

الجمهورية الجزائرية الديمقراطية الشعبية  
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
جامعة أبي بكر بلقايد- تلمسان  
Université Aboubakr Belkaïd-Tlemcen  
كلية التكنولوجيا  
Faculté de Technologie

Département de Génie Electrique et Electronique



## MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

**En** : Automatique

**Spécialité** : Automatique et Informatique Industrielle

**Par** : Kouloughli Abdelkarim et Smahi Benali

### Sujet

Commande floue avec optimisation PSO d'un véhicule volant type quadrirotor.

Soutenu publiquement, le 12/06/2024, devant le jury composé de :

Mme Kahouadji Salima	MCA	Université de Tlemcen	Président
Mme Choukchou-Braham Amal	Professeur	Université de Tlemcen	Examineur
Mr. Benariba Hassan	MCA	Université de Tlemcen	Encadreur

# Dédicace

À nos précieux parents, pour leurs encouragements et leur motivation tout au long de nos parcours éducatifs. Nous souhaitons également exprimer notre gratitude envers nos frères et sœurs pour leur amour inconditionnel et leur soutien constant qui ont accompagné cette aventure académique.

# Remerciement

Nous souhaitons exprimer notre profonde gratitude à Dieu le Tout-Puissant pour ses bienfaits inestimables, qui nous ont permis de mener à bien nos études et de réaliser ce modeste travail dans des conditions propices. Nous tenons également à remercier chaleureusement notre encadrant, M. Benariba Hassane, pour ses conseils éclairés et sa bienveillance tout au long de notre parcours. Sa rigueur scientifique et son professionnalisme ont été des atouts précieux qui nous ont permis de progresser dans notre réflexion et de mener à bien ce projet. Enfin, nous exprimons nos sincères remerciements à Mme Choukchou-Braham Amal et à Mme Kahouadji Khadidja pour l'honneur qu'elles nous font en acceptant de faire partie du jury et de juger ce travail.

# Table des matières

Table des figures	i
Liste des tableaux	iv
Introduction Générale	1
<b>1 Présentation et modélisation d'un quadrirotor</b>	<b>3</b>
1.1 Introduction	3
1.2 Historique	3
1.3 Classification	6
1.3.1 Classification selon la taille	6
1.3.2 Classification selon le mode de propulsion	6
1.4 Modélisation du quadrirotor	11
1.4.1 Mouvements du quadrirotor	11
1.4.2 Modèle dynamique du quadrirotor	14
1.4.3 Développement du Modèle mathématique selon Newton-Euler	20
1.4.4 Représentation d'état du système	24
1.5 Dynamique des moteurs	25
1.6 Commande en boucle ouverte	25
1.6.1 Définition	25
1.6.2 Principe de fonctionnement	25
1.6.3 Simulation en boucle ouverte	25
1.7 Conclusion	27
<b>2 Commande linéaire et optimisation du PID</b>	<b>28</b>
2.1 Introduction	28
2.2 Approche Linéaire	28
2.3 Régulateur PID	29
2.3.1 Structure d'un régulateur PID	30

2.3.2	Choix des paramètres du régulateur . . . . .	31
2.3.3	Schéma de régulation . . . . .	31
2.4	Optimisation de la commande linéaire par PSO . . . . .	33
2.4.1	Introduction . . . . .	33
2.4.2	Problème d'optimisation . . . . .	33
2.4.3	Fonction objectif . . . . .	33
2.4.4	Principe de fonctionnement de PSO . . . . .	35
2.4.5	Mécanisme de confinement . . . . .	37
2.4.6	Optimisation des paramètres de régulateur PID par PSO . . . . .	38
2.4.7	Implémentation de l'algorithme dans le système . . . . .	38
2.5	Résultats de simulations . . . . .	39
2.5.1	Test de poursuite de trajectoire . . . . .	39
2.5.2	Tests de robustesse . . . . .	43
2.6	Conclusion . . . . .	49
<b>3</b>	<b>Commande par logique floue</b>	<b>50</b>
3.1	Introduction . . . . .	50
3.2	Historique . . . . .	50
3.3	Définition de la logique floue . . . . .	50
3.4	Principe de la logique floue . . . . .	51
3.4.1	Ensemble flou . . . . .	51
3.4.2	Variable linguistique . . . . .	51
3.4.3	Fonction d'appartenance . . . . .	52
3.4.4	Opérateurs de la logique floue . . . . .	52
3.4.5	Règles d'inférence floue . . . . .	53
3.4.6	Structure d'un régulateur flou . . . . .	53
3.5	Application de la commande floue sur le quadrirotor . . . . .	55
3.6	Résultats de simulation . . . . .	58
3.6.1	Test de poursuite de trajectoire . . . . .	58
3.6.2	Tests de robustesse . . . . .	62
3.7	Commande PID floue adaptatif . . . . .	65
3.7.1	Application de la commande PID floue adaptatif sur le quadrirotor	65
3.7.2	Tests de robustesse . . . . .	69
3.8	Etude comparative entre les commandes synthétisées . . . . .	75
3.9	Conclusion . . . . .	76

<b>Conclusion générale</b>	<b>77</b>
<b>Bibliographie</b>	<b>78</b>

# Table des figures

1.1	Gyroplane n° 01 . . . . .	4
1.2	Gyroplane n°02 . . . . .	4
1.3	Oemichen . . . . .	4
1.4	De Bothezat hélicoptère . . . . .	5
1.5	Convertawing . . . . .	5
1.6	Drone de grande taille . . . . .	6
1.7	Drone de taille réduite . . . . .	7
1.8	Drone à Voilure fixe . . . . .	7
1.9	Drone à ailes battantes . . . . .	7
1.10	Drone Mono-rotor . . . . .	8
1.11	Drone birotor . . . . .	8
1.12	Drone Tri rotor . . . . .	9
1.13	Drone Quadri-rotor . . . . .	9
1.14	Drone Hexa-rotor . . . . .	10
1.15	Drone Octorotor . . . . .	10
1.16	Mouvement verticale [14] . . . . .	12
1.17	Mouvement de Roulis [14] . . . . .	12
1.18	Mouvement de tangage [14] . . . . .	13
1.19	Mouvement de Lacet [14] . . . . .	13
1.20	Mouvement de translation horizontale [14] . . . . .	14
1.21	Géométrie du quadrirotor [14] . . . . .	15
1.22	Schéma de commande en boucle ouverte du quadrirotor . . . . .	25
1.23	Réponse en boucle ouverte du quadrirotor . . . . .	27
2.1	Réglage par PID d'un système linéaire . . . . .	30
2.2	Schéma de régulation PID de rotation et de position . . . . .	32
2.3	Déplacement d'une particule [5]. . . . .	35
2.4	Schéma d'approche d'optimisation . . . . .	38

2.5	Réponse en X . . . . .	40
2.6	Réponse en Y. . . . .	40
2.7	Réponse en Z. . . . .	41
2.8	Réponse en angle Phi. . . . .	41
2.9	Réponse en angle Teta. . . . .	42
2.10	Réponse en angle Psi. . . . .	42
2.11	Effet du vent sur X . . . . .	44
2.12	Effet du vent sur Y . . . . .	44
2.13	Effet du vent sur Z . . . . .	45
2.14	Effet de la masse sur X . . . . .	46
2.15	Effet de la masse sur Y . . . . .	46
2.16	Effet de la masse sur Z . . . . .	47
2.17	Effet de bruits externes sur X . . . . .	48
2.18	Effet de bruits externes sur Y . . . . .	48
2.19	Effet de bruits externes sur Z . . . . .	49
3.1	Variable Linguistique. . . . .	52
3.2	Fonction D'apprtenance [16] . . . . .	52
3.3	Structure d'un régulateur floue . . . . .	54
3.4	Commande Floue . . . . .	55
3.5	L'erreur de la position (entrée 1) . . . . .	56
3.6	Dérivée de l'erreur (entrée 2) . . . . .	56
3.7	sortie de Régulateur . . . . .	56
3.8	L'erreur de la position Z (entrée 1) . . . . .	56
3.9	Dérivée de l'erreur (entrée 2) . . . . .	56
3.10	Sortie de régulateur . . . . .	57
3.11	L'erreur de la position X,Y(entrée 1) . . . . .	57
3.12	Dérivée de l'erreur (entrée 2) . . . . .	57
3.13	Sortie de régulateur . . . . .	57
3.14	Réponse en X. . . . .	59
3.15	Réponse en Y. . . . .	59
3.16	Réponse en Z. . . . .	60
3.17	Réponse en Phi. . . . .	60
3.18	Réponse en angle Teta. . . . .	61
3.19	Réponse en angle Psi. . . . .	61
3.20	Les trajectoires des positions avec un effet de vent . . . . .	62
3.21	Les trajectoires des positions avec augmentation de la masse . . . . .	63

3.22	Les trajectoires des positions avec un effet du bruit de mesure . . . . .	64
3.23	Type PID floue auto réglable [1]. . . . .	65
3.24	Schéma de la commande adaptative PID logique floue [3] . . . . .	66
3.25	Entrée régulateur (erreur) . . . . .	67
3.26	Entrée régulateur (dérivé de l'erreur) . . . . .	67
3.27	Sortie du régulateur ( $K_p$ ) . . . . .	67
3.28	Sortie du régulateur ( $K_i$ ) . . . . .	67
3.29	Réponse en X . . . . .	67
3.30	Réponse en Y . . . . .	68
3.31	Réponse en Z . . . . .	68
3.32	Effet de vent sur X . . . . .	69
3.33	Effet de vent sur Y . . . . .	70
3.34	Effet de vent sur Z . . . . .	70
3.35	Effet de changement de la masse sur X . . . . .	71
3.36	Effet de changement de la masse sur Y . . . . .	72
3.37	Effet de changement de la masse sur Z . . . . .	72
3.38	Effet de bruit de mesure sur X . . . . .	73
3.39	Effet de bruit de mesure sur Y . . . . .	73
3.40	Effet de bruit de mesure sur Z . . . . .	74

# Liste des tableaux

1.1	Valeurs des paramètres du quadrirotor . . . . .	26
2.1	Paramètres et performances du réglage par PID du quadrirotor . . . . .	33
2.2	Paramètres et valeurs utilisées dans l'algorithme . . . . .	38
2.3	Paramètres et performances du réglage par PID optimisé du quadrirotor . . . . .	39
3.1	Performances du quadrirotor avec réglage flou. . . . .	58

# Introduction Générale

L'essor des technologies de drones a ouvert la voie à une multitude d'applications dans divers domaines, allant de la surveillance et de la photographie aérienne à la livraison de colis et à la recherche scientifique. Parmi ces drones, les quadrirotors qui se distinguent par leur maniabilité et leur stabilité, ce qui les rend particulièrement adaptés à une multitude de missions complexes. Cependant, pour exploiter pleinement le potentiel des quadrirotors, il est crucial de développer des stratégies de commande efficaces, robustes et capables de répondre aux exigences de stabilité et de précision en présence de perturbations et d'incertitudes dynamiques.

Les contrôleurs linéaires sont adaptés au vol quasi-stationnaire, mais ne parviennent pas à offrir de bonnes performances dans des conditions non nominales où les angles d'attitude sont importants, ainsi que lors de vols à grande vitesse et de manœuvres agressives. En revanche, les contrôleurs non linéaires, bien que potentiellement plus efficaces dans ces situations, sont souvent complexes à implémenter sur de petits microprocesseurs et difficiles à ajuster en temps réel.

Plusieurs méthodes et techniques sont proposées telles que , la commande à base de backstepping , la commande par mode glissant ou encore la commande par réseaux de neurones, la commande par logique floue, etc.

Ce mémoire se concentre sur la modélisation et la commande d'un véhicule volant de type quadrirotor, en utilisant des approches linéaires et non linéaires. Ce travail se divise en trois chapitres principaux, chacun abordant des aspects spécifiques de la commande des drones.

Le premier chapitre pose les fondations de cette étude en présentant une définition détaillée des drones et en proposant une classification des différents types de drones existants. Il se concentre ensuite sur la modélisation spécifique d'un drone quadrirotor, en décrivant ses caractéristiques physiques et dynamiques. Cette modélisation est essentielle pour comprendre les défis associés à la commande de ce type de drone.

Le deuxième chapitre explore les techniques de commande linéaire, en particulier l'utilisation des régulateurs PID (Proportionnel-Intégral-Dérivé). Il examine en détail la mé-

thode d'optimisation par essaims particulaires (PSO - Particle Swarm Optimization) pour l'ajustement des gains du PID, afin d'améliorer la performance du système de commande. Cette approche métaheuristique permet d'obtenir des réglages optimaux des paramètres PID, garantissant ainsi une stabilité et une réactivité accrues du quadrirotor.

Enfin, le troisième chapitre introduit une approche de commande non linéaire basée sur la logique floue. Cette technique offre une flexibilité accrue dans la gestion des incertitudes et des non-linéarités inhérentes au système de quadrirotor. En utilisant des règles floues et des ensembles flous, il est possible de concevoir un système de commande capable de s'adapter dynamiquement aux variations des conditions de vol, offrant ainsi une robustesse et une performance supérieure par rapport aux méthodes linéaires traditionnelles.

En résumé, ce mémoire propose une exploration approfondie des stratégies de commande pour les drones quadrirotors, en combinant des techniques linéaires et non linéaires optimisées par des méthodes métaheuristiques. Les résultats obtenus démontrent le potentiel de ces approches pour améliorer significativement les performances des drones, ouvrant ainsi de nouvelles perspectives pour leur utilisation dans des applications encore plus variées et complexes.

# Chapitre 1

## Présentation et modélisation d'un quadrirotor

### 1.1 Introduction

Les drones sont des robots volants sans pilote dotés d'un certain degré d'autonomie et d'un certain degré de capacités de décision pour fonctionner correctement. Les avancées technologiques en matière d'automatisation et de télécommunications ont permis de concevoir des drones de plus en plus performants. Ils existent en différentes tailles, des petits drones récréatifs aux grands drones militaires, et sont utilisés à des fins très diverses, notamment la vidéographie, la surveillance, la recherche et le sauvetage, la sécurité publique, les opérations militaires. . . .

Ce chapitre introduit les drones, en explorant leur historique, leurs différentes classifications, ainsi que leurs champs d'application. Ensuite, une description détaillée du principe de fonctionnement des drones sera présentée, accompagnée d'une modélisation. Enfin, une simulation en boucle ouverte du modèle proposé sera réalisée sous MATLAB-Simulink.

### 1.2 Historique

La première génération de *quadricoptères* était connue pour sa grande taille et ses faibles performances dynamiques. Le premier *quadricoptère gyrocoptère n° 01 « Breguet-Richet »* a été conçu en 1907 par les scientifiques français Louis Breguet et Jacques Breguet, en collaboration avec le professeur Charles Richet.

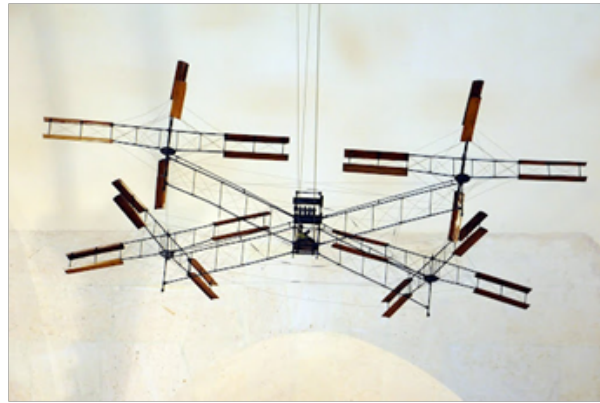


Figure 1.1: Gyroplane n° 01

Un an plus tard, un autre *quadricoptère*, le « 02 Rotorcraft », était né. Cette dernière était équipée d'un moteur plus puissant du constructeur d'engins Renault, atteignant soudain 55CV.



Figure 1.2: Gyroplane n°02

En 1920, Etienne *Oemichen* commence à expérimenter la conception d'un giravion. Le jeune ingénieur de Peugeot a conçu six modèles, dont un énorme *quadricoptère* de 800 kg construit en 1922 avec quatre rotors et huit hélices, tournant à l'aide d'un seul moteur.

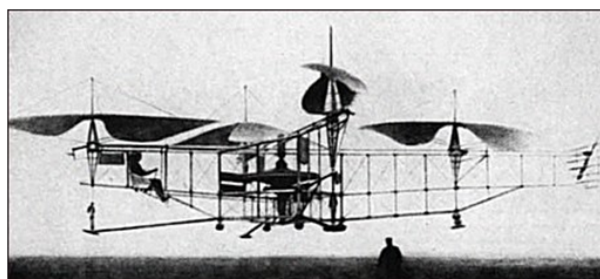


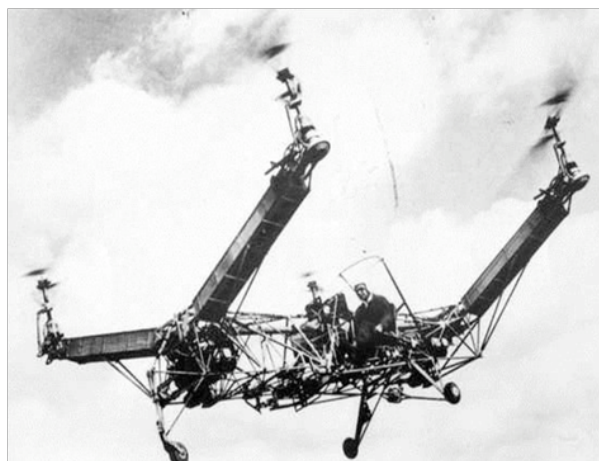
Figure 1.3: Oemichen

En 1921, les ingénieurs George de *Bothezat* et Ivan *Jérôme* construisent une autre plate-forme de décollage vertical pour l'US Army Air Corps à Dayton, Ohio. Il s'agit d'une structure transversale pesant 1678 kilogrammes. Bien que le grand *quadricoptère* ait subi plus de 100 essais en vol, le contrôle du vol s'est avéré difficile et le prototype n'a pas répondu aux exigences de performance de l'armée américaine.



**Figure 1.4:** De Bothezat hélicoptère

Dans les années 1950, à *Amityville*, New York, *Convertawing* a développé un *quadri-coptère*. L'avion est contrôlé en modifiant la poussée fournie par chaque rotor. Le *quadri-coptère* de *Convertawing* a volé avec succès, mais la production a été suspendue en raison d'un manque d'intérêt commercial. [15]



**Figure 1.5:** Convertawing

## 1.3 Classification

Il n'existe pas une manière unique de classer les drones, ils peuvent être répartis selon plusieurs critères : autonomie, portée, altitude, mission, système de contrôle, etc.

### 1.3.1 Classification selon la taille

Les drones sont souvent répartis selon leur taille (ou leur capacité d'emport, à peu près équivalente), on distingue donc les drones HALE, MALE, le micro drones et les mini drones.

- HALE (High Altitude Long Endurance) : Grand drone, généralement à ailes fixes. Ils sont capables de rester en vol pendant de longues périodes et de collecter des informations sur de longues périodes (12 à 48 heures).
- MALE (Medium Altitude Long Endurance) : Pour les vols de longue durée à moyennes altitudes opérationnelles avec un haut degré d'autonomie.

Ces deux types de drones appartiennent à la grande catégorie des drones. Ils peuvent porter des armes, ce qui nécessite généralement la présence de quelqu'un, ce dernier doit préserver la décision de tirer et pouvoir annuler la mission à tout moment.



**Figure 1.6:** Drone de grande taille

- Micro drones : Drones dont la taille varie d'un centimètre à plusieurs dizaines de centimètres. Généralement alimenté par l'électricité. Ils autorisent donc le vol en intérieur. Ils se caractérisent par de faibles charges.
- Mini Drones : ce sont des drones petits (quelques kilogrammes, envergure jusqu'à 1 à 2 mètres) et légers qui permettent une autonomie relativement faible (10 à 30 minutes), généralement utilisés pour les zones d'observation difficiles d'accès. [22]

### 1.3.2 Classification selon le mode de propulsion

La manipulation aérodynamique offre une autre possibilité de classification. Ainsi, les drones peuvent être principalement divisés en trois séries :



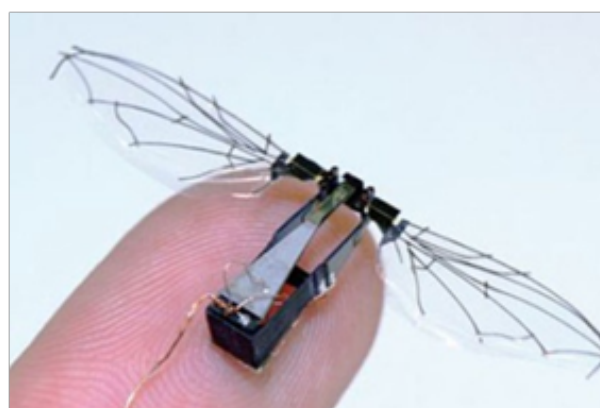
**Figure 1.7:** Drone de taille réduite

- Drone à voilure fixe : est un drone qui utilise une voilure fixe dans son mode de mouvement, qui est :
  - Plus lourd que l'air : type d'avion.
  - Plus léger que l'air : type dirigeable.



**Figure 1.8:** Drone à Voilure fixe

- Drone à ailes battantes : type oiseau ou insecte.



**Figure 1.9:** Drone à ailes battantes

- Drone à Voilure tournantes : (UAV à rotor) La recherche documentaire montre que la recherche sur les robots aériens se concentre principalement sur ce type, qui présente les avantages suivants :

- Décollage et atterrissage verticaux.
- Capacité de planer à basse vitesse et altitude.

Les drones à rotor sont subdivisés en 6 sous-catégories, on peut donc les classer comme suit selon le nombre d'hélices :

- Rotor unique : Il se caractérise par l'utilisation d'un seul rotor comme actionneur principal. Dans cette catégorie on retrouve principalement les convertibles.



**Figure 1.10:** Drone Mono-rotor

- Avion bi-rotor : Il existe de nombreux types de configurations bi-rotor, comme l'hélicoptère classique composé d'un rotor principal et d'un rotor de queue. Les équipements sans plateau cyclique (surtout sauf pour les hélicoptères) utilisent des ailerons pour faire tourner le rotor.

Il existe également des unités qui ont deux rotors sur le même axe tournant dans des directions opposées, et les ailettes sont baignées par le flux d'air de ces rotors.



**Figure 1.11:** Drone birotor

- Tricoptère : L'efficacité de vol est inférieure à celle du quadricoptère. Le tricoptère se compose de deux rotors à l'avant. Les deux rotors tournent dans des directions opposées pour modifier l'angle de tangage et le rotor arrière ajuste le roulis.



**Figure 1.12:** Drone Tri rotor

- Quadricoptère : est un avion doté de quatre rotors montés aux deux extrémités d'un châssis en forme de croix. Ces quatre rotors fournissent la force verticale (portance) qui lui permet de décoller. Deux rotors avec le même axe tournent dans le sens des aiguilles d'une montre, tandis que les deux autres tournent dans des directions opposées : roulis ou tangage, qui est la direction que prend le quad rotor, et lacet, qui est la rotation du quad rotor sur lui-même.



**Figure 1.13:** Drone Quadri-rotor

- Rotor hexagonal : C'est un avion équipé d'un rotor hexagonal. [15]



**Figure 1.14:** Drone Hexa-rotor

— Octocopter : C'est un avion équipé d'un huit rotors.



**Figure 1.15:** Drone Octorotor

## 1.4 Modélisation du quadrirotor

Dans cette section, nous allons explorer le concept de drone miniature, un véhicule aérien sans pilote de petite taille, de type quadrirotor.

Un quadrirotor est un type spécifique de véhicule aérien sans pilote (UAV) doté de quatre rotors. Chaque rotor est situé à l'extrémité d'un bras qui s'étend à partir du centre de la structure. Cette disposition symétrique permet au quadrirotor d'assurer sa stabilité et sa maniabilité dans les trois dimensions de l'espace. Les rotors, généralement alimentés par des moteurs électriques, produisent la poussée nécessaire pour maintenir le quadrirotor en vol et lui permettre de se déplacer dans différentes directions [2]. Afin de concevoir un quadrirotor, on doit d'abord comprendre profondément ses mouvements, sa dynamique et par conséquent les équations dynamiques. Cette compréhension est essentielle non seulement pour la conception du contrôleur, mais aussi pour s'assurer que les simulations du comportement de véhicule sont plus proches que possible de la réalité lorsque la commande est appliquée.

### 1.4.1 Mouvements du quadrirotor

Le changement de la vitesse de chaque rotor individuel génère une poussée, ce qui rend le mouvement possible. Le quadrirotor s'incline dans la direction du rotor le plus long et se déplace ensuite le long de cet axe. Par conséquent, c'est l'angle d'inclinaison qui permet au quadrirotor de se déplacer, ce qui implique que les changements de vitesse du rotor entraînent un mouvement dans au moins trois degrés de liberté. Par exemple, en augmentant la vitesse du rotor avant, le quadrirotor effectue un mouvement de tangage. Ainsi, nous pouvons contrôler les six degrés de liberté du quadrirotor avec cinq mouvements principaux :

- Mouvement vertical
- Mouvement de roulis
- Mouvement de tangage
- Mouvement de lacet
- Mouvement de translation horizontale

### 1.4.1.1 Mouvement vertical

La force combinée des quatre rotors, projetée sur l'axe  $Z$ , s'oppose à la force de gravité. La somme de ces deux forces détermine si le quadrirotor monte ou descend. Si la force générée par les rotors est supérieure au poids du quadrirotor, le quadrirotor s'élève ; si elle est inférieure, il descend. Si les deux forces sont égales, le quadrirotor planera.

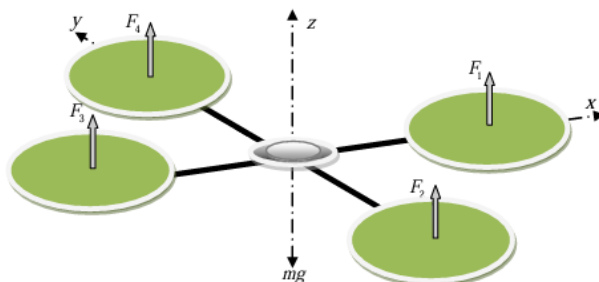


Figure 1.16: Mouvement verticale [14]

### 1.4.1.2 Mouvement de roulis

Pour exécuter ce mouvement, nous appliquons une différence de vitesse entre le rotor 2 et 4 tout en maintenant la même vitesse pour les rotors 1 et 3. Cela crée un couple autour de l'axe  $x$ , ce qui entraîne une rotation autour de cet axe. Ce mouvement est associé à un déplacement le long de l'axe  $y$ .

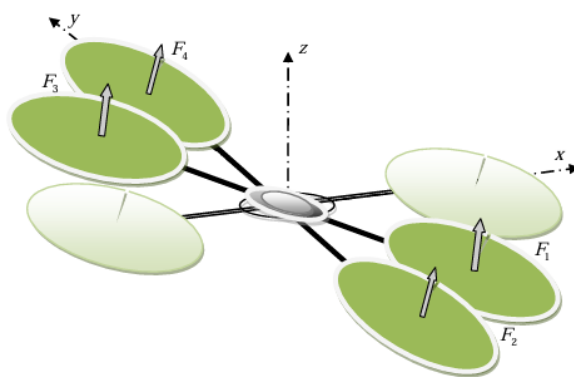


Figure 1.17: Mouvement de Roulis [14]

### 1.4.1.3 Mouvement de tangage

Pour exécuter ce mouvement, nous appliquons une différence de vitesse entre le rotor 1 et 3 tout en maintenant la même vitesse pour les rotors 2 et 4. Cela crée un couple autour de l'axe  $y$ , ce qui entraîne une rotation autour de cet axe. Ce mouvement est associé à un déplacement le long de l'axe  $x$ .

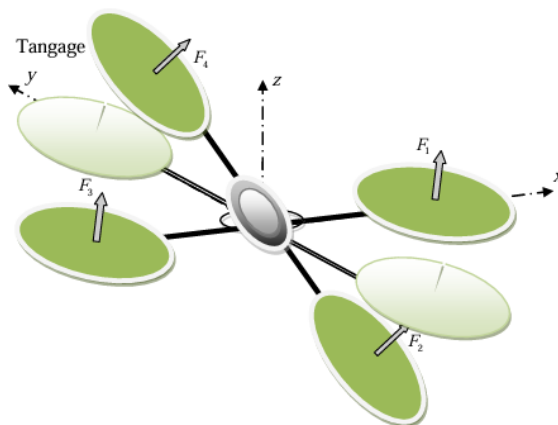


Figure 1.18: Mouvement de tangage [14]

### 1.4.1.4 Mouvement de Lacet

Pour exécuter ce mouvement, nous appliquons une différence de poussée entre les rotors (1,3) et (2,4). Cela crée un couple autour de l'axe  $z$ , ce qui entraîne une rotation autour de cet axe. Pour éviter que ce mouvement de rotation ne soit associé à d'autres mouvements de rotation, il est essentiel que les vitesses des rotors situés sur le même bras soient égales. De plus, pour maintenir une force de poussée constante, il est nécessaire que l'augmentation de la vitesse de rotation d'une paire de rotors soit compensée par une diminution équivalente de la vitesse de rotation de l'autre paire.

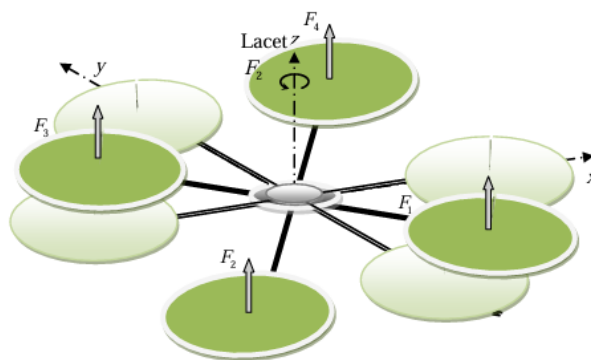


Figure 1.19: Mouvement de Lacet [14]

### 1.4.1.5 Mouvement de translation horizontale

Dans ce cas, nous souhaitons appliquer une force le long de l'axe  $x$  ou  $y$  en inclinant le corps (par le tangage ou le roulis) et en augmentant la poussée globale produite. Cela permet de maintenir la composante de la poussée le long de l'axe  $z$  égale à la force de gravité.

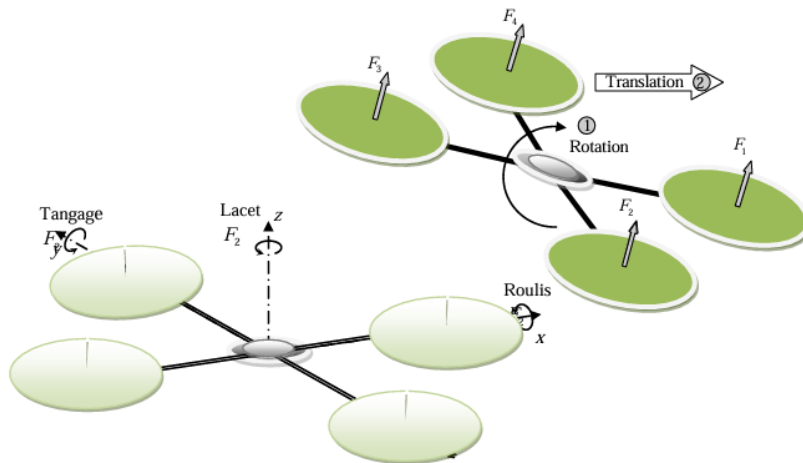


Figure 1.20: Mouvement de translation horizontale [14]

### 1.4.2 Modèle dynamique du quadrirotor

La modélisation d'un quadrirotor est une tâche complexe en raison de la dynamique fortement non linéaire et entièrement couplée du système. Pour mieux comprendre le modèle dynamique présenté ci-dessous, voici les différentes hypothèses de travail :

- \* Le quadrotor est un corps rigide.
- \* Les hélices sont supposées rigides.
- \* Le centre de gravité coïncide avec l'origine du repère fixe du corps.
- \* Les forces de portance et de trainée sont proportionnelles aux carrés de la vitesse de rotation des rotors.

Afin d'élaborer le modèle mathématique, on utilise deux repères, un repère fixe lié à la terre  $R^b$  et un autre mobile lié au centre de masse du quadri-rotor  $R^m$ . Pour passer du repère fixe au repère mobile, nous définissons une matrice de transformation  $T$  qui représente la position et l'orientation du repère mobile par rapport au repère fixe.

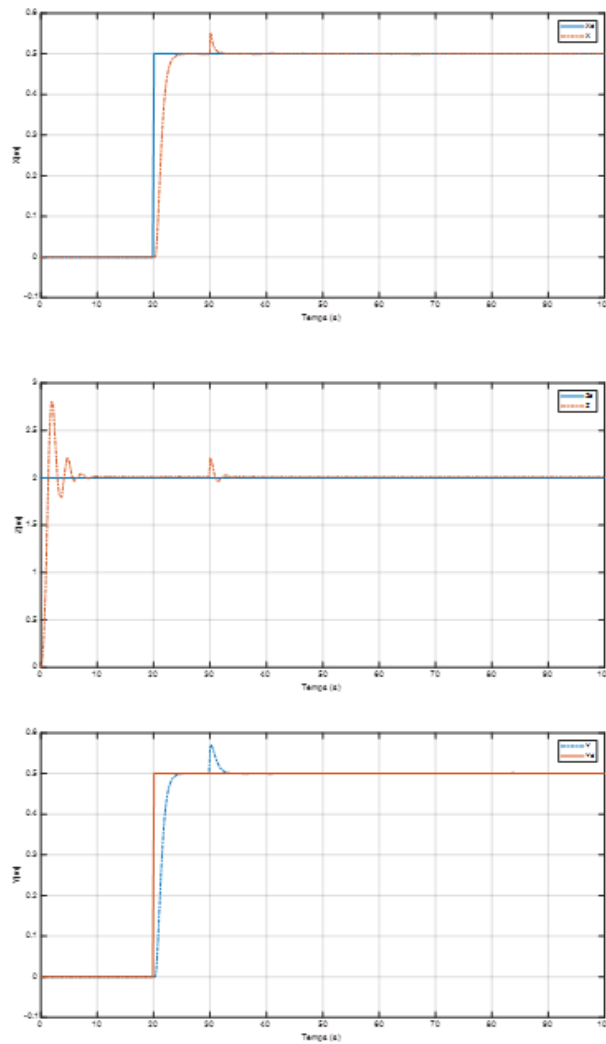


Figure 1.21: Géométrie du quadrirotor [14]

$$T = \begin{bmatrix} R & \xi \\ 0 & 1 \end{bmatrix} \quad (1.1)$$

Avec  $R$  la matrice de rotation de l'objet mobile.

Le vecteur de position est donné par :

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (1.2)$$

Pour déterminer les éléments de la matrice  $R$ , nous utiliserons les angles d'Euler.

### 1.4.2.1 Angles d'Euler

Initialement, le repère mobile coïncide avec le repère fixe. Ensuite, il subit trois rotations : une rotation autour de l'axe X avec un angle de roulis  $\phi$ , une rotation autour de l'axe Y avec un angle de tangage  $\theta$ , et enfin une rotation autour de l'axe Z avec un angle de lacet  $\psi$ .

Nous supposons que  $(-\frac{\pi}{2} < \phi < \frac{\pi}{2})$  et  $(-\frac{\pi}{2} < \theta < \frac{\pi}{2})$  et  $(-\pi < \psi < \pi)$ . La rotation autour de l'axe x par un angle  $\phi$  est représentée par la matrice :

$$Rot_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \quad (1.3)$$

La rotation autour de l'axe y par un angle  $\theta$  est représentée par la matrice :

$$Rot_y(\theta) = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \quad (1.4)$$

La rotation autour de l'axe z par un angle  $\psi$  est représentée par la matrice :

$$Rot_z(\psi) = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.5)$$

La matrice de rotation globale  $\mathbf{R}(\phi, \theta, \psi)$  relie les centres des repères  $R_b$  et  $R_m$ . Elle est obtenue en multipliant les trois matrices de rotation selon la formule suivante :

$$\mathbf{R}(\phi, \theta, \psi) = Rot_z(\psi) \times Rot_y(\theta) \times Rot_x(\phi) \quad (1.6)$$

$$\mathbf{R}(\phi, \theta, \psi) = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \quad (1.7)$$

$$\mathbf{R}(\phi, \theta, \psi) = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\theta c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (1.8)$$

Avec :  $c = \cos x$  et  $s = \sin x$

### 1.4.2.2 Vitesses angulaires

Les vitesses de rotation dans le repère fixe  $(\Omega_1, \Omega_2, \Omega_3)$  sont exprimées en fonction des vitesses de rotation dans le repère mobile  $(\phi, \theta, \psi)$  comme suit :

$$\Omega = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \text{Rot}_x(\phi)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + (\text{Rot}_y(\theta)\text{Rot}_x(\phi))^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (1.9)$$

En effet, la rotation en roulis se produit lorsque les repères sont encore confondus. Ensuite, pour le tangage, le vecteur représentant la rotation doit être exprimé dans le repère fixe, ce qui nécessite de le multiplier par  $\text{Rot}_z(\phi)^{-1}$ . De même, le vecteur représentant la rotation en lacet doit être exprimé dans le repère fixe qu'il a déjà subi deux rotations. On obtient ainsi :

$$\Omega = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\theta}c\phi \\ -\dot{\theta}s\phi \end{bmatrix} + \begin{bmatrix} -\dot{\psi}s\theta \\ \dot{\psi}s\phi c\theta \\ \dot{\psi}c\phi c\theta \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \dot{\psi}s\theta \\ \dot{\theta}c\phi + \dot{\psi}s\phi c\theta \\ \dot{\psi}c\phi c\theta - \dot{\theta}s\phi \end{bmatrix} \quad (1.10)$$

$$\Omega = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (1.11)$$

donc :

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s\phi \tan \theta & c\phi \tan \theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \quad (1.12)$$

Lorsque le quadrirotor effectue de petites rotations, les approximations suivantes peuvent être appliquées :

$$c\phi \approx c\theta \approx 1 \text{ et } s\phi \approx \phi, s\theta \approx \theta$$

Donc la vitesse angulaire sera :

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \quad (1.13)$$

### 1.4.2.3 Effets physiques agissants sur le quadrirotor

#### a) Forces

Les forces agissant sur le système sont :

— **Force de pesanteur**

La force de pesanteur est la force qui attire le quadrirotor vers le centre de la Terre. Elle est calculée comme le produit de la masse totale du quadrirotor ( $m$ ) et de l'accélération due à la gravité ( $g$ ). Cette force agit verticalement vers le bas et est donnée par l'expression

$$P = mg \tag{1.14}$$

— **Forces de poussée**

Les forces de poussée, résultant de la rotation des moteurs, sont perpendiculaires au plan des hélices. Elles sont proportionnelles au carré de la vitesse de rotation des moteurs.

$$F_i = b\omega_i^2 \quad (1.15)$$

Avec  $i = (1, 2, 3, 4)$  et  $b$  est le coefficient de portance, il dépend de la forme et le nombre des pales et la densité de l'air.

— **Forces de traînée**

La traînée est la force qui s'oppose au mouvement d'un corps dans un fluide ou un gaz, agissant comme un frottement. Pour les hélices, cette force est proportionnelle au carré de leur vitesse de rotation, et elle est décrite par la relation :

$$T_h = d\omega_i^2 \quad (1.16)$$

Avec  $d$  coefficient de traînée, sa valeur dépend de la fabrication spécifique de l'hélice. Pour la traînée selon les axes  $(x, y, z)$ , elle est due au mouvement du corps du quadrirotor et elle est décrite par la relation :

$$F_t = K_{ft}V^2 \quad (1.17)$$

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

**b) Moments**

Le quadrirotor subit plusieurs moments, résultants des forces de poussée, de traînée et des effets gyroscopiques.

**- Moments dus aux forces de poussée**

La variance de force entre les rotors 2 et 4 génère un moment qui provoque une rotation autour de l'axe X, Ce moment est exprimé par la relation suivante :

$$M_x = l(F_4 - F_2) = lb(\omega_4^2 - \omega_2^2) \quad (1.18)$$

La variable "l" représente la longueur du bras entre le rotor et le centre de gravité du quadrirotor.

La variance de force entre les rotors 1 et 3 génère un moment qui provoque une rotation autour de l'axe Y, Ce moment est exprimé par la relation suivante :

$$M_y = l(F_3 - F_1) = lb(\omega_3^2 - \omega_1^2) \quad (1.19)$$

**- Moments dus aux forces de traînée**

Il résulte d'un couple réactif engendré par les forces de traînée agissant sur chaque hélice. Ce moment est décrit par la relation suivante :

$$M_z = d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \quad (1.20)$$

Moment résultant des frottements aérodynamiques, défini par :

$$M_a = k_{fa}\Omega^2 \quad (1.21)$$

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

**- Effet gyroscopique**

La rotation simultanée autour de deux axes perpendiculaires entraîne une troisième rotation autour d'un axe perpendiculaire aux deux précédents. Cet effet engendre des rotations non souhaitées de l'engin.

Dans notre cas, nous avons deux moments gyroscopiques.

**- Moment gyroscopique des hélices**

Il est donné par la relation suivante :

$$M_{gh} = \sum_{i=1}^4 \Omega \wedge J_r \left[ 0 \quad 0 \quad (-1)^{i+1} \omega_i \right]^T \quad (1.22)$$

Avec :  $J_r$  l'inertie des rotors.

**- Moment gyroscopique dû aux mouvements de quadri-rotor**

Il est donné par la relation suivante :

$$M_{gm} = \Omega \wedge J\Omega \quad (1.23)$$

Avec  $J$  l'inertie du système.

### 1.4.3 Développement du Modèle mathématique selon Newton-Euler

Après avoir décrit en détail toutes les forces et les différents moments qui agissent sur le système, nous pouvons maintenant formuler les équations dynamiques en utilisant le formalisme de Newton-Euler.

Le système peut être exprimé de la manière suivante :

$$\begin{cases} \dot{\xi} = v \\ m\ddot{\xi} = F_t + F_g + F_f \\ \dot{R} = RS(\Omega) \\ J\dot{\Omega} = -\Omega \wedge J\Omega + M_f - M_a - M_{gh} \end{cases} \quad (1.24)$$

Avec :

$\xi$  : le vecteur de position du quadrirotor.

$m$  : la masse totale du quadrirotor.

$\Omega$  : la vitesse angulaire exprimée dans le repère fixe

$R$  : la matrice de rotation

$\wedge$  : Le produit vectoriel

$J$  : la matrice d'inertie symétrique :

$$J = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (1.25)$$

$S(\Omega)$  : la matrice antisymétrique associé au vecteur de velocity  $\Omega = [\Omega_1, \Omega_2, \Omega_3]^T$ , elle est donnée par :

$$S(\Omega) = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \quad (1.26)$$

$F_f$  : la force totale générée par les quatre rotors, elle est donnée par :

$$F_f = R \times \left[ 0 \quad 0 \quad \sum_{i=1}^4 F_i \right]^T \quad (1.27)$$

$$F_i = b\omega_i^2 \quad (1.28)$$

$F_g$  : la force de gravité, elle est donnée par :

$$F_g = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \quad (1.29)$$

$F_t$  : la force de traînée selon les axes (x, y, z), elle est donnée par :

$$F_t = \begin{bmatrix} -K_{ftx} & 0 & 0 \\ 0 & -K_{fity} & 0 \\ 0 & 0 & -K_{ftz} \end{bmatrix} \dot{\xi} \quad (1.30)$$

$M_f$  : Représente le moment provoqué par les forces de poussée et de traînée, il est donné par :

$$M_f = \begin{bmatrix} l(F_4 - F_2) \\ l(F_3 - F_1) \\ d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \quad (1.31)$$

$M_a$  : représente le moment résultant des frottements aérodynamiques, il est donné par :

$$M_a = \begin{bmatrix} K_{fax}\Omega_x^2 \\ K_{fay}\Omega_y^2 \\ K_{faz}\Omega_z^2 \end{bmatrix} \quad (1.32)$$

Avec :

$K_{fax}, K_{fay}, K_{faz}$  : Les coefficients des frottements aérodynamiques.

### 1.4.3.1 Équations de mouvement de translation

D'après le modèle de Newton-Euler, on a :

$$m\ddot{\xi} = F_f + F_g + F_t \quad (1.33)$$

En remplaçant chaque force par sa formule, nous obtenons :

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} c\phi s\theta c\psi + s\phi s\psi \\ c\phi s\theta s\psi - s\phi c\psi \\ c\phi c\theta \end{bmatrix} \sum_{i=1}^4 F_i - \begin{bmatrix} K_{ftx}\dot{x} \\ K_{fity}\dot{y} \\ K_{ftz}\dot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (1.34)$$

Ainsi les équations différentielles décrivant le mouvement de translation sont :

$$\begin{cases} \ddot{x} = \frac{1}{m} (c\phi s\theta c\psi + s\phi s\psi) (\sum_{i=1}^4 F_i) - \frac{K_{ftx}}{m} \dot{x} \\ \ddot{y} = \frac{1}{m} (c\phi s\theta s\psi - s\phi c\psi) (\sum_{i=1}^4 F_i) - \frac{K_{fity}}{m} \dot{y} \\ \ddot{z} = \frac{1}{m} (c\phi c\theta) (\sum_{i=1}^4 F_i) - \frac{K_{ftz}}{m} \dot{z} - g \end{cases} \quad (1.35)$$

### 1.4.3.2 Équations de mouvement de rotation

D'après le modèle de Newton-Euler, on ma :

$$J\dot{\Omega} = -M_{gm} - M_{gh} + M_f - M_a \quad (1.36)$$

En remplaçant chaque moment par sa formule, nous obtenons les équations différentielles décrivant le mouvement de rotation sont les suivantes :

$$\begin{aligned} I_x \ddot{\phi} &= -\dot{\theta}\dot{\psi} (I_z - I_y) - J_r \bar{\Omega} \dot{\theta} - K_{fax} \dot{\phi}^2 + lb (\omega_4^2 - \omega_2^2) \\ I_y \ddot{\theta} &= \dot{\phi}\dot{\psi} (I_x - I_z) + J_r \bar{\Omega} \dot{\phi} - K_{fay} \dot{\theta}^2 + lb (\omega_3^2 - \omega_1^2) \\ I_z \ddot{\psi} &= -\dot{\phi}\dot{\theta} (I_y - I_x) - K_{faz} \dot{\psi}^2 + d (\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{aligned} \quad (1.37)$$

Avec

$$\bar{\Omega}_r = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4 \quad (1.38)$$

Ainsi, le modèle dynamique complet qui régit le quadrirotor est le suivant :

$$\begin{cases} \ddot{\phi} = \frac{(I_y - I_z)}{I_x} \dot{\theta}\dot{\psi} - \frac{J_r \bar{\Omega}_r}{I_x} \dot{\theta} - \frac{K_{fax}}{I_x} \dot{\phi}^2 + \frac{1}{I_x} U_2 \\ \ddot{\theta} = \frac{(I_z - I_x)}{I_y} \dot{\phi}\dot{\psi} + \frac{J_r \bar{\Omega}_r}{I_y} \dot{\phi} - \frac{K_{fay}}{I_y} \dot{\theta}^2 + \frac{1}{I_y} U_3 \\ \ddot{\psi} = \frac{(I_x - I_y)}{I_z} \dot{\phi}\dot{\theta} - \frac{K_{faz}}{I_z} \dot{\psi}^2 + \frac{1}{I_z} U_4 \\ \ddot{x} = -\frac{K_{ftx}}{m} \dot{x} + \frac{1}{m} U_x U_1 \\ \ddot{y} = -\frac{K_{fty}}{m} \dot{y} + \frac{1}{m} U_y U_1 \\ \ddot{z} = -\frac{K_{ftz}}{m} \dot{z} - g + \frac{c\phi c\theta}{m} U_1 \end{cases} \quad (1.39)$$

Avec :

$$\begin{cases} U_x = c\phi s\theta c\psi + s\phi s\psi \\ U_y = c\phi s\theta s\psi - s\phi c\psi \end{cases} \quad (1.40)$$

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -lb & 0 & lb \\ -lb & 0 & lb & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (1.41)$$

### 1.4.4 Représentation d'état du système

Pour un même système physique, il existe de nombreuses façons pour exprimer la représentation d'état. Dans notre étude, nous choisissons le vecteur d'état suivant :

$$X = [\varphi \ \theta \ \phi \ \dot{\varphi} \ \dot{\theta} \ \dot{\phi} \ \psi \ x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]^T \quad (1.42)$$

$$X = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10} \ x_{11} \ x_{12}]^T \quad (1.43)$$

Finalement, nous obtenons la représentation d'état suivante :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1(x) + g_1(x)U_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2(x) + g_2(x)U_3 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = f_3(x) + g_3(x)U_4 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = f_4(x) + g_4(x)U_xU_1 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = f_5(x) + g_5(x)U_yU_1 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = f_6(x) + g_6(x)U_1 \end{cases} \quad (1.44)$$

$$\begin{cases} f_1(x) = \left(\frac{I_y - I_z}{I_x}\right) x_4 x_6 - \frac{J_r \bar{\Omega}_r}{I_x} x_4 - \frac{K_{fax}}{I_x} x_2^2, & g_1(x) = \frac{1}{I_x} \\ f_2(x) = \left(\frac{I_z - I_x}{I_y}\right) x_2 x_6 - \frac{J_r \bar{\Omega}_r}{I_y} x_2 - \frac{K_{fay}}{I_y} x_4^2, & g_2(x) = \frac{1}{I_y} \\ f_3(x) = \left(\frac{I_x - I_y}{I_z}\right) x_2 x_4 - \frac{K_{faz}}{I_z} x_6^2, & g_3(x) = \frac{1}{I_z} \\ f_4(x) = -\frac{K_{ftx}}{m} x_8, & g_4(x) = \frac{1}{m} \\ f_5(x) = -\frac{K_{fity}}{m} x_{10}, & g_5(x) = \frac{1}{m} \\ f_6(x) = -\frac{K_{ftz}}{m} x_{12} - g, & g_6(x) = \frac{c\phi c\theta}{m} \end{cases} \quad (1.45)$$

## 1.5 Dynamique des moteurs

Les moteurs DC sans balais, comme ceux utilisés dans notre projet, sont connus par leur petite constante de temps. Ainsi, afin de simplifier l'étude, la dynamique des actionneurs est négligée dans cette étude. Cette approche est également évoquée dans plusieurs articles de recherche dans la littérature, tels que [20]

## 1.6 Commande en boucle ouverte

### 1.6.1 Définition

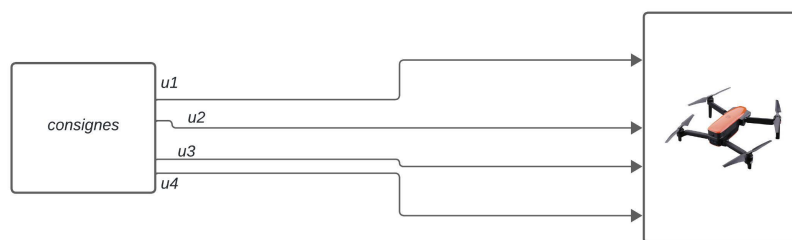
La commande en boucle ouverte est une méthode de contrôle où les signaux de commande sont envoyés aux actionneurs sans utiliser de retour d'information pour ajuster ces commandes. Cette méthode repose sur des commandes prédéterminées ou fournies par un opérateur, sans tenir compte des variations ou perturbations externes qui pourraient affecter le comportement du drone.

### 1.6.2 Principe de fonctionnement

Dans une commande en boucle ouverte, les actions du drone sont entièrement dictées par des commandes programmées à l'avance. Le drone suit ces commandes de manière rigide, sans ajustements basés sur les conditions réelles rencontrées en vol.

### 1.6.3 Simulation en boucle ouverte

Afin de décrire le comportement dynamique du quadrirotor sans aucune régulation, on a utilisé la simulation en boucle ouverte sous MATLAB. Dans ce test on a introduit les vitesses de rotation de quatre moteurs comme des entrées de commande et on a visualiser les états de translation et de rotation du système.



**Figure 1.22:** Schéma de commande en boucle ouverte du quadrirotor

Les paramètres du quadrirotor sont donnés dans le tableau 1.1

Paramètre	Valeur
$m$	1.62 kg
$l$	0.25 m
$g$	9.81 m/s <sup>2</sup>
$b$	$2.98 \times 10^{-5}$ N · m <sup>2</sup> /rad
$d$	$3.23 \times 10^{-7}$ N · m/rad
$I_x$	$3.82 \times 10^{-3}$ kg · m <sup>2</sup>
$I_y$	$3.82 \times 10^{-3}$ kg · m <sup>2</sup>
$I_z$	$7.65 \times 10^{-3}$ kg · m <sup>2</sup>
$I_r$	$2.82 \times 10^{-4}$ kg · m <sup>2</sup>
$K_{fax}$	$5.5670 \times 10^{-4}$ N/rad/s
$K_{fay}$	$5.5670 \times 10^{-4}$ N/rad/s
$K_{faz}$	$6.3540 \times 10^{-4}$ N/m/s
$K_{ftx}$	$5.5670 \times 10^{-4}$ N/m/s
$K_{fty}$	$5.5670 \times 10^{-4}$ N/m/s
$K_{ftz}$	$6.3540 \times 10^{-4}$ N/m/s

**Tableau 1.1:** Valeurs des paramètres du quadrirotor

Nous observons d'après les figures ci-dessous que le système présente une instabilité, que ce soit en ce qui concerne la position ou l'altitude. Une régulation en boucle fermée s'avère nécessaire pour stabiliser le système et améliorer ses performances dynamiques et statiques.

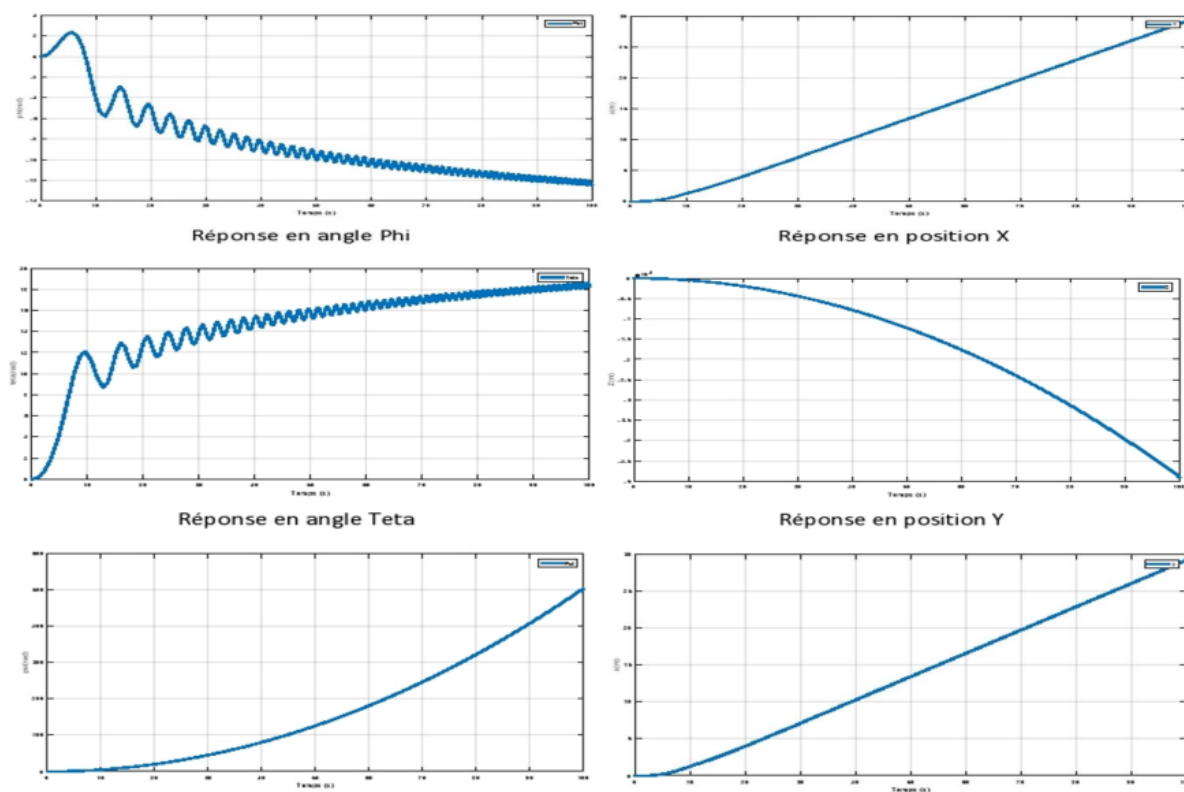


Figure 1.23: Réponse en boucle ouverte du quadrirotor

## 1.7 Conclusion

Dans ce chapitre, nous avons d'abord clarifié le concept de quadrirotor, suivi d'un bref aperçu historique pour mettre en contexte son évolution. Ensuite, nous avons examiné sa classification en fonction de sa taille et de son mode de propulsion, fournissant ainsi un cadre de compréhension pour notre étude approfondie. Nous avons ensuite abordé le fonctionnement général du quadrirotor et les différentes manoeuvres qu'il peut exécuter en vol. Après avoir examiné les effets physiques influant sur son vol, nous avons établi les bases d'un modèle mathématique décrivant sa dynamique, basé sur le formalisme de Newton-Euler. Ce modèle a été ensuite représenté dans un modèle d'état pour une compréhension plus approfondie. Enfin, nous avons exploré la commande en boucle ouverte comme une approche simple et directe pour contrôler le quadrirotor. Toutefois, bien que cette méthode soit simple, elle est limitée en termes d'adaptabilité et de précision et ne permet pas d'ajuster les commandes en fonction des variations rencontrées en vol. Cela met en évidence la nécessité de passer à une approche de contrôle plus avancée.

# Chapitre 2

## Commande linéaire et optimisation du PID

### 2.1 Introduction

Le rôle du contrôleur est de transformer l'écart entre la sortie mesurée du système et la valeur de référence en un signal de contrôle visant à réduire cet écart à zéro. Les performances du contrôleur et du système asservi sont évaluées selon plusieurs critères : rapidité, stabilité, dépassement et précision.

L'asservissement doit trouver un compromis entre performance et stabilité. La performance augmente avec le gain du correcteur, mais au-delà d'une certaine valeur, cela risque de perturber la stabilité du système.

Dans cette partie, on s'intéresse à la commande linéaire par PID du quadrirotor et à l'optimisation des gains de ce régulateur afin d'aboutir à de meilleures performances sur le système.

### 2.2 Approche Linéaire

Ce contrôleur est basé sur la linéarisation du modèle dynamique autour d'un point de fonctionnement spécifique et sur le calcul des actions proportionnelles, intégrales et dérivées. Etant donné que la formulation de l'espace d'état jusqu'ici est non linéaire et que la méthode de contrôle proposée repose sur un modèle linéaire, il est nécessaire de linéariser le système pour obtenir une représentation d'état linéaire. Ce contrôleur est élaboré en linéarisant le modèle dynamique autour d'un point de fonctionnement spécifié et en calculant les actions proportionnelle, intégrale et dérivée. Pour linéariser le modèle dynamique du quadrirotor, nous considérons le point de fonctionnement comme étant le

point de stabilisation de l'attitude à 0 degré pour tous les angles. Ainsi, nous travaillons avec de petits angles, ce qui nous permet d'approximer  $\sin(\theta)$  par  $\theta$  et  $\cos(\theta)$  par 1. Sous ces hypothèses, nous obtenons le modèle réduit linéarisé du quadrirotor, donné comme suit :

$$\begin{cases} \ddot{\phi} = \frac{U_2}{I_x} \\ \ddot{\theta} = \frac{U_2}{I_y} \\ \ddot{\psi} = \frac{U_3}{I_z} \end{cases} \quad (2.1)$$

## 2.3 Régulateur PID

Le contrôleur « proportionnel-intégral-dérivé » est la logique de contrôle la plus largement utilisée parmi les régulateurs à boucle de rétroaction. Ce contrôleur fonctionne de trois manières :

- Action proportionnelle : erreur multipliée par le gain  $K_p$ . Cette action consiste à augmenter l'erreur par un facteur constant pour permettre au système de réagir plus rapidement aux variations de consigne.
- Action d'intégration : Cette action vise à réduire, voire à éliminer, l'erreur statique en régime permanent. Pour ce faire, le régulateur intègre l'erreur dans le temps et multiplie le résultat par une constante  $K_i$
- Action de dérivation : Afin d'obtenir une action dérivée, on multiplie la dérivée de l'erreur par un coefficient  $K_d$ . Cette action aide à réduire le dépassement de la réponse et à améliorer la stabilité du système. Ces paramètres peuvent être compris en termes d'erreur : P dépend de l'erreur actuelle, I de l'accumulation des erreurs passées, et D est une estimation des erreurs futures. La fonction de transfert d'un régulateur PID s'exprime :

$$C(p) = K_p + \frac{1}{K_i} \cdot \frac{1}{p} + K_d p \quad (2.2)$$

La combinaison pondérée de ces trois actions est utilisée pour ajuster le processus. En ajustant les trois paramètres dans l'algorithme du contrôleur selon des conditions spécifiques, celui-ci peut fournir une action de commande qui répond à nos attentes

Le régulateur, représenté par sa fonction de transfert  $C(p)$ , est positionné en amont du système à réguler  $H(p)$ . L'entrée du régulateur inclut nécessairement la consigne  $r(p)$ . Ainsi, la mesure  $y(p)$  de la grandeur à réguler est comparée en temps réel à la référence (la consigne)  $r(p)$ , ce qui nous donne un paramètre important appelé écart ou erreur.

$$e(p) = r(p) - y(p) \quad (2.3)$$

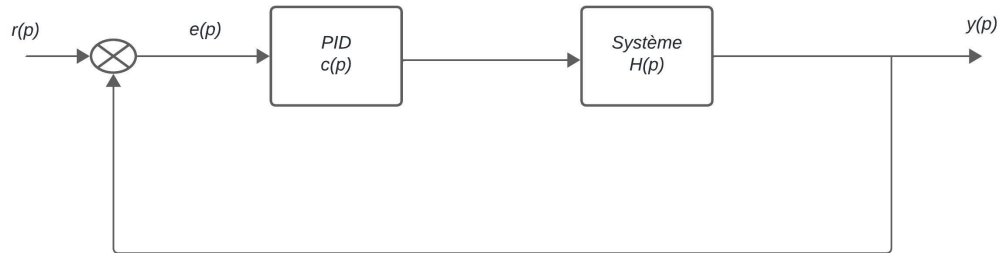


Figure 2.1: Réglage par PID d'un système linéaire

### 2.3.1 Structure d'un régulateur PID

Les différentes actions élémentaires d'un régulateur peuvent être combinées de diverses manières, formant ce qu'on appelle la structure de l'algorithme du régulateur. Les valeurs des paramètres PID n'entraînent pas le même comportement du processus en fonction de la structure choisie, qu'elle soit parallèle ou mixte. Il est donc essentiel que l'ingénieur en automatique soit familiarisé avec ces structures. En conséquence, il existe plusieurs architectures possibles pour combiner les effets des actions (série, parallèle ou mixte), les deux premières étant les plus couramment utilisées.

1. **Structure parallèle** : Dans cette structure la sortie  $y(t)$  est donné par :

$$Y(t) = K_p * e(t) + \frac{1}{T_i} \cdot \int_0^t e(\tau) \cdot d\tau + T_d \cdot \frac{de(t)}{dt} \quad (2.4)$$

2. **Structure série** : Dans cette structure la sortie  $y(t)$  est donné par :

$$Y(t) = \alpha * K_p * e(t) + \frac{K_p}{T_i} \cdot \int_0^t e(\tau) \cdot d\tau + T_d \cdot \frac{de(t)}{dt} \quad (2.5)$$

Avec :  
 $\alpha = \frac{T_i + T_d}{T_i}$  : le coefficient théorique d'interaction entre action intégrale et action dérivée.

3. **Structure mixte** : Actuellement, c'est la structure la plus couramment utilisée par les fabricants. L'expression de la sortie du régulateur PID est :

$$Y(t) = K_p * e(t) + \frac{K_p}{T_i} \cdot \int_0^t e(\tau) \cdot d\tau + K_p \cdot T_d \cdot \frac{de(t)}{dt} \quad (2.6)$$

Dans notre étude on a utilisé la structure parallèle.

Nous avons utilisé trois contrôleurs PID pour les angles de roulis ( $\phi$ ), de tangage ( $\theta$ ) et de lacet ( $\psi$ ). Les commandes  $U_2$ ,  $U_3$  et  $U_4$  sont donc calculées comme suit :

$$\begin{cases} K_p(\phi_d - \phi) + K_d(\dot{\phi}_d - \dot{\phi}) + \int_0^t K_i(\phi_d - \phi)dt \\ K_p(\theta_d - \theta) + K_d(\dot{\theta}_d - \dot{\theta}) + \int_0^t K_i(\theta_d - \theta)dt \\ K_p(\psi_d - \psi) + K_d(\dot{\psi}_d - \dot{\psi}) + \int_0^t K_i(\psi_d - \psi)dt \end{cases} \quad (2.7)$$

### 2.3.2 Choix des paramètres du régulateur

Pour réguler un système, le choix des paramètres est crucial. Ces paramètres déterminent la façon dont le système réagit aux entrées et aux perturbations. Un choix appropriée garantit une réponse rapide, stable et précise du système. Il existe plusieurs méthodes pour définir les gains d'un régulateur PID, parmi lesquelles on trouve :

- le réglage manuel (essais et erreurs).
- la méthode de Ziegler-Nichols.
- la méthode de Cohen-Coon.

Dans notre étude, nous allons procédé au réglage manuel (méthode des essais et erreurs). Cette méthodes est basée sur l'essaie. Dans le premier lieu on choisit des gains d'une manière aléatoire, selon les résultats obtenus on choisit des autres gains jusqu'à ce que nous obtenions les résultats souhaités.

### 2.3.3 Schéma de régulation

En analysant le modèle d'état, nous pouvons constater que certains états réagissent de manière sensible à certaines commandes. Par exemple :  $\dot{x}_2$  dépend de  $U_2$ ,  $\dot{x}_4$  dépend de  $U_3$ , et  $\dot{x}_6$  dépend de  $U_4$ , tandis que  $\dot{x}_8$ ,  $\dot{x}_{10}$ , et  $\dot{x}_{12}$  dépendent tous de  $U_1$ . De plus, nous savons que  $\dot{x}_2$ ,  $\dot{x}_4$ , et  $\dot{x}_6$  représentent respectivement  $\phi$ ,  $\theta$ , et  $\psi$ , et que  $\dot{x}_{12}$  représente  $z$ . Par ailleurs, il est établi que :

- $U_1$  représente la commande des variables de position X,Y,Z.
- $U_2$  représente la commande de la variables de rotation  $\phi$  .
- $U_3$  représente la commande de la variables de rotation  $\theta$  .
- $U_4$  représente la commande de la variables de rotation  $\psi$  .

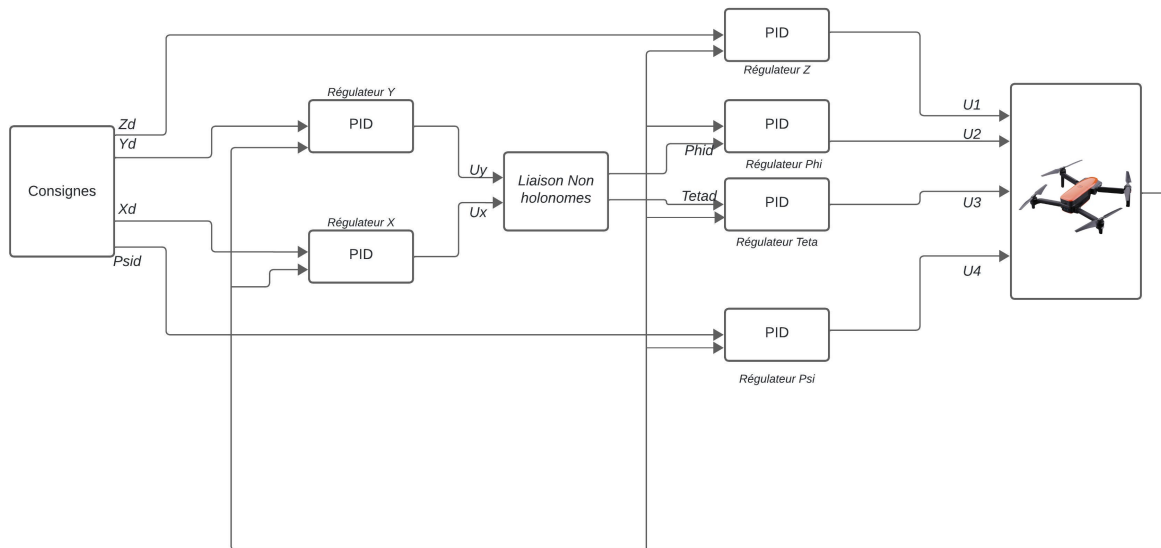
D'après ces informations on déduit que nous avons deux boucles a réguler ; la première

pour commander les variables  $\mathbf{Z}$  et  $\mathbf{Psi}$ , la deuxième pour les variables  $\mathbf{X}$  et  $\mathbf{Y}$  à l'aide des angles  $\mathbf{Theta}$ , et  $\mathbf{Phi}$ .

À partir de l'équation (1.40) du chapitre 1, on obtient :

$$\begin{cases} \phi_d = \arcsin(U_x \sin(\psi_d) - U_y \cos(\psi_d)) \\ \theta_d = \arcsin\left(\frac{U_x \cos(\psi_d) + U_y \sin(\psi_d)}{\cos(\phi_d)}\right) \end{cases} \quad (2.8)$$

Le schéma de régulation du quadrirotor par PID est représenté par la figure (2.1) :



**Figure 2.2:** Schéma de régulation PID de rotation et de position

Un premier test de poursuite de trajectoire sur le système en boucle fermée a été effectué en adoptant le réglage par essais-erreur des gains de PID. Les résultats de ce test sont inclus dans le tableau (2.1), qui présente les gains et les performances du système obtenus.

D'après ce tableau, on remarque que le choix des paramètres des gains du PID par essais-erreur nous offre de bonnes performances. Cependant, cette méthode demande beaucoup de temps et comporte un risque de dommages au système à cause des dépassements élevés. Afin d'optimiser les gains des PID et avoir de meilleures performances du système, nous procéderons par la suite à une méthode d'optimisation métaheuristique dite PSO (Particule Swarm Optimisation, Optimisation par essais particulaires).

Régulateur PID	Paramètres			Performances		
	$K_p$	$K_i$	$K_d$	Temps de réponse (s)	Dépassement (%)	Erreur statique
Régulateur Phi	5	0.1	1	0.5	14	0
Régulateur Téta	5	0.1	1	0.5	6	0
Régulateur Psi	6	0.1	1	0.16	35	0
Régulateur Z	50.43	40	10	1.5	13.3	0
Régulateur X	2.26	0.1	3.26	2.5	1	0.005
Régulateur Y	3.26	0.1	2.26	2.5	1	0.005

Tableau 2.1: Paramètres et performances du réglage par PID du quadrirotor

## 2.4 Optimisation de la commande linéaire par PSO

### 2.4.1 Introduction

L'optimisation par essaim particulaire (PSO) est un algorithme évolutionnaire qui utilise une population de solutions candidates pour trouver une solution optimale. Proposé par Russel Eberhart et James Kennedy en 1995 [12], il s'inspire du comportement social des animaux en essaim, comme les poissons et les oiseaux. Chaque particule, représentant une solution, a une position et une vitesse, et se souvient de sa meilleure performance ainsi que de celle des particules voisines. Cette approche permet d'explorer efficacement l'espace des solutions. [5]

### 2.4.2 Problème d'optimisation

Un problème d'optimisation consiste à rechercher l'optimum, c'est-à-dire le minimum ou le maximum, d'une fonction donnée. Les variables de cette fonction sont souvent contraintes à évoluer dans une région spécifique de l'espace de recherche, définissant ainsi un problème d'optimisation sous contraintes [5].

Mathématiquement, on cherche à minimiser la fonction objectif  $f$  sur  $E$ , c'est-à-dire, on cherche  $x \in E$  tel que :

$$f(x)^* = \min f(x) \quad (2.9)$$

### 2.4.3 Fonction objectif

L'algorithme d'optimisation doit optimiser cette fonction pour améliorer les performances de la boucle de régulation, telles que la précision dynamique, le dépassement, l'erreur statique, etc. Il existe plusieurs critères pour choisir cette fonction parmi lesquels on cite :

### 2.4.3.1 Critère de performance de la précision dynamique

Pour obtenir une bonne précision dynamique d'un système soumis à une entrée échelon, le régime transitoire doit présenter un faible dépassement et un temps de réponse optimal. Les paramètres du régulateur sont donc choisis pour minimiser l'erreur dynamique.

$e(t) = y(t) - y_d(t)$ . Il est nécessaire de minimiser l'un des critères suivants :

#### - Intégrale de la valeur absolue de l'erreur (IAE)

L'intégrale de la valeur absolue de l'erreur se définit par :

$$\text{IAE} = \int_0^{\infty} |e(t)| dt \quad (2.10)$$

Ce critère représente la surface créée par la différence entre la valeur de consigne et la valeur réelle, et il sert à éliminer les petites erreurs.

#### - Intégrale du carré de l'erreur (ISE)

L'intégrale du carré de l'erreur est définie par :

$$\text{ISE} = \int_0^{\infty} e(t)^2 dt \quad (2.11)$$

Cet indice de performance vise à rectifier les systèmes dont le régime transitoire s'étend excessivement, tout en accordant moins d'importance aux dépassements inférieurs à 1.

#### - Intégrale de la valeur absolue de l'erreur pondérée par le temps (ITAE)

L'intégrale de la valeur absolue de l'erreur pondérée par le temps est définie par :

$$\text{ITAE} = \int_0^{\infty} t|e(t)| dt \quad (2.12)$$

L'ajout du paramètre temps vise à stabiliser les systèmes présentant une réponse excessivement oscillatoire.

#### - Intégrale du carré de l'erreur pondérée par le temps (ITSE)

L'intégrale du carré de l'erreur pondérée par le temps se définit par :

$$\text{ITSE} = \int_0^{\infty} t \cdot e(t)^2, dt \quad (2.13)$$

Ce critère accorde peu d'importance aux erreurs initiales et pénalise fortement les erreurs survenant vers la fin de la réponse transitoire à une entrée échelon.

### 2.4.4 Principe de fonctionnement de PSO

Dans cet algorithme, les solutions candidates, appelées particules, évoluent simultanément en partageant des connaissances avec leurs voisines. Chaque particule génère une solution en utilisant son vecteur vitesse pendant son déplacement dans l'espace de recherche. Pour améliorer sa position, chaque particule ajuste sa vitesse en se basant sur sa propre expérience (la meilleure position trouvée précédemment) et sur l'expérience des autres particules (la meilleure solution trouvée par l'essaim).

Le déplacement d'une particule est influencé par trois composantes [9] [6] :

Composante d'inertie : la particule tend à suivre sa direction actuelle.

Composante cognitive : la particule s'appuie sur sa propre expérience et se dirige vers sa meilleure position antérieure.

Composante sociale : la particule se fie à l'expérience de l'essaim et se dirige vers la meilleure position collective.

Dans un espace de recherche de dimension  $D$ , la particule  $i$  de l'essaim est modélisée par son vecteur position  $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})^T$  et son vecteur vitesse  $\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})^T$ , la qualité de sa position est déterminée par la valeur de la fonction objective en ce point. Cette particule mémorise la meilleure position atteinte notée  $\mathbf{p}_i = (p_{i1}, p_{i2}, \dots, p_{iD})^T$  la meilleur position atteinte par ses voisines est notée  $\mathbf{g}_i = (g_{i1}, g_{i2}, \dots, g_{iD})^T$  Le principe de la méthode d'essaim de particules est illustré par la figure suivante : [5]

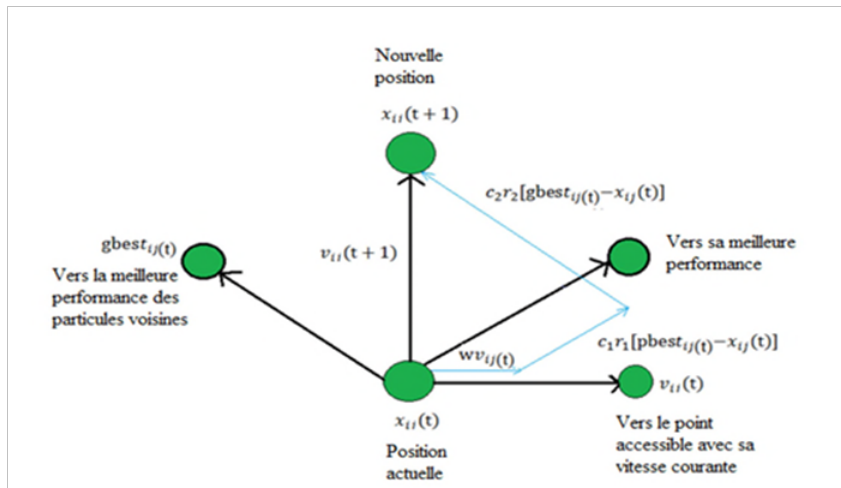


Figure 2.3: Déplacement d'une particule [5].

Le vecteur vitesse est déterminé à partir de l'équation suivante :

$$v_{ij}(k) = \omega v_{ij}(k-1) + c_1 r_1 (p_{ij}(k-1) - x_{ij}(k-1)) + c_2 r_2 (g_{ij}(k-1) - x_{ij}(k-1)) \quad (2.14)$$

La position de la particule  $i$  à l'itération  $k$  est définie par l'équation suivante :

$$x_{ij}(k) = x_{ij}(k-1) + v_{ij}(k) \quad (2.15)$$

$$i = 1, 2, 3, \dots, N_p; j = 1, 2, 3, \dots, N_d; k = 1, 2, 3, \dots, k_{max}$$

Avec :

- $N_p$  représente le nombre de particules dans l'essaim.
- $N_d$  représente le nombre de variables dans l'essaim.
- $k_{max}$  représente le nombre d'itérations maximales.
- $P_{ij}$  est la  $j^{\text{ième}}$  composante de la meilleure position occupée par la  $i^{\text{ième}}$  particule de l'essaim enregistrée dans les itérations précédentes (local best).
- $G_j$  est la  $j^{\text{ième}}$  composante de la meilleure position occupée globalement par l'essaim (global best).
- $x_{ij}(k)$  est la  $j^{\text{ième}}$  coordonnée de la position actuelle de la particule  $i$  à la  $k^{\text{ième}}$  itération.
- $\omega$  est généralement une constante appelée coefficient d'inertie.
- $c_1$  et  $c_2$  sont deux constantes appelées coefficients d'accélération.
- $r_1$  et  $r_2$  sont deux nombres aléatoires tirés uniformément dans  $[0, 1]$  à chaque itération et pour chaque dimension.
- $\omega v_{ij}(k-1)$  correspond à la composante physique du déplacement. Le paramètre  $\omega$  contrôle l'influence de la direction de déplacement futur. Il est à noter que dans certaines applications, le paramètre  $\omega$  peut être variable.
- $c_1 r_1 (p_{ij} - x_{ij}(k-1))$  correspond à la composante cognitive du déplacement où  $c_1$  contrôle le comportement cognitif de la particule.
- $c_2 r_2 (g_j - x_{ij}(k-1))$  correspond à la composante sociale du déplacement où  $c_2$  contrôle l'amplitude sociale de la particule.
- Les gammes appropriées de valeurs pour  $c_1$  et  $c_2$  sont de 1 à 2, bien que 2 soit souvent le plus approprié dans de nombreux cas. [7]

Le coefficient d'inertie peut être variable selon [6] :

$$w = w_{\max} - \left( \frac{w_{\max} - w_{\min}}{k_{\max}} \right) \times k \quad (2.16)$$

Le terme  $k_{\max}$  désigne le nombre maximal d'itérations, tandis que  $k$  représente l'itération actuelle.  $w_{\min}$  et  $w_{\max}$  correspondent respectivement aux valeurs minimale et maximale du coefficient d'inertie  $w$ . Ce coefficient est crucial dans le processus de recherche car il équilibre les aspects locaux et globaux de celle-ci. Un choix judicieux de ce coefficient améliore l'efficacité de la méthode pour obtenir une solution optimale. Les expériences ont démontré qu'une réduction linéaire de la valeur de  $w$  de 0.4 à 0.9 au cours de la recherche conduit à de meilleurs résultats [7].

### 2.4.5 Mécanisme de confinement

Il arrive parfois qu'une particule tende à sortir de l'espace de recherche en raison de sa position et de sa vitesse actuelles. Dans ce cas, l'algorithme utilise un mécanisme de confinement, appelé confinement d'intervalle, pour gérer le déplacement de la particule. Ce mécanisme ramène la particule à un nouveau point appartenant à l'espace de recherche, en la repositionnant généralement au point admissible le plus proche [11] [18].

$$\left\{ \begin{array}{l} \text{si } x_{ij} > x_{j_{\max}}, \text{ alors } x_{ij} = x_{j_{\max}} \\ \text{si } x_{ij} < x_{j_{\min}}, \text{ alors } x_{ij} = x_{j_{\min}} \\ V_{ij} = 0 \end{array} \right. \quad (2.17)$$

Ou :  $x_{ij_{\min}}$  et  $x_{ij_{\max}}$  sont les valeurs limites du paramètre  $x_{ij}$ .

### 2.4.6 Optimisation des paramètres de régulateur PID par PSO

Afin de trouver des gains optimaux du régulateur, on prend la boucle de régulation classique on ajoute une couche supérieur a base d'un PSO qui va fournir les paramètres du PID souhaités pour améliorer les performances du système. Le schéma suivant explique l'approche d'optimisation.

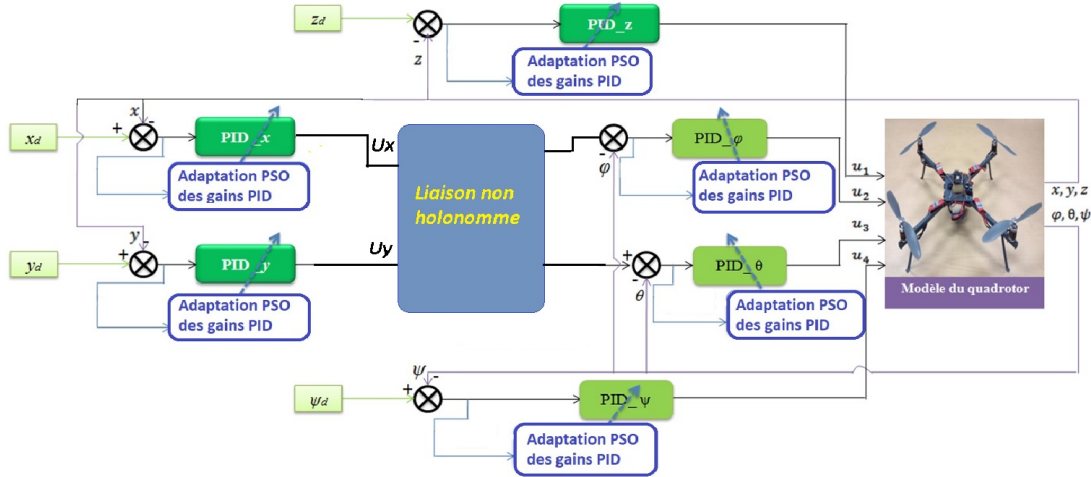


Figure 2.4: Schéma d'approche d'optimisation

### 2.4.7 Implémentation de l'algorithme dans le système

En appliquant plusieurs tests de PSO avec utilisation de différentes fonctions objectives nous avons choisi la fonction intégrale de carré de l'erreur ISE en raison de sa capacité à minimiser efficacement les grandes erreurs, ce qui est important pour la stabilité et la précision d'un quadrirotor. Le tableau suivant donne les valeurs des paramètres PSO utilisées lors de la simulation :

Paramètre	Nbr de Particule	Nbr d'itération	Dimension Problème	$c_1$	$c_2$	$w$
Valeur	10	5	3	2	2	0.7

Tableau 2.2: Paramètres et valeurs utilisées dans l'algorithme

## 2.5 Résultats de simulations

Afin de vérifier les performances des régulateurs PID synthétisés nous avons considéré le schéma de régulation présenté à la figure (2.4) sur lequel nous avons effectué plusieurs tests sous des conditions de simulation différées.

### 2.5.1 Test de poursuite de trajectoire

Dans ce test, on s'intéresse au problème de la poursuite de trajectoire. Le drone est tenue a effectuer un mouvement verticale en début de simulation puis il devait suivre une trajectoire désirée suivant les axes x,y a l'instant  $t=20s$  et  $t=30s$ .

Le tableau (2.3) résume les paramètres de régulateurs PID optimisés ainsi que les performances du système trouvées.

Régulateur PID optimisé	Paramètres			Performances		
	$K_p$	$K_i$	$K_d$	Temps de réponse (s)	Dépassement (%)	Erreur statique
Régulateur Phi	12.78	0.1	10	0.3	22	0
Régulateur Téta	12.24	0.1	10	0.3	6	0
Régulateur Psi	13.68	0.1	10	0.10	35	0
Régulateur Z	50.43	40	10	1	12.5	0
Régulateur X	14.2	0.12	3.26	1.9	0.9	0.003
Régulateur Y	13.7	0.12	2.26	1.9	0.9	0.003

**Tableau 2.3:** Paramètres et performances du réglage par PID optimisé du quadrirotor

Les figures suivantes représentent les différentes réponses des états de système ainsi que les commandes correspondantes.

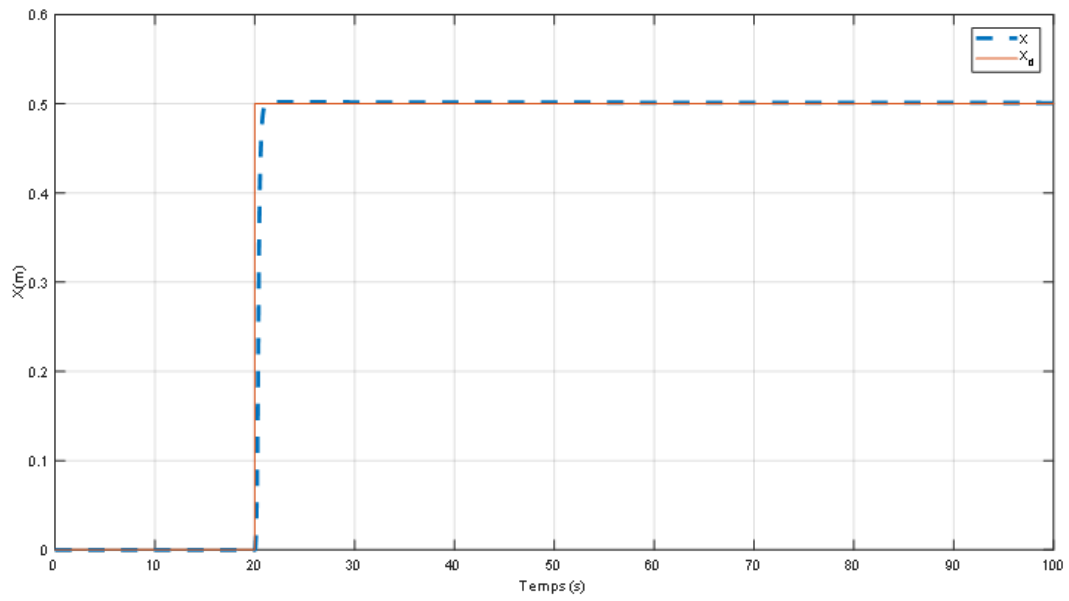


Figure 2.5: Réponse en X

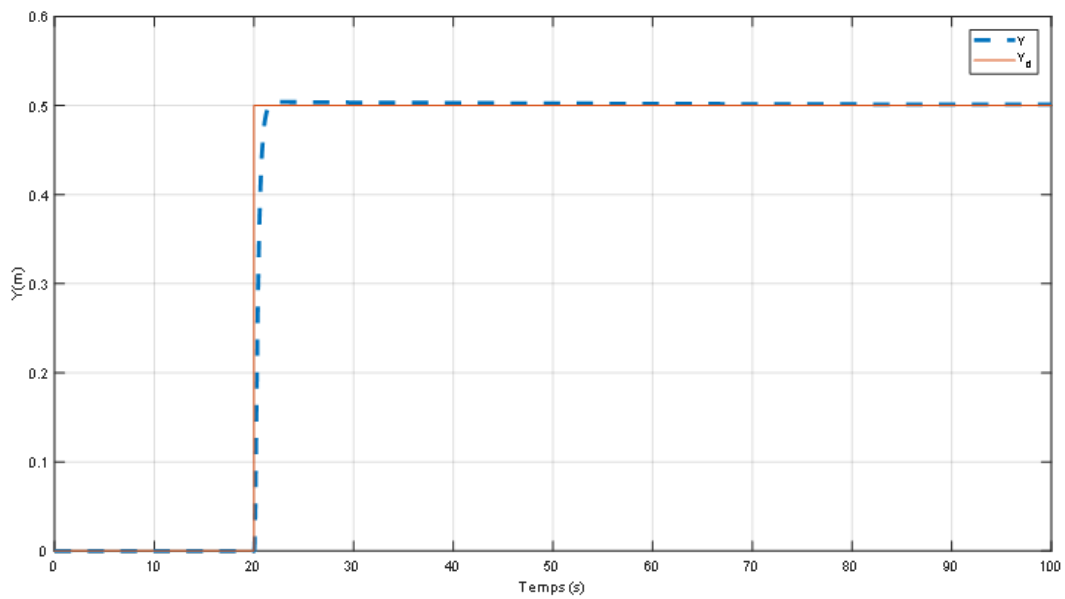


Figure 2.6: Réponse en Y.

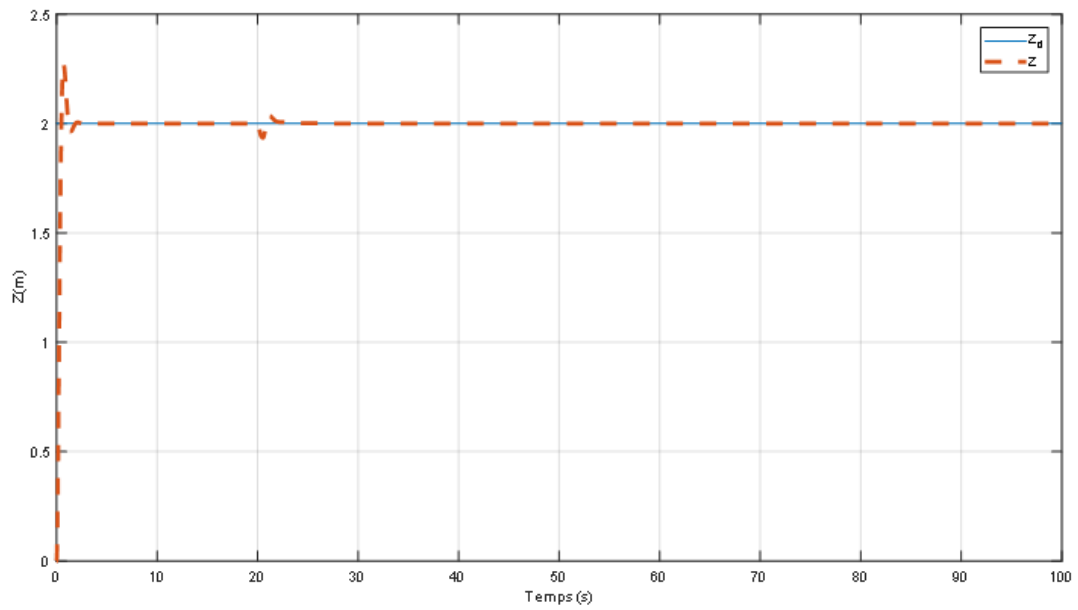


Figure 2.7: Réponse en  $Z$ .

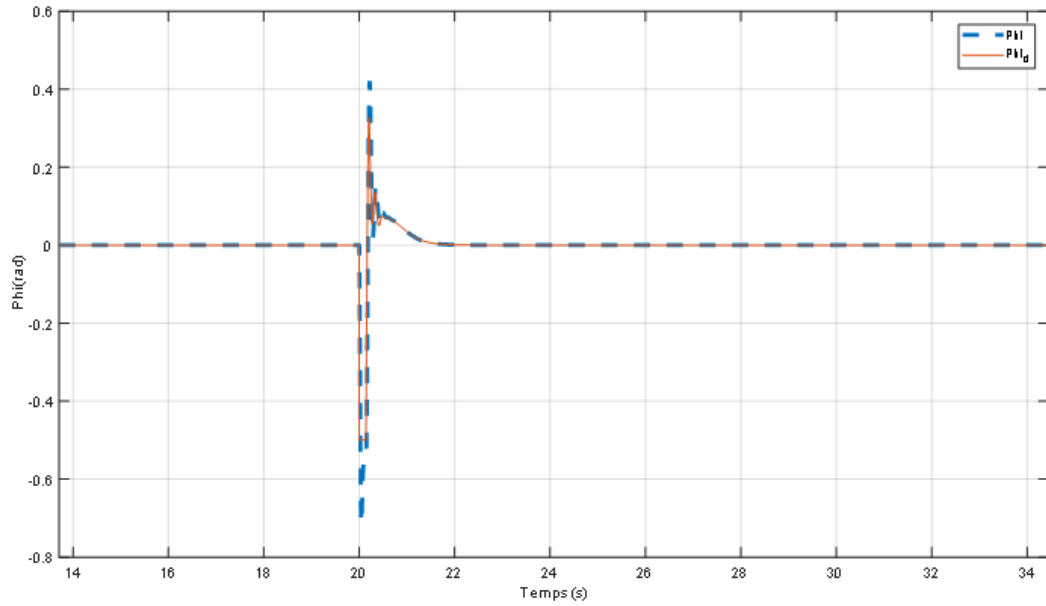


Figure 2.8: Réponse en angle  $\Phi$ .

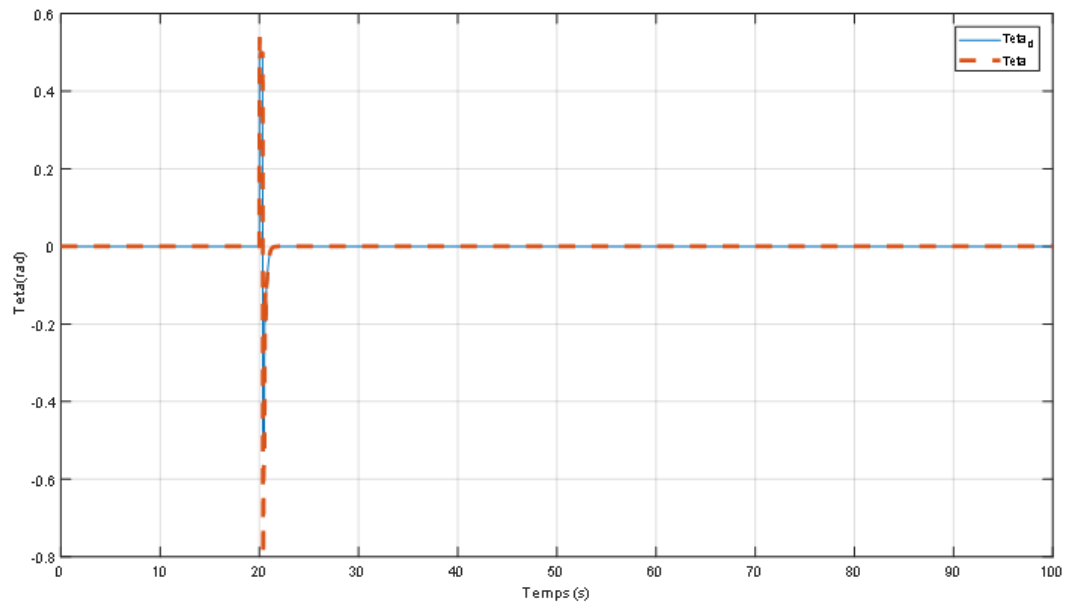


Figure 2.9: Réponse en angle Teta.

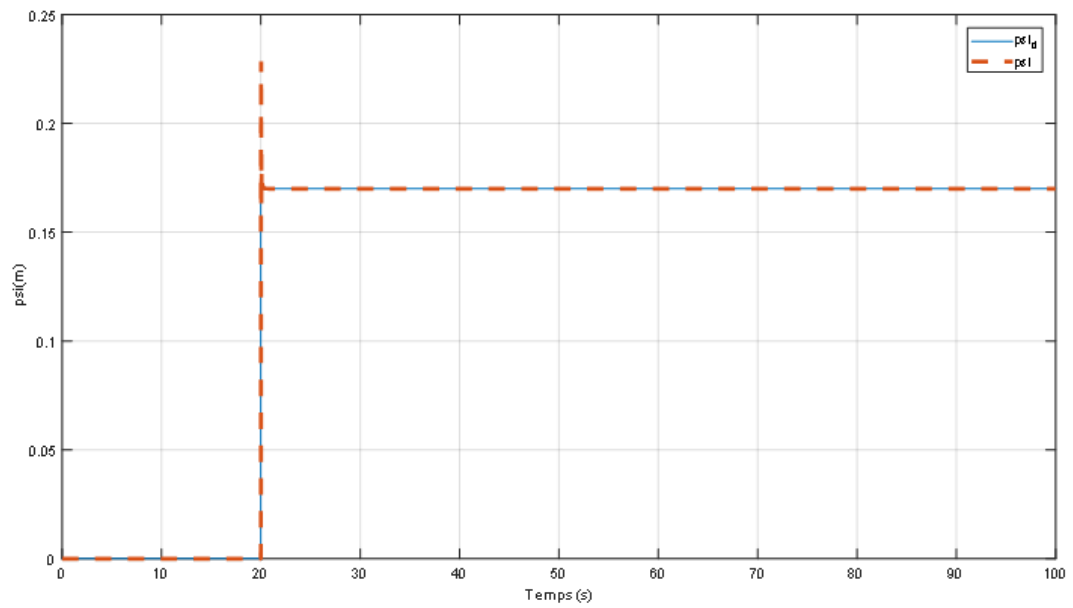


Figure 2.10: Réponse en angle Psi.

## 2.5.2 Tests de robustesse

Dans cette partie, nous allons évaluer la validité de nos commandes à travers des tests de robustesse face aux perturbations extérieures. Les perturbations prises en compte dans cette études sont : [8]

- L'effet du vent.
- La variation des paramètres.
- Le bruit de mesure.

### a) Effet du vent

Il est connu que la pression du vent  $P$  peut être approximée par :

$$P = \frac{1}{2}\rho v^2 \quad (2.18)$$

Où :

$\rho$  représente la masse volumique de l'air ( $\rho = 1.25 \text{ kg/m}^3$ ).

$v_w$  représente la vitesse du vent ( $v_w = 3.3 \text{ m/s}$ ).

$\zeta$  est un facteur de forme ( $\zeta = 1$ ).

Ces valeurs vont produire une pression  $P_w = 6.8 \text{ N/m}^2$ . Dans le but de convertir cette pression en une force, la surface d'influence doit être trouvée. Cette surface a été calculée [23]  $S = 0.02 \text{ m}^2$ . La relation entre la force du vent  $F_w$  et la pression est :  $F_w = P_w S$ , ce qui nous donne  $F_w$  exercée sur le quadricoptère :  $F_w = 0.136 \text{ N}$ . Pour voir l'influence de cette force sur le comportement du système en présence du régulateur, on l'ajoute sur les deux axes  $x$  et  $y$  entre les instants 60s comme suit : [13]

$$\begin{cases} \dot{x}_7 = x_8 + F_w \\ \dot{x}_9 = x_{10} + F_w \end{cases} \quad (2.19)$$

Les figures suivantes présentent les résultats de simulation :

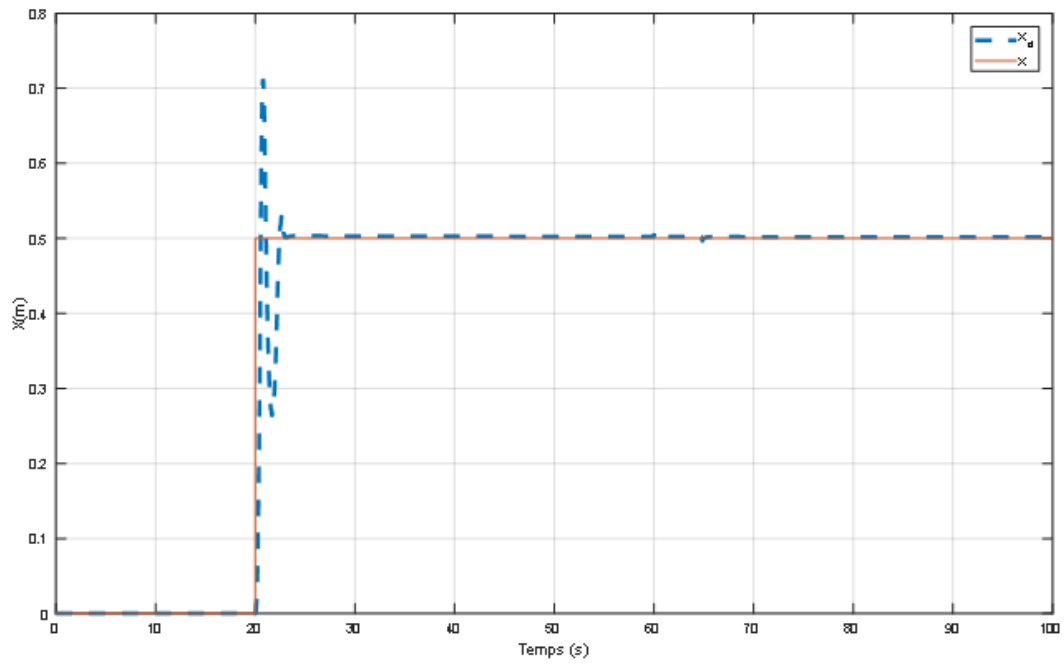


Figure 2.11: Effet du vent sur  $X$

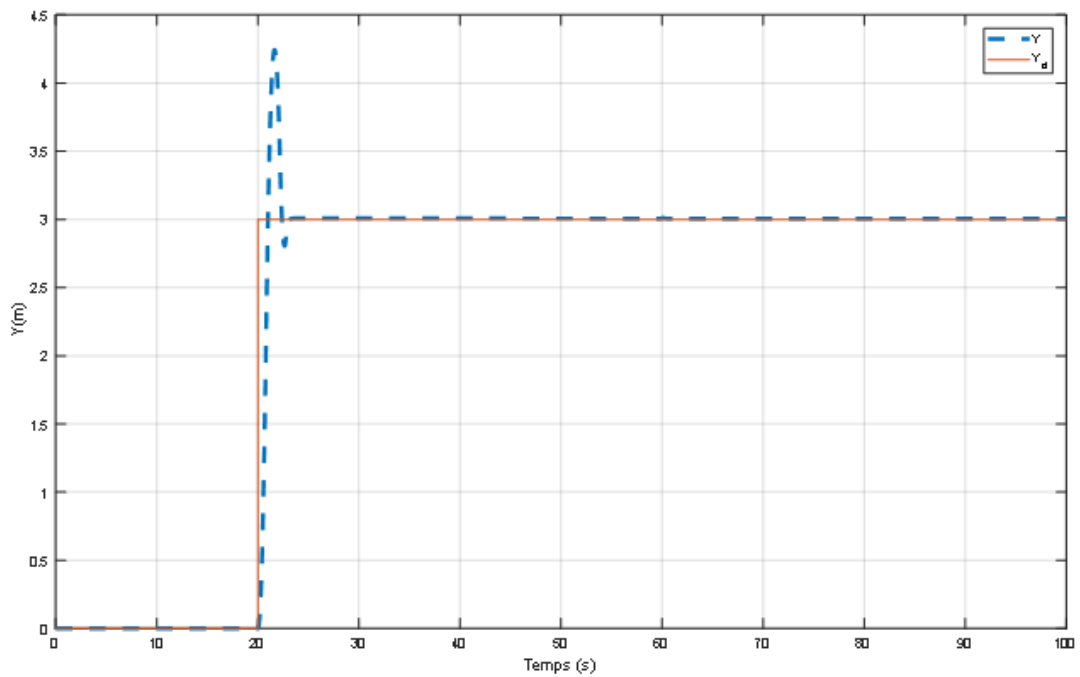


Figure 2.12: Effet du vent sur  $Y$

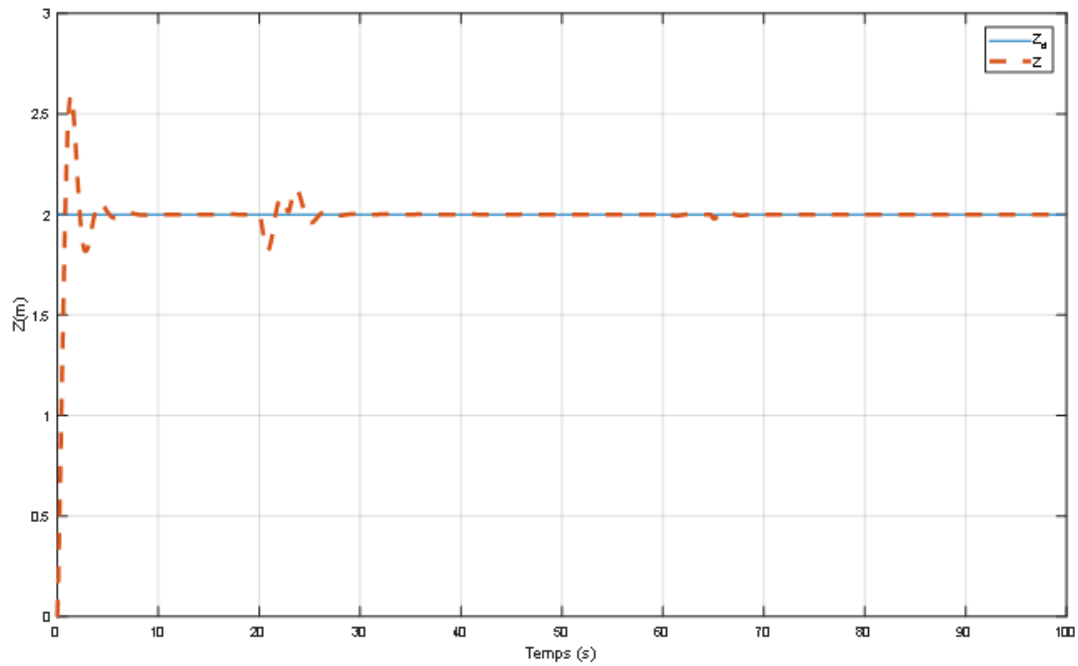


Figure 2.13: Effet du vent sur  $Z$

## b) Variation paramétrique

Quand un quadricoptère transporte un objet, tel qu'une caméra par exemple, entraînant ainsi une augmentation de sa masse initiale de 50%. La figure suivante présente le résultat de simulation [8] :

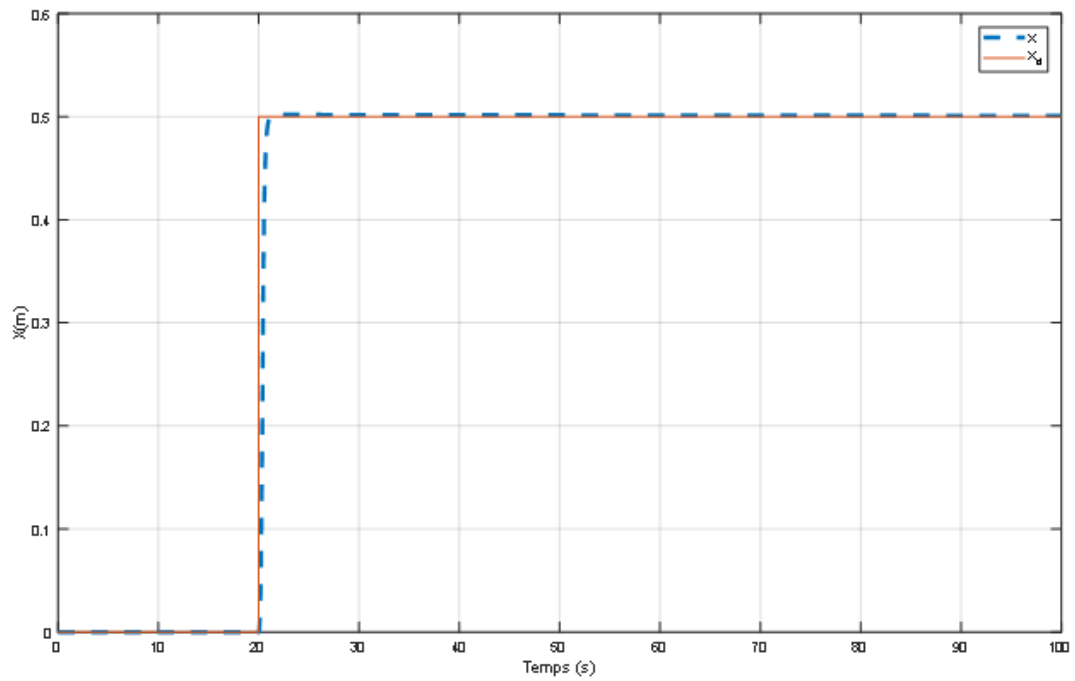


Figure 2.14: Effet de la masse sur X

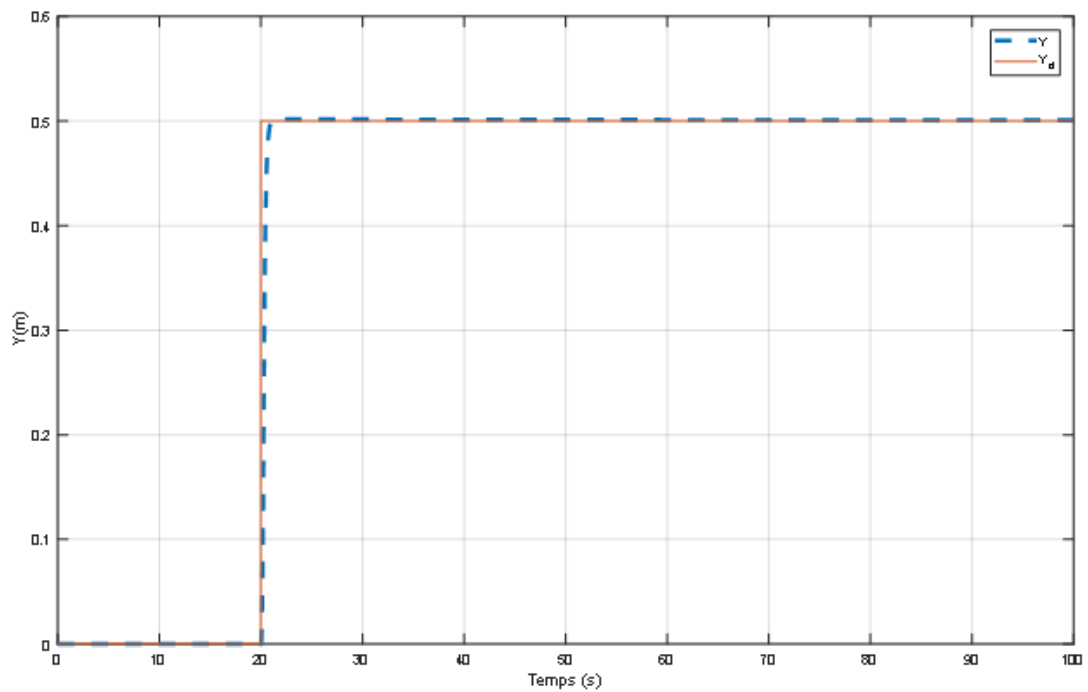


Figure 2.15: Effet de la masse sur Y

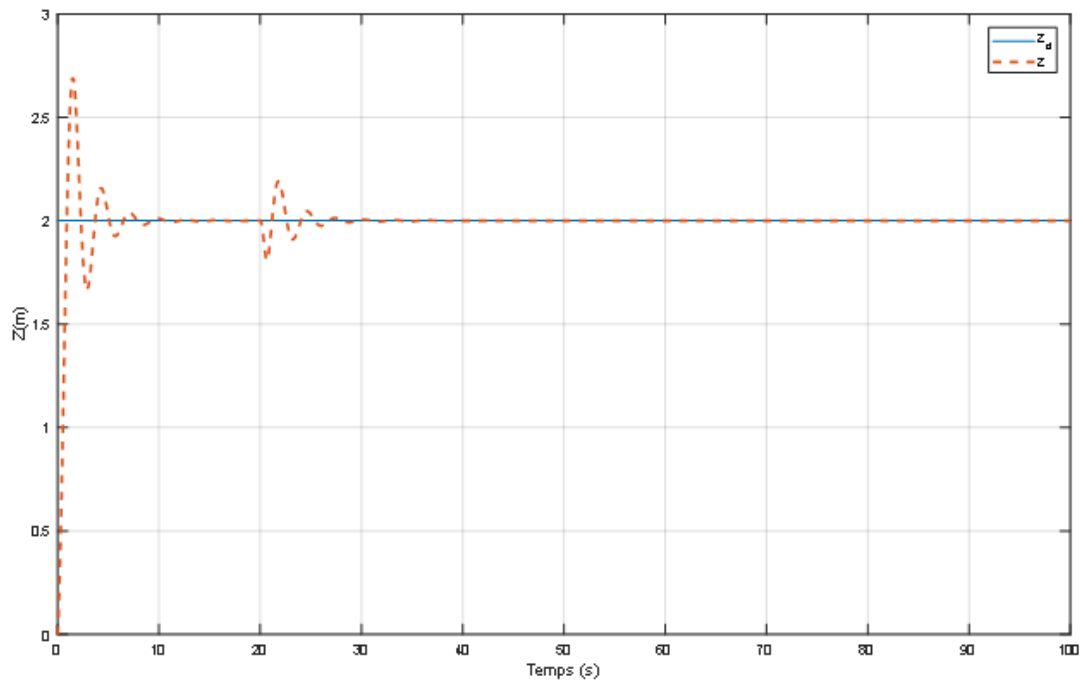


Figure 2.16: Effet de la masse sur  $Z$

### c) Bruit de mesure

Le bruit provient à la fois des imperfections des capteurs et des conditions ambiantes. Pour illustrer, considérons un bruit de 0.1 pour les coordonnées cartésiennes [8]. La figure suivante présente le résultat de simulation :

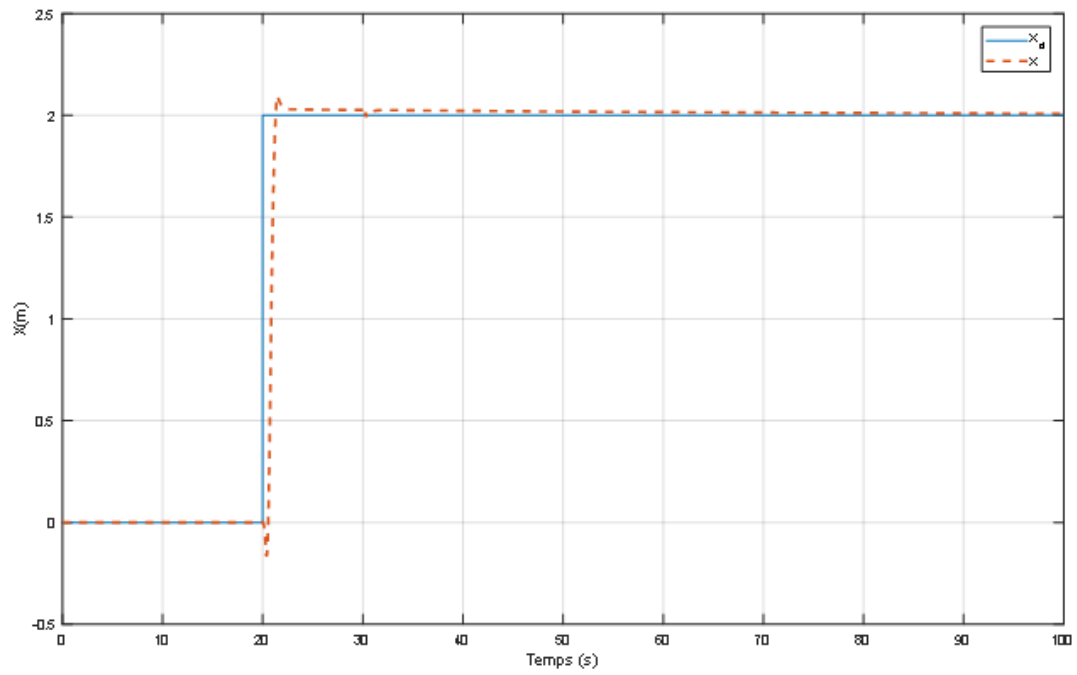


Figure 2.17: Effet de bruits externes sur  $X$

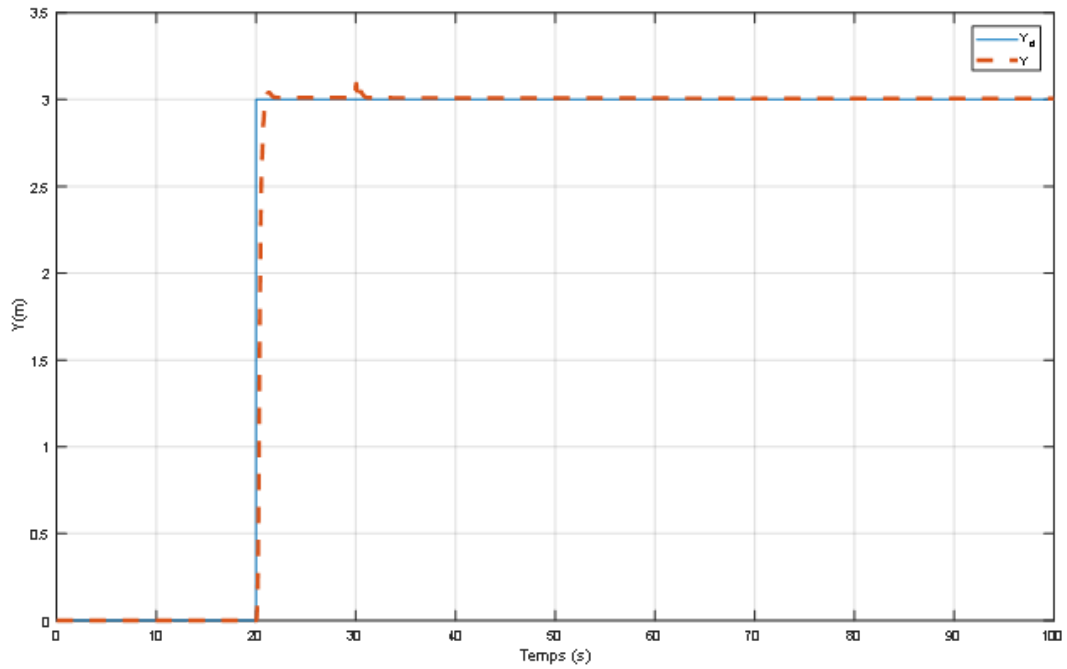


Figure 2.18: Effet de bruits externes sur  $Y$

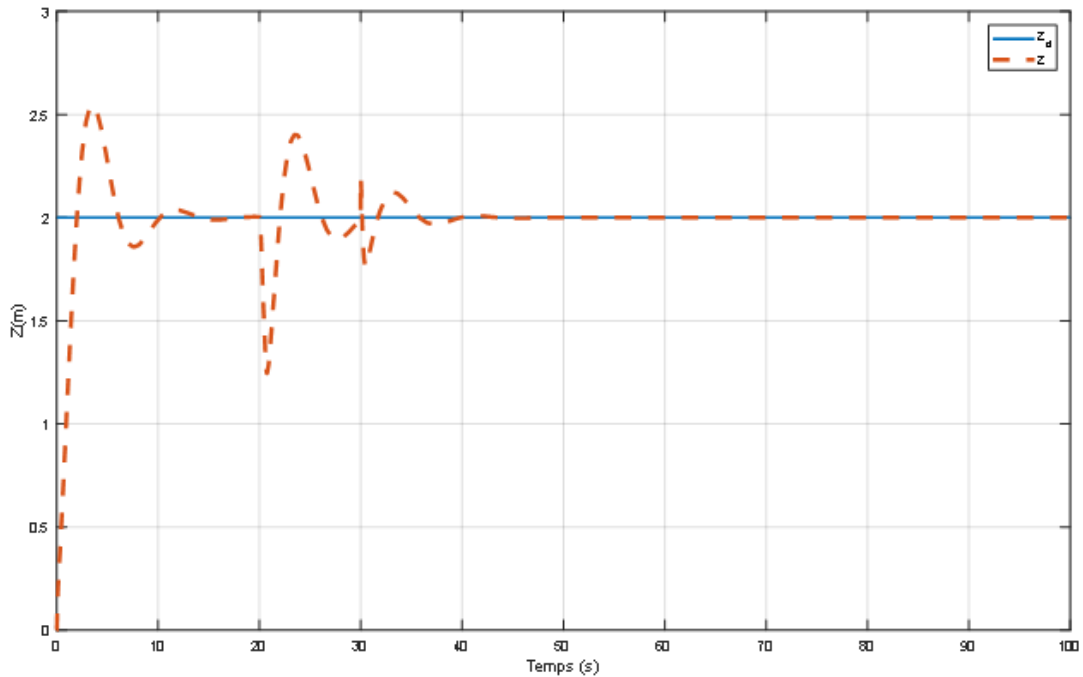


Figure 2.19: Effet de bruits externes sur  $Z$

Après avoir réalisé les tests de robustesse du quadrirotor (bruit de mesure, effet de vent et variation paramétrique) nous avons observé que le système a réussi à rejeter les perturbations et à revenir à son état de stabilité, avec un temps de réponse satisfaisant. Cependant, la variable  $Z$  présente un temps de réponse plus lent que souhaité.

## 2.6 Conclusion

Ce chapitre a examiné en profondeur la commande par PID (Proportionnel-Intégral-Dérivé) ainsi que son optimisation dans le contexte du contrôle de quadrirotors. Les résultats de simulation obtenus avec la commande PID ont démontré une stabilité, une précision satisfaisantes du système. Cependant, en comparaison, les résultats obtenus avec la commande PID optimisée ont surpassé ceux du PID classique, offrant une performance améliorée. Malgré cela, lors des tests de robustesse, le système n'a pas toujours donné des résultats satisfaisants en présence de fortes perturbations. Ces résultats soulignent que le PID peut ne pas être suffisamment flexible pour s'adapter à ces conditions. Pour obtenir de meilleurs résultats, nous allons tester une autre méthode de commande : la logique floue. Dans le prochain chapitre, nous explorerons plus en détail cette méthode et son application au contrôle des quadrirotors.

# Chapitre 3

## Commande par logique floue

### 3.1 Introduction

La modélisation et le contrôle des drones sont cruciaux pour garantir leur sécurité et leur performance. Alors que les approches traditionnelles reposent sur des modèles rigides, la logique floue offre une alternative flexible, capable de traiter les données imprécises et les environnements complexes. Dans ce chapitre nous examinerons les principes de la logique floue, son application dans la commande des drones.

### 3.2 Historique

La logique floue, introduite par L. Zadeh en 1965, a connu un grand succès en permettant la représentation et la manipulation de connaissances imparfaites via des sous-ensembles flous. En 1973, Zadeh a suggéré d'appliquer ces principes à la résolution des problèmes de réglage, ce que Mamdani a concrétisé en 1974 en créant un premier contrôleur flou pour une turbine à vapeur, marquant ainsi une des premières applications industrielles de la commande floue. En 1985, M. Sugeno a introduit la logique floue dans les applications industrielles au Japon, apportant un changement conceptuel significatif. En 1995, J. S. R. Jang a étendu la logique floue aux systèmes de réseaux de neurones et à l'intelligence artificielle, élargissant ainsi son domaine d'application [24].

### 3.3 Définition de la logique floue

La logique floue, créée par Lotfi Zadeh en 1965, étend la logique classique pour traiter des données imprécises et des incertitudes. À la différence de la logique booléenne, qui ne reconnaît que des valeurs vraies ou fausses (0 ou 1), la logique floue emploie des

degrés d'appartenance variant entre 0 et 1. Cette méthode flexible permet de modéliser des situations aux informations floues ou incomplètes, facilitant ainsi la prise de décisions dans des environnements complexes.

## 3.4 Principe de la logique floue

### 3.4.1 Ensemble flou

La notion d'ensemble flou introduit un degré d'appartenance progressif des éléments à un ensemble, permettant ainsi à un élément d'appartenir partiellement à une classe prédéfinie. Cela offre une meilleure représentation des termes et des connaissances vagues que nous utilisons quotidiennement [10].

Mathématiquement, un ensemble flou  $A$  dans un univers de discours  $U$  est défini par une fonction d'appartenance, notée  $\mu_A$ , qui prend des valeurs dans l'intervalle  $[0, 1]$ . Cette fonction attribue à chaque élément  $x$  de  $U$  un degré d'appartenance  $\mu_A(x)$ , indiquant le niveau auquel  $x$  appartient à  $A$ . Un  $\mu_A(x)$  de 1 signifie une appartenance complète, tandis qu'un  $\mu_A(x)$  de 0 signifie une non-appartenance totale.

### 3.4.2 Variable linguistique

Une variable linguistique est décrite par un triplet  $(V, X, T_v)$ , où  $V$  est la variable (comme la taille) définie sur un ensemble de référence  $X$ .  $T_v$  est un ensemble de sous-ensembles flous de  $X$  utilisés pour caractériser  $V$ , limitant ainsi les valeurs que  $V$  peut prendre dans  $X$ . Par exemple, une variable linguistique pour la taille pourrait être représentée par  $T$ ,  $[0m, 2m]$ , ("très petit", "petit", "moyen", "grand", "très grand"). Les termes "petit", "moyen", etc., sont appelés valeurs linguistiques ou termes flous. En général, les supports des sous-ensembles flous associés aux valeurs linguistiques se chevauchent et leur union couvre tout l'univers de discours [16].

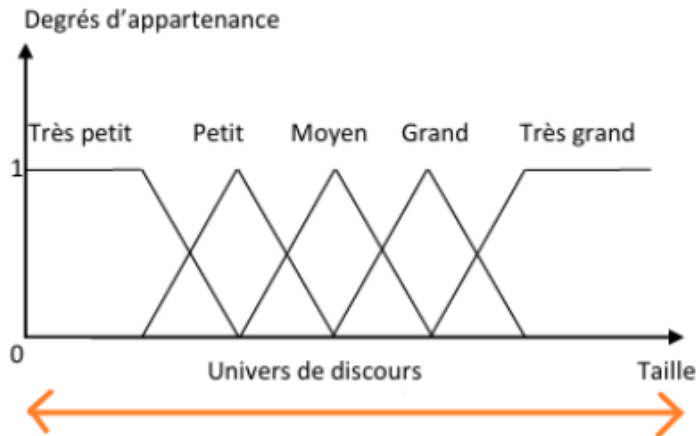


Figure 3.1: Variable Linguistique.

### 3.4.3 Fonction d'appartenance

Chaque ensemble flou peut être décrit par sa fonction d'appartenance. La forme de cette fonction, qui varie en fonction de l'application, peut être monotone, triangulaire, trapézoïdale ou en forme de cloche.

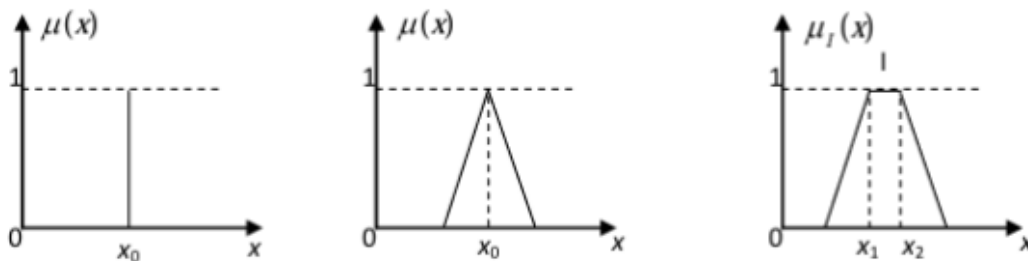


Figure 3.2: Fonction D'appartenance [16]

### 3.4.4 Opérateurs de la logique floue

Les opérateurs flous sont essentiels dans la logique floue, qui gère l'incertitude et l'imprécision en utilisant des degrés de vérité plutôt que des valeurs binaires. Les principaux opérateurs flous OU, ET et NON permettent de combiner et manipuler des ensembles flous.

**- Opérateur OU**

L'opérateur OU : représente l'union de deux ensembles flous  $A$  et  $B$  dans l'univers de discours  $X$ . L'union  $(A \cup B)$  est également un ensemble flou de  $X$ , avec une fonction d'appartenance définie par :

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (3.1)$$

Ici, «  $\cup$  » est l'opérateur maximum, similaire au OR en logique booléenne.

**- Opérateur ET**

L'opérateur ET représente l'intersection de deux ensembles flous  $A$  et  $B$  dans l'univers de discours  $X$ . L'intersection  $(A \cap B)$  est également un ensemble flou de  $X$ , avec une fonction d'appartenance définie par :

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (3.2)$$

Ici, «  $\cap$  » est l'opérateur minimum.

**- Opérateur NON**

Le complément d'un ensemble flou  $A$  dans l'univers de discours  $X$ , noté  $\bar{A}$ , a une fonction d'appartenance définie par [1] :

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (3.3)$$

Cela correspond à l'opérateur NOT en logique booléenne.

### 3.4.5 Règles d'inférence floue

La prise de décision repose sur les connaissances humaines ou expertes, formulées sous la forme de règles d'inférence floue, qui se présentent comme suit : "SI (prémisse) ALORS (conclusion)" ou "SI (prémisse) ET (prémisse) ALORS (conclusion)". Dans ces règles, les prémisses représentent les conditions d'entrée et la conclusion indique l'action à effectuer en sortie [21].

### 3.4.6 Structure d'un régulateur flou

La configuration classique d'un régulateur flou est illustrée dans la figure (3.3), conçue par Mamdani, et repose sur trois étapes essentielles : la fuzzification, l'inférence floue et la défuzzification.

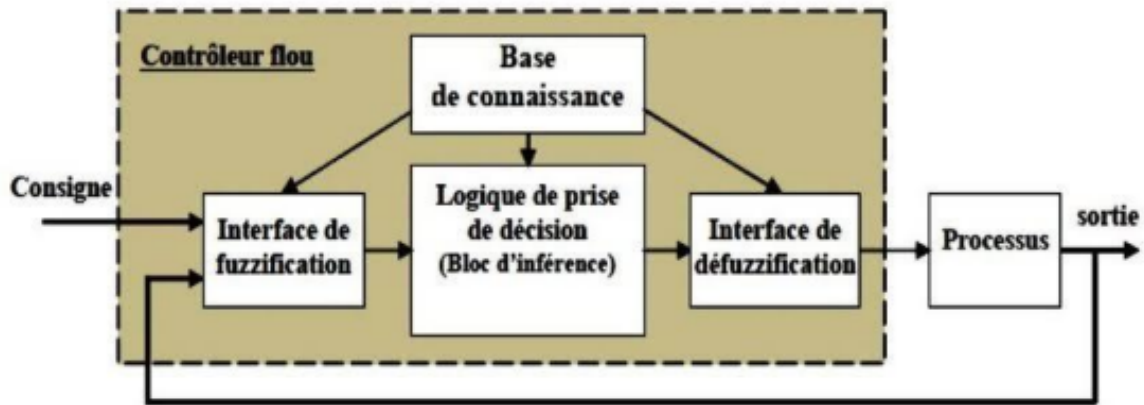


Figure 3.3: Structure d'un régulateur flou

### 3.4.6.1 Fuzzification

Le module de fuzzification vise à convertir les variables d'entrée en variables linguistiques en définissant des fonctions d'appartenance pour chaque variable.

### 3.4.6.2 Inférence et règles floues

L'opération d'inférence relie les valeurs des variables linguistiques d'entrée à celles des variables linguistiques de sortie à l'aide d'une table de règles, qui doivent prendre en compte le comportement statique et dynamique du système à contrôler. Ce mécanisme, en manipulant les règles floues, permet de prendre des décisions. En réglage par logique floue, on utilise généralement l'une des méthodes suivantes [19] :

- La méthode max-min.
- La méthode max-produit.
- La méthode somme-produit.

Cette inférence est alors appliquée aux différentes règles floues établies par le concepteur, afin de parvenir à une conclusion qui sera ensuite transmise au module de défuzzification [4].

### 3.4.6.3 Défuzzification

La défuzzification est le processus de déduction d'une valeur numérique exploitable à partir de la conclusion obtenue grâce aux règles floues. Pour cela, nous utilisons la méthode du centre de gravité ou méthode du centroid, donnée par la formule suivante [17] [21]

$$x_R = \frac{\int x \cdot \mu(x) dx}{\int \mu(x) dx} \quad (3.4)$$

### 3.5 Application de la commande floue sur le quadrirotor

Nous avons choisi d'utiliser la même structure de régulation que celle utilisée pour la commande par contrôleur PID. Ainsi, nous avons deux boucles de régulation simples pour contrôler les variables  $Z$  et  $\psi$ , et deux boucles de régulation doubles : l'une pour contrôler la position  $X$  à l'aide de l'angle  $\theta$ , et l'autre pour contrôler la position  $Y$  à l'aide de l'angle  $\phi$ . Le schéma est présenté comme suit :

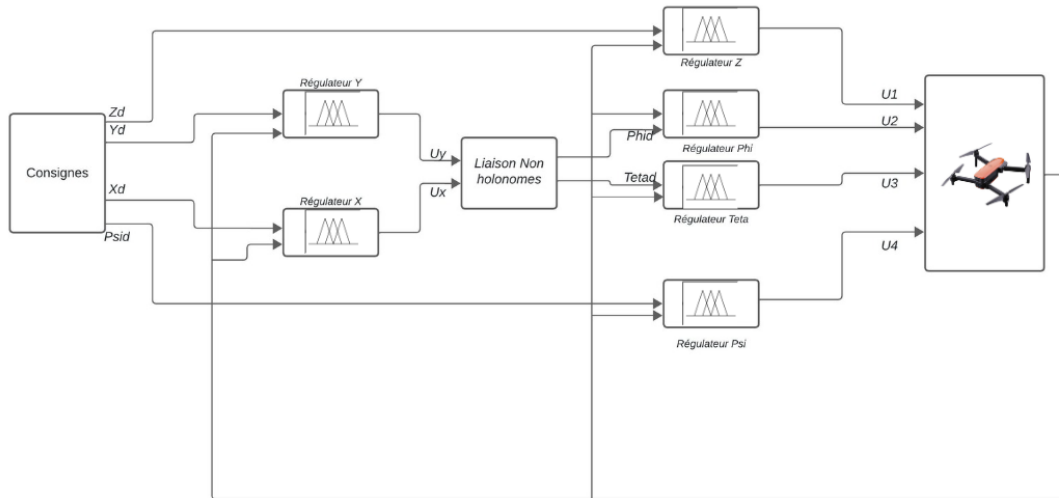


Figure 3.4: Commande Floue

Nous avons choisi d'utiliser trois fonctions d'appartenance triangulaires pour chaque boucle de régulation. Ces fonctions ont été choisies en raison de leur représentation pratique, de leur calcul facile et de leur réponse rapide. Chaque régulateur reçoit en entrée l'erreur de position et la vitesse de déplacement ou de rotation. Grâce à neuf règles d'inférence nous obtenons une sortie du régulateur. Cette sortie est ensuite défuzzifiée. La table des règles est la même pour tous les contrôleurs flous et elle est présentée comme suit :

$e/de$	N2	Z2	P2
N1	NG	N	Z
Z1	N	Z	P
P1	Z	P	PG

Avec : NG : Négative grand, N : Négative

Z : Zero

PG : Positive grand, P : Positive

Après plusieurs essais et ajustements, nous avons déterminé les fonctions d'appartenance pour les entrées et les sorties de chaque variables d'état de système.

**Pour les angles  $\phi, \theta, \psi$  :** nous avons utilisé les fonctions d'inférence comme indiqué dans les figures (3.5), (3.6) et (3.7).

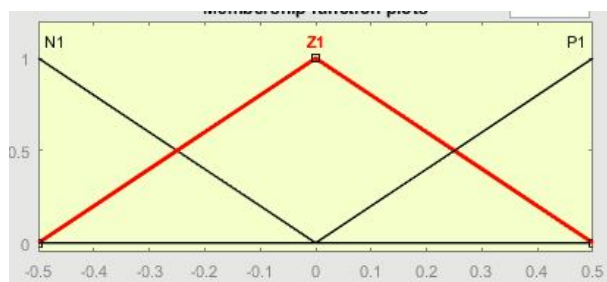


Figure 3.5: L'erreur de la position (entrée 1)

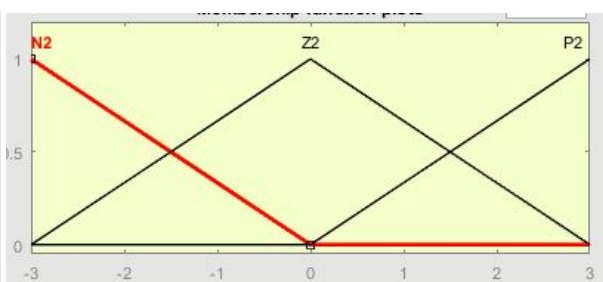


Figure 3.6: Dérivée de l'erreur (entrée 2)

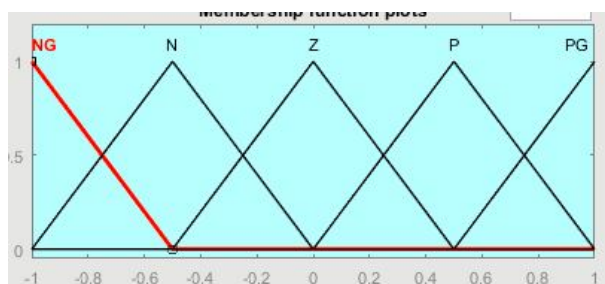


Figure 3.7: sortie de Régulateur

**Pour la variable  $Z$**  nous avons utilisé les fonctions d'inférence comme indiqué dans les figures (3.8), (3.9) et (3.10).

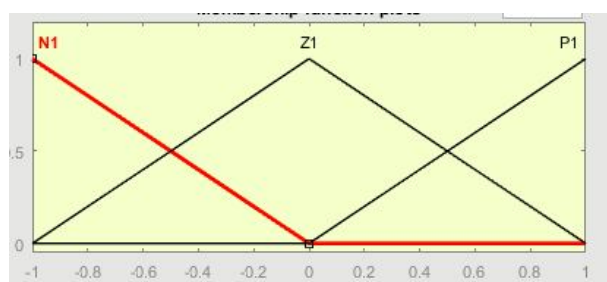


Figure 3.8: L'erreur de la position  $Z$  (entrée 1)

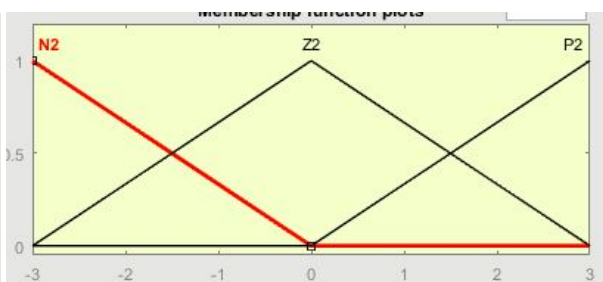


Figure 3.9: Dérivée de l'erreur (entrée 2)

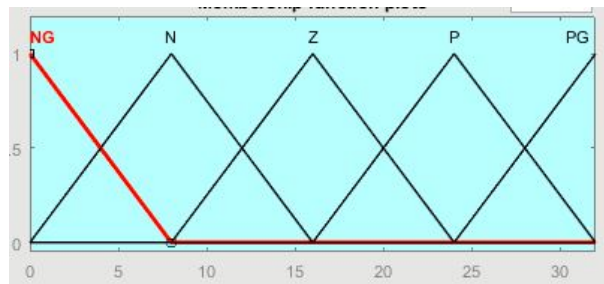


Figure 3.10: Sortie de régulateur

Pour les variables  $X$  et  $Y$  nous avons utilisé les fonctions d'inférence comme indiqué dans les figures (3.11), (3.12) et (3.13).

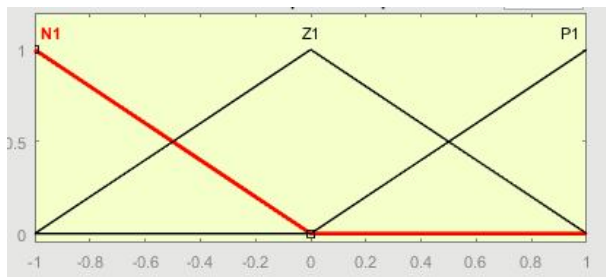


Figure 3.11: L'erreur de la position X,Y(entrée 1)

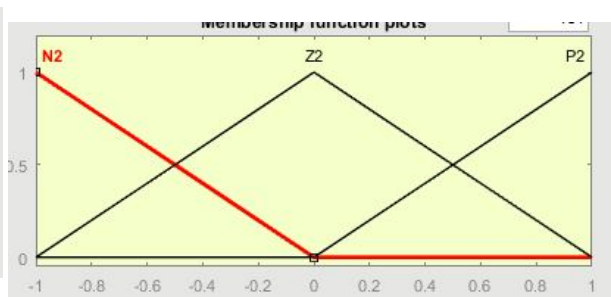


Figure 3.12: Dérivée de l'erreur (entrée 2)

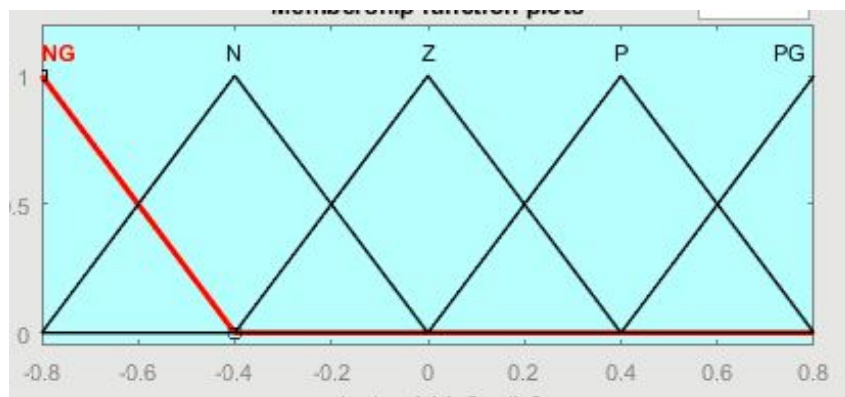


Figure 3.13: Sortie de régulateur

## 3.6 Résultats de simulation

### 3.6.1 Test de poursuite de trajectoire

Après avoir implanter la commande floue sur le quadrirotor, nous remarquons que les réponses en positions X, Y, et Z, ainsi qu'en angles Phi, Thêta, et Psi sont rapides avec une stabilisation efficace.

Cependant l'altitude Z présente des dépassements notables avant de se stabiliser à sa valeur désirée. Ainsi que le temps de réponse est important (par rapport au réglage PID). Globalement, le contrôleur flou a montré de bonnes performances, meilleures que le régulateur PID classique, surtout en régime dynamique, où le dépassement des réponses est réduit.

Les performances du système régulé par logique floue sont données dans le tableau (3.1).

Régulateur flou	Performances		
	Temps de réponse (s)	Dépassement (%)	Erreur statique
Réglateur Phi	0.9	13	0.08
Réglateur Téta	0.9	0.64	0.08
Réglateur Psi	0.4	2.94	0
Réglateur Z	4.1	40	0
Réglateur X	1.9	0	0
Réglateur Y	1.9	0	0

**Tableau 3.1:** Performances du quadrirotor avec réglage flou.

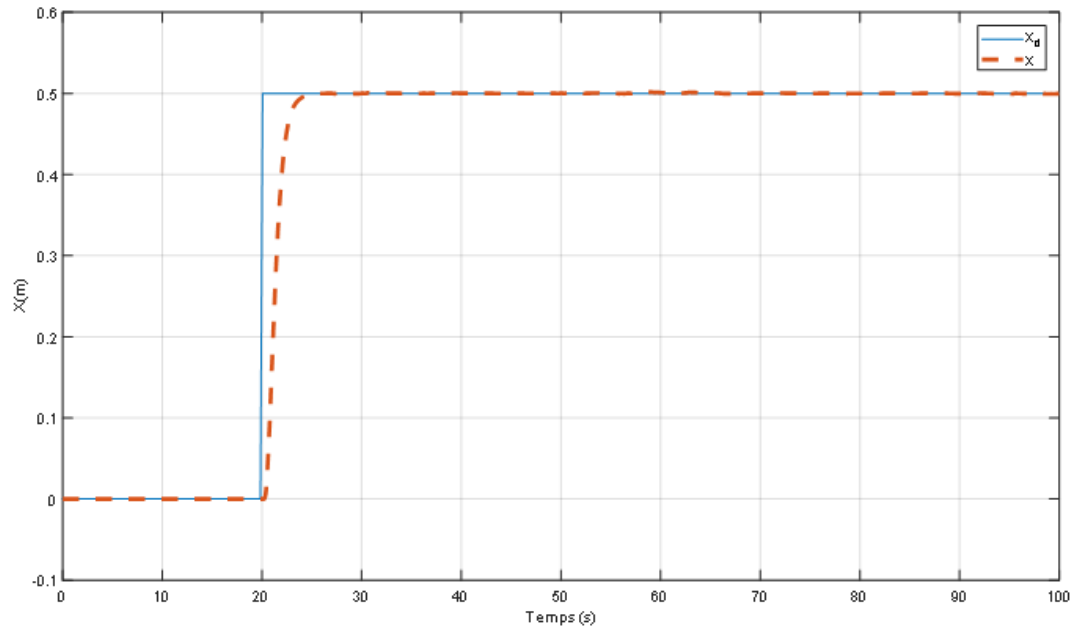


Figure 3.14: Réponse en X.

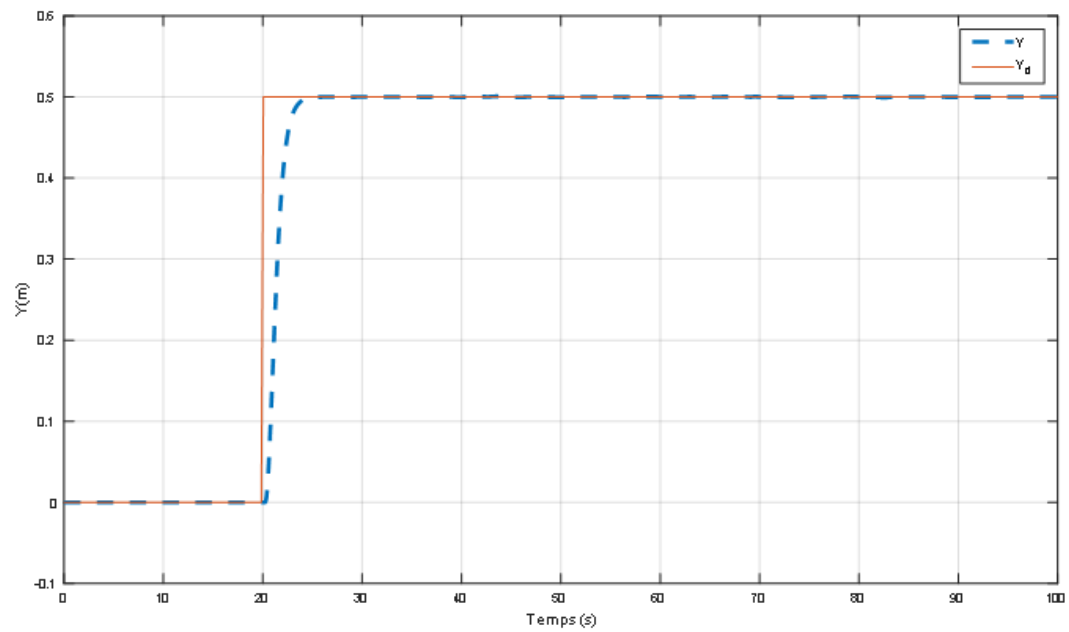


Figure 3.15: Réponse en Y.

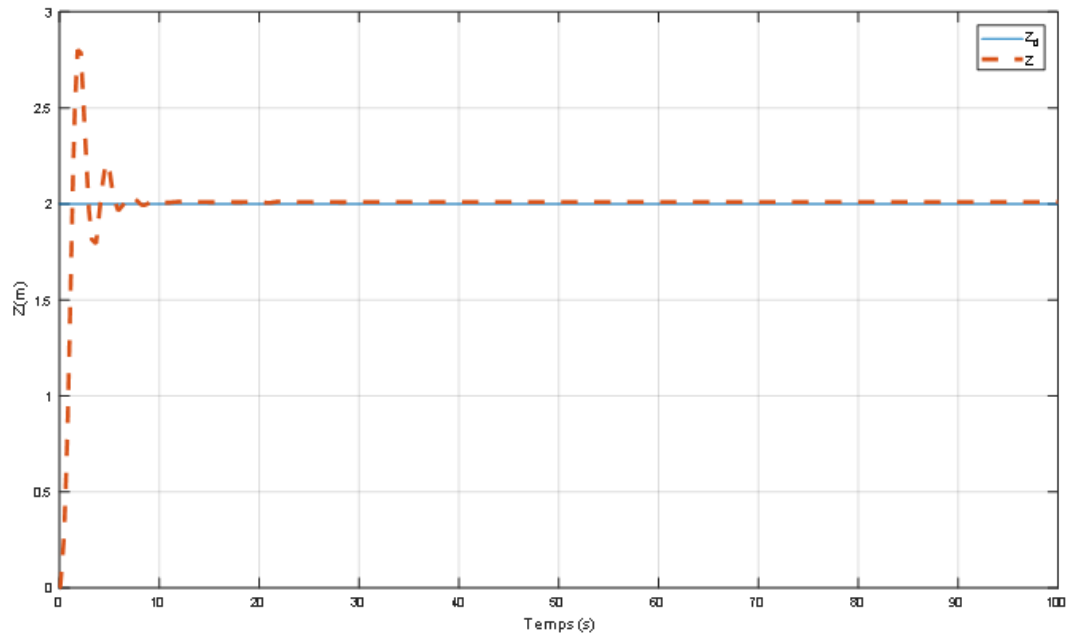


Figure 3.16: Réponse en  $Z$ .

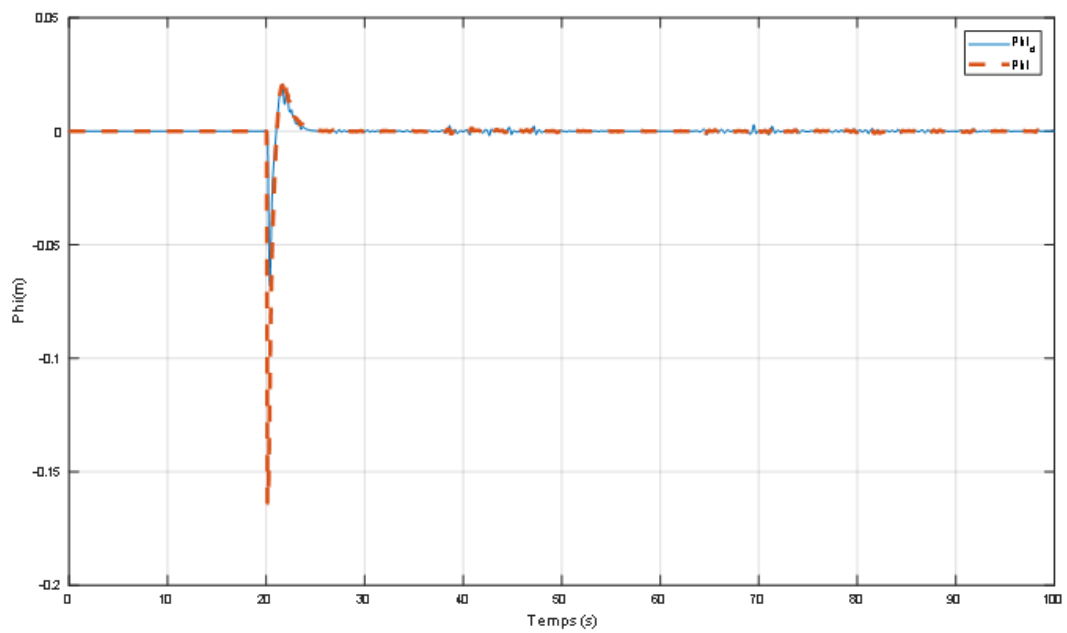


Figure 3.17: Réponse en  $\Phi$ .

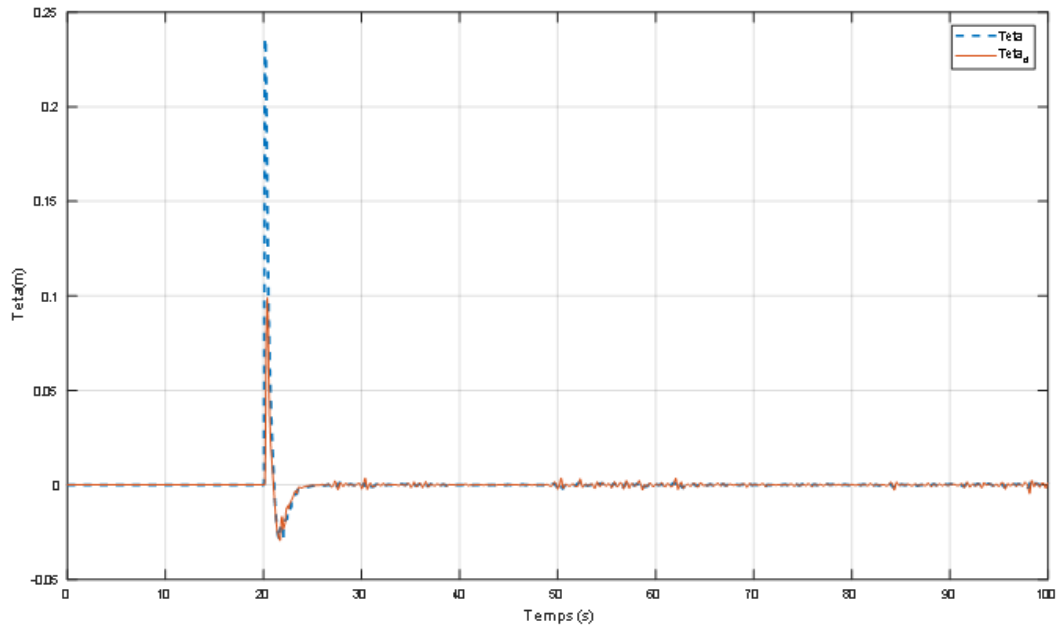


Figure 3.18: Réponse en angle Teta.

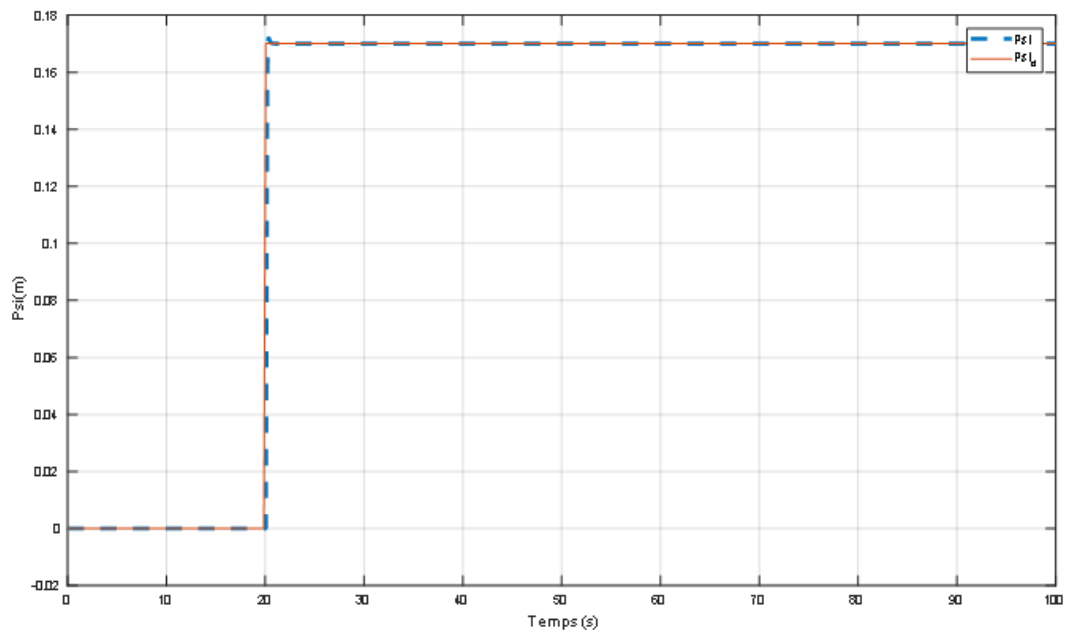


Figure 3.19: Réponse en angle Psi.

### 3.6.2 Tests de robustesse

#### — Perturbations du vent

La figure(3.20) représente l'effet du vent sur le quadrirotor.

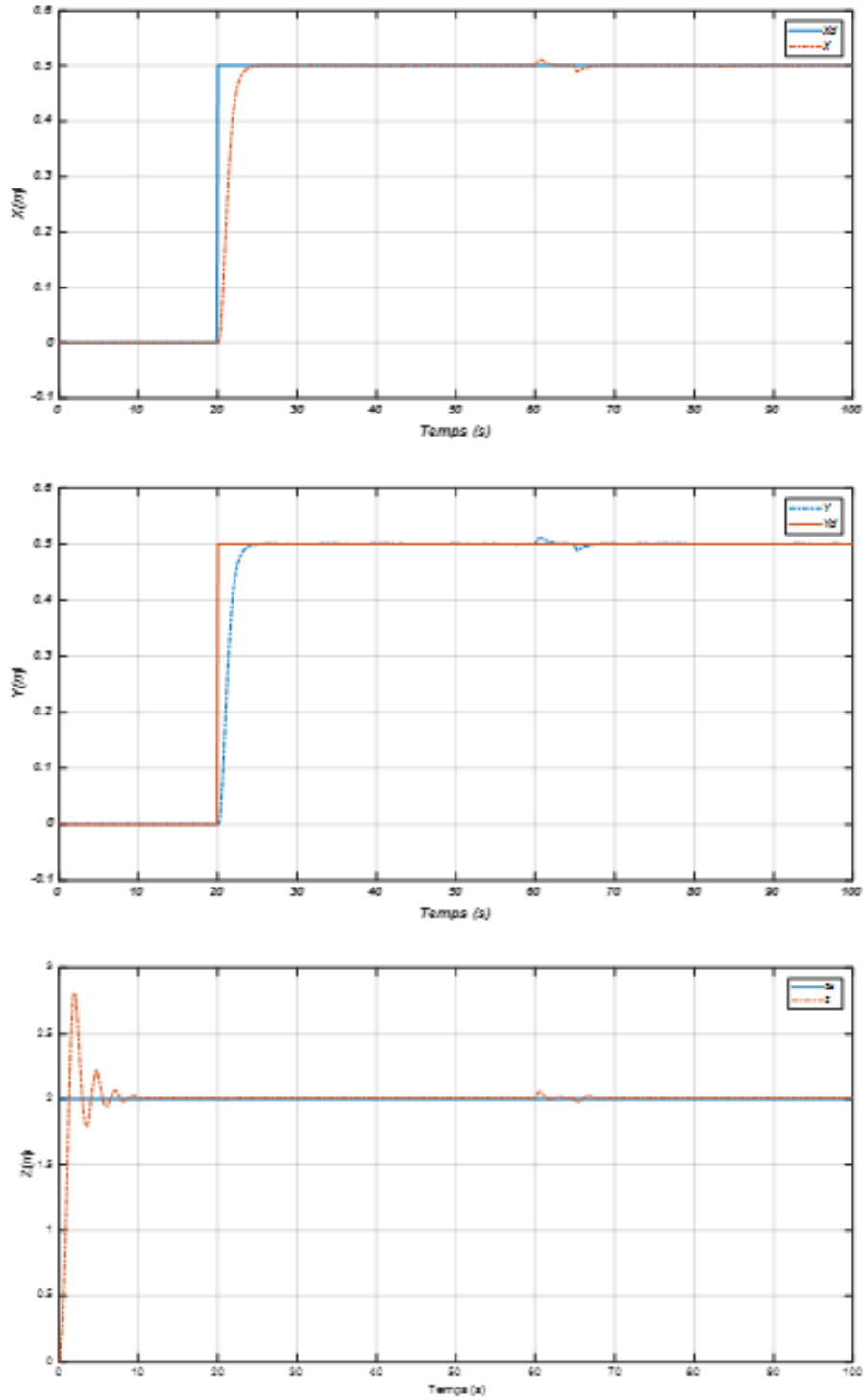


Figure 3.20: Les trajectoires des positions avec un effet de vent

— Variation Paramétrique (la masse)

La figures (3.21) représentent l'effet du changement de masse sur le quadrirotor.

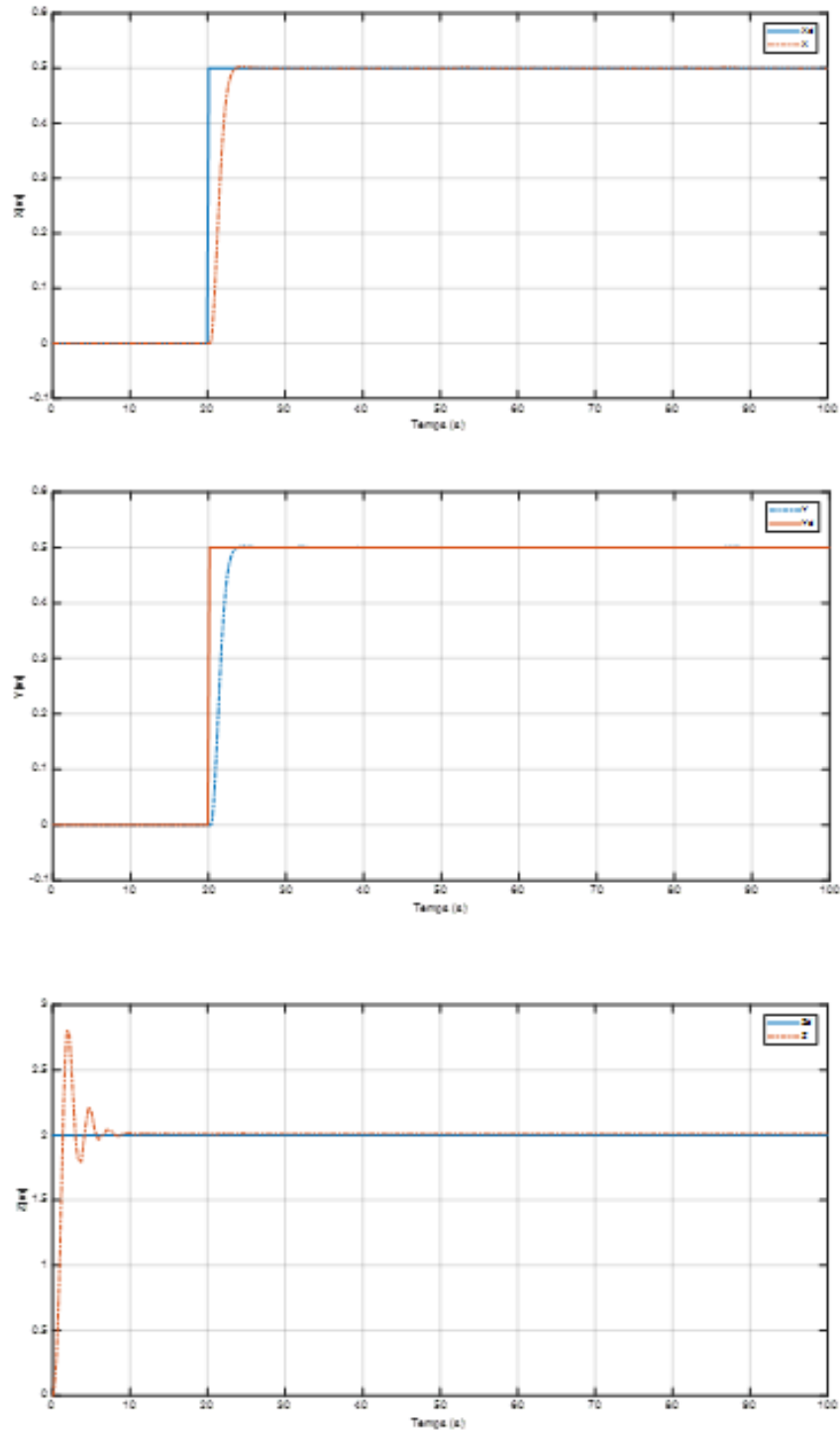
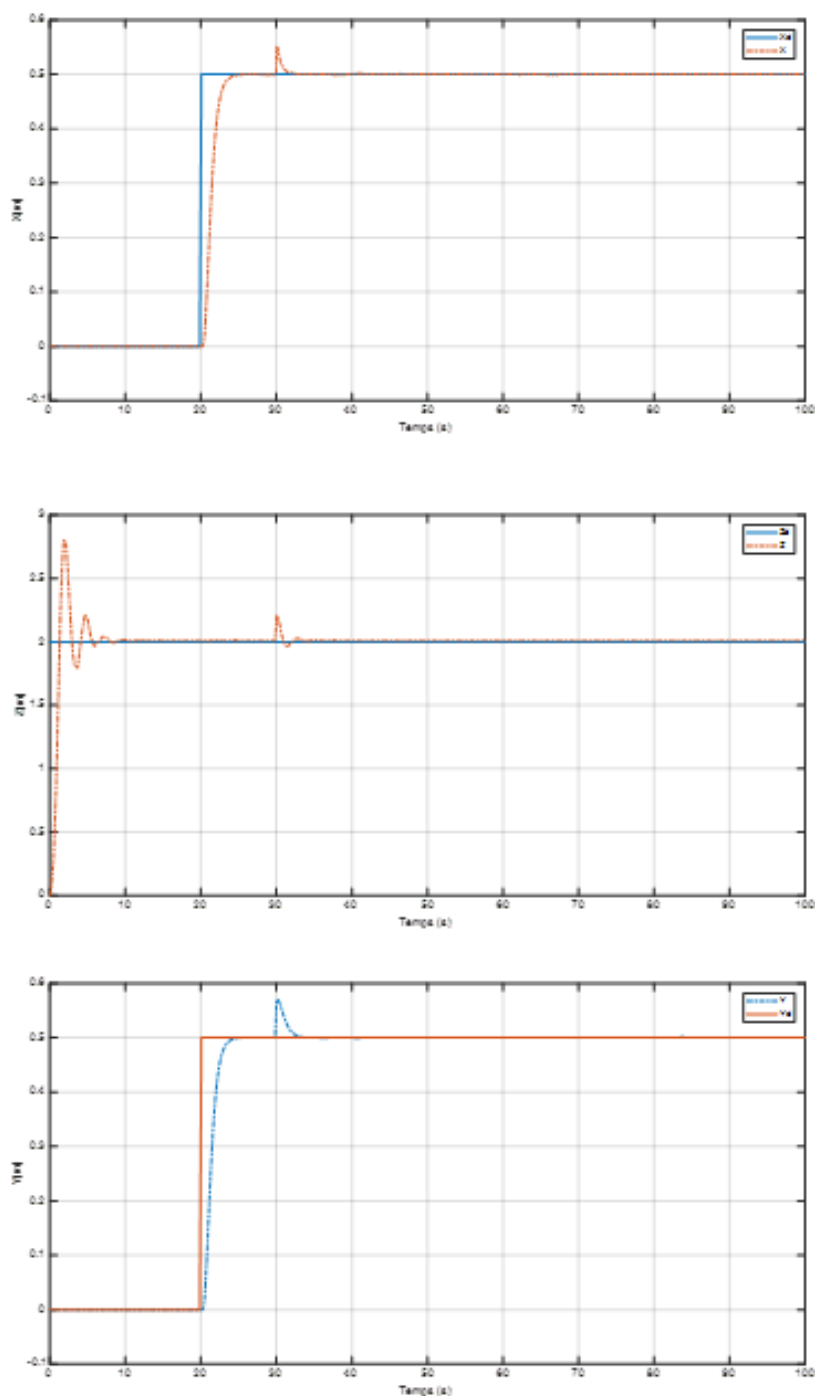


Figure 3.21: Les trajectoires des positions avec augmentation de la masse

La figures (3.22) représente l'effet du bruit de mesure sur le quadrirotor.



**Figure 3.22:** Les trajectoires des positions avec un effet du bruit de mesure

Après avoir effectué les tests de robustesse du quadrirotor, nous observons que même en présence de perturbations et de variations des paramètres, le régulateur peut maintenir une performance stable et satisfaisante. Les résultats obtenus lors des tests confirment cette observation, montrant une performance améliorée du système.

### 3.7 Commande PID floue adaptatif

La commande PID floue est une méthode de régulation qui combine les principes du régulateur PID classique avec la logique floue pour ajuster dynamiquement les paramètres du PID en fonction des conditions du système. Contrairement au PID classique, où les coefficients du contrôleur sont fixes, le PID floue utilise des ensembles de règles floues pour ajuster les coefficients du PID en temps réel en fonction des entrées et des sorties du système. Cela permet une régulation plus précise et adaptative, en particulier dans des environnements où les conditions changent rapidement ou sont mal définies.

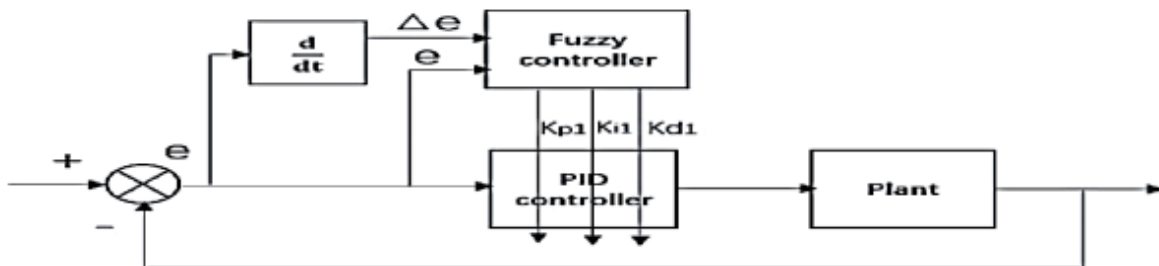


Figure 3.23: Type PID floue auto réglable [1].

#### 3.7.1 Application de la commande PID floue adaptatif sur le quadrirotor

Nous avons maintenu la même structure de régulation que celle utilisée pour les commandes précédentes. La figure (3.20) illustre la configuration de commande du modèle de quadrirotor utilisant six régulateurs PID. Les gains proportionnels  $K_p$  et d'intégration  $K_i$  des régulateurs PID sont ajustés en ligne à l'aide d'un système d'inférence floue. En revanche, les valeurs des gains dérivatifs  $K_d$  de ces régulateurs restent constantes.

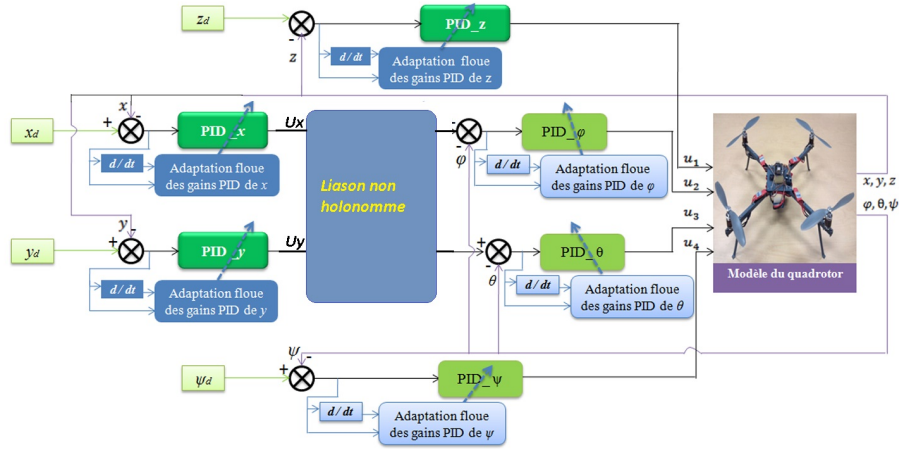


Figure 3.24: Schéma de la commande adaptative PID logique floue [3]

Nous avons maintenu l'utilisation des fonctions d'appartenance triangulaires pour chaque variable d'entrée et de sortie. Cependant, dans cette itération, nous avons opté pour vingt-cinq règles d'inférence pour chaque régulateur. La table des règles est la même pour tous les contrôleurs flous et elle est présentée comme suit :

e/de	NG	NP	Z	PP	PG
NG	$K_p = vb, K_i = vs$	$K_p = b, K_i = s$	$K_p = b, K_i = s$	$K_p = b, K_i = s$	$K_p = m, K_i = m$
NP	$K_p = mb, K_i = ms$	$K_p = mb, K_i = ms$	$K_p = b, K_i = s$	$K_p = m, K_i = m$	$K_p = m, K_i = b$
Z	$K_p = mb, K_i = ms$	$K_p = mb, K_i = ms$	$K_p = b, K_i = s$	$K_p = m, K_i = m$	$K_p = s, K_i = mb$
PP	$K_p = b, K_i = s$	$K_p = m, K_i = m$	$K_p = m, K_i = m$	$K_p = s, K_i = b$	$K_p = vs, K_i = mb$
PG	$K_p = m, K_i = m$	$K_p = m, K_i = b$	$K_p = s, K_i = mb$	$K_p = vs, K_i = vb$	$K_p = m, K_i = m$

Avec :

NG : négative grand, NP : négative petit

Z : zéro

PP : positif petit, PG : positif grand

vb : very big, vs : very small

mb : medium big, ms : medium small,

b : big, m : medium, s : small

Les figures ci-dessous représentent les fonctions d'appartenance d'entrées et de sorties pour l'ajustement des gains du régulateur PID flou adaptatif des variables X et Y.

Après avoir tester la commande PID flou adaptatif sur le quadrirotor on a remarqué

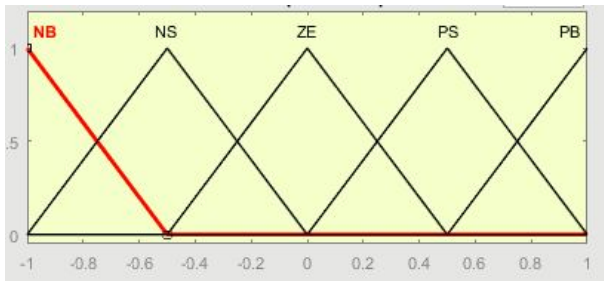


Figure 3.25: Entrée régulateur (erreur)

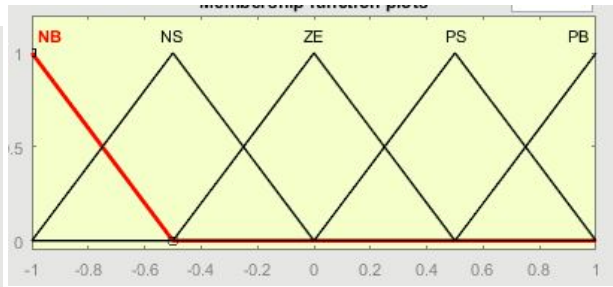


Figure 3.26: Entrée régulateur (dérivé de l'erreur)

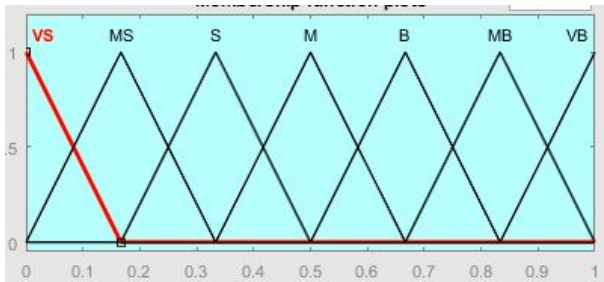


Figure 3.27: Sortie du régulateur ( $K_p$ )

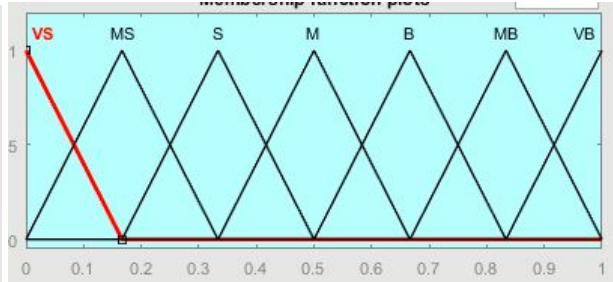


Figure 3.28: Sortie du régulateur ( $K_i$ )

une diminution de temps de réponse par rapport à la commande floue et une précision parfaite. Cette commande a combiné les avantages de la régulation PID classique (rapidité et stabilité) avec ceux de la logique floue (gestion des incertitudes).

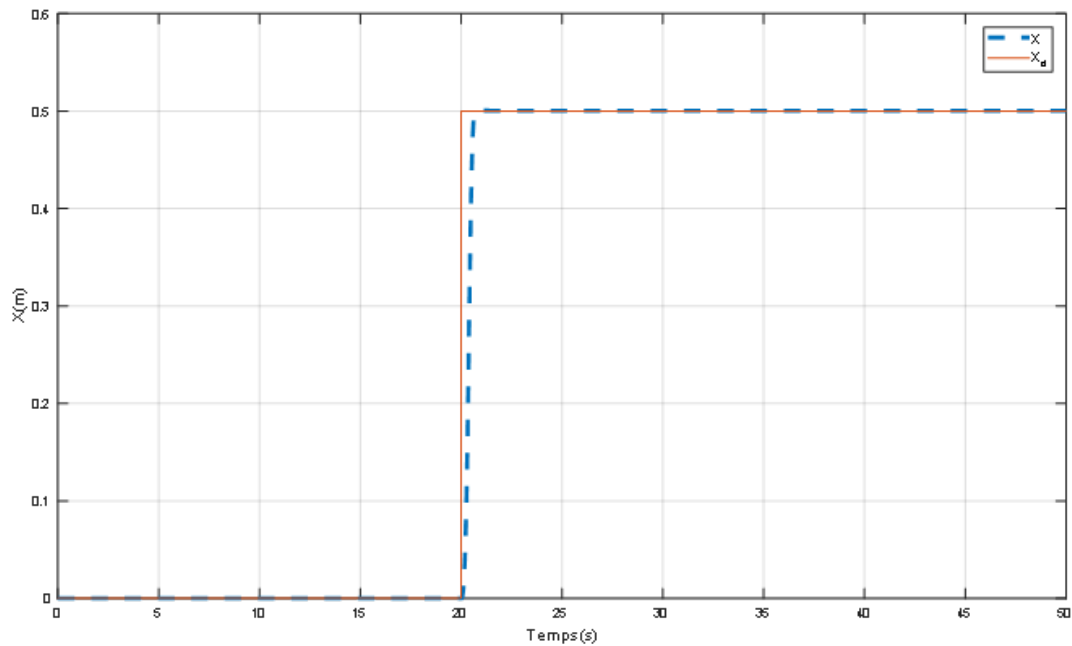


Figure 3.29: Réponse en X

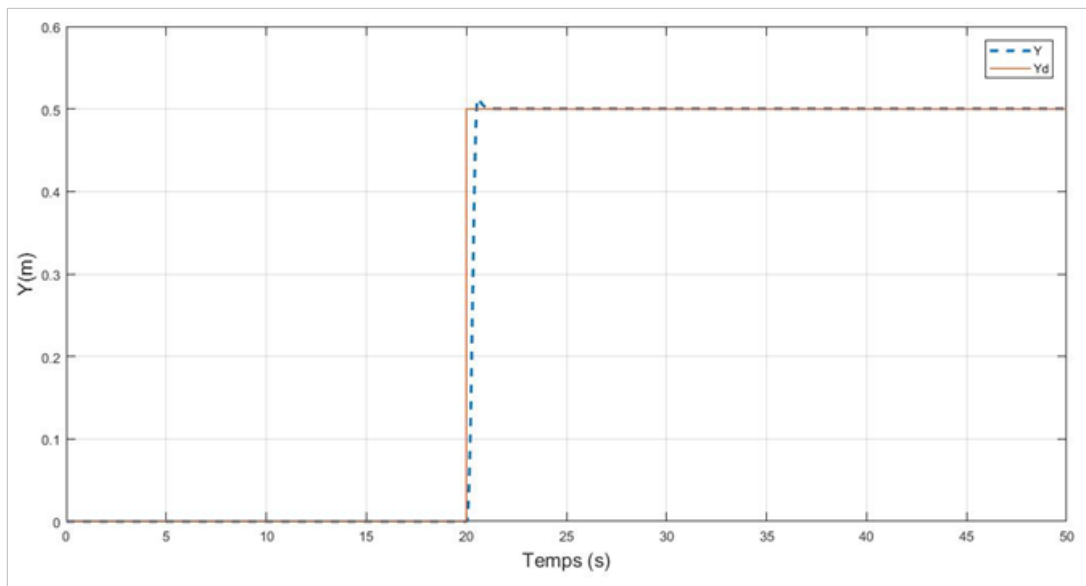


Figure 3.30: Réponse en Y

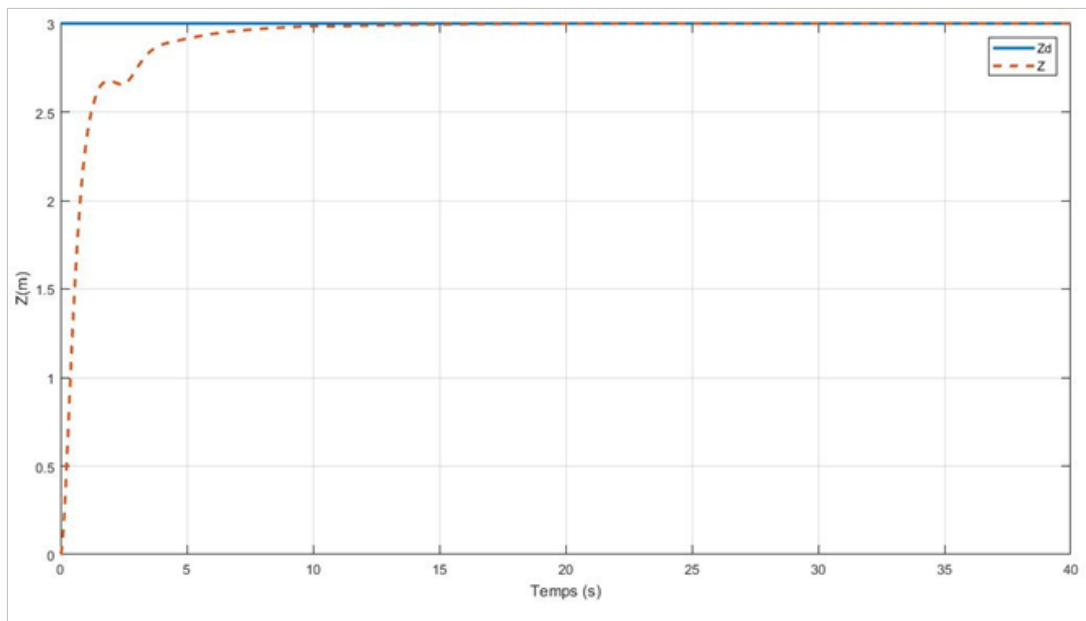


Figure 3.31: Réponse en Z

### 3.7.2 Tests de robustesse

#### — Perturbations du vent

Les figures (3.31) (3.32) et (3.33) représente l'effet du vent sur le quadrirotor.

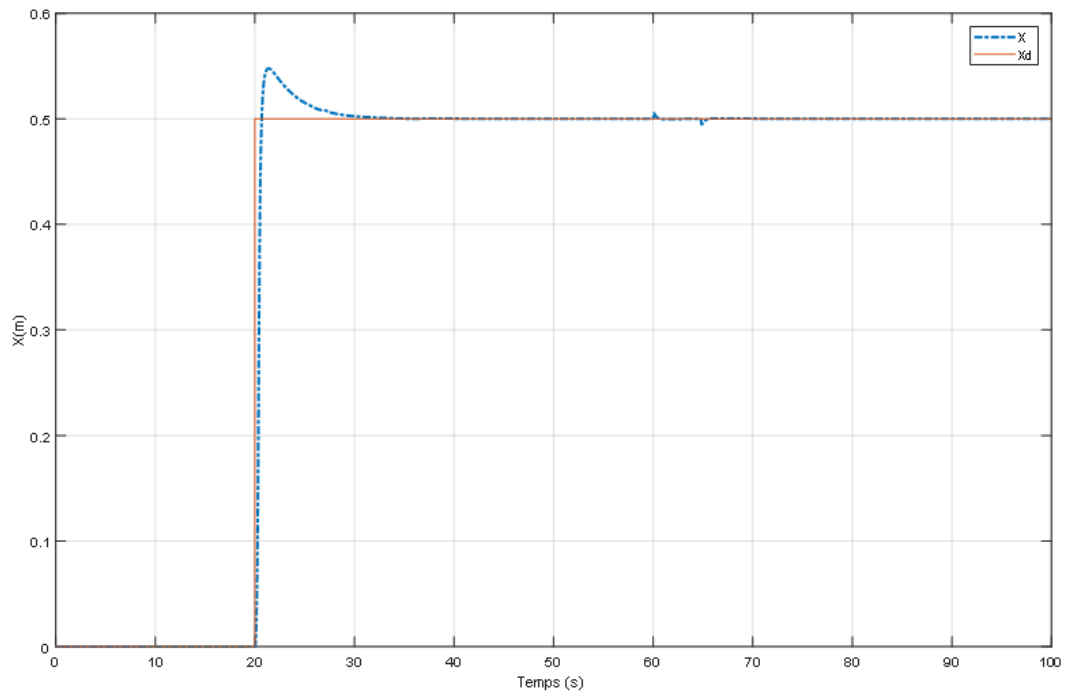


Figure 3.32: Effet de vent sur  $X$

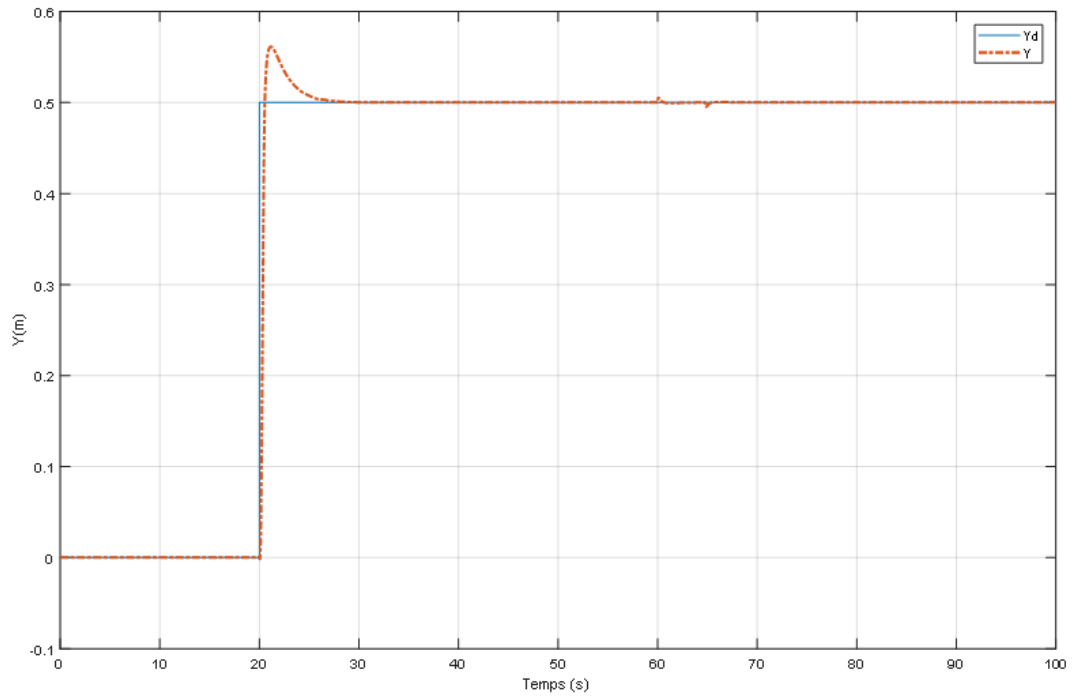


Figure 3.33: Effet de vent sur  $Y$

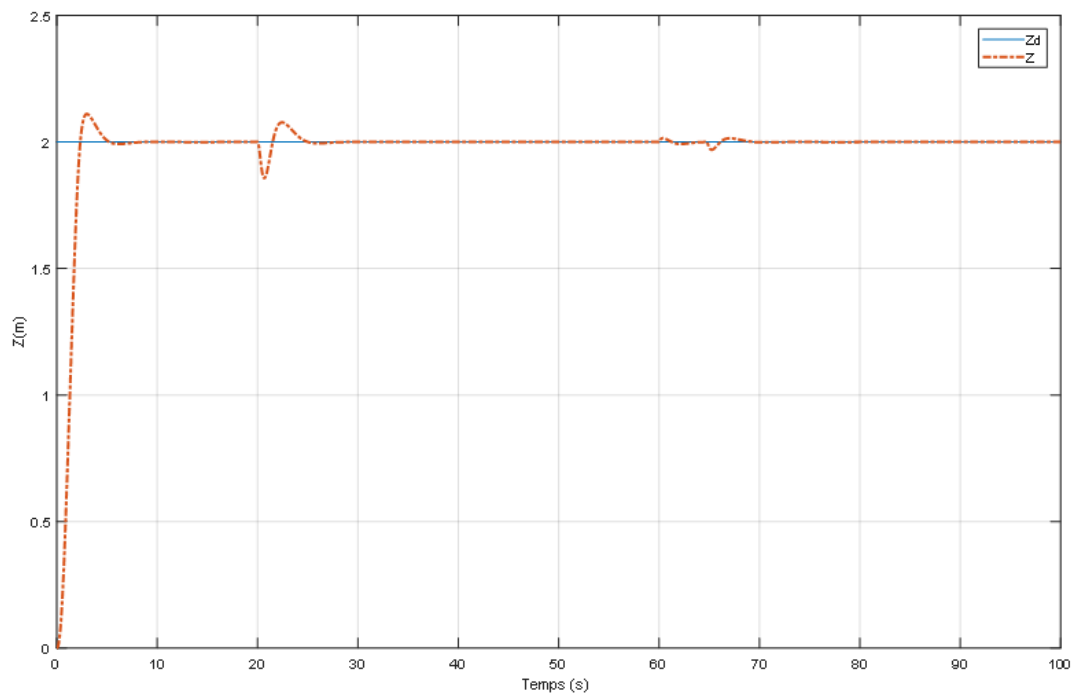
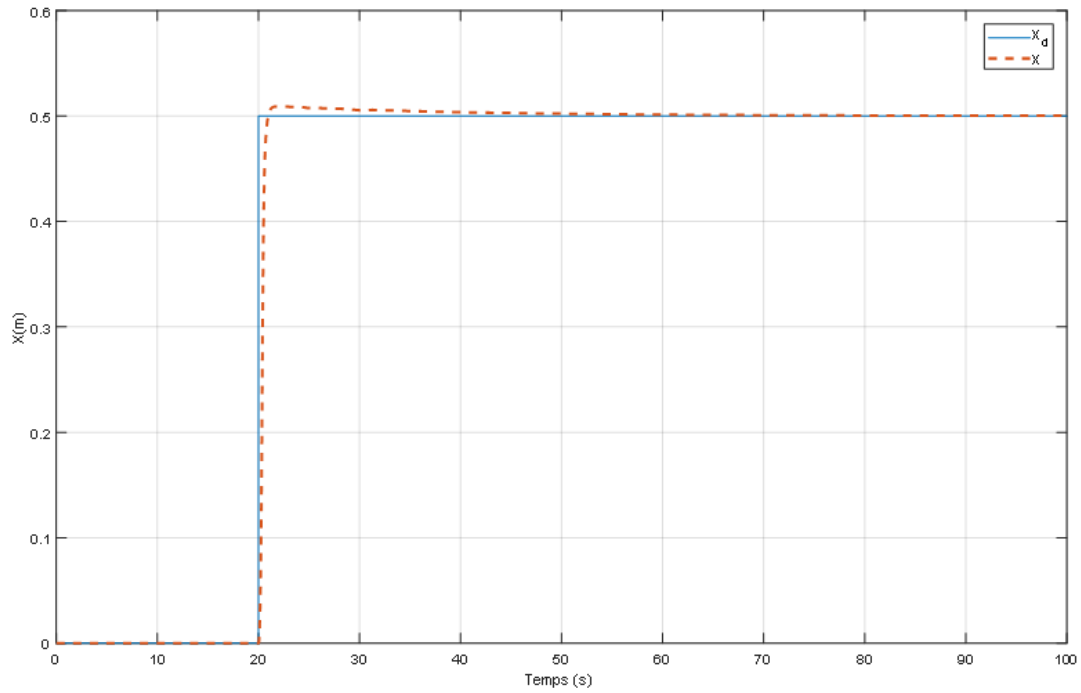


Figure 3.34: Effet de vent sur  $Z$

— **Variation Paramétrique (la masse)**

La figures (3.35) (3.36) et (3.37) représentent l'effet du changement de masse sur le quadrirotor.



**Figure 3.35:** Effet de changement de la masse sur  $X$

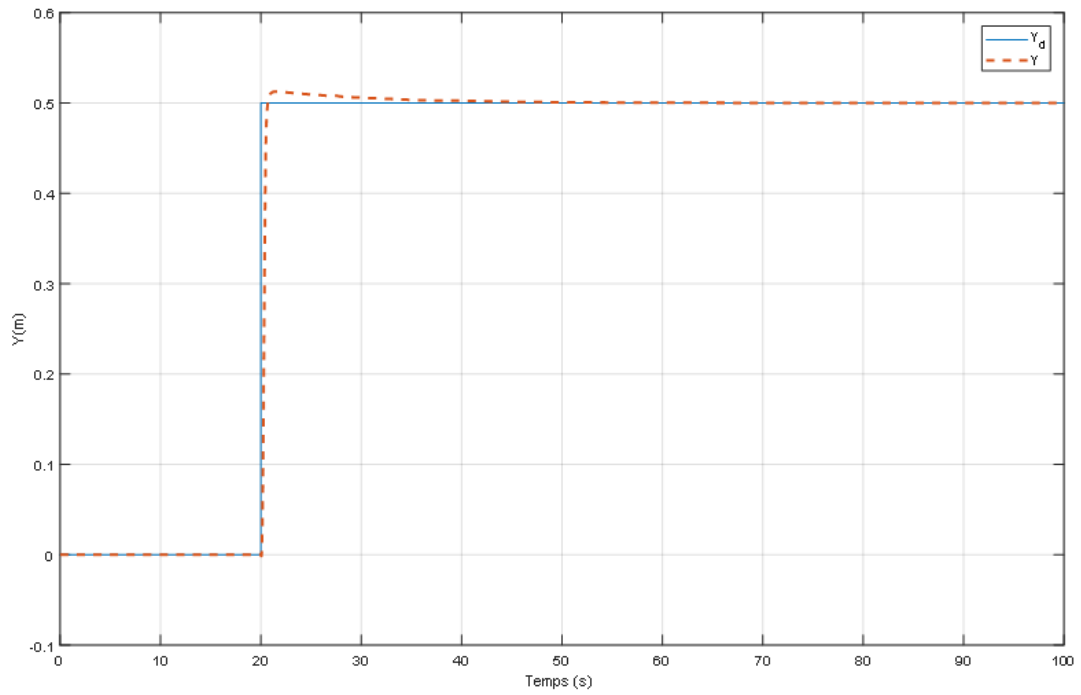


Figure 3.36: Effet de changement de la masse sur  $Y$

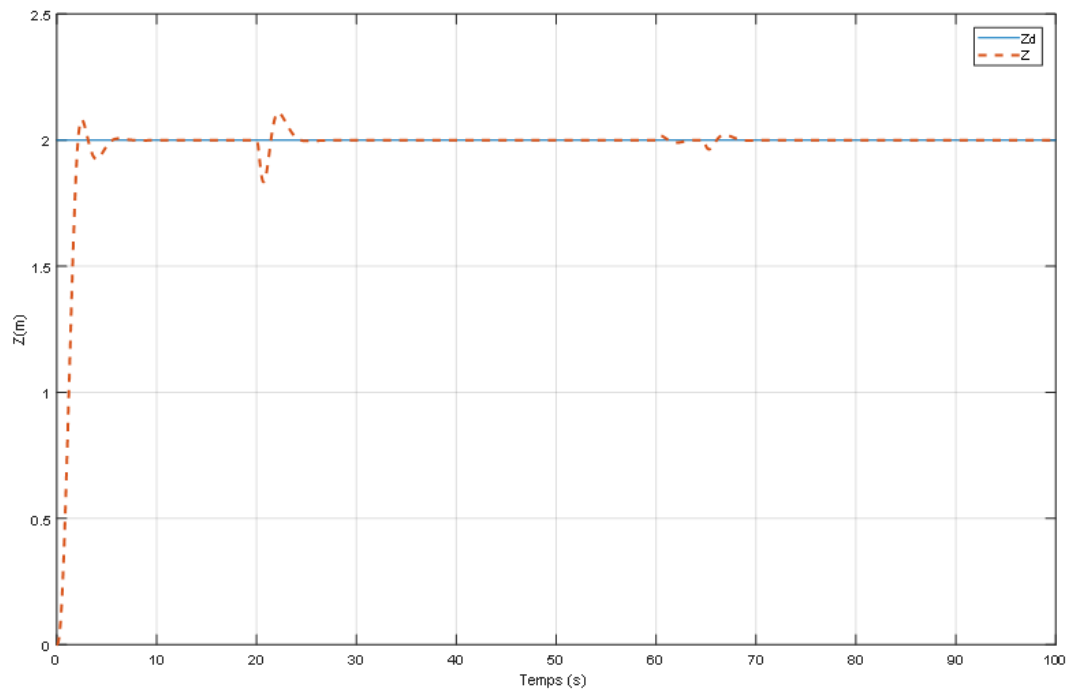
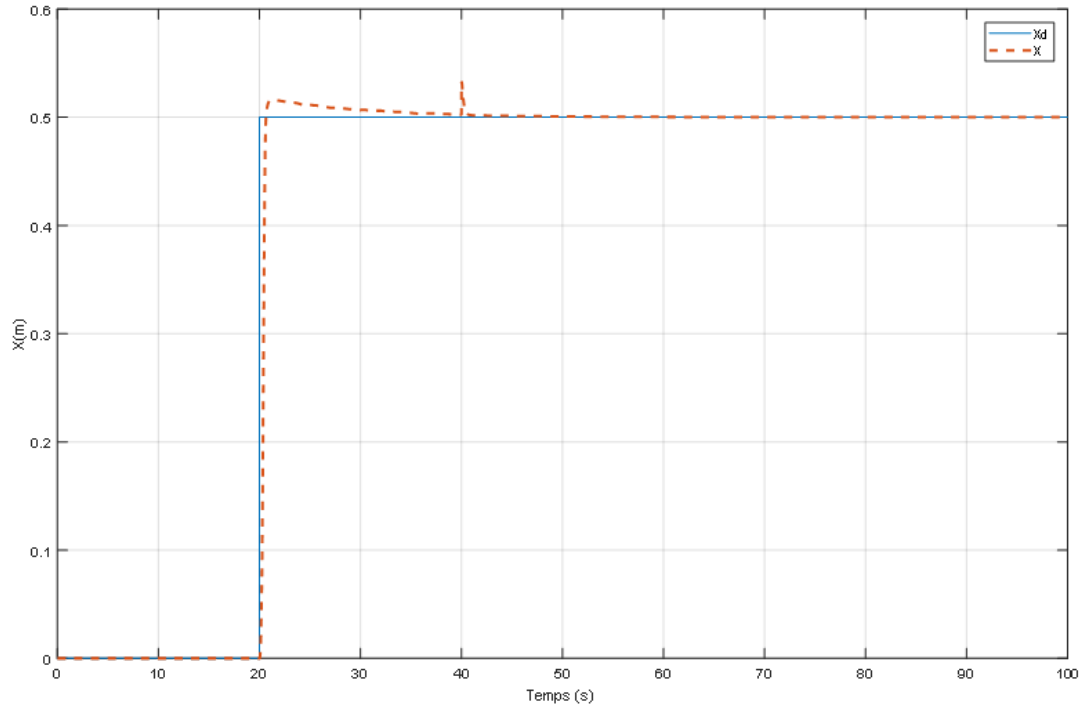


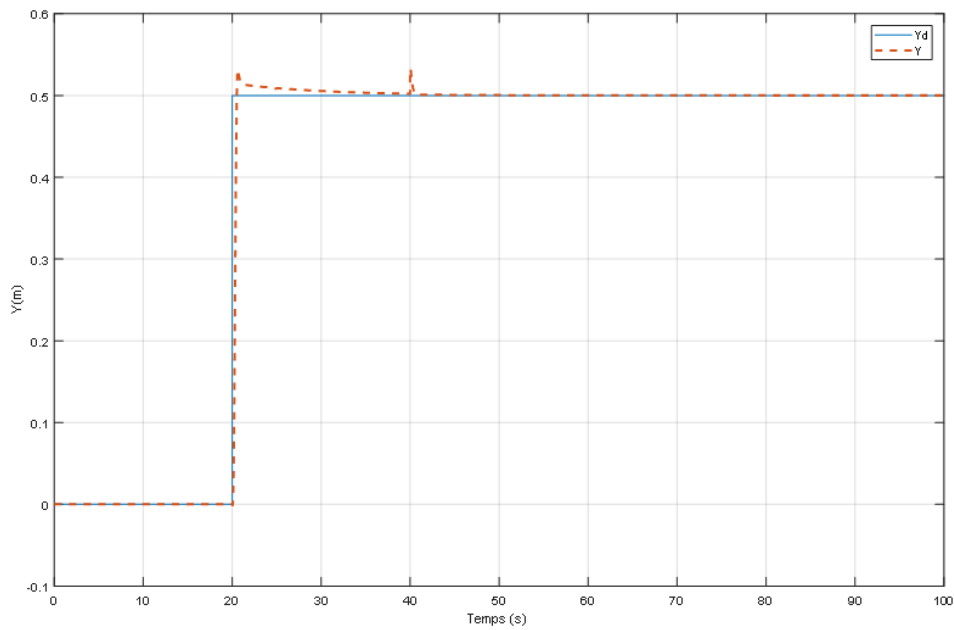
Figure 3.37: Effet de changement de la masse sur  $Z$

— **Bruit de mesures**

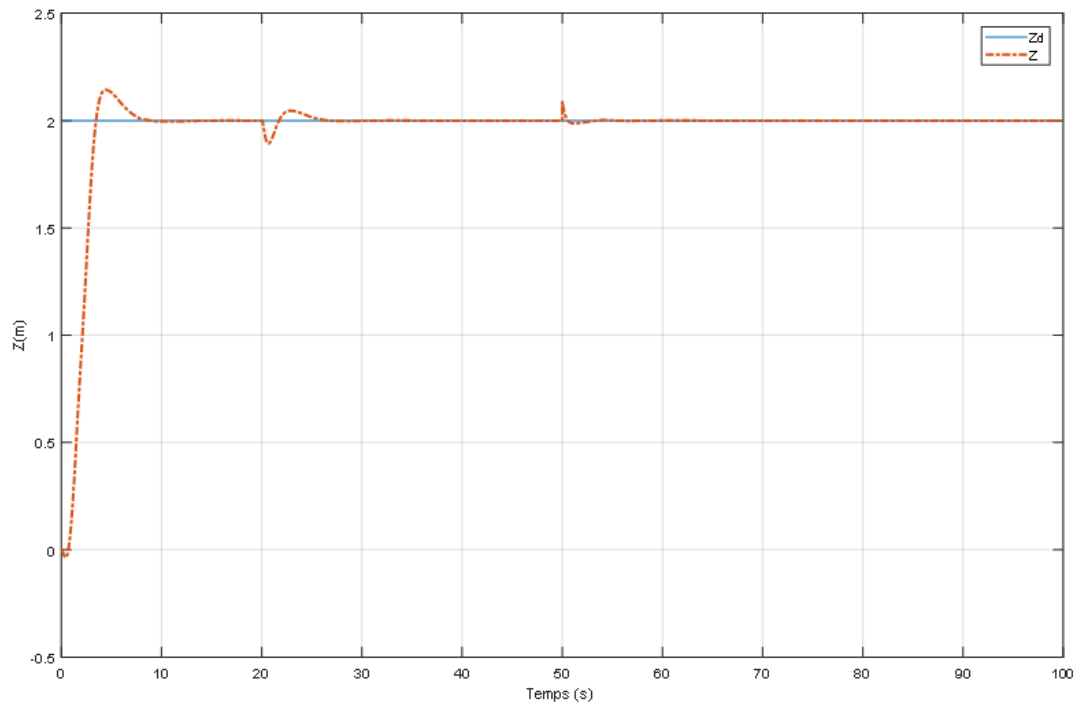
La figures (3.38) (3.39) et (3.40) représente l'effet du bruit de mesure sur le quadrirotor.



**Figure 3.38:** Effet de bruit de mesure sur X



**Figure 3.39:** Effet de bruit de mesure sur Y



**Figure 3.40:** Effet de bruit de mesure sur  $Z$

Après avoir effectué les tests de robustesse du quadrirotor, nous avons constaté que le système pouvait surmonter les perturbations et retrouver sa stabilité avec un temps de réponse court.

## 3.8 Etude comparative entre les commandes synthétisées

### a) Temps de Réponse

- **PID classique** : Rapide dans des conditions idéales, mais peut varier avec les perturbations.
- **PID optimisé** : Plus rapide et stable grâce à des gains optimisés.
- **Régulation floue** : Variable, dépend des règles définies, généralement moins rapide que PID optimisé.
- **PID flou adaptatif** : Temps de réponse amélioré grâce à l'adaptabilité en temps réel.

### b) Stabilité

- **PID classique** : Peut devenir instable si mal réglé.
- **PID optimisé** : Plus stable grâce à des réglages optimisés.
- **Régulation floue** : Très stable, même avec des variations de paramètres.
- **PID flou adaptatif** : Très stable, avec une capacité d'adaptation aux perturbations.

### c) Robustesse

- **PID classique** : Moins robuste face aux variations de paramètres.
- **PID optimisé** : Plus robuste, mais dépend de la qualité de l'optimisation.
- **Régulation floue** : Très robuste face aux incertitudes.
- **PID flou adaptatif** : Très robuste, avec une excellente capacité d'adaptation.

Chaque méthode de régulation a ses propres avantages et inconvénients. La régulation PID classique est simple et efficace pour des systèmes bien modélisés, tandis que la régulation PID optimisée offre de meilleures performances en termes de temps de réponse et de stabilité. La régulation floue est très robuste face aux incertitudes, et la régulation PID floue adaptatif combine cette robustesse avec une excellente adaptabilité en temps réel.

Le choix de la méthode de régulation dépendra des exigences spécifiques de l'application, de la complexité, et des ressources disponibles pour la mise en œuvre et le calcul en temps réel. Pour un quadricoptère nécessitant une performance optimale dans des conditions variées, la régulation PID floue adaptatif pourrait offrir la meilleure combinaison de robustesse et de réactivité.

## 3.9 Conclusion

Dans ce chapitre nous avons étudié les concepts de la logique floue adaptatif et du PID flou dans le contexte du contrôle des systèmes, en se concentrant particulièrement sur leur application aux quadrirotors. Les résultats des simulations ont démontré une performance satisfaisante du système, caractérisée par sa stabilité, sa précision et sa réactivité. Aussi, une amélioration a été observée avec l'utilisation du PID flou adaptatif, qui a surpassé la logique floue dans plusieurs aspects, notamment lors des tests de robustesse en présence de perturbations. Finalement, nous avons comparé les diverses approches de commande étudiées et avons observé que le PID flou adaptatif a donné les meilleurs résultats.

# Conclusion générale

La réalisation de ce mémoire sur la commande d'un quadrirotor nous a permis d'explorer en profondeur les différents aspects et défis liés à ce domaine passionnant de la robotique aérienne. À travers chaque chapitre, nous avons progressivement construit notre compréhension, en commençant par une clarification du concept de quadrirotor, suivi d'une analyse approfondie de sa classification et de son fonctionnement général. Nous avons également développé un modèle mathématique pour décrire sa dynamique, mettant en évidence l'importance de comprendre les effets physiques influant sur son vol. Dans le deuxième chapitre, nous nous sommes concentrés sur la commande par PID et son optimisation, mettant en lumière les avantages et les limitations de cette approche dans le contrôle des quadrirotors. Bien que la commande PID ait démontré une stabilité et une précision satisfaisantes, nous avons constaté que son adaptabilité face à des conditions changeantes reste limitée, notamment en présence de fortes perturbations. Le troisième chapitre a été consacré à l'étude de la logique floue et le PID flou adaptatif. Nos simulations ont révélé une performance satisfaisante du système, mais une amélioration significative a été observée avec l'utilisation du PID flou adaptatif, qui s'est avéré plus efficace, notamment lors des tests de robustesse. En résumé, notre exploration des différentes techniques de commande nous a permis de reconnaître l'importance de trouver un équilibre entre stabilité, précision et rapidité pour garantir le bon fonctionnement des quadrirotors dans des environnements variés et complexes. Ces résultats soulignent également l'importance de poursuivre la recherche dans le domaine des approches non linéaires, telles que le modes glissant et le Backstepping, pour relever les défis futurs de la commande des quadrirotors.

# Bibliographie

- [1] Abdelmoumen Abidli and Fares Khengaoui. *Variation de La Vitesse du Moteur à Courant Continu par PID Flou*. PhD thesis, Université Kasdi Merbah Ouargla.
- [2] Reg Austin. *Unmanned aircraft systems : UAVS design, development and deployment*. John Wiley & Sons, 2011.
- [3] Kheira Belhadri. Contrôle robuste d'un quadrotor, 2021. Présenté par : BELHADRI Kheira, Année Universitaire : 2020-2021.
- [4] Hansruedi Bühler. *Réglage par logique floue*. Presses polytechniques et universitaires romandes, 1994.
- [5] I Chalane, T Ouari, and O Guenounou. Optimisation des paramètres d'un pid par essais particuliers (pso). *Mémoire 2eme master, université de Bejaïa*, 2014, 2013.
- [6] Yann Cooren. *Perfectionnement d'un algorithme adaptatif d'Optimisation par Essaim Particulaire : application en génie médical et en électronique*. PhD thesis, Université Paris-Est, 2008.
- [7] Russell C Eberhart and Yuhui Shi. Comparing inertia weights and constriction factors in particle swarm optimization. In *Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512)*, volume 1, pages 84–88. IEEE, 2000.
- [8] Benattia Seif Eddine and Oudjit Mohamed Amine. Modélisation, commande et observation d'un uav de type quadrirotor. Master's thesis, École Nationale Polytechnique - Alger, 2012.
- [9] Abdelmalek Gacem. Utilisation des méthodes d'optimisations métaheuristiques pour la résolution du problème de répartition optimale de la puissance dans les réseaux électriques. *Mémoire de Magister, Université Mohamed Lakhdar Ben Amara d'El Oued Algerie*, 2010.
- [10] Rahem Hakima. *Commande par logique floue d'un robot manipulateur à articulation flexible*. PhD thesis, Université Mouloud Mammeri, 2017.
- [11] MT Huynh. Application de l'optimisation par essaim particulairea la représentation de matrices de préférence valuées. *Université Libre De Bruxelles 2009*, 2009.

- [12] James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. iee, 1995.
- [13] Laib Khaled and Maamria Djamaledine. Commande d'un quadrirotor. Master's thesis, École Nationale Polytechnique - Alger, 2011.
- [14] Hicham Khebbache. *Tolérance aux défauts via la méthode backstepping des systèmes non linéaires Application : Système UAV de type Quadrirotor*. PhD thesis, Université de Sétif 1-Ferhat Abbas, 2012.
- [15] YAAKOUB KHEDDAR and MOHAMED BELGHOUL. *Modélisation et commande d'un drone Quadrirotor*. PhD thesis, Université Ibn Khaldoun, 2017.
- [16] D'ATTITUDE D'UN LA STABILISATION and UAV DE. Thème : Commande par mode glissant-floue pour.
- [17] Tahar Laamayad. *Commande Optimale d'une Machine Asynchrone Apport de La Logique Floue*. PhD thesis, Université de Batna 2, 2008.
- [18] R Mansouri. *Contribution à l'analyse et la synthèse des systèmes d'ordre fractionnaire par la représentation d'état*. PhD thesis, Ph. D. Thesis, Université de Tizi-Ouzou, Algeria, 2008.
- [19] Abd Elmounaim Mekki. Commande non linéaire d'un hélicoptère à six degrés de liberté type quadrotor basée sur le pid à gains adaptatifs, 2016. Présenté par : MEKKI Abd Elmounaim, Année Universitaire : 2015/2016.
- [20] Olaf Moseler and Rolf Isermann. Application of model-based fault detection to a brushless dc motor. *IEEE Transactions on industrial electronics*, 47(5) :1015–1020, 2000.
- [21] Azouz Mustapha. *Modélisation et commande d'un quadrirotor : Etude comparative de la commande floue et PID*. PhD thesis, Université Mouloud Mammeri, 2016.
- [22] Vasile PRISACARIU. A brief history of uavs in the 1970s. *Scientific Research & Education in the Air Force-AFASES*, 2022, 2022.
- [23] N RACHEDI. Commande hybrid avec observation d'un uav de type quadrotor. *mémoire de magistère, EMP Bordj EL-Bahri, algerie*, 2011.
- [24] Lotfi Asker Zadeh. Fuzzy sets. *Information and control*, 8(3) :338–353, 1965.

## Résumé

Le mémoire explore deux approches de commande pour un quadrirotor : linéaire et non linéaire par logique floue. La première partie aborde la modélisation et la commande en boucle ouverte, ainsi que la commande linéaire avec un régulateur PID optimisé par PSO. La deuxième partie se concentre sur la commande non linéaire par logique floue, incluant sa modélisation, sa mise en œuvre sur le quadrirotor, et une comparaison avec la commande linéaire.

Mots clés: Quadrirotor; linéaire, non linéaire, PID, PSO, logique floue, commande, PID floue adaptatif

## Abstract

The thesis explores two control approaches for a quadrotor: linear and nonlinear using fuzzy logic. The first part addresses modeling and open-loop control, along with linear control using a PID regulator optimized by PSO. The second part focuses on nonlinear control using fuzzy logic, including its modeling, implementation on the quadrotor, and a comparison with linear control.

Keywords: Quadrotor, linear, non-linear, PID, PSO, fuzzy logic, control, adaptive fuzzy PID

## ملخص

تستكشف الأطروحة طريقتين للتحكم في رباعي المرواح: الخطية وغير الخطية باستخدام المنطق الضبابي. يعالج الجزء الأول النمذجة والتحكم في الحلقة المفتوحة، إلى جانب التحكم الخطي باستخدام منظم بي اي دي المحسن بواسطة بي اس او. أما الجزء الثاني فيركز على التحكم غير الخطي باستخدام المنطق الضبابي، بما في ذلك نمذجته وتنفيذه على رباعي المرواح ومقارنته بالتحكم الخطي.

كلمات مفتاحية: كوادريوتور، خطي، غير خطي، ضبابي متكيف، PID، منطق ضبابي، تحكم، PSO، PID