j_x$	7.5e-3	Kg.m <sup>2</sup>
Moment d'inertie du quadrirotor par rapport à y	j <sub>y</sub>	7.5e-3	Kg.m <sup>2</sup>
Moment d'inertie du quadrirotor par rapport à z	j <sub>z</sub>	1.3e-2	Kg.m <sup>2</sup>
Coefficient de portance	В	3.13e-5	N.s <sup>2</sup>
Coefficient de trainée	k	7.5e-5	$N .m.s^2$
Moment d'inertie du rotor par rapport à z	j <sub>rz</sub>	6e-5	Kg.m <sup>2</sup>
Distance entre le rotor et le centre de gravité	L	0.23	М

Tableau 3-1: Paramètres du quadrirotor.

## 3.1 Synthèse des lois de commande forwarding pour le quadrirotor

L'objectif principal est de synthétiser une loi de commande de forwarding pour un hélicoptère quadrirotor forçant la position {x (t), y (t), z (t)} et l'angle { $\psi$  (t)} à poursuivre les valeurs souhaitées {x<sub>d</sub> (t), y<sub>d</sub> (t), z<sub>d</sub> (t),  $\psi$ <sub>d</sub> (t)}.

Introduisons les entrées virtuelles basées sur les angles d'inclinaison ( $\phi$ ,  $\theta$ ) donnés comme suit:

$$\begin{cases} U_{x} = Cx_{7}Sx_{8}Cx_{9} + Sx_{7}Sx_{9} \\ U_{y} = Cx_{7}Sx_{8}Sx_{9} - Sx_{7}Cx_{9} \end{cases}$$
(III.1)

Toutes les erreurs de poursuite sont écrites sous la forme suivante :

$$e_{i} = \begin{cases} x_{i} - x_{id} / i\epsilon \{1, 2, 3, 7, 8, 9\} \\ x_{i} - \dot{x}_{(i-1)d} / i\epsilon \{4, 5, 6, 10, 11, 12\} \end{cases}$$
(III.2)

Toutes les fonctions de Lyapunov prennent la forme :

$$V_{i} = \begin{cases} \frac{1}{2} e_{i}^{2} / i\epsilon \{4, 5, 6, 10, 11, 12\} \\ V_{i+1} + \frac{1}{2} \lim_{t \to \infty} \tilde{e}_{i}^{2} / i\epsilon \{1, 2, 3, 7, 8, 9\} \end{cases}$$
(III.3)

Avec  $\tilde{e}_i$  représente la solution temporelle du système.

Les étapes de la synthèse de la commande sont expliquées pour le sous-système :

$$\begin{cases} \dot{x}_7 = x_{10} \\ \dot{x}_{10} = a_1 x_{11} x_{12} + b_1 U_2 \end{cases}$$
(III.4)

Soit l'erreur de poursuite définie par  $e_7 = x_7 - x_{7d}$ Sa dynamique est décrite par

$$\dot{e}_7 = x_{10} - \dot{x}_{7d}$$

Si on pose  $e_{10} = x_{10} - \dot{x}_{7d}$ , le système (III.4) devient :

$$\begin{cases} \dot{e}_7 = e_{10} \\ \dot{e}_{10} = a_1 x_{11} x_{12} - \ddot{x}_{7d} + b_1 U_2 \end{cases}$$
(III.5)

La synthèse s'effectue en deux étapes :

#### Première étape :

On ne considère que la deuxième équation du système (III.5). Soit la fonction de Lyapunov

$$V_{10} = \frac{1}{2}e_{10}^2$$

Sa dérivée est  $\dot{V}_{10} = e_{10}\dot{e}_{10}$ 

Pour rendre  $\dot{V}_{10}$  définie négative, on peut choisir la commande :

$$U_2 = \alpha_{10}(x) = \frac{1}{b_1} \left( -a_1 x_{11} x_{12} - \lambda_{10} e_{10} + \ddot{x}_{7d} \right) \ avec\lambda_{10} \ge 0 \tag{III.6}$$

On aura  $\dot{V}_{10} = -\lambda_{10}e_{10}^2$  qui est définie négative sur R. Cette commande va assurer la convergence de  $e_{10}$  vers l'origine.

#### Deuxième étape :

En remplaçant la commande trouvée dans l'étape précédente dans le système (III.5), on obtient

$$\begin{cases} \dot{e}_7 = e_{10} \\ \dot{e}_{10} = -\lambda_{10} e_{10} \end{cases}$$
(III.7)

La solution temporelle de ce système est

$$\begin{cases} \tilde{e}_{10}(t) = e_{10}(0)e^{-\lambda_{10}t} \\ \tilde{e}_{7}(t) = -\frac{1}{\lambda_{10}}e_{10}(0)e^{-\lambda_{10}t} + e_{7}(0) + \frac{1}{\lambda_{10}}e_{10}(0) \end{cases}$$
(III.8)

On pose

$$\begin{cases} e_{10}(0) = e_{10} \\ e_{7}(0) = e_{7} \end{cases}$$
(III.9)

Soit la fonction de Lyapunov

$$V_7 = V_{10} + \frac{1}{2} \lim_{t \to \infty} \tilde{e}_7^2(t)$$
(III.10)

$$V_7 = \frac{1}{2} \left[ e_{10}^2 + (e_7 + \frac{1}{\lambda_{10}} e_{10})^2 \right]$$
(III.11)

Si on pose  $U_2 = \alpha_{10}(x) + \omega_{10}$  la dérivée de  $V_7$  est

$$\dot{V}_7 = e_{10}\dot{e}_{10} + (e_7 + \frac{1}{\lambda_{10}}e_{10})(\dot{e}_7 + \frac{1}{\lambda_{10}}\dot{e}_{10})$$
(III.12)

$$\dot{V}_7 = -\lambda_{10}e_{10}^2 + \left[\frac{e_7}{\lambda_{10}} + \left(\frac{1}{\lambda_{10}^2} + 1\right)e_{10}\right]b_1\omega_{10}$$
(III.13)

Pour rendre  $\dot{V}_7$  définie négative, on peut prendre :

$$\omega_{10} = -\frac{\lambda_7}{b_1} \left[ \frac{e_7}{\lambda_{10}} + \left( \left( \frac{1}{\lambda_{10}^2} + 1 \right) e_{10} \right] \quad \text{Avec } \lambda_7 \ge 0 \tag{III.14}$$

La commande  $U_2$  devient :

$$U_{2} = \frac{1}{b_{1}} \left( -a_{1} x_{11} x_{12} - \frac{\lambda_{7}}{\lambda_{10}} e_{7} - \left( \frac{\lambda_{7}}{\lambda_{10}^{2}} + \lambda_{7} + \lambda_{10} \right) e_{10} + \ddot{x}_{7d}$$
(III.15)

Cette commande va assurer la stabilisation du système.

Les mêmes étapes sont reprises afin de déterminer $U_1$ ,  $U_3$ ,  $U_4$ ,  $U_x$  et  $U_y$ .

Les commandes du système avec forwarding sont données comme suit :

$$\begin{cases} U_{2} = \frac{1}{b_{1}} \left( -a_{1}x_{11}x_{12} - \frac{\lambda_{7}}{\lambda_{10}}e_{7} - \left(\frac{\lambda_{7}}{\lambda_{10}^{2}} + \lambda_{7} + \lambda_{10}\right)e_{10} + \ddot{x}_{7d} \\ U_{3} = \frac{1}{b_{2}} \left( -a_{1}x_{11}x_{12} - \frac{\lambda_{8}}{\lambda_{11}}e_{8} - \left(\frac{\lambda_{8}}{\lambda_{11}^{2}} + \lambda_{8} + \lambda_{11}\right)e_{11} + \ddot{x}_{8d} \\ U_{4} = \frac{1}{b_{3}} \left( -a_{5}x_{10}x_{11} - \frac{\lambda_{9}}{\lambda_{12}}e_{9} - \left(\frac{\lambda_{9}}{\lambda_{12}^{2}} + \lambda_{9} + \lambda_{12}\right)e_{12} + \ddot{x}_{9d} \\ U_{1} = \frac{m}{Cx_{7}Cx_{8}}\left(g - \frac{\lambda_{3}}{\lambda_{6}}e_{3} - \left(\frac{\lambda_{3}}{\lambda_{6}^{2}} + \lambda_{3} + \lambda_{6}\right)e_{6} + \ddot{x}_{3d} \\ U_{x} = \frac{m}{U_{1}}\left(-\frac{\lambda_{1}}{\lambda_{4}}e_{1} - \left(\frac{\lambda_{1}}{\lambda_{4}^{2}} + \lambda_{1} + \lambda_{4}\right)e_{4} + \ddot{x}_{1d} \\ U_{y} = \frac{m}{U_{1}}\left(-\frac{\lambda_{2}}{\lambda_{5}}e_{2} - \left(\frac{\lambda_{2}}{\lambda_{5}^{2}} + \lambda_{2} + \lambda_{5}\right)e_{5} + \ddot{x}_{2d} \end{cases}$$
(III.16)

Avec  $\lambda_i \ge 0$  pour i  $\in [1, 12]$ 

#### 3.1.1 Résultats de simulation

Pour évaluer l'efficacité de la commande par la méthode de forwarding, quelques résultats de simulations sont présentés pour la commande en position et en orientation du quadrirotor. Une application de l'approche proposée est évaluée sur le modèle dynamique complet du drone. Les valeurs initiales sont définies comme suit :  $\xi(0) = [0; 0; 0]^T$  m et  $\eta(0) = [0; 0; 0]^T$  rad, les vitesses linéaire et angulaire initiales sont supposées nulles .

Plusieurs tests de vol virtuels ont été faits pour démontrer les performances de la commande forwarding.

La première application ne prend pas en compte les perturbations. Dans la deuxième application, les perturbations externes sont données par un échelon de 10% de la consigne. Une autre perturbation qui est un bruit de mesure aléatoire. Puis, on considère le cas d'une perturbation interne qui est la variation des paramètres.

#### 3.1.1.1 Test de vol sans perturbations externes

Dans ce cas, on ne prend pas en compte les termes de perturbations externes. La figure 3-1 montre l'évolution de la position et l'orientation du véhicule pendant le vol. On constate que, le contrôleur proposé est capable de suivre les trajectoires de référence.







Figure 3-1: Résultats de poursuite dans le cas de trajectoires de référence de type échelon.



Figure 3-2: Erreurs de poursuite.



Figure 3- 3: Signaux de commande.

#### 3.1.1.2 Test de vol en présence des perturbations externes

Dans ce cas, nous introduisons des perturbations externes de nature constante et de nature aléatoire ainsi que des incertitudes paramétriques.

a. Réponse du système en présence de perturbations de type échelon :

La Figure 3-4 montre les résultats de simulation avec une perturbation constante appliquée à partir de la  $25^{\text{ème}}$  seconde, nous constatons que cette perturbation est éliminée en un temps fini et le système garde ses performances et présente une faible erreur de poursuite.

b. Réponse du système en présence de perturbations de type aléatoire :

Dans ce cas, nous remarquons à la Figure 3-6 que le système garde ses performances de convergence vers la valeur désirée sans être affecté par le bruit de mesure, et cela est assuré avec un effort minimal.

c. Réponse du système en présence des variations paramétriques par rapport aux valeurs nominales

Nous présentons les résultats de simulation pour une variation de 10% des paramètres du système. Nous constatons que, malgré les variations des paramètres du quadrirotor, les erreurs de poursuite s'annulent et la trajectoire désirée est atteinte rapidement comme la montre les Figures 3-8, 3-9 et 3-10. Nous déduisons que le régulateur reste toujours robuste aux variations paramétriques.



Figure 3- 4: Rejet d'une perturbation constante appliquée à la 25ème seconde



Figure 3- 5: Réponse système avec la commande forwarding en présence d'une perturbation aléatoire.


Figure 3- 6: Signaux de commande pour la perturbation aléatoire.

### **Remarque :**

Après la réponse transitoire, qui est de l'ordre de quelques secondes, l'erreur de poursuite est acceptable, ce qui permet une manipulation sûre même en présence de perturbations et de bruits de mesure. La réponse transitoire peut être encore réduite par un réglage fin des régulateurs.



Figure 3-7: Réponse du système avec la commande forwarding dans le cas de variations des paramètres.



Figure 3-8: Signaux de commande pour une variation paramétrique.



Figure 3-9: Erreurs poursuite par la commande forwarding dans le cas de variations des paramètres.

• Les entrées de commande du système sont représentées sur les Figures 3-5,3-7,3-9. Elles représentent les vitesses de rotation des moteurs. Il apparaît clairement que les signaux obtenus sont acceptables et physiquement réalisables. En conclusion, l'approche par la commande robuste par forwarding est très efficace et permet d'assurer la stabilité du système en boucle fermée.

### Deuxième simulation de vol :

Cette deuxième simulation de vol est effectuée pour valider les performances de l'approche forwarding à l'aide d'un autre type de trajectoire variant au cours du temps.

### Fonction de la trajectoire désirée :

Les équations de mouvement de la trajectoire désirée sont données comme suit :

$$x_d = 8\cos\left(\frac{t}{2}\right), y_d = 8\cos(\frac{t}{2}), z_d = 8\cos(\frac{t}{2})$$

#### Simulation

La Figure 3-10 présente le suivi de la trajectoire désirée effectuée par le quadrirotor dans le plan X Y Z. On remarque que le drone converge rapidement entre l'instant initial et l'instant final de la trajectoire.



Figure 3-10: Poursuite de trajectoires de référence sinusoïdales.

### 3.2 Synthèse des lois de commande PID\_Backstepping pour le quadrirotor

Après avoir réalisé la commande forwarding, nous passons à la deuxième commande qui est une commande combinée, sur les six sorties du système (x, y, z,  $\varphi$ ,  $\theta$ ,  $\Psi$ ). Cette méthode consiste à subdiviser le système en deux sous-systèmes ; le premier englobe la dynamique rapide (orientation) commandé par un PID tandis que le second concerne le sous-système à dynamique lente (position) contrôlé par la commande backstepping. Afin de relier ces deux dynamiques, on propose les contraintes non holonomes suivantes :

$$\begin{cases} \varphi_d = \arcsin(\frac{u_x \sin(\Psi) - u_y \cos(\Psi)}{u_1}) \\ \theta_d = \arcsin(\frac{u_x \cos(\Psi) + u_y \sin(\Psi)}{\cos(\varphi_d) u_1}) \end{cases}$$
(III.17)

Le schéma de la Figure 3-11 illustre la stratégie de commande adoptée.



Figure 3-11: Commande d'un quadri-rotor en attitude et en position.

### **3.2.1** Commande en position par la commande Backstepping

Le but est de trouver une loi de commande qui assure la poursuite d'une référence  $y_d$ , cela s'effectue en plusieurs étapes :

Dans notre cas, nous avons appliqué la commande Backstepping pour stabiliser la position du quadrirotor suivant les axes x et y, donc on l'applique aux deux sous-systèmes.

Nous avons utilisé la commande par Backstepping pour stabiliser la boucle externe et le PID pour stabiliser la boucle interne.

### **3.2.1.1** Etapes de la synthèse

Le premier sous-système est donné par les équations suivantes :

$$\begin{cases} \dot{x}_{1} = x_{4} & (III.18) \\ \dot{x}_{4} = C_{x7} S_{x8} \frac{U_{1}}{m} & \end{cases}$$

Considérons  $e_1$  l'erreur de suivi de trajectoire de l'axe x, et sa dérivée temporelle  $\dot{e}_1$ :

$$e_1 = \mathbf{x}_d - \mathbf{x}_1 \tag{III.19}$$

$$\dot{e}_1 = \dot{x}_d - \dot{x} \tag{III.20}$$

L'analyse de la stabilité est traitée par le théorème de Lyapunov en considérant une fonction V (e1) définie positive et sa dérivée temporelle semi-définie négative.

$$V_1 = \frac{1}{2} (x_d - x_1)^2$$
(III.21)

$$\dot{V}_1 = (x_d - x_1)(\dot{x}_d - \dot{x}_1)$$
 (III.22)

$$\dot{V}_1 = (x_d - x_1)(\dot{x}_d - x_4)$$
 (III.23)

Pour assurer la stabilité pour cette équation, prenons

$$x_4 = \Phi_4 = -\lambda (x_d - x_1) + \dot{x}_{1d}$$
(III.24)

La nouvelle référence désirée sera la variable de commande pour le sous-système précédant

$$x_4 = \Phi_4 \tag{III.25}$$

L'erreur de régulation est :

$$e_2 = \Phi_4 - x_4 \tag{III.26}$$

Sa dérivée est

$$\dot{e}_2 = \dot{\Phi}_4 - \dot{x}_4 \tag{III.27}$$

La fonction de Lyapunov étendue pour ce système est

$$V_4 = V_1 + \frac{1}{2}e_2^2 \tag{III.28}$$

$$V_4 = \frac{1}{2}(e_1^2 + e_2^2) \tag{III.29}$$

Dont la dérivée est :

(III.30)

$$\dot{V}_4 = \dot{V}_1 + e_2 \dot{e}_2$$

$$\dot{V}_4 = (\mathbf{x}_d - \mathbf{x}_1)(\dot{\mathbf{x}}_d - \dot{\mathbf{x}}_1) + (\Phi_4 - \mathbf{x}_4)(\dot{\Phi}_4 - \dot{\mathbf{x}}_4)$$
(III.31)

$$\dot{V}_4 = (x_d - x_1)(\dot{x}_d - \Phi_4 - x_4) + (\Phi_4 - x_4)(\dot{\Phi}_4 - C_{x7}S_{x8}\frac{U_1}{m})$$
(III.32)

$$\dot{V}_4 = (\mathbf{x}_d - \mathbf{x}_1)(\dot{\mathbf{x}}_d - \Phi_4) + (\Phi_4 - \mathbf{x}_4 - \dot{\Phi}_4 - C_{\mathbf{x}7} S_{\mathbf{x}8} \frac{U_1}{m})$$
(III.33)

$$C_{x7} S_{x8} \frac{U_1}{m} = -\lambda_4 (\Phi_4 - x_4) + (x_{1d} - x_1) + \dot{\Phi}_4 - x_1 + x_{1d}$$
(III.34)

$$\theta_d = x_8^* = \arcsin\left[\frac{[-\lambda_4(\Phi_4 - x_4) + (x_d - x_1) + \dot{\Phi}_4 - x_1 + x_{1d}]m}{C_{x7} U_1}\right]$$
(III.35)

De la même manière pour le sous système

$$\begin{cases} \dot{x}_2 = x_5 \\ \dot{x}_5 = -S_{x8} \frac{U_1}{m} \end{cases}$$
(III.36)

On aboutit à la commande qui stabilise  $x_5$  qui est

$$\Phi_5 = -\lambda_2 (\mathbf{x}_{2d} - \mathbf{x}_2) + \dot{\mathbf{x}}_2 \tag{III.37}$$

$$\Phi_d = x_7^* = \arcsin\left[\frac{-m\left[-\lambda_5(\Phi_5 - x_5) + \dot{\Phi}_5 - x_2 + x_{2d}\right]}{U_1}\right]$$
(III.38)

### 3.3 Commande en attitude par la technique PID

### 3.3.1 Expression du modèle linéarisé

Nous considérons la linéarisation des équations du modèle autour du point d'équilibre choisi.

L'architecture du PID en cascade a été étendue au cas non linéaire en séparant les dynamiques de translation et de rotation des équations du mouvement. Un contrôle linéaire est alors appliqué à la dynamique de rotation et un contrôle non linéaire a été appliqué à la dynamique de translation par la commande Backstepping.

### 3.3.2 Commande en attitude et altitude par PID

La commande linéaire par un PID est bien adaptée au vol quasi-stationnaire, pour lequel les angles d'inclinaison du véhicule sont faibles. Ce qui permet d'obtenir un modèle découplé en plusieurs chaînes SISO (Single Input Single Output) de la dynamique du drone.

Soit

$$\begin{cases} e_{\Phi} = \Phi_d - \Phi\\ U_2 = K_{p\Phi}e_{\Phi}(t) + K_{I\Phi}\int_0^t e_{\Phi}(\tau) d\tau + K_{D\Phi}\frac{\mathrm{d}e_{\Phi}(t)}{\mathrm{d}t} \end{cases}$$
(III.39)

De même pour l'angle de tangage, on a

$$\begin{cases} e_{\theta} = \theta_{d} - \theta \\ U_{3} = K_{p\theta}e_{\theta}(t) + K_{I\theta}\int_{0}^{t} e_{\theta}(\tau) d\tau + K_{D\theta}\frac{\mathrm{d}e_{\theta}(t)}{\mathrm{d}t} \end{cases}$$
(III.40)

Et pour l'angle de lacet

$$\begin{cases} e = \Psi_d - \Psi \\ U_4 = K_{p\Psi} e_{\Psi}(t) + K_{I\Psi} \int_0^t e_{\Psi}(\tau) \, d\tau + K_{D\Psi} \frac{de_{\Psi}(t)}{dt} \end{cases}$$
(III.41)

Et pour la commande en altitude

$$\begin{cases} e = z_d - z \\ U_1 = K_{pz} e_z(t) + K_{Iz} \int_0^t e_z(\tau) d\tau + K_{Dz} \frac{de_z(t)}{dt} \end{cases}$$
(III.42)

Les paramètres  $K_{px}$ ,  $K_{Ix}$  et  $K_{Dx}$ , définissent respectivement les gains proportionnel, intégral et dérivé des angles  $\Phi$ ,  $\theta$ ,  $\Psi$  et d'altitude Z.

### 3.3.3 Lois de commande par PID\_Backstepping

Les lois de commande PID sont données comme suit :

$$\begin{cases}
U_1 = K_{pz}e_z(t) + K_{Iz}\int_0^t e_z(\tau) d\tau + K_{Dz}\frac{de_z(t)}{dt} \\
U_2 = K_{p\Phi}e_{\Phi}(t) + K_{I\Phi}\int_0^t e_{\Phi}(\tau) d\tau + K_{D\Phi}\frac{de_{\Phi}(t)}{dt} \\
U_3 = K_{p\theta}e_{\theta}(t) + K_{I\theta}\int_0^t e_{\theta}(\tau) d\tau + K_{D\theta}\frac{de_{\theta}(t)}{dt} \\
U_4 = K_{p\Psi}e_{\Psi}(t) + K_{I\Psi}\int_0^t e_{\Psi}(\tau) d\tau + K_{D\Psi}\frac{de_{\Psi}(t)}{dt}
\end{cases}$$
(III.43)

$$\begin{cases} \theta_d = x_8^* = \arcsin\left[\frac{[-\lambda_4(\Phi_4 - x_4) + (x_d - x_1) + \dot{\Phi}_4 - x_1 + x_{1d} ]m}{C_{X7} * U_1}\right] \\ \Phi_d = x_7^* = \arcsin\left[\frac{-m \left[-\lambda_5(\Phi_5 - x_5) + \dot{\Phi}_5 - x_2 + x_{2d}\right]}{U_1}\right] \end{cases}$$
(III.44)

### 3.3.4 Résultats de simulation de la commande PID\_backstepping

Nous présentons ici les résultats de simulations menés sur le modèle dynamique complet du quadrirotor afin d'évaluer l'efficacité de la commande proposée. Les paramètres du Tableau 3.1 sont représentatifs de ce type de véhicule.

L'évaluation est réalisée pour voir la capacité de la commande à garantir une bonne exécution d'une manœuvre de poursuite de trajectoire. Les valeurs initiales sont définies comme suit :

 $\varepsilon(0) = [0; 0; 0]^{T}$  m et  $\eta(0) = [0; 0; 0]^{T}$ rad, les vitesses linéaires et angulaires initiales sont prises égales à zéro. Les paramètres du contrôleur sont donnés par le Tableau 3-2 :

Altitude	K <sub>Pz</sub> =1.3356	K <sub>Iz</sub> =2.1444	K <sub>Dz</sub> =2.9622
Roulis	$K_{P\Phi} = 0.6054$	К <sub>ІФ</sub> =0.2834	$K_{D\Phi} = 0.3175$
Tangage	Крө=0.6054	K <sub>10</sub> =0.2834	К <sub>D</sub> =0.3175
Lacet	К <sub>РΨ</sub> =0.1383	К <sub>ІΨ</sub> =0.0029	$K_{D\Psi} = 0.2660$

Tableau 3-2: Paramètres des régulateurs PID.

Plusieurs tests de vol ont été faits pour démontrer les performances de la commande proposée. Dans la première application, on ne prend pas en compte les perturbations externes. Dans le deuxième cas, avec les termes des perturbations externes considérées qui sont :

Une perturbation constante dans le temps appliquée au drone, elle est de l'ordre de 10% de la consigne, et une autre perturbation qui est un bruit aléatoire borné.

### 3.3.4.1 Cas a : Test de vol sans perturbations externes

Dans ce cas, on ne prend pas en compte les termes de perturbations externes.

La Figure 3-12 montre l'évolution du véhicule pendant le vol. Sur la base de ces résultats, on peut voir que, la commande PID\_Backstepping est capable de forcer les trajectoires du système à converger vers les trajectoires de référence.

### 3.3.4.2 Cas b : Test de vol en présence des perturbations externes

Dans ce cas, les perturbations externes sont prises en compte dans le modèle dynamique du quadrirotor. On a représenté sur la Figure 3-13 l'évolution de la position de translation et de lacet et des angles de roulis et tangage au cours du temps. Lorsque les termes de perturbations aérodynamiques sont introduits, le drone est capable de suivre les trajectoires de référence. Ceci démontre l'efficacité de la commande.

Les entrées de commande du système sont représentées sur les Figures 3-16. Nous constatons que les réponses obtenues sont acceptables et physiquement réalisables. En conclusion, la commande proposée est efficace et permet d'assurer la stabilité du système en boucle fermée.



Figure 3-12: Poursuite des trajectoires de références de type échelon



Figure 3- 13: Réponse du système dans le cas d'une commande PID \_Backstepping avec perturbations.



### 3.3.4.3 Variation paramétrique PID\_Backstepping





Figure 3- 15: Réponse système avec la commande PID\_backstepping en présence d'une perturbation aléatoire.

PID 1 z



# PID 2 roulis





# PID 3 tangage





# 3.4 Conclusion

Dans ce chapitre, nous avons appliqué la commande forwarding au contrôle de l'attitude et de la position du drone quadrirotor. Nous avons comparés les résultats obtenus par cette commande non linéaire à une commande hybride PID\_Backstepping. À travers ces simulations sous Simulink, nous pouvons dire que la précision est meilleure dans le cas de la technique de commande forwarding., et que la commande forwarding est plus plus robuste par rapport à la commande mixte.

# Chapitre 4

# Mise en œuvre expérimentale

### **4** Introduction

Dans le cadre de la mise en œuvre des résultats de simulation obtenus nous avons implémenté les approches de commande sur un drone quadrirotor DIY de la Figure 4-1 c'est une plateforme qui est mise à notre disposition par l'LISV de Versailles. L'utilisation de ce type de projet apporte une solution fonctionnelle avec la possibilité de réaliser toutes sortes de changements, tant matériels que logiciels. L'un des projets multirotors open-hardware et open-soft les plus célèbres est le projet de drone DIY [77]. Le matériel est développé et commercialisé par 3DR [78].



Figure 4-1: Quadrirotor DIY utilisé.



Figure 4-2: Banc d'essai Quad équipé du pilote automatique Pixhawk.

Le développement du logiciel suit l'évolution du matériel, et deux types de pilotes automatiques sont disponibles, en l'occurrence l'APM2.6 et le PixHawk, utilisables sur n'importe quelle plateforme. La solution la moins chère a été sélectionnée sur la base du DIY Quad et Pilote automatique APM2.6. Le coût de tout le matériel nécessaire ne dépasse pas 800 euros. La Figure 4-2 présente le drone DIY développé [43].

### 4.1 Présentation du banc d'essai

Les expériences ont été menées sur la base du projet de drone DIY open-hardware et opensoftware. Nous avons utilisé la plate-forme illustrée par la Figure 4-1. C'est un banc d'essai avec DIY Quad utilisé pour l'essai en intérieur, et particulièrement pour valider la commande PID développée pour le contrôle d'attitude. Dans cette expérience, du fait de la qualité des mesures, on ne considère que le mouvement de roulis, de tangage et de lacet. Le banc d'essai pour le contrôle d'attitude est composé d'un support et du DIY Quad. Le DIY Quad est un Quadcopter équipé de :

- 1. Support avec rotule rotative,
- 2. Corps quadrirotor,
- 3. Pilote automatique Pixhawk,
- 4. Quatre régulateurs de vitesse électroniques (ESC),
- 5. Quatre moteurs brushless avec hélices,
- 6. Récepteur Graupner RC,
- 7. Radios de télémétrie 3DR,
- 8. GPS u-blox avec boussole,

9. Tableau de distribution électrique et batterie LiPo,

10. Module d'alimentation APM pour la consommation de courant et les mesures de tension de batterie.



Figure 4- 3: Montage du pixhawk2.

De plus, l'émetteur RC est utilisé pour effectuer le contrôle manuel et la sélection du mode. La station sol open source (Q Ground Control ou Mission Planner) est utilisée pour visualiser les données de télémétrie et pour configurer les paramètres du mode de vol et pour étalonner les capteurs. Le montage du Pixhawk 2 avec les différents autres composants est donné par la Figure 4-3.

### 4.1.1 Contrôleur de vol pixhawk

Le pilote automatique Pixhawk est basé sur un cœur Cortex M4F STM32F427 32 bits, 168 Mhz/256 Ko de RAM/2 Mo Flash et un coprocesseur à sécurité intégrée STM32F103 32 bits. Le système embarqué est équipé d'un accéléromètre/gyroscope 6 degrés de liberté MPU-6000 et d'un accéléromètre/magnétomètre 3 axes ST Micro LSM303D. Le pixhawk est également équipé d'une carte microSD pour une journalisation à haut débit sur de longues périodes de temps. La fréquence de fonctionnement de la boucle principale du firmware et la fréquence d'acquisition des mesures de l'accéléromètre et des gyroscopes sont toutes deux de 400 Hz.

Ainsi, la stabilité sera meilleure et les mesures seront plus fiables que d'utiliser l'APM2.6 qui fonctionne à 100Hz.



Figure 4- 4: Entrées/Sorties d'un Pixhawk.

### 4.1.1.1 Rôle du contrôleur de vol pixhawk

C'est le cœur du drone, il permet entre autres de prélever les mesures nécessaires à partir des différents capteurs (accéléromètre, GPS) d'estimer les différents paramètres nécessaires au contrôle du drone et bien évidemment de contrôler les actionneurs du drone à savoir l'ensemble moteurs/hélices. Ce contrôleur est généralement sous forme d'un calculateur embarqué que nous pouvons configurer et programmer. Il assure plusieurs types de régulation principalement pour le contrôle de la position, de l'attitude et des vitesses des quatre moteurs.



Figure 4- 5: Schéma de principe du rôle de pixhawk.

### 4.1.2 Structure logicielle

Certaines plateformes expérimentales utilisent une interface graphique pour l'utilisateur, et d'autres sont basées sur la ligne de commande pour fournir plus de flexibilité. PX4 Autopilot est un système de pilotage automatique open source capable de contrôler les véhicules automatiques avec une variété de scenarios d'exploitation d'aéronefs [38, 39, 40, 78]. Pixhawk est la plateforme matérielle de référence pour PX4 sur le système d'exploitation temps réel Nutts [78].

#### 4.1.3 Les moteurs sans balais (brushless motors)

Dans le cadre de ce projet, les moteurs doivent être de petite masse et à la fois de très haute vitesse avec un rapport couple-masse élevé. Comme le montre la Figure 4-6, les moteurs brushless sont en fait un type de moteur synchrone à aimants permanents qui tournent avec un courant alternatif triphasé AC. Ils présentent de nombreux avantages par rapport aux moteurs DC classiques à savoir, leur longue durée de fonctionnement et leur insensibilité aux parasites en plus leur réponse dynamique élevée. Ce type de moteur permet de tourner et de générer de la poussée pour permettre au drone de voler. La force totale F délivrée par les quatre moteurs (F = F1 + F2 + F3 + F4) doit être supérieure au poids total du drone (avec tous les éléments embarqués) pour qu'il puisse voler. Dans le cas des multi rotor, les moteurs génèrent une poussée vers le haut.



Figure 4- 6: Conception d'un moteur brushless.

### 4.1.4 Variateurs de vitesse (ESC)

ESC signifie Electronique Speed Control, c'est un circuit électronique qui gère la vitesse des moteur brushless. Sur un multi-rotor, chaque moteur obtient son propre ESC, dont chacun se connecta au contrôleur de vol. Les ESC convertissent les séquences des signaux de commande PWM venant du contrôleur de vol ou de récepteur en un signal alternatif triphasé, et entraine le moteur brushless en fournissant le niveau d'énergie électrique approprié.



Figure 4-7: Variateur de vitesse 30A (ESC).

### 4.1.5 Batterie lipo

Les batteries utilisées sur un quadrotor sont essentiellement des « Lithium Polymère » (Li-Po). Elles sont issues d'une technologie qui permet d'avoir un très bon rapport poids/puissance. Un élément Li-Po (1S) fournit une tension de 3,7v. Sur un drone, on utilise en général des batteries à 3 ou 4 éléments (3S ou 4S). L'intensité est aussi un critère de choix. Un chargeur spécifique est à prévoir ainsi que des règles de sécurité très strictes car ces batteries peuvent exploser en cas de mauvaise manipulation.



Figure 4-8: Batterie lipo (lithium).

### 4.1.6 Hélices

Concernant les hélices, il faut les choisir en fonction de la taille des moteurs et du châssis du quad rotor utilisé. Les hélices utilisées sont DJI 8045 comme la montre la Figure 4-9. Les hélices sont fabriquées avec la fibre de carbone qui est résistante et légère, elles ont 8 pouces (20 cm) de diamètres, un pas de tangage de 4.5 pouces (un pas de tangage signifie combien de distance que le quadrotor va aller de l'avant pour une révolution à 360 degrés de l'hélice). On prendra deux de sens horaire et deux de sens anti horaire. Ces hélices seront fixées aux moteurs et leur rôle sera de brasser l'air pour générer la force de poussée. La taille de l'hélice varie avec la puissance du moteur. Pour bien comprendre, voici quelques règles à respecter pour le bon choix des différents accessoires : Plus une hélice est petite, plus elle a besoin de tourner vite pour être efficace, plus elle est grande, moins elle a besoin de tourner vite. Sur un gros modèle

d'avion par exemple, une petite hélice même tournant très vite (beaucoup de KV), il ne suffira pas à le faire voler correctement (pas assez puissante). Donc il faudra l'équiper d'une grande hélice qui aura besoin de tourner moins vite (moins de KV). Les grandes hélices consomment plus que les petites hélices, une grande hélice devra être montée sur un moteur puissant et vice versa. P(watt)=U(volt)I(A) :P(Watt) : la puissance max admise par le moteur brushless U(Volt) : la tension continue de la batterie LIPO, I(A) : le courant max qui doit être délivré par l'ESC au moteur brushless en général un moteur avec fort KV, prévu pour petite hélice qui consomme peu à peu de Watts, et inversement : moteur avec faible KV, prévu pour grande hélice qui va consommer beaucoup de Watts.



Figure 4-9: Hélices.

### 4.1.7 Radiocommande

La FlySKy FS-i6 est une radio 2.4Ghz 6 voies (ce qui est tout à fait suffisant pour la plupart des utilisations). Elle est équipée de 3 interrupteurs à 2 positions, 1 interrupteur à 3 positions et de 2 potentiomètres rotatifs. L'écran de la FS –i6 offre une résolution de 128\*64 pixels. 4 piles LR-6 suffisent à l'alimenter. Cette radio utilise l'AFHDS (Automatic Frequency Hopping Digital System) 2A, un nouveau protocole utilisant 2 antennes qui devrait offrir une liaison encore plus fiable que les précédentes versions. Sa configuration se fait d'abord via le menu qui s'affiche sur son écran, une fois allumée, ensuite le reste dans Mission Planner, notamment pour la radio-calibration.

Bande passante : 500kHz Puissance RF : moins de 20dBm Système 2.4ghz Type de code : GFSK Sensibilité : 1024 Avertissement de basse tension : moins de 4.2 V Poids du paquet : 676g Émetteur taille : 174 x 89 x 190 mm Plage de contrôle : 500m



Figure 4- 10: Radiocommande FlySky.

### 4.1.7.1 Récepteur

Le récepteur FlySky FS-iA6B dispose de six (6) voies et fonctionnera parfaitement avec les radiocommandes FlySky comme les FS-i4, FS-i6, FS-i10, etc. Il a une portée de plus de 500 mètres grâce à ses deux antennes, il reste très compact et ses connecteurs permettent une installation propre. Le FS-iA6B fonctionne en PWM, PPM et iBus.

Fréquence : 2.4055--2.475ghz

Numéro de la bande largeur : 140

Transmission de puissance :  $\leq 20$ dBm

RF sensibilité du récepteur : -105dBm

2. Mode 4G : la deuxième génération d'une version améliorée du système numérique fm automatique

Encodage : GFSK

Longueur de l'antenne : 26mm \* 2 (double antennes)

Puissance d'entrée : DC 4.0-6.5

Interface d'acquisition de données : oui

Type de modèle : avion / planeur / hélicoptère

Émetteur compatible avec fs-i4. fs-i6. fs-i10. fs-gt2e, fs-gt2g



Figure 4-11: Récepteur de la radiocommande.



Figure 4-12: Schéma de câblage du drone.



Figure 4-13: Liaison entre MATLAB/Simulink et le système à partir de PIXHAWK.

• Faire la compilation des algorithmes de commandes conçus dans Simulink tel que le PID, et les déployer sur le PIXHAWK Figure 4-15. Calibration des capteurs (gyroscope, accéléromètre et magnétomètre), radio et compas sont nécessaires, après compilation. Cette étape peut être effectuée en utilisant un programme appelé 'QGroundControl'. Le QGroundControl est une unité de contrôle de l'opérateur / logiciel pour les véhicules aériens. Il permet de calibrer, visualiser et de contrôler un véhicule ou un micro-véhicule aérien (MAV) pendant le développement et l'exploitation, tant à l'intérieur qu'à l'extérieur. Avec une architecture logicielle flexible, elle prend en charge plusieurs projets MAV / pilote automatique.



Figure 4-1: Compilation réussie du modèle Simulink

- Nous avons réussi à valider expérimentalement, sous un banc d'essai, un algorithme de commande de type PID pour la commande des orientations (roulis, tangage et lacet) du quadrirotor. Un banc d'essai adéquat a été développé pour les expériences préliminaires. Cela permet de verrouiller certains degrés de liberté afin de réduire la complexité de la commande.
- récupérer des données du système réel commandé vers MATLAB/Simulink.

### 4.2 Récupération des données du drone vers MATLAB/Simulink

La récupération des données permettra d'afficher et d'analyser les résultats validés en expérimental sous MATLAB.

### 4.2.1 Validation expérimentale de la commande PID en attitude

Dans cette expérience, le contrôleur PID est conçu dans Matlab/Simulink en utilisant le support PX4 de Embedded Coder. On peut ainsi générer le code C / C ++ à partir de Simulink et ensuite déployer le code dans le microcontrôleur PIXHAWK.

Le contrôleur PID est appliqué au drone avec les paramètres du tableau 3-2. Les paramètres de simulation sont utilisés dans ces expériences.

Les Figures 4-16, 4-17, 4-18, 4-19, 4-20 et 4-21 illustrent les résultats expérimentaux.



Figure 4- 14: Signal de commande de tangage.



Figure 4- 15: Signal de commande de roulis.



Figure 4- 16: Erreur de poursuite de tangage.





Figure 4-18: Erreur de poursuite en roulis.







Figure 4-20: Différence entre la vitesse mesurée et la vitesse désirée en lacet.

Les tests sont effectués afin de montrer l'efficacité du régulateur proposé. Un bon suivi est obtenu pour les angles de roulis et de tangage comme la montre les figures ci-dessus. L'erreur de suivi est relativement faible.

## 4.3 Conclusion

Le marché des véhicules aériens sans pilote connait un développement exponentiel. Surtout en raison de l'élargissement des domaines d'application civile et aux hauts potentiels de la robotique aérienne. Généralement, il est difficile d'utiliser un robot aérien commercial à des fins de recherche, en raison du fait que les chercheurs doivent développer leur propre contrôleur de vol pour qu'ils puissent accéder à toutes les informations des capteurs dont ils ont besoin et contrôlent, correctement, le robot.

Un tel problème est résolu par l'émergence de projets open source dans le domaine de la navigation des drones. L'un des projets open source les plus connus dans la communauté des robots aériens est le projet ArduCopter. Sponsorisé par 3DRobotics, le projet offre une solution open source complète pour contrôler différents types de robots aériens. Pour cette raison, nous avons utilisé ce projet comme point de départ pour faire de la recherche expérimentale. Le projet ArduCopter a été modifié et utilisé avec succès pour valider les résultats théoriques. En effet, le régulateur PID appliqué a été validé à l'aide de matériel DIY Quadcopter et Projet logiciel ArduCopter. Un banc d'essai, pour le contrôle d'attitude, a été assemblé à l'aide d'un support avec rotule rotative à des fins de sécurité. Les résultats expérimentaux obtenus montrent l'efficacité et les performances des solutions proposées.

# **Conclusion générale**

## **Conclusion générale**

Malgré le nombre considérable de solutions et en raison de leur importance, l'amélioration des performances de vol des drones quadrirotors et les problèmes de contrôle sont toujours d'actualité. Cette thèse a apporté quelques contributions à la commande non linéaire robuste des drones quadricopter.

Dans ce travail, nous avons développé une loi de commande de stabilisation de la trajectoire d'un quadrirotor synthétisée par la commande forwarding qui est basée, en premier lieu, sur le développement du modèle dynamique du quadrotor en prenant en considération les différentes forces et couples qui peuvent influencer l'évolution de ce drone. Cette loi de contrôle permet de réaliser le suivie des différentes trajectoires désirées exprimées en termes de coordonnées du centre de masse du système. En dépit de la complexité du modèle proposé, la commande forwarding a donné des résultats satisfaisants de suivi de trajectoires imposées tout en assurant un rejet des perturbations. Les simulations montrent de bonnes performances du contrôleur forwarding proposé.

Pour s'assurer de la robustesse de la commande forwarding, nous avons comparé ses performances avec une autre commande hybride qui est basée sur la commande PID\_Backstepping. L'effet des perturbations sur les réponses du système est étudié en effectuant plusieurs simulations. Les approches proposées ont prouvé leur robustesse et efficacité en simulation pour la stabilité de la position et de l'orientation du système.

Le quadrotor suit une trajectoire référencée de x, y, et z avec des petites variations des angles de tangage, roulis et lacet. Nous avons observé que la trajectoire désirée est atteinte après quelques secondes, en présence ou en absence de perturbations, ce qui confirme que les approches proposées sont robustes.

En conclusion, la commande forwarding a une forte capacité de compensation des perturbations.

L'implémentation, en temps réel, de la commande PID en orientation du quadrirotor sur le banc d'essai du LISV (Laboratoire d'Ingénierie des Systèmes de Versailles), en utilisant un contrôleur de vol PIXHAWK et sa plateforme logicielle PX4, et confirmer les résultats de simulation.

Pour les travaux futurs, nous envisageons :
- Implémenter, en temps réel, la commande forwarding et la commande combinée PID\_Backstepping développées et simulées dans le chapitre 3, qui n'ont pas été effectuée à cause de la courte durée du stage.
- Simuler les commandes proposées dans cette thèse en accrochant une charge au centre de masse du drone.
- Introduire les perturbations externes avec des paramètres variables et rapprocher le modèle dynamique d'un véritable système, en introduisant des rafales de vent, et de valider expérimentalement les résultats.

## **Références bibliographiques**

- [1] ALBERTOS, Pedro; BLANKE, Mogen; ISIDORI, Alberto; SCHAUFELBERGER, Walther; SANZ, Ricardo. An introduction to forwarding. [En ligne]. Published as Chapter 4 in: Control of complex systems, K. ASTROM. Editors, Springer 2001 L. Praly CAS \_Ecole des Mines de Paris Octobre 29, 1999. [Consulté le 22juin 2022]. Disponible sur : https://books.google.dz/books?hl=fr&lr=&id=cublBwAAQBAJ&oi=fnd&pg=PA2&dq=A LBERTOS,+Pedro%3B+BLANKE,+Mogen%3B+ISIDORI,+Alberto%3B+SCHAUFEL BERGER,+Walther%3B+SANZ,+Ricardo.+An+introduction+to+forwarding.+published+ as+Chapter+4
- [2] AMEHO, Yann. Du pilotage d'une famille de drones à celui d'un drone hybride via la commande adaptative. [En ligne]. Thèse de doctorat : Automatique. Université de Toulouse : Institut Supérieur de l'Aéronautique et de l'Espace, 2013, 144 p. [consulté le 2 juillet 2022]. Disponible à l'adresse : https://tel.archives-ouvertes.fr/tel-01020984/document
- [3] ANTONELLI, Gianluca ; ARRICHIELLO, Filippo. Adaptive trajectory tracking for quadrotor MAVs in presence of parameter uncertainties and external disturbances. IEEE/ASME International conference on advanced intelligent mechatronics[en ligne]. 2013, pp.1337-1342. [Consulté le 22juin 2022]. Doi : 10.1109/TCST.2017.2650679
- [4] ASHTON, Roza; MAGGIORE, Manfredi. A class of position controllers for underactuated VTOL vehicles. IEEE Transactions on Automatic Control [En ligne]. 2014, vol.59, n°9, pp.2580-2585. [Consulté le 2 décembre 2022]. DOI : 10.1109/TAC.2014.2308609.
- [5] BARKANA, Itzhak. Comments on design of strictly positive real systems using constant output feedback. In IEEE Transactions on Automatic Control [en ligne].2004, vol. 49, n°11, pp.2091-2093. doi: 10.1109/TAC.2004.837565. Disponible sur : https://ieeexplore.ieee.org/abstract/document/1356136.
- [6] BARKANA, Itzhak; TEIXEIRA, Marcelo. Carvelho. Minhoto; HSU, Liu. Mitigation of symmetry condition in positive realness for adaptive control. Automatica. 2006, vol 42, pp. 1611-1616. [Consulté le 22juin 2022]. Disponible sur : https://www.sciencedirect.com/science/article/abs/pii/S0005109806002184
- [7] BASRI, Mohd; HUSAIN, Abdul-Rashid. A hybrid optimal backstepping and adaptive fuzzy control for autonomous quadrotor helicopter with time-varying disturbance. J Aerospace Engineering. [en ligne]. 2015, vol. 229, n°12, pp.2178-2195. [Consulté le 22juin 2022]. Disponible sur :https://doi.org/10.1177/0954410015569583
- [8] BENALLEGUE, Abdelaziz; FRIDMAN, L; MOKHTARI, Rida. High-order sliding mode observer for a quadrotor uav. Int.J. Robust nonlinear control [En ligne].2008, vol.18, n°4, pp. 427-440[Consulté le 23 juin 2022]. Disponible sur : http:// doi.org/10.1002/rnc.1225.
- [9] BÉRARD, C ; BIANNIC, J-M ; et SAUSSIÉ, D. La commande multi variable : Application

au pilotage d'un avion. Dunod. 2012

- [10]BOLANDI, Hamed; REZAEI, Mohammad; ZADEH, Smail. Attitude control of a quadrotor with optimized pid controller. Intelligent Control and Automation, [en ligne]. 2013, vol.04, n° 3, pp. 342-349[consulté le 22juin 2022]. DOI: 10.4236/ica.2013.43040
- [11]BOUABDALLAH, Samir; SIEGWARE, Roland. PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor. Proceedings of IEEE/RSJ international conference on intelligent Robots and Systems [en ligne]. 2004, pp.2451-2456. [Consulté le 30 juin 2022] DOI: 10.1109/IROS.2004.1389776
- [12]BOUABDALLAH, Samir ; SIEGWART, Roland. Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain [En

ligne].pp. 2247-2252, doi: 10.1109/ROBOT.2005.1570447. disponible sur: https://ieeexplore.ieee.org/abstract/document/1570447

[13] BOUCHOUCHA, Mouloud; TADJINE, Mohamed; TAYEBI, Abdelhamid; BOUABDALLAH, Samir. Robust Nonlinear PI for Attitude Stabilisation of a Four-Rotor Mini-Aircraft from Theory to Experiment. Archives of control sciences [en ligne]. 2008, n°1. pp.99-120. [Consulté le 25juin 2022]. Disponible vol. 18 sur https://www.infona.pl/resource/bwmeta1.element.baztech-article-BSW3-0045-0007. [14] CHAUPRADE, Lionel. Les Drones Aériens. Cépaduès Editions. 2014.152p. [Consulté le

[14] CHAUPRADE, Lionel. Les Drones Aeriens. Cepadues Editions. 2014.152p. [Consulte le 25juin 2022]. Disponible sur : https://www.cairn.info/revue-defense-nationale-2015-4-page-129.htm

- [15] CHEN, Fang; JIANG, Bin. Robust Backstepping Sliding-Mode Control and Observer-Based Fault Estimation for a Quadrotor UAV. IEEE Transactions on Industrial Electronics.
  [en ligne].2016, vol. 63, n°8, pp.5044-5056. [Consulté le 22juin 2022]. DOI: 10.1109/TIE.2016.2552151
- [16]CHEN, Yanmin; HE, Yongling; ZHOU, Minfeng. Modeling and control of a quadrotor helicopter system under impact of wind field. Res J Appl Sci. Eng Technol[en ligne]. 2013, vol. 6, n°17, pp. 3214-3221. [Consulté le 28juin 2022]. ISSN: 2040-7459; e-ISSN: 2040-7467
- [17] DONG, W; GU, G; ZHU, X; DING, H. High-performance trajectory tracking control of a quadrotor with disturbance observer. Sensors Actuators A Physical[en ligne]. 2014, vol. 211, n°5, pp.67-77. [Consulté le 20mai 2022]. Disponible sur : https://doi.org/10.1016/j.sna.2014.03.011.
- [18]DONG, Z; FAN, H; WANG, Y; XU, L. Adaptive Backstepping Controller Design for Quadrotor Aircraft with Unknown Disturbance.14th International Conference on Control, Automation, Robotics & Vision Phuke [en ligne]. 2016. [consulté le 20mai 2022]. Disponible sur : DOI : 10.1109/ICARCV.2016.7838694.
- [19]DUC, M-N; TRONG,T.N; XUANG, S. The quadrotor MAV system using PID control. Mechatronics and Automation (ICMA), IEEE International Conference [en ligne]. 2015, pp. 506-510. [Consulté le 20juin 2022]. DOI : 10.1109/ICMA.2015.7237537.
- [20] FEYEL, Philippe. La commande robuste par loop-shaping. Herms Lavoisier 2013. 280p .ISBN :2746245280.
- [21] FLEISCHMANN, Sébastien ; THEODOULIS, Spilios; Édouard, LAROCHE, et al. Controller d
- esign point selection for linearized gain scheduling. In American Control Conference (ACC). 2017, pp.1574-1579. [Consulté le 20juin 2022]. doi: 10.23919/ACC.2017.7963177.
- [22]GARCIA, L. R; Carrillo, A. E; DZUL, López; LOZANO,R. Modeling the quadrotor minirotorcraft en Quad Rotorcraft Control. Springer [en ligne]. 2013. [Consulté le 20mai 2022]. Disponible sur : https://doi.org/10.1007/978-1-4471-4399-4\_2
- [23]GONZALEZ-Hernandez, I; PALACIOS,F; CRUZ, S. Real-time altitude control for a quadrotor helicopter using a super-twisting controller based on high-order sliding mode observer. International Journal of Advanced Robotic System [en ligne]. 2017, vol.14, n°1, pp. 2450-2455. [Consulté le 20mai 2018]. Disponible sur https://doi.org/10.1177/1729881416687113
- [24] HAMEL, Tarek; MAHONY, Robert. Image based visual servo control for a class of aerial robotic systems Automatica [en ligne]. 2007, vol 43 pp. 1975-1983.Disponible sur: https://doi.org/10.1016/j.automatica.2007.03.030.

[25] HASSENI, Seif-el-islam ; ABDOUL, L ; GLIDA, Hossam-Eddine. Parameters tuning of a quadrotor PID controllers by using nature-inspired algorithms, Evolutionary Intelligence [En ligne].2021, vol.14, pp. 61-73. [Consulté le 20mai 2022]. Disponible sur : https://doi.org/10.1007/s12065-019-00312-8.

[26] HÉRISSÉ, Bruno. Asservissement et Navigation Autonome d'un drone en environnement incertain par flot optique. Thèse de doctorat, Université Nice Sophia Antipolis Français [en ligne]. [Consulté le 30 juin 2022]. Disponible sur : https://theses.hal.science/tel-00558203/.

 [27] HUANG, HM ; PAVEK, K ; NOVAK, B et al. A Framework for Autonomy Levels for Unmanned Systems (ALFUS) [En ligne]. 2005, pp. 849-863. [Consulté le 30 juin 2022]. Disponible sur : https://d1wqtxts1xzle7.cloudfront.net/41420818/A\_framework\_for\_autonomy\_levels\_for \_unma20160122-29920-mjbf2h-libre.pdf?1453467585=&response-content-

 $disposition = inline\%\,3B + filename\%\,3DA\_framework\_for\_autonomy\_levels\_for\_unma.pdf$ 

- [28] IDRES, Mohamed. Quadrotor trajectory tracking using PID cascade control. IOP Conf. Series: Materials Science and Engineering. [En ligne].2017, 270. [Consulté le 20mai 2022]. DOI 10.1088/1757-899X/270/1/012010
- [29]ISLAM, Mohamed. Trajectory tracking in quadrotor platform by using PD controller and LQR control approach. 6th International Conference on Mechatronics. ICOM'17 IOP Publishing IOP Conf. Series: Materials Science and Engineering [En ligne]. 2017, 260. [Consulté le 20mai 2022]. DOI 10.1088/1757-899X/260/1/012026
- [30] KAHOUADJI, Mouad; MOKHTARI, Rida; CHOUKCHOU-BRAHAM, Amal; CHERKI, Brahim. Real-Time Attitude Control of 3 DOF Quadrotor UAV using Modified Super Twisting Control of 3 DOF Quadrotor UAV using Modified Super Twisting Algorithm. Journal of the Franklin Institute 2019, https://doi.org/10.1016/j.jfranklin.2019.11.038.
- [31] KENDOUL. Farid. Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. Journal of Field Robotics [En ligne]. 2012, vol. 29, n°2. pp.315-378. [Consulté le 24 mai 2022]. Disponible sur : https://doi.org/10.1002/rob.20414.
- [32] KHALIL, H.K. Nonlinear Systems. 2éme édition. Prentice-Hall. Upper Saddle River, NJ, USA, 1996.
- [33] KHATOON, Shahida; SHAHID, Mohammad; CHADHARY, Himanshu. Dynamic modeling and stabilization of quadrotor using PID controller. Advances in Computing, Communications and Informatics ICACCI, 2014 International Conference IEEE [En ligne]. 2014, pp.746-750. [Consulté le 20mai 2022. Disponible sur : doi: 10.1109/ICACCI.2014.6968383. Disponible sur : https://ieeexplore.ieee.org/abstract/document/6968383.
- [34] KHEBBACHE, Hicham. Tolérance aux défauts via la méthode backstepping des systèmes non linéaires - Application : Système UAV de type Quadrirotor. Phd, UNIVERSITE FERHAT ABBAS DE SETIF, 2012. Disponible sur : http://dspace.univsetif.dz:8888/jspui/handle/123456789/1300.
- [35]KOSZEWNIK, A. The parrot UAV controlled by PID controllers. acta mechanica et automatic [En ligne]. 2014, vol.8, n°2, [Consulté le 24 mai 2022]. Disponible sur : https://sciendo.com/downloadpdf/journals/ama/8/2/article-p65.xml.
- [36] LIM, H; PARK, J ; LEE, D et al. Construisez votre propre quadrirotor : projets open source sur des véhicules aériens sans pilote, dans IEEE Robotics & Automation Magazine [En ligne].2012, vol. 19, n°3, pp. 33-45. doi : 10.1109/MRA.2012.2205629.

- [37] MATOUK, D; GHEROUAT, D. Quadrotor Position and Attitude Control via Backstepping Approach. 8th International Conference on Modelling, Identification and Control [En ligne]. 2016. [Consulté le 20mai 2022]. DOI: 10.1109/ICMIC.2016.7804228.
- [38] MEIER, L. Mavlink Micro air vehicle communication protocol.2013. [Consulté le 20 juin 2022]. Disponible sur : http://qgroundcontrol.org/mavlink/start
- [39] MEIER, Lorenz ; HONEGGER, Dominik; POLLEFEYS, Marc. PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms. IEEE International Conference on Robotics and Automation (ICRA)[En ligne]. 2015, pp. 6235-6240. Disponible sur : https://ieeexplore.ieee.org/abstract/document/7140074.
- [40] MEIER, Lorenz. COMMUNITY PX4. The professional autopilot.2017. [Consulté le 20 juin 2022. Disponible sur : http://px4.io/
- [41] METTLER, Bernard; TISCHLER, Mark; KANADE, Takeo. System identification modeling of a small scale unmanned rotorcraft for flight control design. Journal of the American Helicopter Society [en ligne]. 2002, vol. 47, n°1, pp.50-63. [Consulté le 20mai 2022]. Disponible sur : https://doi.org/10.4050/JAHS.47.50.
- [42] MOHANTA, JC ; PARHI, D. Un schéma de contrôle pour la navigation et l'évitement d'obstacles d'un agent de vol autonome. Arab J Sci Eng 43 [en ligne].2018, pp. 1395-1407. [Consulté le 20mai 2022]. Disponible sur : https://doi.org/10.1007/s13369-017-2899-y
- [43] MOKHTARI, Khalil; ELHADRI, Abdelhafid; ABDELAZIZ, Mourad. A passivity based simple adaptive synergetic control for a class of nonlinear system. Int J Adapt Control Signal Process [en ligne]. 2019, pp.1-15. Disponible sur: https://doi.org/10.1002/acs.3035.
- [44]MOKHTARI, Rida; CHERKI, Brahim. Robust control for attitude tracking problem for a quadrotor unmanned aerial vehicle. 3rd International Conference on Systems and Control, Algiers, Algeria [en ligne]. 2013, pp.862-867. [Consulté le 20mai 2022]. Disponible sur: https://ieeexplore.ieee.org/abstract/document/6750958.
- [45] MORELLI, Eugene. Aircraft system identification: theory and practice. American Institute of Aeronautics and Astronautics Reston [en ligne]. 2006, Va, USA. [Consulté le 25 juin 2022].Disponible

http://www.nianet.org/wp.Content/uploads/2016/05/Aircraft\_Systems\_Identification\_Shor t\_Course\_Flyer1.pdf.

- [46] ORTIZ, J.P; L. I. MINCHALA; REINOSO, M.J. Nonlinear robust h-infnity PID controller for the multivariable system quadrotor. IEEE Latin America Transactions. [en ligne] 2016,14(3), 1176-1183. [Consulté le 20mai 2018].DOI: 10.1109/TLA.2016.7459596
- [47] ORTIZ, A; SANTOS, N; LAROCHE, E; KIEFER, R. Controller tuning strategy for quadrotor mav carrying a cable-suspended load. In International Micro Air Vehicle Conference and Flight Competition IMAV.2017.
- [48] OUENDI, Fatima; HOUTI, Smail; TADJINE, Mohamed. Comparison of Quadrotor Performance Using Forwarding and PID-Backstepping. Control Nonlinear Dynamics and Systems Theory [en ligne]. 2019, vol.19, n°3, pp. 427-443. [Consulté le 5juillet 2022]. Disponible sur : https://e-ndst.kiev.ua/v19n3/V19N3.pdf.
- [49] PAREDES, J.A; JACINTO, C; RAMIREZ,R; Vargas,I.Contrôleur PD flou simplifié pour le mélange de comportement et l'amélioration des performances dans les systèmes de contrôle d'attitude quadcopter IEEE ANDESCON, Arequipa, Pérou. [en ligne].2016, pp.1-4.[consulté le 20mai 2022].disponible sur :doi : 10.1109/ANDESCON.2016.7836217.
- [50] PULVER, Aaron, WEI, R; MANN, C. Locating AED enabled medical drones to enhance cardiac arrest response times. Prehospital Emerg Care [en ligne]. 2016, vol.20, n°3, pp.378-389. [Consulté le 20mai 2022]. Disponible sur: https://doi.org/10.3109/10903127.2015.1115932.
- [51] RASHAD, R; ABOUDOUNIA, A ; EL-BADAWY, A. Backstepping trajectory tracking control of a quadrirotor with disruption removal. XXV Conférence

internationale sur les technologies de l'information, de la communication et de l'automatisation (ICAT).Sarajevo, Bosnie-Herzégovine [en ligne]. 2015, pp.1-7. [Consulté le 20mai 2022]. Disponible sur: http:// doi.org/10.1109/ICAT.2015.7340523.

[52]ROUSSEL, Emmanuel ; GASSMANN, Vincent ; LAROCHE, Édouard. Accuracysimplicity trade-off for miniature helicopter models: A comparative study based on flight data. Control Engineering Practice (CEP) [en ligne].2015. [Consulté le 20mai 2022] disponible sur : https://doi.org/10.1016/j.conengprac.2018.01.001.

[53] RESPONDEK, W; TALL, IA. Strict feedforward form and symmetries of nonlinear control systems, 2004 43rd IEEE Conference on Decision and Control CDC IEEE Cat. No.04CH37601, Nassau, Bahamas [En ligne]. 2004, Vol.2, pp.1611-1616. [Consulté le 20mai 2022]. doi : 10.1109/CDC.2004.1430275.

- [54] SEPULCHRE, R ; JANKOVIC, M ; KOKOTOVIC.PV. Constructive Nonlinear Control. Springer-Verlag, London, 1997. [Consulté le 20mai 2022].
- [55] SEPULCHRE.R ; JANKOVIC. M ; KOKOTOVIC, PV. Systèmes entrelacés et conceptions récursives pour la stabilisation globale, 1997 Conférence européenne de contrôle (ECC), Bruxelles, Belgique 1997, pp. 3857-3861, [consulté le 20mai 2018]. doi : 10.23919/ECC.1997.7082719.
- [56] SHEN,S; MICHAEL,N et KUMAR, V Autonomous multi-floor indoor navigation with a computationally limited MAV, 2011 IEEE International Conference on Robotics and Automation, Shanghai, Chine. 2011, pp.20-25. doi: 10.1109 /ICRA.2011.5980357.
- [57] SHI, X ; HU, C. YIN ; CHENG, Y et HUANG, X. Conception d'un contrôleur de suivi de trajectoire avec méthode de recul pour les véhicules aériens sans pilote quadrotor, 2018 Chinese Control And Decision Conference (CCDC), Shenyang, Chine, 2018, pp. 3124-3128, [consulté le 20mai 2018]. doi : 10.1109/CCDC.2018.8407661.
- [58]SKOGESTAD, Sigurd; POSTLETHWAITE, Ian. Multivariable Feedback Control Analysis and Design. https://dl.acm.org/doi/abs/10.5555/525126.
- [59] SLOTINE, Jean-Jacques. Applied nonlinear control. Prentice Hall, Englewood Cliffs, 1991Disponi sur :

https://d1wqtxts1xzle7.cloudfront.net/33582713/AppliedNonlinear\_Control\_Slotine-libre.pdf.

- [60] W. Su et M. Fu. Contrôle non linéaire robuste : au-delà du retour en arrière et du transfert non linéaire, Actes de la 38e conférence IEEE sur la décision et le contrôle (Cat. No.99CH36304), Phoenix, AZ, États-Unis, 1999vol.1, pp. 831- 836 [consulté le 20mai 2022]. doi: 10.1109/CDC.1999.832893.
- [61] TAN, W; LIU, J; TAM, P.K.S. PID tuning based on loop-shaping H control. IEE Proc-Control Theory. 1998, vol. 145, n°6, pp. 831- 836

[62]TAYEBI. Abdelhamid, MCGILVRAY, S. Attitude stabilization of a vtol quadrotor aircraft. IEEE Transactions on Control Systems Technology [en ligne]. 2006, vol. 14, n°3, pp. 562-571. [Consulté le 26juin 2022]. doi: 10.1109/TCST.2006.872519.

- [63] TAYEBI, Abdelhamid; ROBERTS, Andrew; BENALLEGUE, Abdelaziz. Inertial vector measurements based velocity free attitude stabilization. IEEE Transactions on Automatic Control [En ligne]. 2013, vol. 58, n°11, pp. 2893-2898. [Consulté le 30mai 2022]. doi: 10.1109/TAC.2013.2256689.
- [64] WANG, L; JIA, H. The trajectory-tracking problem of quadrotor UAV: Global stability analysis and control design based on the cascade theory. Asian Journal of Control [En ligne]. 2014, vol.16, n°2, pp.574-588. [Consulté le 30 mai 2022]. Disponible sur : https://doi.org/10.1002/asjc.746.

- [65] WANG, P ; MAN, Z ; CAO, Z ; ZHENG, J. Modélisation dynamique et contrôle linéaire du quadricoptère, Conférence internationale 2016 sur les systèmes mécatroniques avancés (ICAMechS), Melbourne, VIC, Australie [En ligne]. 2016, pp. 498-503. [Consulté le 22 juin2022] doi : 10.1109/ICAMechS.2016.7813499.
- [66] WANG, X ; LIN,X ; YU,Y; WANG, Q. Contrôle en arrière pour quadrirotor avec modèle de poussée basé sur le réseau neuronal BP, Conférence annuelle académique des jeunes de l'Association chinoise de l'automatisation (YAC), Hefei, Chine[En ligne]. 2017, pp. 292-297. Consulté le 20juin2022]. doi: 10.1109/YAC.2017.7967422.
- [67] XIONG, Jing-Jing; ZHENG, En-Hui. Position and attitude tracking control for a quadrotor UAV. ISA transactions[En ligne]. 2014, vol. 53, n° 3, pp. 725-731. [Consulté le 22juin 2022] Disponible sur https://doi.org/10.1016/j.isatra.2014.01.004
- [68] YIN,C; HUANG,C; Chen,Y; Dadras,S; ZHONG,S.M; Y,CHENG. Fractional-order exponential switching technique to enhance sliding mode control. Applied Mathematical Modelling.2017, vol. 44, pp.705-726. https://doi.org/10.1016/j.apm.2017.02.034.
- [69] ZEMALACHE MEGUENNI, Kadda. Commande d'un système sous-actionné : Application à un drone à Quatre Hélices [En ligne]. Thèse de doctorat : Robotique. Université d'Evry Val d'Essonne, 2006, 119 p. Disponible sur : https://www.theses.fr/2006EVRY0039.
- [70] ZHAO, B; XIAN, B; ZHANG, Y. Contrôle de suivi adaptatif robuste non linéaire d'un drone quadrirotor via la méthodologie d'immersion et d'invariance, dans IEEE Transactions on Industrial Electronics[En ligne].2014, vol. 62, n°5, pp. 2891-2902. [Consulté le 6juillet2022]. doi : 10.1109/TIE.2014.2364982.
- [71]ZHOU, Kemin. Essentials of robust control. Prentice hall Upper Saddle River, NJ [En ligne]. 1998, vol. 104. [Consulté le 22juin 2022]. Disponible sur : https://ostad.nit.ac.ir/payaidea/ospic/file1503.pdf.
- [72]ZHANG, N; GAI, W; ZHONG, M. Aerospace Science and Technology 86, 204 2019. DOI 10.1016/j.ast. 2019.01.021.
- [73] ZHANG, C ; ZHOU, X ; ZHAO, H ; DAI, A. Contrôle flou tridimensionnel du suivi de la trajectoire d'un mini quadrirotor UAV sous l'impact de la perturbation du vent, Conférence internationale 2016 sur les systèmes mécatroniques avancés (ICAMechS), Melbourne, VIC, Australie[En ligne]. 2016, pp. 372-377. [Consulté le 6 juillet 2022]. doi : 10.1109/ICAMechS.2016.7813477.
- [74]ZHANG, X; HUANG, X; LU, H. Contrôle de suivi de trajectoire basé sur le transfert pour les systèmes non linéaires avec des perturbations inconnues bornées. Int. J. Contrôle Autom. Syst[En ligne]. 2016, vol.14, pp.1231-1243. [Consulté le 6 juillet 2022]. Disponible sur : https://doi.org/10.1007/s12555-015-0083-2.
- [75] ZHANG, X, HUANG, X ; LU, H. Contrôle d'immersion et d'invariance basé sur le transfert pour les systèmes non linéaires à rétroaction stricte dimensionnels non linéaire, 2016, vol.83, pp. 483-496. [Consulté le 6 juillet 2022]. Disponible sur: https://doi.org/10.1007/s11071-015-2342-4.
- [76] ZHANG, X; HUANG, L; LU, H. Q. Mapping filtered forwarding-based trajectory tracking control. Journal of the Franklin Institute[En ligne].2015, vol.352, n°1, pp.5735-5757. [Consulté le 6 juillet 2022]. Disponible sur: https://doi.org/10.1016/j.jfranklin.2015.10.004.
- [77]<u>https://docs.px4.io/en/flight\_controller/mro\_pixhawk.html</u> [En ligne]. [Consulté le 16juin 2022].
- [78] jmavsim simulation. [En ligne]. [Consulté le 6 juillet 2022]. Diponible sur https://dev.px4.io/v1.9.0/en/simulation/jmavsim.html