

## Remerciements

A faire en dernier :-)



# Table des matières

<b>Liste des figures</b>	<b>ix</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Liste des algorithmes</b>	<b>xiii</b>
<b>Liste des abréviations</b>	<b>xv</b>
<b>Introduction</b>	<b>1</b>
<b>1 Généralité sur les drones convertible</b>	<b>7</b>
1.1 Micros drones convertibles . . . . .	7
1.1.1 Domaine de vol . . . . .	7
1.1.2 Types d'architecture des drones convertibles . . . . .	8
1.2 Propriétés des <i>tailsitters</i> et des <i>freewings</i> . . . . .	12
1.2.1 Actionnement . . . . .	12
1.2.2 Aérodynamique . . . . .	12
1.3 Modélisations . . . . .	13
1.4 Actionnements . . . . .	15
1.5 Architectures de commande de vol . . . . .	16
1.6 Méthodes de commande . . . . .	17
1.7 Technologies et réalisations . . . . .	18
<b>2 Modélisation d'un drone convertible : DarkO</b>	<b>19</b>
2.1 Modèle du drone DarkO . . . . .	19
2.1.1 Modèle non-linéaire complet . . . . .	21
2.1.2 Modèle non linéaire simplifié à basse vitesse . . . . .	24
2.2 Identification des paramètres du modèle . . . . .	26
2.2.1 Modélisation des actionneurs . . . . .	29
2.3 Équilibres stationnaires . . . . .	30
2.3.1 Équilibre stationnaire sans vent . . . . .	30
2.3.2 Équilibre stationnaire en présence de vent . . . . .	31
2.4 Dynamiques linéarisées . . . . .	36
2.4.1 Dynamique linéarisée sans vent . . . . .	36
2.4.2 Dynamique linéarisée en présence de vent . . . . .	37

---

2.5	Conclusion du Chapitre 2 . . . . .	43
<b>3</b>	<b>Commande hybride</b>	<b>47</b>
3.1	Motivation . . . . .	47
3.2	Contrôleur par retour d'état non-linéaire . . . . .	48
3.3	Contrôleur par retour d'état linéaire . . . . .	49
3.4	Conception d'une commande locale-globale basée sur l'hystérésis . .	51
3.5	Conclusion du Chapitre 3 . . . . .	54
<b>4</b>	<b>Commande longitudinale d'une maquette de DarkO</b>	<b>57</b>
4.1	Motivation . . . . .	57
4.2	Présentation de la maquette expérimentale . . . . .	58
4.2.1	Description physique, capteurs et actionnements . . . . .	58
4.2.2	Simulation des mouvements du drone . . . . .	61
4.3	Contrôle linéaire, proportionnel-intégral à 3 DOF . . . . .	62
4.3.1	Description du schéma de contrôle . . . . .	62
4.3.2	Optimisation $H_\infty$ . . . . .	64
4.4	Résultats . . . . .	65
4.5	Conclusion du Chapitre 4 . . . . .	67
<b>5</b>	<b>Commande proportionnelle intégrale de DarkO</b>	<b>69</b>
5.1	Motivation . . . . .	69
5.2	Schéma de commande linéaire proportionnelle intégrale . . . . .	69
5.2.1	Description du schéma de contrôle . . . . .	69
5.2.2	Contrôleur optimisé sous contraintes $H_\infty$ , cas sans vent . .	73
5.2.3	Contrôleur optimisé sous contrainte $H_\infty$ , cas multimodèles .	75
5.3	Vol expérimental en soufflerie ouverte . . . . .	81
5.4	Conclusion du Chapitre 5 . . . . .	84
<b>6</b>	<b>Modélisation et commande d'un drone à aile libre rotation libre</b>	<b>85</b>
6.1	Motivation . . . . .	85
6.2	Design et modélisation d'un drone : Colibri . . . . .	85
6.3	State estimation . . . . .	90
6.3.1	Placement des capteurs . . . . .	90
6.3.2	Estimation de la vitesse angulaire . . . . .	92
6.3.3	Estimation de l'état de l'aile . . . . .	94
6.3.4	Estimation de l'orientation du fuselage . . . . .	95
6.4	Inversion non linéaire incrémentale de la dynamique du drone . . .	95

---

6.5	Expérimentations . . . . .	97
6.6	Vol avec un contrôleur unifié . . . . .	97
6.7	Commande Udwadia-Kalaba . . . . .	97
6.8	Vols expérimentaux . . . . .	97
6.9	Conclusion du Chapitre 6 . . . . .	97
	<b>Conclusion</b>	<b>99</b>
	<b>A Annexe technique sur les drones</b>	<b>101</b>
A.1	Système de drone : Paparazzi . . . . .	101
A.1.1	Les capteurs d'un autopilote . . . . .	101
A.1.2	Le microcontrôleur d'un autopilote . . . . .	102
A.1.3	Évolutions . . . . .	102
A.1.4	Les logiciels d'un autopilote . . . . .	103
A.1.5	Le segment sol . . . . .	103
A.1.6	Le logiciel embarqué . . . . .	103
A.2	AM32 . . . . .	103
	<b>Bibliographie</b>	<b>105</b>



# Liste des figures

1	Phases de vol d'un drone <i>tailsitters</i> , DarkO. . . . .	2
1.1	Structure <i>quadplanes</i> proposée par [Avy 2023]. . . . .	9
1.2	Structure <i>Tiltrotor</i> proposée par [Flores 2014], dans deux configurations, vol stationnaire et d'avancement. . . . .	10
1.3	Structure <i>tiltrotor</i> proposée par [Smeur 2020, Fernandez 2023b, Pfimlin 2006]. . . . .	10
1.4	Structure <i>tiltwings</i> proposée par [Dufour 2024, Ostermann 2012]. . .	11
1.5	Représentation de la pression dynamique sur la voilure de DarkO lors d'un vol d'avancement. . . . .	13
1.6	Représentation temporelle des modèles de perturbations de vent. . .	15
1.7	Todo. . . . .	16
1.8	Architecture classique de contrôle hiérarchique pour les drones. . .	17
1.9	Méthodes de commande utilisées sur les architectures <i>tailsitters</i> et <i>freewings</i> . . . . .	17
2.1	Vue de dessus d'un autopilote Apogee v1.00. . . . .	20
2.2	Repère de référence de DarkO avec une représentation schématique des actionneurs. . . . .	21
2.3	Réponse entrée-sortie de l'ensemble moteur/hélice. . . . .	27
2.4	Montage d'un pendule bifilaire pour l'identification de l'inertie ( $\mathbf{J}$ ) de DarkO. . . . .	27
2.5	Identification de l'inertie ( $\mathbf{J}$ ), à partir des mesures issues du pendule bifilaire 2.4. . . . .	28
2.6	Montage de DarkO sur un banc de mesure face à une soufflerie ouverte.	29
2.7	Les paramètres ( $\tau$ , $\delta$ , $\theta$ ) de l'ensemble des points d'équilibre (surface) obtenus à l'aide du Théorème 1 et de l'Algorithm 1 pour un vent constant horizontal et vertical ( $w_{rx}, w_{rz}$ ), avec les saturations des actionneurs (rose). . . . .	35
2.8	Section des surfaces de la Figure 2.7 pour $w_{rz} = 0 \text{ m s}^{-1}$ . . . . .	36
3.1	Simulation de la loi de commande non-linéaire avec la dynamique de DarkO (2.1). . . . .	50

3.2	Simulation du modèle complet (2.1) (ligne continue) et simplifié (2.14) (ligne pointillée) avec $\mathbf{u} = \mathbf{u}_{\text{lin}}$ défini dans (3.2) et comme une condition initiale $\tilde{\mathbf{x}}_0$ dans le bassin d'attraction. . . . .	52
3.3	Simulation divergente du modèle complet (2.1) avec $\mathbf{u} = \mathbf{u}_{\text{lin}}$ défini dans (3.2) et une condition initiale $\tilde{\mathbf{x}}_0$ en dehors du bassin d'attraction. . . . .	53
3.4	Simulation en boucle fermée avec le contrôleur hybride (3.4). . . . .	55
4.1	Montage à un degré de liberté. . . . .	58
4.2	Modèle de DarkO à un seul degré de liberté devant le WindShape. . . . .	59
4.3	Architecture d'essai en vol virtuel : WindShape (a) ; capteur de vitesse (b) ; capteur de forces et moments (c) ; encodeur rotatif (d) ; servomoteur (e) ; moteur <i>brushless</i> + ESC (f) ; LabJack (g) ; ordinateur de contrôle (h) . . . . .	60
4.4	Schéma de commande linéaire, proportionnel-intégral. . . . .	63
4.5	Résultats expérimentaux. . . . .	66
5.1	Schéma de commande intégrale avec la perturbation de vent $\mathbf{w}$ , la perturbation du système à l'entrée $\mathbf{d}$ et à la sortie $\nu$ . . . . .	70
5.2	Simulation du modèle non linéaire (2.1) (ligne continue) et du modèle linéarisé (2.32) (ligne pointillée) avec des incrément de vent constants croissants et le contrôleur basé sur l'optimisation sans vent de la section 5.2.2. . . . .	74
5.3	Diagrammes des valeurs singulières des fonctions de transfert dans (5.8) à la première itération de l'algorithme 3. . . . .	79
5.4	Simulation du modèle non linéaire (2.1) (ligne continue) et du modèle linéarisé (2.32) (ligne pointillée) avec des incrément de vent constants croissants, le contrôleur étant réglé à l'aide de l'optimisation multimodèle de l'algorithme 3 dans la section 5.2.3. . . . .	80
5.5	Vol expérimental de DarkO devant la soufflerie ouverte. . . . .	81
5.6	Expérience du drone DarkO devant la soufflerie avec des incrément de vent constants croissants (graphique du bas). . . . .	82
5.7	Visualisation statistique des performances de vol stationnaire. . . . .	83
6.1	Les repères, inertiel (I) et aile (W), attaché à l'architecture de Colibri. . . . .	86
6.2	Repère inertiel (I), du fuselage (F) et de l'aile (W) et forces agissant sur le drone Colibri. . . . .	87
6.3	Simulation de la position et de l'orientation du drone multicorps Colibri en boucle fermée avec un contrôleur à double boucle. . . . .	91

---

6.4	Mesure de l'angle (noir, graphique du haut), mesure de la vitesse angulaire avec le gyroscope de l'aile (bleu, graphique du bas), estimation de la vitesse angulaire par différence finie (vert, graphique du bas) et estimation avec le filtre grand gain (courbes rouges). . . . .	93
6.5	Prototype : Colibri. . . . .	97
6.6	Position et orientation de l'aile dans les six premiers graphiques et mesure de l'angle entre l'aile et le fuselage sur le dernier graphique lors d'un vol réel. . . . .	98
A.1	Todo. . . . .	104



# Liste des tableaux

1.1	Comparaison des architectures de drone. . . . .	8
2.1	Paramètres numériques identifiés du modèle DarkO. . . . .	22
2.2	Longueur et espacement des fils du pendule pour chacun des axes. . .	27
5.1	Écart-type du bruit pour la modélisation des capteurs en simulation.	73
5.2	Valeurs des scalaires de pondération positifs $W_1$ – $W_5$ utilisés dans l'exécution de l'Algorithm 3. . . . .	78
6.1	Paramètres numériques du modèle Colibri. . . . .	86
A.1	Évolution des autopilotes paparazzi sur dix ans. . . . .	103



# Liste des algorithmes

1	Obtention des paramètres d'équilibre en (2.17). . . . .	32
2	Détermination des matrices de la linéarisation de (2.32) . . . . .	44
3	Réglage itératif et multimodèle des gains du contrôleur. . . . .	77



# Liste des abréviations

*ABAG* École Nationale de l'Aviation Civile

*API* Interface de programmation d'application (*Application Programming Interface*)

*DOF* Degrés de liberté (*Degrees of Freedom*)

*EKF* Filtre de Kalman étendu (*Extended Kalman Filter*)

*ENAC* École Nationale de l'Aviation Civile

*ESC* Contrôleurs électroniques de vitesse (*Electronic Speed Controller*)

*GPS* Géo-positionnement par satellite (*Global Positioning System*)

*IMU* Centrales inertielles (*Inertial Measurement Units*)

*INDI* Inversion incrémentale non-linéaire (*Incremental Non-linear Dynamic Inversion*)

*LQR* Commande linéaire quadratique (*Linear Quadratic Regulators*)

*MAV* Micro drone (*Micro Air Vehicle*)

*MCU* Microcontrôleurs (*Microcontroller Unit*)

*MFC* Commande sans modèle (*Model free control*)

*MIMO* Entrées et sorties multiples (*Multiple-Input Multiple-Output*)

*NED* Nord Est Bas (*North, East, Down*)

*PID* Proportionnel Intégral Dérivé

*PLA* Thermoplastique : acide polylactique

*PWM* Modulation de largeur d'impulsions (*Pulse Width Modulation*)

*UAV* Drone autonomes (*Unmanned Aerial Vehicle*)

*VTOL* Drones à décollage et atterrissage verticaux (*Vertical Take-Off and landing*)



# Introduction

## Contexte

Ces dernières années, le domaine des drones s'est considérablement développé. En effet, de nombreux progrès ont été réalisés dans la conduite de vols autonomes, lesquels permettent de réaliser de nombreuses tâches longues, répétitives ou dangereuses, de manière plus sûre que des avions ou des systèmes télépilotés. Les drones ont fait leurs preuves dans de nombreuses applications civiles, alors qu'ils étaient auparavant conçus à des fins de surveillance et de destruction dans le secteur militaire. Tout leur intérêt réside dans leur capacité à se maintenir stabilisé sans intervention humaine. Ainsi, les opérateurs peuvent se concentrer sur la mission, sans devoir consacrer une grande attention au pilotage du drone.

La possibilité d'utiliser des systèmes de vols autonomes dans le secteur civil a été rendu possible par l'accessibilité croissante, proposée par l'industrie, de solutions à faible coût pour les applications d'imagerie aérienne. Ainsi, ce sont dans des domaines aussi variés que l'agriculture de précision, l'inspection des infrastructures civiles ou encore les opérations de sécurité que les drones autonomes sont aujourd'hui mobilisés, devenant alors un riche sujet de recherche.

La miniaturisation des équipements électroniques et mécaniques est à l'origine de l'essor d'une classe de drones de plus en plus petits. Souvent qualifiés de *Micro Air Vehicle* (MAV) ou de *Unmanned Aerial Vehicle* (UAV), leur petite taille leur permet d'intervenir dans des espaces confinés ou contraints. Ils n'ont, cependant, qu'une charge utile restreinte, souvent limitée à l'emport d'une caméra ou d'un colis de faible masse. Leur faible autonomie restreignant leur usage, la recherche s'est alors concentrée sur une solution permettant d'optimiser leur utilisation. En cela, les drones à décollage et atterrissage verticaux (*Vertical take-off and landing*; VTOL) répondent aux exigences.

Dans l'ensemble de VTOL, plusieurs architectures existent et seront détaillé dans la section 1.1.2. Toutefois, nos travaux se sont concentré sur les classes des *tailsitters* et des *freewings*.

De nombreux travaux ont été menés sur les *tailsitters*, avec l'objectif de couvrir l'intégralité du domaine de vol. Ce dernier est constitué des phases de vol suivantes :

1. Décollage vertical
2. Transition entre le vol stationnaire et le vol d'avancement

3. Vol d'avancement
4. Transition entre le vol d'avancement et le vol stationnaire
5. Atterrissage vertical

Bien que l'on puisse observer une symétrie dans la phase ① et ⑤, qui correspondent au décollage et à l'atterrissage vertical, une différence fondamentale est observée. Lors du décollage, la vitesse du drone engendrera un flux d'air sur l'aile orienté dans le même sens que le flux d'air généré par les hélices. Cependant, lors de l'atterrissage, le flux d'air va se trouver inversé, le drone devant descendre, ce qui engendre une vitesse opposée à la direction du flux d'air des hélices. Cette inversion génère une instabilité qui doit être compensée par le contrôleur.

Le vecteur  $\vec{W}$  représente la perturbation de vent qui peut affecter le vol sur l'intégralité des cinq phases de vol. Toutefois, on observe que dans les phases décollage ①, de transition ② et ④ et d'atterrissage ⑤, le drone offre une grande surface verticale sujette au vent. Ainsi, il est nécessaire de traiter l'impact du vent sur cette architecture.

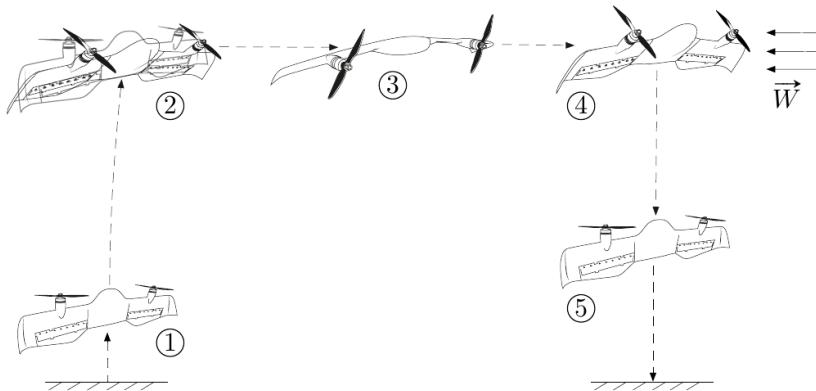


FIGURE 1 – Phases de vol d'un drone *tailsitters*, DarkO.

Nous pouvons citer en exemple le *tailsitter* à double rotors appelé « T-Wing » [Stone 2002, Stone 2008], un autre tail-sitter appelé « MavIon » [Lustosa 2015], ou le « JLion » et le « KH-Lion » [Ke 2017b]. Ces drones partagent une architecture similaire basée sur une aile supportant deux moteurs sur le bord d'attaque et soufflant deux élévons situés sur le bord de fuite. Cette architecture offre une plus grande robustesse que les *tiltrotors*, composés de pièces mobiles, ce qui les rend plus fragiles et d'un actionneur puissant pour faire tourner l'ensemble moteur-hélice. La complexité inhérente à ces architectures nécessite un travail de modélisation en raison des nombreuses non-linéarités et couplages impliqués, en par-

ticulier en termes de modélisation des effets aérodynamiques. Dans ce contexte, l'interférence aérodynamique entre l'aile fixe et les rotors a été modélisée dans [Droandi 2015, Simmons 2022, Aref 2018], et les forces et moments d'hélice générés à des angles d'attaque élevés sont abordés dans [Fernandez 2023a]. Cependant, ces modèles sont complexes et ne sont que partiellement utilisables pour la conception des commandes.

Un autre point important est la représentation de l'attitude du drone. Aussi, il est possible de représenter son orientation par des angles d'Euler [Escareno 2006, Guerrero 2009, Zhang 2017], ce qui permet une compréhension intuitive. Toutefois, une singularité apparaît dans certaines phases de vol. Compte tenu de la grande manœuvrabilité, il est préférable de représenter l'attitude par un quaternion unitaire, ce qui élimine toute singularité [Ke 2017a]. De nombreuses publications modélisent les effets aérodynamiques en fonction de l'angle d'attaque et du dérapage générés par les hélices [Escareno 2007, Chiappinelli 2018]. Il est possible de choisir un autre modèle pour les interactions aérodynamiques entre les moteurs, les ailes et les éléments, comme présenté dans [Lustosa 2019]. La technique de modélisation présentée dans [Lustosa 2019] permet de disposer d'un modèle global couvrant l'ensemble de l'enveloppe de vol, grâce à ce que l'on appelle l'approche  $\Phi$ -théorie. Bien que cette dernière ne permette pas de prédire la chute brutale de la force de portance avec un angle d'attaque (AoA) croissant (qui est causée par un flux d'air turbulent) [Tal 2022], elle permet de représenter le drone avec suffisamment de précision pour capturer le comportement lors de manœuvres agressives.

Actuellement, nous pouvons mentionner deux types d'architecture de commande ayant fonctionné sur ce tailsitter. La première est basée sur une inversion incrémentale non-linéaire de la dynamique du drone (*Incremental Non-linear Dynamic Inversion*, INDI) et la seconde est basée sur la technique sans modèle (*Model free control*, MFC).

Les deux architectures sur lesquelles se sont concentré nos recherches sont celle de DarkO, un *tailsitters* et celle de Colibri un *freewings* basé sur une aile inspiré de DarkO en rotation libre autour d'un fuselage qui sera maintenu horizontal.

## Question de recherche

Notre travail se concentre sur la recherche d'un contrôleur de vol unifié, pour une architecture de drone fortement non-linéaire et couplé, sur l'intégralité du domaine de vol en environnement perturbé.

Rejet de perturbation et métrique de maintien de position

## Plan et contribution

Notre exposé commencera par une description générale des architectures de drone convertible (1) avec une description des avantages et des inconvénients ainsi que leur mode de fonctionnement. Une description générale de la modélisation, de l'actionnement et des lois de commande proposée sur les *tailsitters* et les *freewings* ouvrira nos propos sur ces architectures.

Le chapitre 2 détaillera le modèle non-linéaire d'un drone *tailsitters*, DarkO, à partir des travaux de [Lustosa 2019] et de [Olszanecki Barth 2020]. Nous proposons un modèle simplifié pour les basses vitesses ainsi que le détail des équilibres stationnaire en présence ou non de vent. De ces équilibres, nous présentons la dynamique linéarisée du drone paramétrée par deux scalaires, le vent horizontal et vertical. Ce modèle étant le point de départ des chacun de nos travaux, il se retrouve expliquer dans [Sansou 2022a, Chapitre 2] et [Sansou 2022c, Section II] dans la condition de vent nulle, dans [Sansou 2022b, Section 2] avec des conditions de vent non nulle et dans [Sansou 2024a, Section II] sous sa forme la plus complète.

Le chapitre 4 fera l'objet d'une proposition de loi de commande hybride permettant d'augmenter le domaine de stabilité d'une loi linéaire avec une loi de commande non-linéaire basée sur une direction de zéro moment. Ces travaux ont été publiés dans [Sansou 2022a] et [Sansou 2022c].

Le chapitre 4 permet de décrire une maquette expérimentale utilisée pour évaluer les performances d'une loi de commande basée sur une architecture proportionnelle dérivative. Cette loi est un retour de sortie permettant de stabiliser une position stationnaire en présence de vent. Cette maquette restreint les degrés de liberté classique d'un drone pour se concentrer sur la réjection de perturbation de vent grâce à un changement d'incidence de la maquette. La description de la loi de commande, son optimisation et les résultats sont disponibles dans [Sansou 2022b].

Le chapitre 5 étend le contrôleur proposé dans le chapitre précédent (chapitre 4) pour stabiliser la dynamique complète d'un drone *tailsitter*. La méthode d'obtention des gains du contrôleur est basée sur un algorithme itératif permettant de [Sansou 2024a]

Le chapitre 6 [Sansou 2024b]

Publications

## **Objectifs fixés pour la thèse**

Étudier le comportement d'un drone *tailsitters* en environnement perturbé, saturation des actionneurs, dynamique linéarisé, impact des non-linéarité et cycle limite.

Proposer des architectures de commande basée modèle pour un drone *tailsitters* permettant d'assurer une robustesse aux perturbations de vent, ouvrant la possibilité de certification de ce type d'architecture.

Utiliser des capteurs pour mesurer les perturbations en avance de phase pour les rejeter (sonde 5 trous, micro, Pitot, etc.) ce qui implique de développer une nouvelle architecture



## CHAPITRE 1

# Généralité sur les drones convertible

---

## Sommaire

<b>1.1</b>	<b>Micros drones convertibles</b>	<b>7</b>
1.1.1	Domaine de vol	7
1.1.2	Types d'architecture des drones convertibles	8
<b>1.2</b>	<b>Propriétés des <i>tailsitters</i> et des <i>freetwings</i></b>	<b>12</b>
1.2.1	Actionnement	12
1.2.2	Aérodynamique	12
<b>1.3</b>	<b>Modélisations</b>	<b>13</b>
<b>1.4</b>	<b>Actionnements</b>	<b>15</b>
<b>1.5</b>	<b>Architectures de commande de vol</b>	<b>16</b>
<b>1.6</b>	<b>Méthodes de commande</b>	<b>17</b>
<b>1.7</b>	<b>Technologies et réalisations</b>	<b>18</b>

## 1.1 Micros drones convertibles

### 1.1.1 Domaine de vol

Tout l'intérêt d'un drone convertible réside dans sa capacité à décoller et atterrir verticalement, tout en conservant une bonne efficacité énergétique en vol d'avancement, grâce à une aile. Cette aile a l'avantage de générer de la portance, laquelle s'oppose au poids du drone et permet d'assurer sa sustentation. La contrepartie de la génération de la portance est la traînée qui s'oppose à l'avancement et doit être contrée par une force de traction générée par les hélices. Nous pouvons définir l'efficacité énergétique comme le ratio entre le temps de vol et l'énergie électrique nécessaire pour effectuer ce vol.

Afin de souligner la prééminence de l'efficacité énergétique de ce modèle convertible, il convient de la comparer avec celle d'un drone quadrirotor qui assure sa

sustentation uniquement grâce à des hélices (à l'instar d'un hélicoptère), et à celle d'un drone à voilure fixe.

Architecture	Vitesse $\text{m s}^{-1}$	Stationnaire	Temps de vol	Consommation
Convertible	[0 - 30]	Possible	Quelques heures	Faible
Quadrirotor	[0 - 16]	Possible	Quelques minutes	Forte
Voilure fixe	[8 - 30]	Impossible	Plusieurs heures	Variable

TABLE 1.1 – Comparaison des architectures de drone.

Nous observons qu'un drone convertible possède un domaine de vol bien plus important qu'un drone à voilure fixe, lequel ne sera pas en mesure de voler à très basse vitesse, et qu'un quadrirotor, dont l'autonomie va être limitée par sa consommation. En alliant autonomie et vol stationnaire, le modèle convertible répond aux exigences des missions civiles et militaires.

### 1.1.2 Types d'architecture des drones convertibles

La conception structurelle et aérodynamique d'un drone est le facteur principal permettant des transitions stables et fluides. De plus, il est nécessaire d'optimiser l'architecture pour une mission, de manière à être le plus efficace. Au vu de la diversité des tâches à réaliser, un grand nombre de modèles ont été proposés, lesquels sont généralement catégorisables en trois classes : *tiltrotor*, *tailsitter*, *tiltwing*. En ce qu'il s'agit des prémisses des convertibles, il paraît opportun d'ajouter aux trois grandes catégories précédemment citées dans les états de l'art [Saeed 2018, Ducard 2021, Misra 2022], la catégorie *quadplanes*.

#### *Quadplanes*

Les *quadplanes* résultent de la fusion d'un avion et d'un quadrirotor (comme visible sur la figure 1.1), ce qui permet un découplage de l'actionnement en fonction de la phase de vol. Le premier système de propulsion est composé de quatre hélices générant une force verticale permettant le contrôle lors des phases de décollage, d'atterrissement et stationnaire. Le second système de propulsion est composé d'une hélice propulsive supplémentaire afin de maintenir le vol d'avancement.

L'avantage de ce type d'architecture est sa grande robustesse. Effectivement, aucune pièce en mouvement n'est nécessaire pour réaliser la transition, ce qui réduit le risque de défaillance mécanique. Toutefois, l'inconvénient réside dans son manque d'efficacité. Lors d'un vol d'avancement, la portance sera générée par l'aile. Ainsi, il est possible de désactiver les rotors qui génèrent des perturbations aérodynamiques



FIGURE 1.1 – Structure *quadplanes* proposée par [Avy 2023].

et des trainées parasites. Les axes des moteurs se retrouvent orthogonaux au flux d'air généré par le déplacement du drone, ce qui correspond au cas le plus défavorable en termes de trainée. De plus, la surcharge engendrée par l'emport de moteurs supplémentaires se traduit par une diminution de la charge utile transportable.

Pour ce qui est du contrôle, un atout indéniable est la séparation des actionnements en fonction des phases de vol. Ainsi, l'architecture de commande sera basée sur un mécanisme de commutation permettant de choisir la loi de commande appropriée sur un critère de vitesse air. Ce critère est pertinent en ce qu'il est lié à la capacité de l'aile à générer de la portance induite par le flux d'air. Ainsi, à basse vitesse, le drone se stabilise avec l'actionnement quadrirotor et la loi de commande associée. Dans les vitesses plus importantes, la commutation de la loi permet de contrôler le drone en mode avion. Toutefois, le passage d'une loi à l'autre reste le point clé de la commande et demeure complexe et critique.

### ***Tiltrotor***

Les *tiltrotors* nécessitent l'utilisation d'un actionneur dédié à la modification de l'orientation des moteurs. Les rotors sont montés sur des arbres basculants actionnés et la transition du vol stationnaire au vol d'avancement (ou inversement) s'effectue progressivement en fonction de l'inclinaison du rotor. Les deux configurations sont représentées sur la figure 1.2. Ainsi, l'angle entre le souffle des hélices et l'aile peut être ajusté à chaque instant. Cet angle joue un rôle important dans le contrôle des forces et des moments aérodynamiques : sa maîtrise permet, non seulement, de mieux gérer les performances aérodynamiques du vol lors des transitions, mais aussi la stabilité du système sur l'ensemble du domaine de vol. Malgré le fait que

les *tiltrotor* embarquent un actionneur uniquement dédié à la transition, ce qui augmente la masse du drone, cette architecture est intéressante car elle permet d'utiliser les mêmes actionneurs pour assurer la sustentation en stationnaire et pour générer la traction en mode avion.

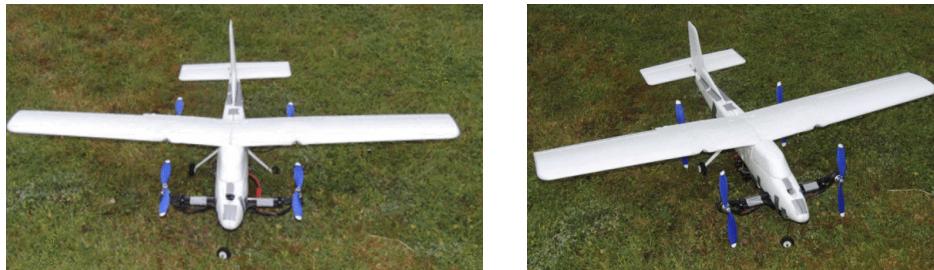


FIGURE 1.2 – Structure *Tiltrotor* proposée par [Flores 2014], dans deux configurations, vol stationnaire et d'avancement.

### ***Tailsitter***

Contrairement aux *tiltrotors* qui se posent sur le fuselage de l'avion (figure 1.2), les *tailsitter* se posent à la verticale (voir figure centrale 1.3). Durant la transition du mode stationnaire au vol d'avancement, la structure entière bascule vers l'avant, modifiant ainsi l'angle d'incidence de la voilure. Selon la configuration du *tailsitter*, la transition peut être réalisée soit par la génération du moment aérodynamique créé par les élévons, soit par le couple créé par le système de propulsion. Pendant le vol d'avancement, en position horizontale, le *tailsitter* vole comme un avion conventionnel sans dérive. En utilisant des techniques aérodynamiques classiques, les concepteurs peuvent optimiser le profil de l'aile du *tailsitter* pour le rendre plus endurant afin de réduire la consommation d'énergie. Grâce à ce processus d'optimisation aérodynamique, le *tailsitter* peut effectuer des missions de vol de plus d'une heure.



FIGURE 1.3 – Structure *tiltrotor* proposée par [Smeur 2020, Fernandez 2023b, Pflimlin 2006].

Ce modèle semble être la configuration la plus intéressante énergétiquement, car il utilise les mêmes actionneurs dans tout le domaine de vol. Ainsi, aucune masse superflue n'est embarquée.

### ***Tiltwing***

La particularité des *tiltwing*s réside dans le fait que les rotors sont inclinés en même temps que les ailes (voir figure 1.4). Un actionneur supplémentaire et puissant est donc nécessaire pour surmonter le couple de l'aile afin de la positionner dans l'orientation souhaitée. La commande de cet actionneur doit être prise en compte lors de la conception des lois de commande. Pendant le décollage, l'atterrissement et les vols stationnaires, les ailes doivent être positionnées vers le haut afin de produire une force de poussée opposée au vecteur gravité. Dans ces phases de vol, lorsque les ailes sont orientées vers le haut, l'aéronef est plus vulnérable au vent et les lois de commande doivent rejeter ces perturbations. Dans la littérature, il existe plusieurs configurations d'ailes basculantes et différentes approches de contrôle conçues pour stabiliser leur dynamique de vol.



FIGURE 1.4 – Structure *tiltwing* proposée par [Dufour 2024, Ostermann 2012].

Par ailleurs, les *freewing* sont une gamme en cours de développement à l'intérieur de l'architecture des *tiltwing*. Ils sont actionnés comme des *tiltwing*, excepté au niveau de l'axe de rotation entre l'aile et le fuselage. Cette rotation est laissée libre, ce degré de liberté permettant à l'aile de changer librement son incidence. Il en résulte un gain de masse car il est possible de supprimer l'actionneur puissant et lourd nécessaire à la rotation de l'aile. De plus, l'aile étant libre de s'orienter, les turbulences ont un impact plus faible sur la structure, ce qui rend les vols plus stables [Welstead 2012, Axtén 2021, Axtén 2023].

## 1.2 Propriétés des *tailsitters* et des *freewings*

D'un point de vue mécanique, les *tailsitters* et les *freewings* sont caractérisés comme des systèmes sous-actionnés avec une dynamique fortement couplée. Ces caractéristiques mécaniques rendent le processus de modélisation et d'identification difficile. Cela peut s'expliquer par le fait que, pour ce type de système, une entrée de commande donnée agit simultanément sur différentes dynamiques. Ainsi, l'identification de l'influence d'une entrée de commande donnée sur une dynamique particulière reste un processus important qui nécessite plus d'attention.

### 1.2.1 Actionnement

Dans ces deux architectures, il est courant de trouver des actionneurs basés sur des effets aérodynamiques. Ces actionneurs ont l'avantage d'être peu consommateurs en énergie. Ils sont mus par des servomoteurs qui consomment peu d'électricité proportionnellement au couple qu'ils génèrent. Dans le cas des ailes volantes, les surfaces aérodynamiques sont souvent placées sur la partie arrière des ailes et peuvent être utilisées symétriquement à l'instar de volets, ou anti-symétriquement comme des ailerons. Dans les phases de décollage, de vol stationnaire ou d'atterrissage, la plateforme est maintenue en vol par les hélices : il est donc nécessaire de dimensionner les groupes moteurs-hélices pour qu'ils puissent générer assez de force. En fonction des configurations, les moments peuvent être obtenus par des différentiels sur l'utilisation des moteurs ou bien par des surfaces aérodynamiques. Dans le cas de surfaces soufflées par le flux d'air des hélices, il existe un couplage des actionneurs qui complexifie la modélisation et le contrôle de ces architectures.

### 1.2.2 Aérodynamique

Lors d'un vol d'avancement, les *tailsitters* et les *freewings* assurent le maintien du vol en palier, en générant une force de portance grâce à leur surface alaire. Cette portance, qui s'opposant au poids, permet de maintenir une trajectoire. La force de trainée engendrée par l'aile et le fuselage est compensée par la composante horizontale de la poussée. En vol stationnaire, le poids est compensé par la traction de l'hélice. La relation d'équilibre est plus complexe lorsque nous évaluons l'ensemble des points d'équilibre lors de la transition d'un mode à l'autre. La transition équilibrée est assurée par le mélange correct des forces aérodynamiques et de propulsion.

Les interactions aérodynamiques entre l'hélice et la surface de l'aile sont connues pour être complexes et difficiles à modéliser. La vitesse induite par le souffle de l'hé-

l'effet de l'hélice entraîne une variation locale de l'angle d'attaque et des variations de pression dynamique au niveau des sections d'aile immergées, d'où une distribution différente de la portance. L'analyse de l'interaction entre l'hélice, l'aile et les élévons doit permettre d'obtenir de bonnes caractéristiques de vol et de tirer profit des combinaisons.

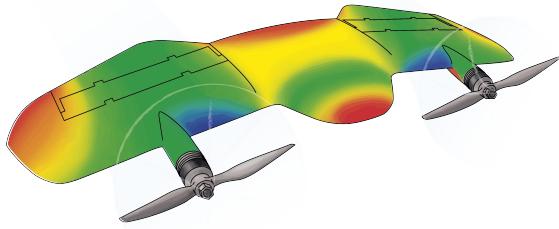


FIGURE 1.5 – Représentation de la pression dynamique sur la voilure de DarkO lors d'un vol d'avancement.

L'identification de ces effets aérodynamiques nécessite, pour chaque point de vol, les informations de l'air environnant, les valeurs des entrées de commande et les mesures des forces et moments aérodynamiques agissant sur le système. Le moyen le plus précis et le plus fiable d'identifier les phénomènes aérodynamiques est probablement de mener des campagnes en soufflerie, lesquelles sont particulièrement chronophages et coûteuses.

### 1.3 Modélisations

Dans le cas des drones convertible basculant, le modèle aérodynamique doit être correct dans les phases de transition. Cela nécessite l'extension de l'aérodynamique classique (faible incidence) à un angle d'attaque élevé et à un fonctionnement à faible vitesse. En outre, l'effet du souffle de l'hélice sur les profils aérodynamiques du véhicule doit être compris et pris en compte dans le modèle du système. Par exemple, les travaux de [Rohr 2021] sur un *tiltwing* identifient les zones d'interaction entre l'hélice et l'aile. Ils séparent clairement la génération de force et de couple lesquelles sont obtenues par la partie soufflée et non-soufflée de l'aile.

Des fonctions non linéaires continues décrivant les coefficients de portance et de traînée aérodynamiques sur toute la plage de l'angle d'attaque pour les *tiltrotors* ont été dérivées dans [Ducard 2014] et pour les *tiltwing* dans [Lustosa 2019, Lustosa 2017].

Il existe de nombreuses conceptions possibles pour un drone convertible. Bien que tous les modèles aient une structure commune, il existe des différences majeures dans la formulation réelle des termes de force et de couple. Celles-ci dépendent de la disposition des moteurs ou des hélices, de l'existence ou non de surfaces de contrôle aérodynamiques et de la forme du véhicule.

Un modèle précis est nécessaire pour les conceptions classiques de contrôle basées sur un modèle et en particulier pour les approches d'inversion dynamique ou de contrôle prédictif de modèle. Cependant, une modélisation précise va nécessiter une campagne d'identification approfondie en soufflerie ou un vol en environnement contraint. Il se peut également que la complexité du modèle ne permette pas une utilisation directe dans le contrôle à cause de limitation matérielle des calculateurs embarqué.

Dans notre cas, nous nous sommes intéressés à l'impact du vent sur les architectures mentionnées précédemment. Il est donc nécessaire de modéliser le vent. Dans le cas de vent constant ou de cisaillement de vent, un modèle à échelons semble tout à fait approprié pour représenter le changement brusque de vitesse de vent. Pour les rafales, nous utiliserons plusieurs modèles, tel que le modèle "Chapeau mexicain" ou les "ondelettes de Morlet".

La fonction définissant le chapeau mexicain est :

$$\Psi_{mex}(t) = w_m - \frac{A_g}{2} (1 - \cos(2\pi f_g(t - t_{0,mex}))) \sin(3\pi f_g(t - t_{0,mex})) \quad (1.1)$$

et la fonction ondelettes de Morlet est défini par :

$$\Psi_{mor}(t) = w_m + A_g \exp(-(t - t_{0,mor})) \cos(5 * (t - t_{0,mor})) \quad (1.2)$$

où  $w_m$  est le vent moyen sans perturbation,  $t_{0,mex}$  représente l'instant de début de perturbation et  $t_{0,mor}$  l'instant où la perturbation est maximale,  $A_g$  est l'intensité maximale de la perturbation et  $f_g$  est la fréquence de la perturbation. Les tracés de la figure 1.6 montrent la représentation temporelle des deux fonctions pour les valeurs  $w_m = 1 \text{ m s}^{-1}$ ,  $t_{0,mex} = 2 \text{ s}$ ,  $t_{0,mor} = 5 \text{ s}$ ,  $A_g = 1 \text{ m s}^{-1}$  et  $f_g = 0.8 \text{ rad s}^{-1}$ .

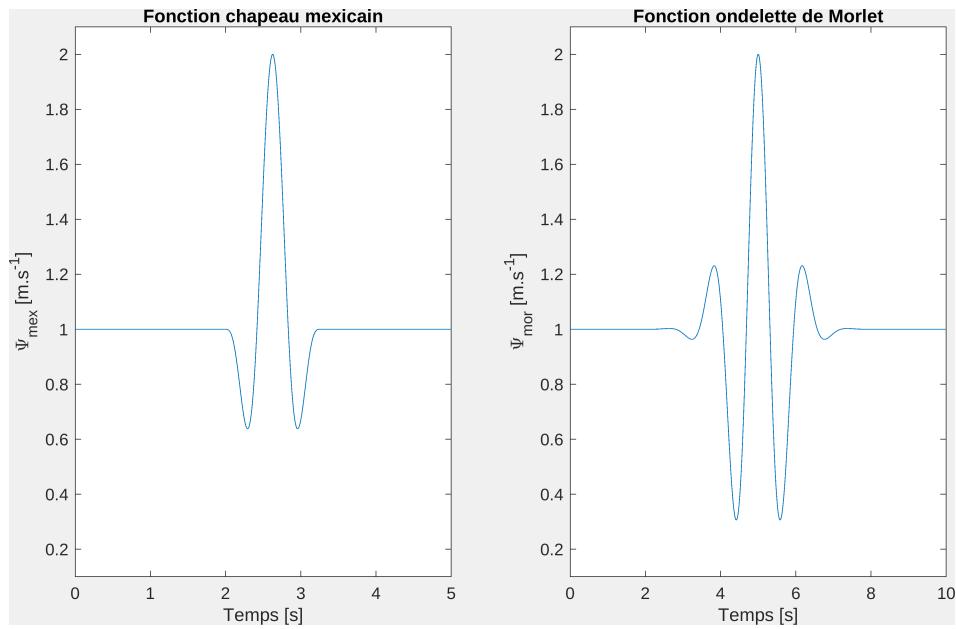


FIGURE 1.6 – Représentation temporelle des modèles de perturbations de vent.

Le modèle de Dryden est défini :

$$\Phi_{u_g}(\Omega) = \sigma_u^2 \frac{2L_u}{\pi} \frac{1}{1 + (L_u\Omega)^2} \quad (1.3a)$$

$$\Phi_{v_g}(\Omega) = \sigma_v^2 \frac{2L_v}{\pi} \frac{1 + 12(L_v\Omega)^2}{(1 + 4(L_v\Omega)^2)^2} \quad (1.3b)$$

$$\Phi_{w_g}(\Omega) = \sigma_w^2 \frac{2L_w}{\pi} \frac{1 + 12(L_w\Omega)^2}{(1 + 4(L_w\Omega)^2)^2} \quad (1.3c)$$

Le modèle de von Kármán :

explication

$$\Phi_{u_g}(\Omega) = \sigma_u^2 \frac{2L_u}{\pi} \frac{1}{(1 + (1.339L_u\Omega)^2)^{\frac{5}{6}}} \quad (1.4a)$$

$$\Phi_{v_g}(\Omega) = \sigma_v^2 \frac{2L_v}{\pi} \frac{1 + \frac{8}{3}(2.678L_v\Omega)^2}{(1 + (2.678L_v\Omega)^2)^{\frac{11}{6}}} \quad (1.4b)$$

$$\Phi_{w_g}(\Omega) = \sigma_w^2 \frac{2L_w}{\pi} \frac{1 + \frac{8}{3}(2.678L_w\Omega)^2}{(1 + (2.678L_w\Omega)^2)^{\frac{11}{6}}} \quad (1.4c)$$

## 1.4 Actionnements

La figure 1.7 illustre les effets des actionneurs sur la dynamique de l'attitude des *tailsitters*. La rotation autours de l'axe  $x_b$  peut être contrôlé par des braquages

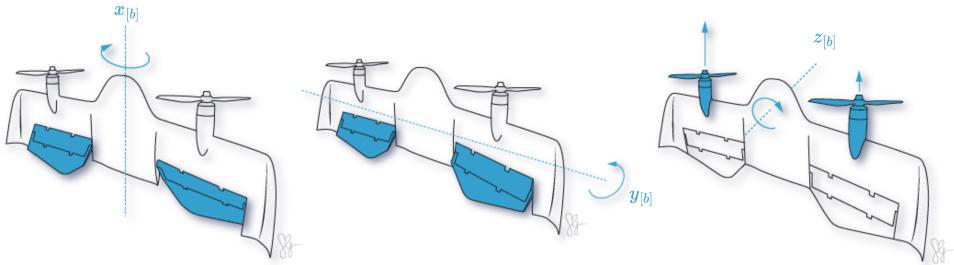


FIGURE 1.7 – Todo.

de volets asymétriques, l'angle de tangage (axe  $y_b$ ) par des braquages de volets symétriques et la rotation autour de l'axe  $z_b$  est contrôlée par un dispositif de poussée différentielle, ce qui permet d'obtenir un angle de tangage plus élevé. La rotation autour de l'axe de lacet est contrôlée par une configuration de poussée différentielle, qui crée un couple temporaire non nul autour de son axe. Le dispositif de poussée différentielle engendre une différence entre les rotations de l'hélice gauche et de l'hélice droite, et modifie ainsi la vitesse de rotation de l'avion et donc le comportement aérodynamique autour des volets.

## 1.5 Architectures de commande de vol

Une approche en cascade est une méthode couramment utilisée dans le cadre de la commande de drone. Cette approche est basée sur l'imbrication de boucle de commande illustrée sur la Figure 1.8. Cette architecture s'appuie sur la séparation des dynamiques d'un drone sous actionné. Effectivement, comme le drone est obligé de modifier son orientation pour changer de position, le schéma de contrôle prend en compte cette spécificité. Ainsi la boucle interne est réglée pour avoir un temps de réponse faible vis-à-vis des autres lois. Cette boucle interne stabilise l'attitude du drone  $att_m$  en la faisant converger vers l'attitude de consigne  $att_c$ . Cette attitude de consigne est le résultat de la boucle de vitesse qui fait converger la vitesse du drone  $vel_m$  vers la vitesse de consigne  $vel_c$ . La vitesse de consigne est la résultante de la boucle de position qui fait converger la position du drone  $pos_m$  vers la position de consigne définie par le pilote  $pos_c$ .

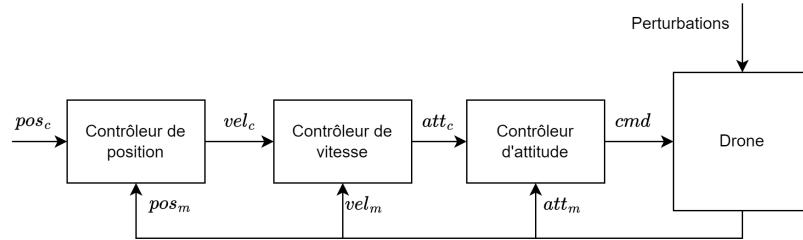
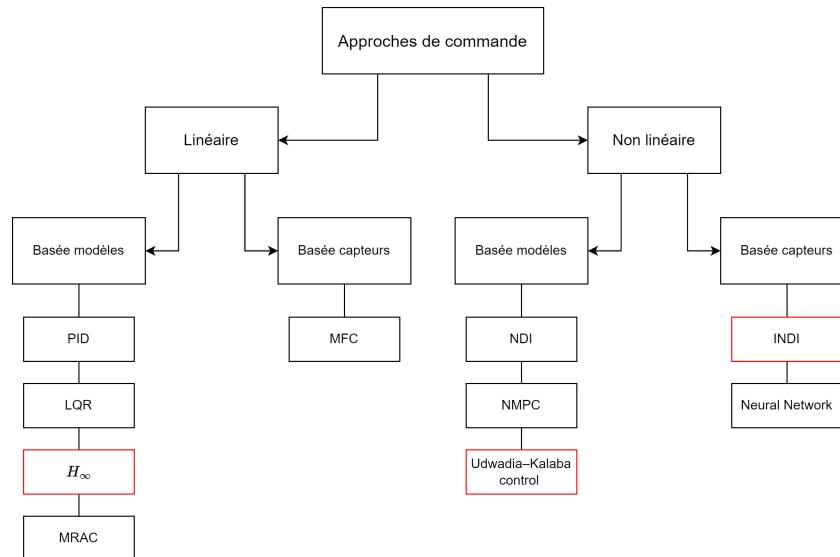


FIGURE 1.8 – Architecture classique de contrôle hiérarchique pour les drones.

FIGURE 1.9 – Méthodes de commande utilisées sur les architectures *tailsitters* et *freewings*.

## 1.6 Méthodes de commande

Faire un schéma avec les méthodes de commande et encadré en rouge nos lois état de l'art

### Contrôleur PID

Le contrôle Proportionnel Intégral Dérivé (PID) (ou P/PD/PI) est couramment utilisé, car son réglage est intuitif et ne nécessite qu'une connaissance limitée du système. Le contrôle PID donne souvent de très bons résultats et constitue un excellent point de départ pour la conception de contrôleurs plus avancés.

La forme générale d'un contrôleur PID est la suivante :

$$u_{PID}(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (1.5)$$

**Contrôleur LQR**

**Contrôleur  $H_\infty$**

**Contrôleur non linéaire**

**Contrôleur séquencé**

**Contrôleur adaptatif**

## 1.7 Technologies et réalisations

Faire un schéma des composants avec nos contributions

quelques lignes sur  
contributions  
ns paparazzi et  
32  
nclusion

## CHAPITRE 2

# Modélisation d'un drone convertible : DarkO

---

## Sommaire

---

<b>2.1</b>	<b>Modèle du drone DarkO</b>	<b>19</b>
2.1.1	Modèle non-linéaire complet	21
2.1.2	Modèle non linéaire simplifié à basse vitesse	24
<b>2.2</b>	<b>Identification des paramètres du modèle</b>	<b>26</b>
2.2.1	Modélisation des actionneurs	29
<b>2.3</b>	<b>Équilibres stationnaires</b>	<b>30</b>
2.3.1	Équilibre stationnaire sans vent	30
2.3.2	Équilibre stationnaire en présence de vent	31
<b>2.4</b>	<b>Dynamiques linéarisées</b>	<b>36</b>
2.4.1	Dynamique linéarisée sans vent	36
2.4.2	Dynamique linéarisée en présence de vent	37
<b>2.5</b>	<b>Conclusion du Chapitre 2</b>	<b>43</b>

---

## 2.1 Modèle du drone DarkO

DarkO, drone conçu et développé à l'École Nationale de l'Aviation Civile (ENAC) de Toulouse (France), est un exemple clair de drone convertible avec une architecture dite *tailsitter*. DarkO est assemblé à partir de plusieurs pièces d'Onyx imprimées en 3D (un matériau très robuste composé de fibres de carbone omnidirectionnelles). Toutes les pièces sont emboîtées sur un seul axe, de sorte que le drone puisse facilement être démonté pour remplacer des pièces ou accéder à l'électronique embarquée.

L'autopilote embarqué est une carte Apogee<sup>1</sup> fabriquée à l'ENAC, voir Fig. 2.1.

---

1. <https://wiki.paparazziuav.org/wiki/Apogee/v1.00>

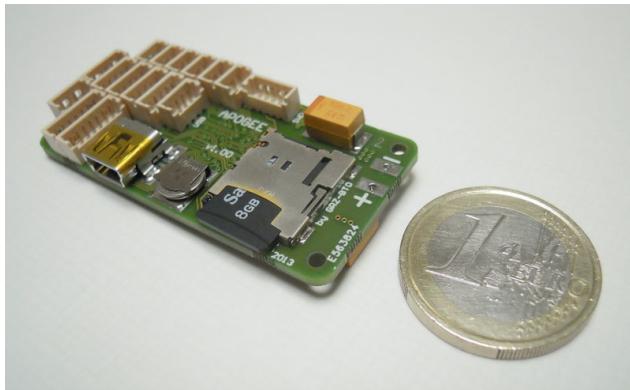


FIGURE 2.1 – Vue de dessus d'un autopilote Apogee v1.00.

L'autopilote offre la possibilité d'enregistrer les données de bord sur une carte mémoire SD, à la fréquence de contrôle de 500 Hz, ce qui permet un post-traitement efficace des données acquises. Le protocole de communication utilisé entre l'autopilote et les contrôleurs électroniques de vitesse (ESC) est le Dshot 600. Les ESC sont des AIKON AK32 35A avec un firmware AM32. La communication sol-bord est réalisée via un canal bidirectionnel basé sur des modules XBee-PRO S1.

Les actionneurs de DarkO peuvent être décomposés en deux catégories. La première est composée de deux hélices (T-Motor T5147) placées symétriquement à l'avant de l'aile (illustrées en **noir** dans la Fig. 2.2) et alimentées par deux moteurs électriques (T-Motor F30 2300kv) générant une traction selon l'axe  $x_b$ . La seconde catégorie est relative aux actionneurs aérodynamiques. Ainsi, le drone possède deux élévons, placés à l'arrière de l'aile (illustrés en **bleu** dans la Fig. 2.2), agissant en tant que surface de contrôle. Les élévons génèrent des forces et des moments en modifiant leur incidence relativement au flux d'air dans lequel ils sont placés. Ce flux d'air peut être généré par le vent relatif (lié à la vitesse du drone), le vent extérieur, mais aussi par le souffle des hélices. Les élévons sont commandés par deux servomoteurs MKS DS65K.

La figure 2.2 montre le modèle de DarkO, ainsi qu'un repère de référence inertiel *North, east, down* (NED) (ou repère terrestre) “i” lié à la surface de la Terre, et un repère corps ‘b’ attaché au drone, avec  $x_b$  correspondant à l'axe de roulis (l'axe des hélices dans le plan  $z_b = 0$ ),  $y_b$  l'axe de tangage (la direction des ailes),  $z_b$  l'axe de lacet. En utilisant la même notation que dans [Lustosa 2019], le couple hélice/élévon gauche et droit est désigné par les indices  $i = 1$  (gauche) et  $i = 2$  (droite). La convention de signe sera définie comme positive pour les positions des élévons  $\delta_1, \delta_2$  lorsqu'ils créent un moment à cabrer avec les hélices tournant dans des

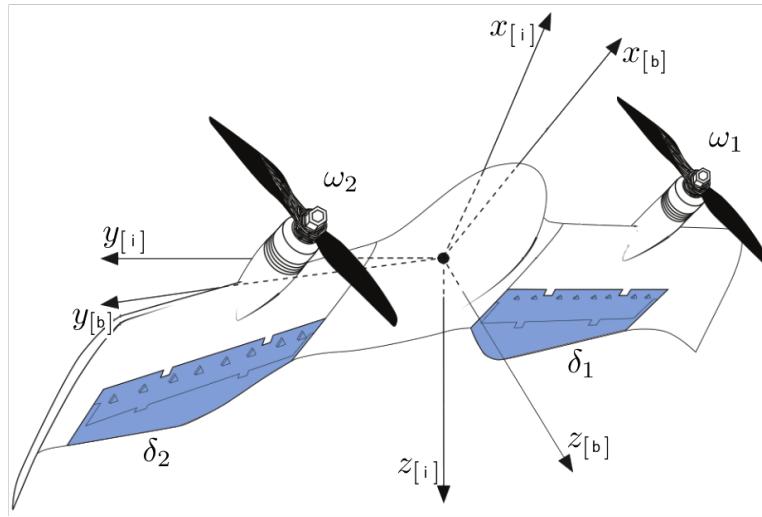


FIGURE 2.2 – Repère de référence de DarkO avec une représentation schématique des actionneurs.

directions opposées avec des vitesses angulaires  $\omega_1 > 0$  et  $\omega_2 < 0$ , respectivement.

### 2.1.1 Modèle non-linéaire complet

En exploitant la modélisation présentée dans [Lustosa 2019] et [Olszanecki Barth 2020], un modèle précis de la dynamique de DarkO décrit la position  $\mathbf{p} \in \mathbb{R}^3$  du centre de gravité et sa vitesse  $\mathbf{v} = \dot{\mathbf{p}} \in \mathbb{R}^3$ , son orientation, bien représentée par un quaternion  $\mathbf{q} \in \mathbb{S}^3 := \{\mathbf{q} \in \mathbb{R}^4 : |\mathbf{q}| = 1\}$ , et sa vitesse angulaire  $\boldsymbol{\omega}_b$ , représentée dans le repère du corps, qui satisfait  $\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes [\begin{smallmatrix} 0 \\ \boldsymbol{\omega}_b \end{smallmatrix}]$ , où  $\otimes$  représente le produit hamiltonien (voir [Lustosa 2019, Olszanecki Barth 2020] ou le tutoriel [Hua 2013] pour plus de détails). En choisissant l'état global comme produit hamiltonien

$\mathbf{x} := (\mathbf{p}, \mathbf{v}, \mathbf{q}, \boldsymbol{\omega}_b)$ , le modèle mathématique, dérivé dans [Lustosa 2019], dépend d'un ensemble de paramètres énumérés dans la Table 2.1, où nous indiquons également la valeur obtenue à partir d'une identification du système [Sansou 2022a]. Le modèle dynamique peut être écrit comme ci-dessous :

Paramètres et coefficients	Valeurs	Unités
$m$ (Masse du drone)	0.519	kg
$b$ (Envergure)	0.542	m
$c$ (Corde aérodynamique)	0.13	m
$\mathbf{B} = \text{diag}(b, c, b)$	$\text{diag}(0.542, 0.13, 0.542)$	m
$S$ (Surface de l'aile)	0.026936	$\text{m}^2$
$S_{\text{wet}}$ (Surface soufflée)	0.0180	$\text{m}^2$
$S_p$ (Surface des hélices)	0.0127	$\text{m}^2$
$\mathbf{J} = \text{diag}(J_x, J_y, J_z)$	$\text{diag}(0.0067, 0.0012, 0.0082)$	$\text{kg m}^2$
$k_f$ (Poussée des hélices)	1.7800e-8	$\text{kg m}$
$k_m$ (Moment des hélices)	2.1065e-10	$\text{kg m}^2$
$p_x$ (Position en $x$ des hélices)	0.065	m
$p_y$ (Position en $y$ des hélices)	0.162	m
$a_y$ (Position en $y$ de la portance)	0.1504	m
$\xi_f$ (Portance des élevons)	0.2	—
$\xi_m$ (Moment des élevons)	1.4	—
$\rho$ (Densité de l'air)	1.225	$\text{kg m}^{-3}$
$C_d$ (Trainé)	0.1644	—
$C_y$ (Latéral)	0	—
$C_\ell$ (Portance)	5.4001	—
$\Delta_r$ (Centrage du drone)	-0.0145	m

TABLE 2.1 – Paramètres numériques identifiés du modèle DarkO.

$$\dot{\mathbf{p}} = \mathbf{v} \quad (2.1a)$$

$$m\dot{\mathbf{v}} = -mg + \mathbf{R}(\mathbf{q})\mathbf{F}_b, \quad (2.1b)$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}_b \quad (2.1c)$$

$$\mathbf{J}\dot{\boldsymbol{\omega}}_b = -[\boldsymbol{\omega}_b]_\times \mathbf{J}\boldsymbol{\omega}_b + \mathbf{M}_b, \quad (2.1d)$$

où  $\mathbf{g} := [0 \ 0 \ 9.81]^\top$  désigne le vecteur de gravité,  $m \in \mathbb{R}$  est la masse,  $\mathbf{J} \in \mathbb{R}^{3 \times 3}$  est le moment d'inertie diagonal (voir Table 2.1) et en partitionnant le quaternion  $\mathbf{q} \in \mathbb{S}^3$  comme  $\mathbf{q} := [\eta \ \boldsymbol{\epsilon}^\top]^\top$ , la matrice de rotation correspondante est  $\mathbf{R}(\mathbf{q}) \in SO(3) := \{\mathbf{R} \in \mathbb{R}^{3 \times 3} : \mathbf{R}^\top \mathbf{R} = \mathbb{I}_3, \det(\mathbf{R}) = 1\}$  est défini comme (voir [Hua 2013])

$$\mathbf{R}(\mathbf{q}) := \mathbb{I}_3 + 2\eta[\boldsymbol{\epsilon}]_\times + 2[\boldsymbol{\epsilon}]_\times^2. \quad (2.2)$$

D'après [Lustosa 2019], le vecteur de force  $\mathbf{F}_b$  et le vecteur de moment  $\mathbf{M}_b$  dans (2.1) dépendent (i) de l'état du système  $\mathbf{x}$ , (ii) de la perturbation  $\mathbf{w} \in \mathbb{R}^3$ , représentant la vitesse du vent dans le référentiel inertiel, et (iii) la commande des

actionneurs (voir Figure 2.2), comprenant la vitesse de rotation des deux hélices  $\omega_1, \omega_2 \in \mathbb{R}$  et la déflexion des élevons  $\delta_1, \delta_2 \in \mathbb{R}$ .

Considérons d'abord l'effet des commandes des actionneurs. Chaque hélice génère une poussée  $\mathbf{T}_i$  orientée dans la direction  $x$  du repère corps et un moment  $\mathbf{N}_i$  selon le même axe :

$$\mathbf{T}_i := \begin{bmatrix} \tau_i \\ 0 \\ 0 \end{bmatrix} := \begin{bmatrix} k_f \omega_i^2 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{N}_i := (-1)^i \frac{k_m}{k_f} \mathbf{T}_i, \quad i = 1, 2. \quad (2.3)$$

La position de chaque élévon  $\delta_i \in \mathbb{R}$  est assignée par un servomoteur qui impose un niveau d'efficacité (en termes de déviation du courant d'air) quantifié par deux matrices antisymétriques :

$$\Delta_i^f := \begin{bmatrix} 0 & 0 & \xi_f \delta_i \\ 0 & 0 & 0 \\ -\xi_f \delta_i & 0 & 0 \end{bmatrix}, \quad \Delta_i^m := \begin{bmatrix} 0 & 0 & \xi_m \delta_i \\ 0 & 0 & 0 \\ -\xi_m \delta_i & 0 & 0 \end{bmatrix}, \quad i = 1, 2. \quad (2.4)$$

Les paramètres constants  $k_f, k_m, \xi_f, \xi_m$  apparaissant dans (2.3) et (2.4) sont listés dans la Table 2.1.

Avec les quantités ci-dessus, nous pouvons réarranger la dynamique donnée dans la Table suivante [Lustosa 2019, eqns (97), (98)] (voir aussi [Sansou 2022a]) et exprimer  $\mathbf{F}_b$  et  $\mathbf{M}_b$  dans (2.1) comme :

$$\begin{aligned} \mathbf{F}_b := \mathbf{T}_1 + \mathbf{T}_2 + \frac{S_{\text{wet}}}{4S_p} \Phi^{(fv)} & \left( (\Delta_1^f - \mathbb{I}_3) \mathbf{T}_1 + (\Delta_2^f - \mathbb{I}_3) \mathbf{T}_2 \right) \\ & + \frac{1}{4} \rho S \Phi^{(fv)} (\Delta_1^f + \Delta_2^f - 2\mathbb{I}_3) \|\mathbf{v}_b\| \mathbf{v}_b \\ & + \frac{1}{4} \rho S \Phi^{(mv)} (\Delta_1^f + \Delta_2^f - 2\mathbb{I}_3) \mathbf{B} \|\mathbf{v}_b\| \boldsymbol{\omega}_b, \end{aligned} \quad (2.5)$$

$$\begin{aligned}
\mathbf{M}_b := \mathbf{N}_1 + \mathbf{N}_2 + \left[ \begin{smallmatrix} p_x \\ p_y \\ 0 \end{smallmatrix} \right]_{\times} \mathbf{T}_1 + \left[ \begin{smallmatrix} p_x \\ -p_y \\ 0 \end{smallmatrix} \right]_{\times} \mathbf{T}_2 & \quad (2.6) \\
& - \frac{S_{\text{wet}}}{4S_p} \left( \mathbf{B} \Phi^{(\text{mv})} (\Delta_1^m - \mathbb{I}_3) + \left[ \begin{smallmatrix} 0 \\ a_y \\ 0 \end{smallmatrix} \right]_{\times} \Phi^{(\text{fv})} (\Delta_1^m + \mathbb{I}_3) \right) \mathbf{T}_1 \\
& - \frac{S_{\text{wet}}}{4S_p} \left( \mathbf{B} \Phi^{(\text{mv})} (\Delta_2^m - \mathbb{I}_3) + \left[ \begin{smallmatrix} 0 \\ -a_y \\ 0 \end{smallmatrix} \right]_{\times} \Phi^{(\text{fv})} (\Delta_2^m + \mathbb{I}_3) \right) \mathbf{T}_2 \\
& + \frac{1}{4} \rho S \left( \left( \left[ \begin{smallmatrix} 0 \\ a_y \\ 0 \end{smallmatrix} \right]_{\times} \Phi^{(\text{fv})} + \mathbf{B} \Phi^{(\text{mv})} \right) \Delta_1^m \right. \\
& \left. + \left( \left[ \begin{smallmatrix} 0 \\ -a_y \\ 0 \end{smallmatrix} \right]_{\times} \Phi^{(\text{fv})} + \mathbf{B} \Phi^{(\text{mv})} \right) \Delta_2^m - 2 \mathbf{B} \Phi^{(\text{mv})} \right) \|\mathbf{v}_b\| \mathbf{v}_b \\
& + \frac{1}{4} \rho S \left( \left( \left[ \begin{smallmatrix} 0 \\ a_y \\ 0 \end{smallmatrix} \right]_{\times} \Phi^{(\text{mv})} + \mathbf{B} \Phi^{(\text{m}\omega)} \right) \Delta_1^m \right. \\
& \left. + \left( \left[ \begin{smallmatrix} 0 \\ -a_y \\ 0 \end{smallmatrix} \right]_{\times} \Phi^{(\text{mv})} + \mathbf{B} \Phi^{(\text{m}\omega)} \right) \Delta_2^m - 2 \mathbf{B} \Phi^{(\text{m}\omega)} \right) \mathbf{B} \|\mathbf{v}_b\| \boldsymbol{\omega}_b,
\end{aligned}$$

où  $\mathbf{v}_b := \mathbf{R}^T(\mathbf{q})(\mathbf{v} - \mathbf{w})$  représente la vitesse de l'air vu par le drone exprimée dans le repère du corps. Dans [Lustosa 2019], la valeur  $\|\mathbf{v}_b\|$ , apparaissant dans les expressions de  $\mathbf{F}_b$  et  $\mathbf{M}_b$ , est remplacé par la valeur  $\eta = \sqrt{\|\mathbf{v}_b\|^2 + \mu c^2 \|\boldsymbol{\omega}_b\|^2}$ , avec  $\mu \in \mathbb{R}$  étant un paramètre lié à l'identification du modèle. Toutefois, dans le cas de DarkO, l'identification fournit  $\mu = 0$  [Sansou 2022a]. Dès lors, nous présentons ici une description simplifiée. La matrice des coefficients aérodynamiques constants  $\Phi := \begin{bmatrix} \Phi^{(\text{fv})} & \Phi^{(\text{mv})} \\ \Phi^{(\text{mv})} & \Phi^{(\text{m}\omega)} \end{bmatrix} \in \mathbb{R}^{6 \times 6}$ , est défini dans [Olszanecki Barth 2020, eqs. (6)–(9)] comme  $\Phi^{(\text{fv})} := \text{diag}(C_d, C_y, C_\ell)$  et

$$\begin{aligned}
\left[ \begin{array}{c|c} \Phi^{(\text{mv})} & \Phi^{(\text{m}\omega)} \end{array} \right] := \\
\left[ \begin{array}{ccc|ccc} 0 & 0 & 0 & 0.1396 & 0 & 0.0573 \\ 0 & 0 & -\frac{\Delta_r}{c} C_\ell & 0 & 0.6358 & 0 \\ 0 & 0 & 0 & 0.0405 & 0 & 0.0019 \end{array} \right].
\end{aligned}$$

Les valeurs numériques des constantes figurant dans la Table 2.1 (ces valeurs numériques n'ont pas été indiquées dans [Lustosa 2019] et [Olszanecki Barth 2020] et sont données ici pour permettre de reproduire les résultats de nos simulations).

### 2.1.2 Modèle non linéaire simplifié à basse vitesse

Dans la mesure où nous allons nous intéresser au maintien du drone en stationnaire, c'est-à-dire avec une vitesse du drone faible, nous pouvons simplifier la dynamique (2.1) en négligeant les effets aérodynamiques quadratiques dus à la vi-

tesse  $\mathbf{v}_b$  et à la vitesse angulaire  $\boldsymbol{\omega}_b$  dans (2.5) et (2.6). Nous définissons le vecteur de commande :

$$\mathbf{u} := \begin{bmatrix} \tau_1 & \tau_2 & \delta_1 & \delta_2 \end{bmatrix}^\top, \quad (2.7)$$

lequel permet d'obtenir le modèle basse vitesse comportant les effets majeurs non-linéaires du vent :

$$\dot{\mathbf{p}} = \mathbf{v}, \quad (2.8a)$$

$$m\dot{\mathbf{v}} = -m\mathbf{g} + \mathbf{R}(\mathbf{q})\left(\mathbf{M}_f(\mathbf{u}) + \mathbf{D}_f(\mathbf{u})\|\mathbf{w}\|\mathbf{R}^\top(\mathbf{q})(\mathbf{v} - \mathbf{w})\right), \quad (2.8b)$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \begin{bmatrix} 0 \\ \omega_b \end{bmatrix}, \quad (2.8c)$$

$$\mathbf{J}\dot{\boldsymbol{\omega}}_b = -[\boldsymbol{\omega}_b]_\times \mathbf{J}\boldsymbol{\omega}_b + \mathbf{M}_m(\mathbf{u}) + \mathbf{D}_m(\mathbf{u})\|\mathbf{w}\|\mathbf{R}^\top(\mathbf{q})(\mathbf{v} - \mathbf{w}), \quad (2.8d)$$

où les vecteurs  $\mathbf{M}_f(\mathbf{u})$  et  $\mathbf{M}_m(\mathbf{u})$ , et les matrices  $\mathbf{D}_f(\mathbf{u})$  et  $\mathbf{D}_m(\mathbf{u})$  proviennent de l'annulation des termes dépendant de la vitesse angulaire dans l'équation (2.5) et (2.6). Ils peuvent être développés en

$$\begin{aligned} \mathbf{M}_f(\mathbf{u}) &:= \mathbf{T}_1 + \mathbf{T}_2 + \frac{S_{\text{wet}}}{4S_p} \Phi^{(\text{fv})} \left( (\Delta_1^f - \mathbb{I}_3) \mathbf{T}_1 + (\Delta_2^f - \mathbb{I}_3) \mathbf{T}_2 \right) \\ &= \begin{bmatrix} \left(1 - \frac{S_{\text{wet}}}{4S_p} C_d\right) (\tau_1 + \tau_2) \\ 0 \\ -\frac{S_{\text{wet}}}{4S_p} C_\ell \xi_f (\delta_1 \tau_1 + \delta_2 \tau_2) \end{bmatrix}, \end{aligned} \quad (2.9)$$

$$\begin{aligned} \mathbf{M}_m(\mathbf{u}) &:= \mathbf{N}_1 + \mathbf{N}_2 + \begin{bmatrix} p_x \\ p_y \\ 0 \end{bmatrix}_\times \mathbf{T}_1 + \begin{bmatrix} p_x \\ -p_y \\ 0 \end{bmatrix}_\times \mathbf{T}_2 \\ &\quad - \frac{S_{\text{wet}}}{4S_p} \left( \mathbf{B} \Phi^{(\text{mv})} (\Delta_1^m - \mathbb{I}_3) + \begin{bmatrix} 0 \\ a_y \\ 0 \end{bmatrix}_\times \Phi^{(\text{fv})} (\mathbb{I}_3 + \Delta_1^m) \right) \mathbf{T}_1 \\ &\quad - \frac{S_{\text{wet}}}{4S_p} \left( \mathbf{B} \Phi^{(\text{mv})} (\Delta_2^m - \mathbb{I}_3) + \begin{bmatrix} 0 \\ -a_y \\ 0 \end{bmatrix}_\times \Phi^{(\text{fv})} (\mathbb{I}_3 + \Delta_2^m) \right) \mathbf{T}_2 \\ &= \begin{bmatrix} \frac{k_m}{k_f} (\tau_1 - \tau_2) + \frac{S_{\text{wet}}}{4S_p} a_y C_\ell \xi_f (\delta_1 \tau_1 - \delta_2 \tau_2) \\ \frac{S_{\text{wet}}}{4S_p} \Delta_r C_\ell \xi_m (\delta_1 \tau_1 + \delta_2 \tau_2) \\ \left(p_y + \frac{S_{\text{wet}}}{4S_p} a_y C_d\right) (\tau_1 - \tau_2) \end{bmatrix}, \end{aligned} \quad (2.10)$$

$$\begin{aligned} \mathbf{D}_f(\mathbf{u}) &:= \frac{1}{4}\rho S \Phi^{(fv)} \left( \Delta_1^f + \Delta_2^f - 2\mathbb{I}_3 \right) \\ &= \frac{1}{4}\rho S \begin{bmatrix} -2C_d & 0 & C_d\xi_f(\delta_1 + \delta_2) \\ 0 & 0 & 0 \\ -C_\ell\xi_f(\delta_1 + \delta_2) & 0 & -2C_\ell \end{bmatrix}, \end{aligned} \quad (2.11)$$

$$\begin{aligned} \mathbf{D}_m(\mathbf{u}) &:= \frac{1}{4}\rho S \left( \left( \begin{bmatrix} 0 \\ a_y \\ 0 \end{bmatrix}_x \Phi^{(fv)} + \mathbf{B}\Phi^{(mv)} \right) \Delta_1^m \right. \\ &\quad \left. + \left( \begin{bmatrix} 0 \\ -a_y \\ 0 \end{bmatrix}_x \Phi^{(fv)} + \mathbf{B}\Phi^{(mv)} \right) \Delta_2^m - 2\mathbf{B}\Phi^{(mv)} \right) \\ &= \frac{1}{4}\rho S \begin{bmatrix} -a_y C_d \xi_m (\delta_1 - \delta_2) & 0 & 0 \\ \Delta_r C_\ell \xi_m (\delta_1 + \delta_2) & 0 & 2\Delta_r C_\ell \\ 0 & 0 & -a_y C_\ell \xi_m (\delta_1 - \delta_2) \end{bmatrix}, \end{aligned} \quad (2.12)$$

où l'on observe l'effet non linéaire d'un vent non nul, lequel est non linéaire avec  $\mathbf{q}$ ,  $\|\mathbf{v}_b\|$  et  $\mathbf{w}$ . Comme dans [Olszanecki Barth 2020, eqn. (10)] et selon la formule de Diederich, nous obtenons  $C_\ell = C_d + \frac{\pi AR}{1 + \sqrt{1 + (\frac{AR}{2})^2}}$  où  $AR = \frac{b^2}{S}$  est l'allongement de l'aile. Nous observons le couplage des actionneurs ( $\delta_1\tau_1 + \delta_2\tau_2$ ) dans les expressions des matrices  $\mathbf{M}_f(\mathbf{u})$  et  $\mathbf{M}_m(\mathbf{u})$ .

## 2.2 Identification des paramètres du modèle

Les valeurs numériques de la Table 2.1 ont été obtenues par une campagne d'identification du modèle [Sansou 2022a]. En particulier, le coefficient  $k_f$  a été identifié à partir de l'équation (2.3), qui relie la vitesse de rotation du moteur  $\omega_i$  à la traction générée, à la vitesse de rotation minimale et maximale et à la constante de temps de la chaîne d'actionnement du moteur.

Pour effectuer l'identification des 3 coefficients principaux (diagonaux) de la matrice d'inertie, nous avons réalisé un montage d'un système de pendule bifilaire. Cette méthode est largement utilisée dans le domaine des drones [Jardin 2007], et est basée sur la période d'oscillation autour de chacun des trois axes ( $x_b$ ,  $y_b$ ,  $z_b$ ) du drone, lequel est suspendu par deux fils, ce qui forme un pendule de torsion comme le montre la Fig. 2.4.

Lors de la mesure, l'autopilote est utilisé pour réaliser une acquisition de l'orientation du drone à 500 Hz. Le drone est positionné avec un angle non nul vis-à-vis de la position d'équilibre du pendule bifilaire puis il est lâché sans vitesse initiale. Le couple de rappel engendré par les deux fils produit des oscillations amorties (Voir

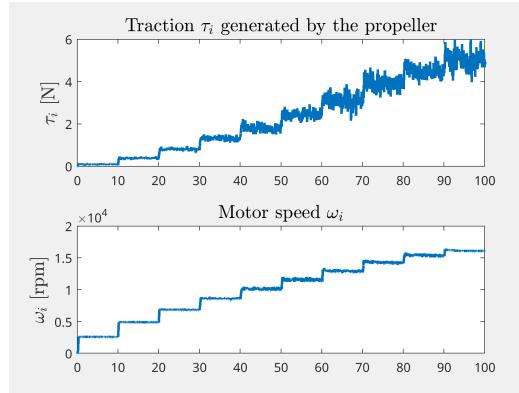
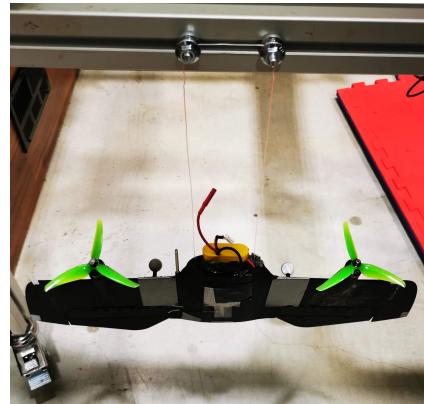


FIGURE 2.3 – Réponse entrée-sortie de l’ensemble moteur/hélice.

FIGURE 2.4 – Montage d’un pendule bifilaire pour l’identification de l’inertie ( $\mathbf{J}$ ) de DarkO.

la figure 2.5). Il est nécessaire de connaître la longueur des fils  $h$  ainsi que leur écartement  $D$  pour réaliser l’identification. Ces valeurs sont mesurées directement sur le banc de mesure pour chacune des trois configurations et reportées dans la table 2.2.

	$h$	$D$
$J_x$	0.962 m	0.142 m
$J_y$	0.415 m	0.051 m
$J_z$	1.018 m	0.149 m

TABLE 2.2 – Longueur et espacement des fils du pendule pour chacun des axes.

Une fois la mesure réalisée, nous utilisons l’outil *Simulink Design Optimization* pour obtenir les valeurs de l’amortissement visqueux  $C$ , et de l’inertie sur l’axe

mesuré  $I$ , à partir du modèle suivant :

$$\ddot{\theta} + \frac{C}{I}\dot{\theta} + \left(\frac{mgD^2}{4Ih}\right) \frac{\sin \theta}{\sqrt{1 - 0.5\left(\frac{D}{h}\right)^2(1 - \cos \theta)}} = 0$$

où  $\theta$  est l'angle mesuré par l'autopilote à l'aide du code d'estimation d'état utilisant le gyroscope, l'accéléromètre et l'Optitrack.

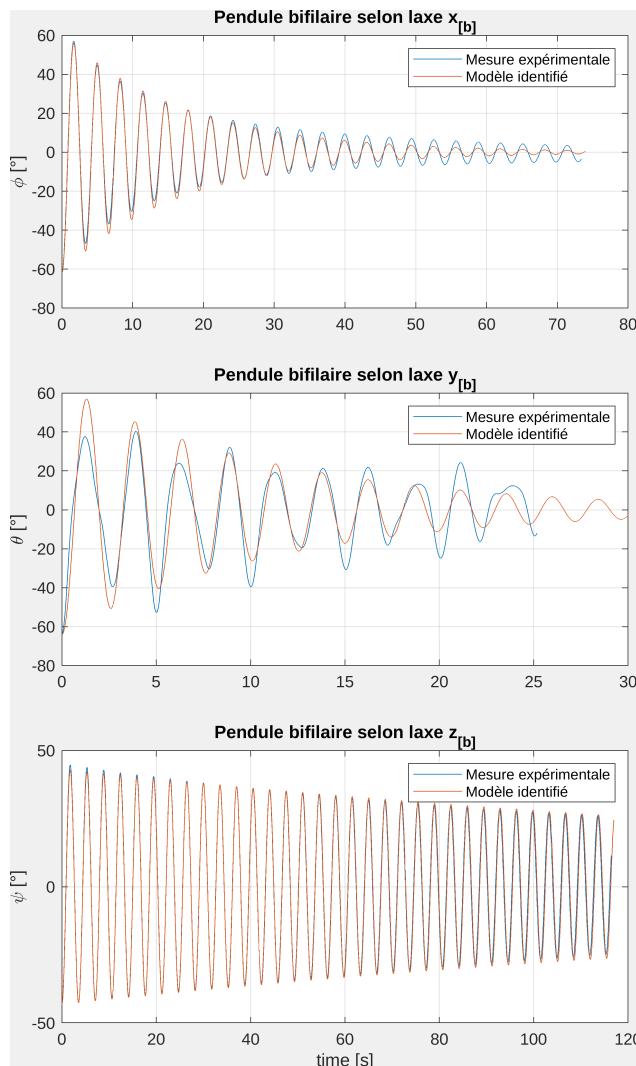


FIGURE 2.5 – Identification de l'inertie ( $J$ ), à partir des mesures issues du pendule bifilaire 2.4.

uter explication  
l'indentification

Les autres coefficients ont été estimés à l'aide d'un montage sur un capteur de forces et moments à 6 degrés de liberté (DOF). Ces mesures permettent d'estimer

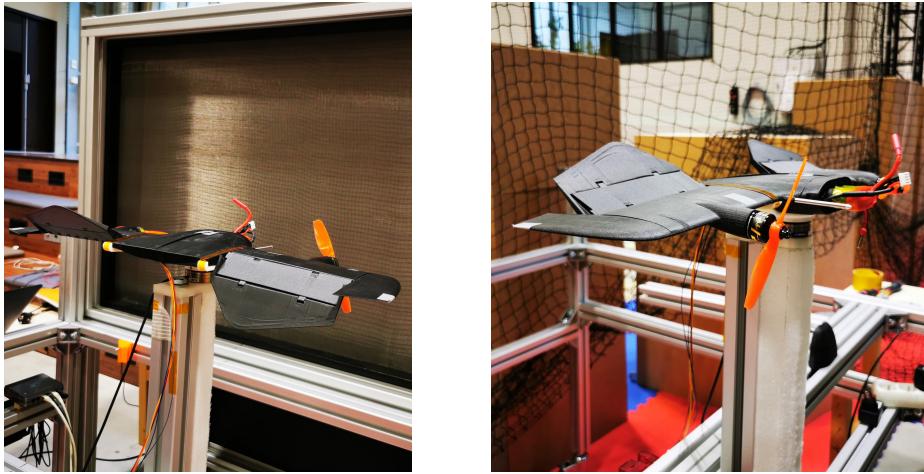


FIGURE 2.6 – Montage de DarkO sur un banc de mesure face à une soufflerie ouverte.

la surface. Il est intéressant de noter que la surface soufflée par les hélices représente 67 % de la surface totale du drone.

### 2.2.1 Modélisation des actionneurs

Les actionneurs de DarkO ont des dynamiques qui limitent leur action en terme d'amplitude et de vitesse.

Pour les moteurs électriques générant la traction par les hélices, il existe deux causes de saturation. Une saturation à haute vitesse liée à la tension maximale du moteur et une saturation basse vitesse liée à la vitesse minimale de commutation de bobine du moteur, pour maintenir la rotation. De plus, ces saturations permettent d'obtenir un modèle réaliste à énergie finie. Elles correspondent à la contrainte suivante :  $\omega_i \in [2500, 16000] \text{ rpm} = [262, 1675] \text{ rad s}^{-1}$ ,  $i = 1, 2$ .

En termes de dynamique, nous avons représenté la chaîne d'actionnement du moteur (composée de l'ESC, du moteur et de l'hélice) par une fonction de transfert du premier ordre ayant une constante de temps égale à 0.0125 s, ce qui fournit un système d'actionnement assez agressif.

Les saturations impactant les élévons proviennent des limites physiques des servomoteurs et du débattement limité par la forme de l'UAV,  $\delta_i \in [-30 ; 30]^\circ$ ,  $i = 1, 2$ . La saturation la plus importante ici est peut-être la bande passante de l'actionneur (due à l'actionnement du servomoteur), qui est modélisée par une fonction de transfert du premier ordre avec une constante de temps 0.05 s.

description du modèle des actionneurs

## 2.3 Équilibres stationnaires

### 2.3.1 Équilibre stationnaire sans vent

Nous proposons une modification du vecteur de commande, dans le cas d'un équilibre sans vent  $\mathbf{w}_{\text{eq}} = 0$ , basé sur le couplage des actionneurs.

$$\mathbf{u}_{\text{nowind}} := \begin{bmatrix} \tau_1 & \tau_2 & \delta_1 \tau_1 & \delta_2 \tau_2 \end{bmatrix}^\top \quad (2.13)$$

Nous soulignons que le vecteur  $\mathbf{u}_{\text{nowind}}$  dans (2.13) correspond à une transformation non inversible des actionneurs de DarkO correspondant à  $\mathbf{u} := [\tau_1 \ \tau_2 \ \delta_1 \ \delta_2]^\top$  ((2.7)). Néanmoins, si l'on impose les contraintes de saturation décrites dans la section 2.2.1, il est possible de déterminer de manière unique  $\mathbf{u}$  à partir d'une valeur souhaitée de  $\mathbf{u}_{\text{nowind}}$  dans (2.13). Les valeurs positives non nulles de  $\tau_1$  et  $\tau_2$  peuvent être déterminées à partir des deux premières composantes de  $\mathbf{u}_{\text{nowind}}$ , puis  $\delta_1$  et  $\delta_2$  sont facilement construites à partir des deux dernières composantes de  $\mathbf{u}_{\text{nowind}}$ .

Nous obtenons un modèle linéaire vis-à-vis de sa commande, dérivé de (2.8) en imposant  $\mathbf{w} = 0$ ,

$$\dot{\mathbf{p}} = \mathbf{v}, \quad m\dot{\mathbf{v}} = -m\mathbf{g} + \mathbf{R}(\mathbf{q})\mathbf{F}\mathbf{u}_{\text{nowind}}, \quad (2.14a)$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \begin{bmatrix} 0 \\ \omega_b \end{bmatrix} \quad \mathbf{J}\dot{\boldsymbol{\omega}}_b = -[\boldsymbol{\omega}_b]_\times \mathbf{J}\boldsymbol{\omega}_b + \mathbf{M}\mathbf{u}_{\text{nowind}}, \quad (2.14b)$$

avec les matrices

$$\left[ \begin{array}{c|c} \mathbf{F} & \mathbf{M} \end{array} \right] := \left[ \begin{array}{cccc|cccc} a_f & a_f & 0 & 0 & a_m & -a_m & b_m & -b_m \\ 0 & 0 & 0 & 0 & 0 & 0 & c_m & c_m \\ 0 & 0 & b_f & b_f & d_m & -d_m & 0 & 0 \end{array} \right] \quad (2.15)$$

et les scalaires

$$\left[ \begin{array}{c|c} a_f & b_f \\ \hline a_m & b_m \\ \hline c_m & d_m \end{array} \right] = \left[ \begin{array}{c|c} 1 - \frac{S_{\text{wet}}}{4S_p} C_d & -\frac{S_{\text{wet}}}{4S_p} C_\ell \xi_f \\ \hline \frac{k_m}{k_f} & \frac{S_{\text{wet}}}{4S_p} a_y C_\ell \xi_f \\ \hline \frac{S_{\text{wet}}}{4S_p} \Delta_r C_\ell \xi_m & p_y + \frac{S_{\text{wet}}}{4S_p} a_y C_d \end{array} \right].$$

Tous les couples d'équilibre  $(\mathbf{u}_{\text{nowind}}, \mathbf{x}) = (\mathbf{u}_{\text{nowind,eq}}, \mathbf{x}_{\text{eq}})$  sont paramétrés par une rotation arbitraire autour de l'axe  $z_{[i]}$  définie par  $\beta \in [-\sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}]$ . Le point

d'équilibre a pour expression :

$$\mathbf{u}_{\text{nowind,eq}} = \frac{mg}{(1 - \frac{S_{\text{wet}}}{4S_p} C_d)} [1 \ 1 \ 0 \ 0]^\top \quad (2.16a)$$

$$\mathbf{q}_{\text{eq}} = [\eta_{\text{eq}} \ \boldsymbol{\epsilon}_{\text{eq}}^\top]^\top = \left[ \sqrt{\frac{1}{2} - \beta} \ \beta \ \frac{2\beta^2 - 1}{2\sqrt{\frac{1}{2} - \beta}} \ \beta \right]^\top. \quad (2.16b)$$

En présence d'un vent nul, le degré de liberté  $\beta$  permet d'orienter le drone dans n'importe quelle direction horizontale.

### 2.3.2 Équilibre stationnaire en présence de vent

À partir des modèles (2.1) et (2.8), nous caractérisons un équilibre stationnaire en présence d'un vent constant  $\mathbf{w}_{\text{eq}} = \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} \in \mathbb{R}^3$  exprimé dans le repère inertiel, tel que  $[w_x \ w_y] \neq 0$ , c'est-à-dire qu'il existe toujours un vent horizontal non nul. Ainsi, pour chaque position de référence  $\mathbf{p}_{\text{eq}} \in \mathbb{R}^3$ , un ensemble de couples état/commande possible est  $(\mathbf{u}_{\text{eq}}, \mathbf{x}_{\text{eq}}) = (\mathbf{u}_{\text{eq}}, \mathbf{p}_{\text{eq}}, \mathbf{v}_{\text{eq}}, \mathbf{q}_{\text{eq}}, \boldsymbol{\omega}_{\text{b,eq}})$  obtenu à l'aide de

$$\mathbf{u}_{\text{eq}} = [\tau \ \tau \ \delta \ \delta]^\top \quad (2.17a)$$

$$\mathbf{q}_{\text{eq}} = \mathbf{q}_{\text{eq}\psi} \otimes \mathbf{q}_{\text{eq}\theta} \quad (2.17b)$$

$$\boldsymbol{\omega}_{\text{b,eq}} = 0, \quad \mathbf{v}_{\text{eq}} = 0. \quad (2.17c)$$

Nous définissons deux quaternions  $\mathbf{q}_{\text{eq}\psi}$  et  $\mathbf{q}_{\text{eq}\theta}$  permettant d'exprimer l'ensemble des conditions de vent dans le repère inertiel vers un repère tourné où le vent est toujours contenu dans le plan  $x - z$ . Grâce à cette transformation, nous exprimons un ensemble continu d'équilibres en présence de vent.

$$\mathbf{q}_{\text{eq}\theta} := \left[ \cos\left(\frac{\theta}{2}\right) \ 0 \ \sin\left(\frac{\theta}{2}\right) \ 0 \right]^\top \quad (2.18)$$

$$\mathbf{q}_{\text{eq}\psi} := \left[ \cos\left(\frac{\psi}{2}\right) \ 0 \ 0 \ \sin\left(\frac{\psi}{2}\right) \right]^\top. \quad (2.19)$$

Les paramètres de l'équilibre sont la rotation horizontale  $\psi = \arctan(w_x, w_y)$ , l'angle d'inclinaison  $\theta$ , la poussée des hélices  $\tau$ , et la déflexion des elevons  $\delta$ . Ils peuvent être obtenus à partir de l'algorithme 1.

**Théorème 1.** Pour tout vent constant,  $\mathbf{w} = [w_x \ w_y \ w_z]^\top \in \mathbb{R}^3$  ayant une composante horizontale non nulle  $[w_x \ w_y]$ , les équations (2.19)–(2.18) avec  $\theta$ ,  $\tau$  et  $\delta$  sélec-

**Algorithme 1** Obtention des paramètres d'équilibre en (2.17).

**Entrée :** Vecteur vent  $\mathbf{w}_{\text{eq}} = [w_x \ w_y \ w_z]^\top$

**Sortie :** Paramètres  $\psi, \theta, \tau, \delta$  dans (2.17)

- 1: Détermine l'angle  $\psi = \text{atan}2(w_x, w_y)$  de manière à obtenir  $\mathbf{q}_{\text{eq}\psi}$  dans (2.19)
- 2: Détermine la perturbation tournée  $\mathbf{w}_r$  avec la composante  $y$  nulle, en utilisant  
 $\mathbf{R}_\psi := \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ , selon

$$\mathbf{w}_{r,\text{eq}} := \begin{bmatrix} w_{rx} \\ 0 \\ w_{rz} \end{bmatrix} := \mathbf{R}^\top(\mathbf{q}_{\text{eq}\psi}) \mathbf{w}_{\text{eq}} = \mathbf{R}_\psi^\top \mathbf{w}_{\text{eq}} \quad (2.20)$$

- 3: Détermine l'angle d'inclinaison  $\theta$  de manière à obtenir  $\mathbf{q}_{\text{eq}\theta}$  dans (2.17b) :

$$\theta = -\tan^{-1} \left( \frac{w_{rz}}{w_{rx}} + \frac{2mg}{\rho S \|\mathbf{w}_{\text{eq}}\| C_\ell (1 - \frac{\xi_f}{\xi_m}) w_{rx}} \right) \quad (2.21)$$

- 4: Pour des raisons de commodité, nous définissons les scalaires

$$\begin{bmatrix} a | b \\ c | d \end{bmatrix} := \begin{bmatrix} 2S_{\text{wet}} C_\ell mg \sin \theta \xi_f & 2S_{\text{wet}} C_d C_\ell \rho \|\mathbf{w}_{\text{eq}}\| w_x^b \\ -4SS_p C_\ell \rho \|\mathbf{w}_{\text{eq}}\| w_x^b \xi_f & \frac{b\xi_f}{2} \end{bmatrix}$$

et grâce à ces scalaires  $(a, b, c, d)$ , déterminons la traction des hélices  $\tau$  dans (2.17a) comme

$$\begin{aligned} \tau = & \frac{S_p}{2S_{\text{wet}} C_\ell \xi_f (4S_p - S_{\text{wet}} C_d)} \left( a + b + c + d + \sqrt{(a + b + c - d)^2 - 4(d^2 + ac - bd)} \right. \\ & \left. - \frac{4w_z^b d}{w_x^b} (d + c) + \frac{4w_z^b ad \cos \theta}{w_x^b C_\ell \sin \theta} \left( C_d - \frac{4S_p}{S_{\text{wet}}} \right) \right)^{\frac{1}{2}}, \end{aligned} \quad (2.22)$$

où

$$\begin{bmatrix} w_x^b \\ w_z^b \end{bmatrix} = \begin{bmatrix} w_{rx} \cos \theta - w_{rz} \sin \theta \\ w_{rx} \sin \theta + w_{rz} \cos \theta \end{bmatrix}.$$

- 5: Déterminons la déflexion des élévons  $\delta$  comme

$$\delta = \frac{2mg \sin \theta}{\rho S \|\mathbf{w}_{\text{eq}}\| C_d \xi_f w_z^b} + \frac{w_x^b}{\xi_f w_z^b} - \frac{(4 - \frac{S_{\text{wet}}}{S_p} C_d)}{\rho S \|\mathbf{w}_{\text{eq}}\| C_d \xi_f w_z^b} \tau. \quad (2.23)$$

**Retourne :**  $\psi, \theta, \tau, \delta$

tionnées selon l’Algorithme 1 caractérisent un couple d’équilibre  $(\mathbf{u}_{eq}, \mathbf{x}_{eq})$  pour la dynamique non linéaire (2.1) et (2.8).

*Démonstration.* Dans un premier temps, notons qu’avec l’expression de  $\mathbf{R}$  (2.2) et l’expression de  $\psi$  dans l’étape 1 de l’Algorithme 1, on peut définir la perturbation à l’équilibre tourné  $\mathbf{w}_{r,eq} := \mathbf{R}_\psi^\top \mathbf{w}_{eq} := \mathbf{R}^\top(\mathbf{q}_{eq\psi}) \mathbf{w}_{eq}$  (voir (2.20) dans l’Algorithme 1), qui correspond à la rotation nécessaire pour aligner l’axe  $x_{[b]}$  du repère corps avec la direction du vent. Une fois que le drone est face au vent, il subit un vent avec une composante latérale  $y$  nulle et il peut ajuster son angle d’inclinaison  $\theta$  afin de générer la poussée et la portance nécessaires pour compenser les effets du vent dans les directions longitudinale et verticale (l’effet latéral est nul en raison de l’orientation spécifique de l’appareil  $\psi$ ). Avec cette rotation  $\psi$ , il est possible d’exprimer le vent dans le repère corps comme étant

$$\begin{aligned}\mathbf{w}_{eq}^b &:= \begin{bmatrix} w_x^b \\ 0 \\ w_z^b \end{bmatrix} = \mathbf{R}^\top(\mathbf{q}_{eq\theta}) \mathbf{w}_{r,eq} \\ &= \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}^\top \begin{bmatrix} w_{rx} \\ 0 \\ w_{rz} \end{bmatrix} = \begin{bmatrix} w_{rx} \cos \theta - w_{rz} \sin \theta \\ 0 \\ w_{rx} \sin \theta + w_{rz} \cos \theta \end{bmatrix}\end{aligned}\quad (2.24)$$

Nous insistons sur le fait que  $w_x^b$  est toujours négatif et différent de zéro, car le drone est orienté dans la direction du vent grâce à la rotation engendrée par  $\mathbf{q}_{eq\psi}$ , et suite à l’hypothèse  $[w_x^b] \neq 0$ .

L’équation (2.8a) montre qu’il est nécessaire d’avoir  $\mathbf{v}_{eq} = 0$  pour maintenir l’équilibre stationnaire. En multipliant (2.8b) par  $\mathbf{R}(\mathbf{q}_{eq})$  donnée dans (2.24), nous l’exprimons dans le repère corps. Comme nous appliquons la même commande  $\tau_1 = \tau_2 = \tau$  aux deux moteurs et la même commande au deux élévons  $\delta_1 = \delta_2 = \delta$ , nous obtenons pour les deux modèles (2.1) et (2.8), l’équilibre des forces selon l’axe  $x_{[b]}$  donné par :

$$(2 - \frac{S_{wet}}{2S_p} C_d) \tau - \frac{1}{2} \rho S \| \mathbf{w}_{eq} \| C_d (w_x^b - \xi_f \delta w_z^b) - mg \sin(\theta) = 0 \quad (2.25)$$

et l’équilibre des forces selon l’axe  $z_{[b]}$  donné par :

$$-\frac{S_{wet}}{2S_p} \xi_f C_\ell \tau \delta - \frac{1}{2} \rho S \| \mathbf{w}_{eq} \| C_\ell (w_z^b + \xi_f \delta w_x^b) + mg \cos(\theta) = 0 \quad (2.26)$$

De manière similaire, à partir de (2.1d) et (2.8d), l’équilibre des moments autour

de l'axe  $y_{[b]}$  permet d'obtenir :

$$\frac{S_{\text{wet}}}{2S_p} \Delta_r \xi_m C_\ell \tau \delta + \frac{1}{2} \rho S \Delta_r \| \mathbf{w}_{\text{eq}} \| C_\ell (w_z^b + \xi_m \delta w_x^b) = 0. \quad (2.27)$$

Pour calculer la solution du triplet  $(\theta, \tau, \delta)$  des trois équations d'équilibre (2.25)–(2.27), ajoutons (2.26) multipliée par  $\Delta_r \xi_m$ , à (2.27) multipliée par  $\xi_f$ , de manière à annuler le premier terme et à obtenir :

$$\begin{aligned} \Delta_r \xi_m \left( -\frac{1}{2} \rho S \| \mathbf{w}_{\text{eq}} \| C_\ell (w_z^b + \xi_f \delta w_x^b) + mg \cos(\theta) \right) \\ + \xi_f \left( \frac{1}{2} \rho S \Delta_r \| \mathbf{w}_{\text{eq}} \| C_\ell (w_z^b + \xi_m \delta w_x^b) \right) = 0, \end{aligned}$$

qui est équivalent à

$$\frac{1}{2} \rho S \Delta_r \| \mathbf{w}_{\text{eq}} \| C_\ell (\xi_f - \xi_m) w_z^b + \Delta_r \xi_m mg \cos(\theta) = 0,$$

où  $(w_x^b, w_z^b)$  sont les première et troisième composantes de  $\mathbf{w}^b$  dans (2.24). Ensuite, en utilisant (2.24) et après calcul, nous obtenons :

$$\begin{aligned} -\frac{1}{2} \rho S \Delta_r \| \mathbf{w}_{\text{eq}} \| C_\ell (\xi_f - \xi_m) w_{rz} \sin \theta + \left( -\frac{1}{2} \rho S \Delta_r \| \mathbf{w}_{\text{eq}} \| C_\ell (\xi_f - \xi_m) w_{rz} + \Delta_r \xi_m mg \right) \cos \theta = 0, \end{aligned}$$

qui est satisfaite par

$$\theta = -\tan^{-1} \left( \frac{\rho S \| \mathbf{w}_{\text{eq}} \| C_\ell (\xi_f - \xi_m) w_{rz} - 2 \xi_m mg}{\rho S \| \mathbf{w}_{\text{eq}} \| C_\ell (\xi_f - \xi_m) w_{rx}} \right). \quad (2.28)$$

Cette dernière expression coïncide avec la sélection (2.21) dans l'Algorithme 1 après quelques développements. À partir de (2.21), nous pouvons calculer les commandes à l'équilibre en substituant (2.25) dans (2.26). Après quelques simplifications, la force nécessaire de traction des hélices  $\tau$  pour maintenir la position d'équilibre correspond à l'expression (2.22). Finalement, avec la valeur de  $\tau$  dans (2.22), nous pouvons obtenir la déflexion des élévons nécessaire  $\delta$  à partir de l'équation (2.26), ce qui nous donne la valeur obtenue dans (2.23).  $\square$

Il est intéressant de noter que pour chaque couple de vent  $(w_{rz}, w_{rx})$  correspond une orientation d'équilibre (2.17b), (2.21) étant indépendante de l'entrée  $\mathbf{u}_{\text{eq}}$ . En outre, il convient de souligner que pour toutes les valeurs de vent raisonnables,

l'équation (2.22) correspond à la racine positive d'un polynôme du second ordre, l'autre racine étant toujours négative, ce qui conduit à une condition de poussée négative physiquement impossible.

À partir de l'expression analytique (2.17) de l'équilibre du drone pour différentes conditions de vent  $\mathbf{w}$ , nous reportons, sur la Fig. 2.7, les valeurs correspondantes de  $\theta$ ,  $\delta$ ,  $\tau$  pour des valeurs de vitesse de vent horizontal allant de 0 à  $-20 \text{ m s}^{-1}$  et pour des valeurs de vitesse de vent vertical allant de  $-6$  à  $6 \text{ m s}^{-1}$ . L'angle d'incidence  $\theta$  diminue de  $90^\circ$  à  $-4.65^\circ$ .  $\theta = 90^\circ$  correspond à un vol stationnaire sans vent. La traction  $\tau$  atteint son minimum à  $w_{rx} = -12.8 \text{ m s}^{-1}$ , ce qui correspond à une condition de vol qui minimise la consommation d'énergie, car les moteurs sont la principale source de consommation électrique.

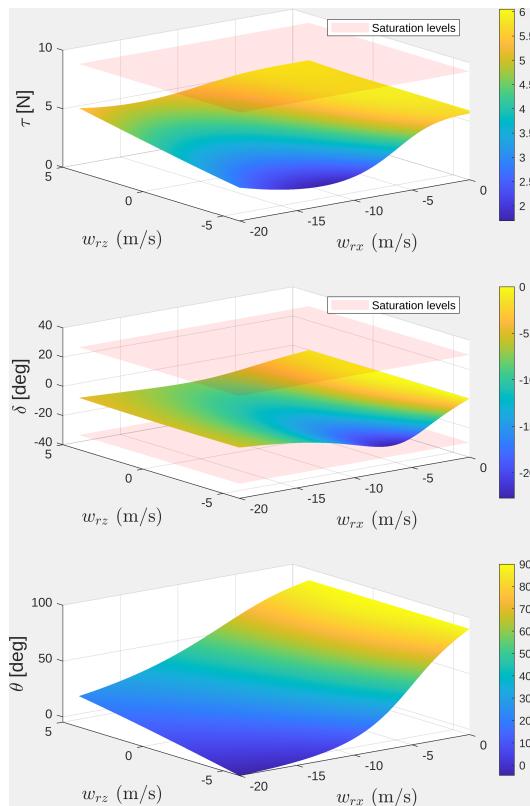


FIGURE 2.7 – Les paramètres ( $\tau$ ,  $\delta$ ,  $\theta$ ) de l'ensemble des points d'équilibre (surface) obtenus à l'aide du Théorème 1 et de l'Algorithm 1 pour un vent constant horizontal et vertical ( $w_{rx}, w_{rz}$ ), avec les saturations des actionneurs (rose).

Il est possible de faire une coupe des surfaces présentée dans (2.7) pour une vitesse verticale nulle  $w_{rx} = 0$ , ce qui nous donne le résultat de la Figure 2.8

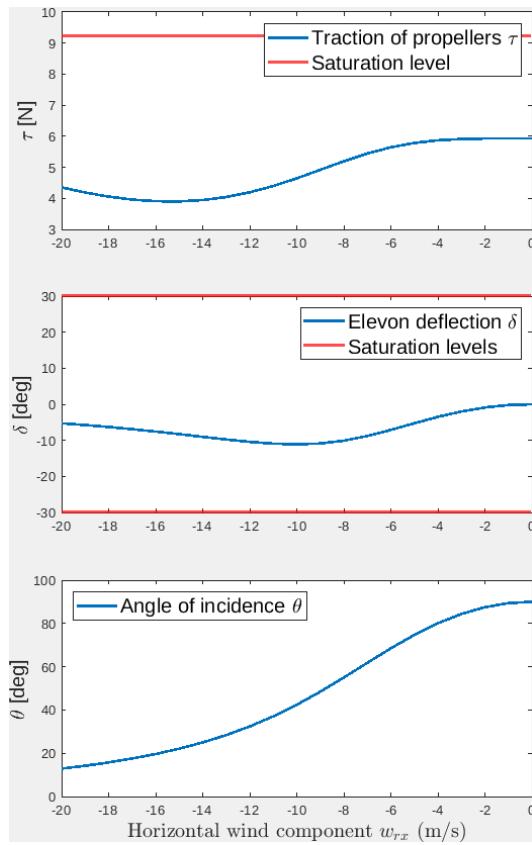


FIGURE 2.8 – Section des surfaces de la Figure 2.7 pour  $w_{rz} = 0 \text{ m s}^{-1}$ .

## 2.4 Dynamiques linéarisées

### 2.4.1 Dynamique linéarisée sans vent

Considérons le cas sans vent discuté dans la section 2.3.1 pour lequel nous utilisons le vecteur de commande  $\mathbf{u}_{\text{nowind}}$  et le vecteur de commande à l'équilibre  $\mathbf{u}_{\text{nowind,eq}}$  défini dans l'équation (2.16a) et rappelons la transformation du vecteur de commande suivante  $\mathbf{u}_{\text{nowind}} := [\tau_1 \ \tau_2 \ \delta_1 \tau_1 \ \delta_2 \tau_2]^{\top}$ . La dynamique linéarisée dans le cas sans vent est :

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A}_0 \tilde{\mathbf{x}} + \mathbf{G}_0 (\mathbf{u}_{\text{nowind}} - \mathbf{u}_{\text{nowind,eq}}), \quad (2.29)$$

où l'expression de  $\mathbf{A}_0$  est :

$$\mathbf{A}_0 = \mathbf{A}_w \Big|_{\mathbf{w}=0} = \begin{bmatrix} \mathbb{0}_3 & \mathbb{I}_3 & \mathbb{0}_3 & \mathbb{0}_3 \\ \mathbb{0}_3 & \mathbb{0}_3 & \mathbf{A}_{v\epsilon} & \mathbb{0}_3 \\ \mathbb{0}_3 & \mathbb{0}_3 & \mathbb{0}_3 & \mathbf{A}_{\epsilon\omega} \\ \mathbb{0}_3 & \mathbb{0}_3 & \mathbb{0}_3 & \mathbb{0}_3 \end{bmatrix}, \quad (2.30)$$

avec les matrices suivantes :

$$\mathbf{A}_{\epsilon\omega} = \frac{\sqrt{2}}{4} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \text{ et } \mathbf{A}_{v\epsilon} = \sqrt{2} \begin{bmatrix} 0 & -2g & 0 \\ g & 0 & g \\ 0 & -2g & 0 \end{bmatrix},$$

alors que l'expression de  $\mathbf{G}_0$  est :

$$\mathbf{G}_0 := \begin{bmatrix} \mathbb{0}_{3 \times 1} & \mathbb{0}_{3 \times 1} & \mathbb{0}_{3 \times 1} & \mathbb{0}_{3 \times 1} \\ 0 & 0 & a_g & a_g \\ 0 & 0 & 0 & 0 \\ b_g & b_g & 0 & 0 \\ \mathbb{0}_{3 \times 1} & \mathbb{0}_{3 \times 1} & \mathbb{0}_{3 \times 1} & \mathbb{0}_{3 \times 1} \\ c_g & -c_g & d_g & -d_g \\ 0 & 0 & e_g & e_g \\ f_g & -f_g & 0 & 0 \end{bmatrix},$$

avec :

$$\left[ \begin{array}{c|c} a_g & b_g \\ \hline c_g & d_g \\ \hline e_g & f_g \end{array} \right] = \left[ \begin{array}{c|c} -\frac{S_{\text{wet}}}{4mS_p} C_\ell \xi_f & \frac{1}{m} (1 - \frac{S_{\text{wet}}}{2S_p} C_d) \\ \hline \frac{k_m}{J_x k_f} & \frac{S_{\text{wet}} a_y}{4J_x S_p} C_\ell \xi_f \\ \hline \frac{S_{\text{wet}} \Delta_r}{4J_y S_p} C_\ell \xi_m & \frac{1}{J_z} (p_y + \frac{S_{\text{wet}}}{4S_p} a_y C_d) \end{array} \right].$$

#### 2.4.2 Dynamique linéarisée en présence de vent

Pour chacun des équilibres caractérisés dans le Théorème 1, nous détaillons les équations linéarisées du mouvement par rapport au modèle non linéaire simplifié à faible vitesse (2.8). Une approche directe conduirait à des équations linéarisées qui dépendent de l'angle  $\psi$  caractérisé à l'étape 1 de l'Algorithme 1. Au lieu de cela, nous définissons ici les coordonnées incrémentales dans un cadre de référence inertiel convenablement tourné, de sorte que la dynamique linéarisée soit indépendante de l'angle  $\psi$ . Plus précisément, pour chaque condition de vent d'équilibre  $\mathbf{w}_{\text{eq}}$  associée à l'équilibre  $(\mathbf{u}_{\text{eq}}, \mathbf{p}_{\text{eq}}, \mathbf{v}_{\text{eq}}, \mathbf{q}_{\text{eq}}, \boldsymbol{\omega}_{\text{b,eq}})$  caractérisée en (2.19)–(2.18), désignant les com-

posantes scalaire et vectorielle du quaternion en (2.17b) comme  $\mathbf{q}_{\text{eq}} = (\eta_{\text{eq}}, \boldsymbol{\epsilon}_{\text{eq}})$ , et à partir de la matrice de rotation  $\mathbf{R}_\psi := \mathbf{R}(\mathbf{q}_{\text{eq}\psi})$  introduite au début de la preuve du Théorème 1, nous étudions ici la dynamique incrémental linéaire du vecteur d'état tourné :

$$\begin{aligned}\tilde{\mathbf{x}} &:= (\tilde{\mathbf{p}}, \tilde{\mathbf{v}}, \tilde{\boldsymbol{\epsilon}}, \tilde{\boldsymbol{\omega}}_b) = \left( \mathbf{R}_\psi^\top (\mathbf{p} - \mathbf{p}_{\text{eq}}), \mathbf{R}_\psi^\top \mathbf{v}, \mathbf{R}_\psi^\top (\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\text{eq}}), \boldsymbol{\omega}_b \right), \\ \tilde{\mathbf{u}} &:= \mathbf{u} - \mathbf{u}_{\text{eq}}, \quad \tilde{\mathbf{w}} := \mathbf{R}_\psi^\top (\mathbf{w} - \mathbf{w}_{\text{eq}}).\end{aligned}\quad (2.31)$$

Nous notons que la rotation en (2.31) possède la propriété  $\mathbf{R}_\psi^\top \boldsymbol{\epsilon}_{\text{eq}} = [0 \sin(\frac{\theta}{2}) 0]^\top$ , ce qui simplifie grandement le mouvement linéarisé.

En exploitant le fait que les vitesses linéaire et angulaire  $(\mathbf{v}_{\text{eq}}, \boldsymbol{\omega}_{b,\text{eq}})$  doit être nulle à l'équilibre (voir (2.17)), nous prouvons ci-dessous que la dynamique linéarisée de l'état (2.31) est donnée par :

$$\begin{aligned}\dot{\tilde{\mathbf{x}}} &= \mathbf{A}_w \tilde{\mathbf{x}} + \mathbf{G}_w \tilde{\mathbf{u}} + \mathbf{E}_w \tilde{\mathbf{w}} \\ &= \begin{bmatrix} \mathbb{0}_3 & \mathbb{I}_3 & \mathbb{0}_3 & \mathbb{0}_3 \\ \mathbb{0}_3 & \mathbf{A}_{vv} & \mathbf{A}_{v\epsilon} & \mathbb{0}_3 \\ \mathbb{0}_3 & \mathbb{0}_3 & \mathbb{0}_3 & \mathbf{A}_{\epsilon\omega} \\ \mathbb{0}_3 & \mathbb{0}_3 & \mathbf{A}_{\omega\epsilon} & \mathbb{0}_3 \end{bmatrix} \tilde{\mathbf{x}} + \begin{bmatrix} \mathbb{0}_{3 \times 4} \\ \mathbf{G}_v \\ \mathbb{0}_{3 \times 4} \\ \mathbf{G}_\omega \end{bmatrix} \tilde{\mathbf{u}} + \begin{bmatrix} \mathbb{0}_{3 \times 3} \\ \mathbf{E}_v \\ \mathbb{0}_{3 \times 3} \\ \mathbf{E}_\omega \end{bmatrix} \tilde{\mathbf{w}},\end{aligned}\quad (2.32)$$

avec les matrices  $\mathbf{A}_{vv}$ ,  $\mathbf{A}_{v\epsilon}$ ,  $\mathbf{A}_{\epsilon\omega_b}$ ,  $\mathbf{A}_{\omega\epsilon}$ ,  $\mathbf{G}_v$ ,  $\mathbf{G}_\omega$ ,  $\mathbf{E}_v$ ,  $\mathbf{E}_\omega$  construites en suivant l'Algorithme 2.

**Théorème 2.** Pour tout vent constant,  $\mathbf{w} = [w_x \ w_y \ w_z]^\top \in \mathbb{R}^3$  ayant une composante horizontale non nulle  $[w_x]$ , et pour le doublet d'équilibre qui découle  $(\mathbf{u}_{\text{eq}}, \mathbf{x}_{\text{eq}})$  de la dynamique (2.8) telle que caractérisée dans (2.19)-(2.18), la dynamique linéarisée du vecteur état incrémental (2.31) est donnée par (2.32), avec les matrices construites comme dans l'Algorithme 2.

*Démonstration.* Tout d'abord, en exploitant la matrice de rotation  $\mathbf{R}_\psi := \mathbf{R}(\mathbf{q}_{\text{eq}\psi})$  utilisée dans (2.31), nous transformons la dynamique non linéaire (2.8) en coordonnées tournées :

$$(\mathbf{p}_r, \mathbf{v}_r, \mathbf{q}_r) := \left( \mathbf{R}_\psi^\top \mathbf{p}, \mathbf{R}_\psi^\top \mathbf{v}, \mathbf{q}_{\text{eq}\psi}^{-1} \otimes \mathbf{q} \right), \quad \mathbf{w}_r := \mathbf{R}_\psi^\top \mathbf{w} \quad (2.33)$$

où  $\boldsymbol{\omega}_b$  reste inchangée car elle est exprimée dans le repère du corps. Quelques observations permettent de simplifier la dynamique transformée (2.8) :

- nous avons  $\mathbf{R}_\psi^\top m\mathbf{g} = m\mathbf{g}$  car la rotation de  $\psi$  est autour de l'axe  $z_{[i]}$  ;
- comme  $\mathbf{q}_r = \mathbf{q}_{\text{eq}\psi}^{-1} \otimes \mathbf{q}$ , alors  $\mathbf{R}_\psi^\top \mathbf{R}(\mathbf{q}) = \mathbf{R}(\mathbf{q}_r)$  ;
- comme  $\mathbf{v}_b := \mathbf{R}^\top(\mathbf{q})(\mathbf{v} - \mathbf{w})$  (comme défini après l'équation (2.6)), alors  $\|\mathbf{v}_b\| = \|\mathbf{v} - \mathbf{w}\| - \|\mathbf{v}_r - \mathbf{w}_r\|$

— enfin  $\mathbf{R}^\top(\mathbf{q})\mathbf{w} = \mathbf{R}^\top(\mathbf{q}_r)\mathbf{R}_\psi^\top\mathbf{R}_\psi\mathbf{w}_r = \mathbf{R}^\top(\mathbf{q}_r)\mathbf{w}_r$ .

Sur la base des observations ci-dessus, nous pouvons dériver la version tournée des équations (2.8) comme étant :

$$\dot{\mathbf{p}}_r = \mathbf{v}_r, \quad (2.34a)$$

$$m\dot{\mathbf{v}}_r = -mg + \mathbf{R}(\mathbf{q}_r) \left( \mathbf{M}_f(\mathbf{u}) + \mathbf{D}_f(\mathbf{u}) \|\mathbf{w}_r\| \mathbf{R}^\top(\mathbf{q}_r)(\mathbf{v}_r - \mathbf{w}_r) \right), \quad (2.34b)$$

$$\dot{\mathbf{q}}_r = \left( \frac{1}{2} \mathbf{q}_r \otimes \begin{bmatrix} 0 \\ \omega_b \end{bmatrix} \right), \quad (2.34c)$$

$$\mathbf{J}\dot{\boldsymbol{\omega}}_b = -[\boldsymbol{\omega}_b]_\times \mathbf{J}\boldsymbol{\omega}_b + \mathbf{M}_m(\mathbf{u}) + \mathbf{D}_m(\mathbf{u}) \|\mathbf{w}_r\| \mathbf{R}^\top(\mathbf{q}_r)(\mathbf{v}_r - \mathbf{w}_r) \quad (2.34d)$$

Avec ces nouvelles coordonnées, le vecteur d'état incrémental (2.31) peut être exprimés comme étant :

$$\begin{aligned} \tilde{\mathbf{x}} &= (\mathbf{p}_r - \mathbf{R}_\psi^\top \mathbf{p}_{eq}, \mathbf{v}_r, \boldsymbol{\epsilon}_r - \mathbf{R}_\psi^\top \boldsymbol{\epsilon}_{eq}, \boldsymbol{\omega}_b), \\ \tilde{\mathbf{u}} &:= \mathbf{u} - \mathbf{u}_{eq}, \quad \tilde{\mathbf{w}} := \mathbf{w}_r - \mathbf{w}_{r,eq} \end{aligned} \quad (2.35)$$

où  $\mathbf{w}_{r,eq} = \mathbf{R}_\psi^\top \mathbf{w}_{eq} = \begin{bmatrix} w_{rx} \\ 0 \\ w_{rz} \end{bmatrix}$ , déjà défini dans (2.20), et  $\mathbf{R}_\psi^\top \boldsymbol{\epsilon}_{eq} = [0 \sin(\frac{\theta}{2}) 0]^\top$  ont tous deux une structure peu dense intéressante.

En se concentrant sur la dynamique tournée (2.34) et l'expression (2.35) des variables incrémentales, la preuve du théorème revient à montrer que la linéarisation de (2.34) autour de l'équilibre tourné :

$$\begin{aligned} \mathbf{x}_{r,eq} &= (\mathbf{p}_{r,eq}, \mathbf{v}_{r,eq}, \boldsymbol{\epsilon}_{r,eq}, \boldsymbol{\omega}_{br,eq}) \\ &= (\mathbf{R}_\psi^\top \mathbf{p}_{eq}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \sin(\frac{\theta}{2}) \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}), \quad \mathbf{w}_{r,eq} = \begin{bmatrix} w_{rx} \\ 0 \\ w_{rz} \end{bmatrix} \end{aligned} \quad (2.36)$$

coïncide avec l'équation (2.32) et les expressions de l'Algorithme 2.

Dans ce but, inspirée par [Tregouet 2015, Proof of Lemma 1], pour linéariser la dynamique du quaternion  $\mathbf{q}_r = [\eta_r \ \boldsymbol{\epsilon}_r^\top]^\top$  évoluant dans  $\mathbb{S}^3$ , nous remplaçons  $\eta_r$  par sa valeur positive liée à la norme unitaire du quaternion. Ainsi,  $\eta_r = (1 - \boldsymbol{\epsilon}_r^\top \boldsymbol{\epsilon}_r)^{\frac{1}{2}}$ . Concentrons-nous d'abord sur la matrice  $\mathbf{A}_w$  dans (2.32). Les trois premières lignes sont simplement  $[\mathbb{0}_3 \ \mathbb{I}_3 \ \mathbb{0}_3 \ \mathbb{0}_3]$  du fait de la linéarité de l'équation (2.34a). Pour le second bloc de lignes, nous nous concentrons sur l'équation (2.34b) et nous commençons par caractériser  $\mathbf{R}(\mathbf{q}_{r,eq})$ , dont la structure est relativement vide dû à  $\boldsymbol{\epsilon}_{r,eq}$ . Comme rappelé dans (2.24) et en utilisant l'expression  $\mathbf{R}$  de (2.2), nous pouvons

écrire :

$$\mathbf{R}(\mathbf{q}_{r,eq}) = \mathbf{R}_\theta := \begin{bmatrix} 1 - 2\bar{\epsilon}_2^2 & 0 & 2\bar{\epsilon}_2\bar{\eta} \\ 0 & 1 & 0 \\ -2\bar{\epsilon}_2\bar{\eta} & 0 & 1 - 2\bar{\epsilon}_2^2 \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix},$$

où  $\bar{\epsilon}_2 = \sin \frac{\theta}{2}$  représente le deuxième élément de  $\boldsymbol{\epsilon}_{r,eq}$  selon (2.36) et  $\bar{\eta} = \sqrt{1 - \bar{\epsilon}_2^2} = \cos \frac{\theta}{2}$ .

Avec cette expression de  $\mathbf{R}_\theta$ , nous pouvons dériver l'expression de (2.34b), en utilisant la notation abrégée  $\cdot|_{eq}$  pour caractériser l'évaluation d'une fonction (matricielle ou vectorielle) à l'équilibre (2.36),

$$\begin{aligned} \mathbf{A}_{vv} &= \frac{\partial}{\partial \mathbf{v}} \left( \frac{1}{m} \mathbf{R}(\mathbf{q}_r) (\mathbf{D}_f(\mathbf{u}) \| \mathbf{w}_r \| \mathbf{R}^\top(\mathbf{q}_r) (\mathbf{v}_r - \mathbf{w}_r)) \right) \Big|_{eq} \\ &= \frac{\partial}{\partial \mathbf{v}} \left( \frac{1}{m} \mathbf{R}_\theta \mathbf{D}_f(\mathbf{u}_{eq}) \| \mathbf{w}_{eq} \| \mathbf{R}_\theta^\top \mathbf{v}_r \right) \Big|_{eq}. \end{aligned} \quad (2.37)$$

Avec cette fonction et compte tenu de l'égalité  $\mathbf{D}_{f,eq} = \mathbf{D}_f(\mathbf{u}_{eq})$ , il est possible de montrer qu'elle coïncide avec la matrice  $\mathbf{A}_{vv}$  donnée en (2.50) dans l'Algorithme 2.

Nous nous concentrons maintenant sur  $\mathbf{A}_{ve}$  de la matrice  $\mathbf{A}_w$ , qui doit être calculée à partir de (2.34b) de manière similaire à (2.37), comme :

$$\mathbf{A}_{ve} = \frac{\partial}{\partial \boldsymbol{\epsilon}} \left( \frac{1}{m} \mathbf{R}(\mathbf{q}_r) (\mathbf{M}_f(\mathbf{u}) + \mathbf{D}_f(\mathbf{u}) \| \mathbf{w}_r \| \mathbf{R}^\top(\mathbf{q}_r) \mathbf{w}_r) \right) \Big|_{eq}. \quad (2.38)$$

Pour évaluer la partie droite de (2.38), nous démarrons de l'expression de  $\mathbf{R}(\boldsymbol{\epsilon}) = \mathbf{R}([\frac{\eta}{\epsilon}])$  dans (2.2). Après la substitution de  $\eta = \sqrt{1 - \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}} \neq 0$  (nous rappelons que pour tous les équilibres caractérisés, nous avons  $\eta \neq 0$ ), nous pouvons calculer la dérivée généralisée :

$$\begin{aligned} \partial \mathbf{R}_\epsilon(\boldsymbol{\epsilon}, \mathbf{v}) &:= \frac{\partial}{\partial \boldsymbol{\epsilon}} \mathbf{R} \left( \begin{bmatrix} \sqrt{1-\boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}} \\ \boldsymbol{\epsilon} \end{bmatrix} \right) \mathbf{v} \\ &= 2\eta [\mathbf{v}]_\times \left( \frac{\boldsymbol{\epsilon} \boldsymbol{\epsilon}^\top}{1 - \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}} - \mathbb{I}_3 \right) - 4\mathbf{v}\boldsymbol{\epsilon}^\top + 2\boldsymbol{\epsilon}\mathbf{v}^\top + 2\boldsymbol{\epsilon}^\top \mathbf{v} \mathbb{I}_3, \end{aligned} \quad (2.39)$$

qui implique donc :

$$\frac{\partial}{\partial \boldsymbol{\epsilon}} \mathbf{R}^\top([\frac{\eta}{\epsilon}]) \mathbf{v} = \frac{\partial}{\partial \boldsymbol{\epsilon}} \mathbf{R} \left( \begin{bmatrix} \sqrt{1-\boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}} \\ \boldsymbol{\epsilon} \end{bmatrix} \right) \mathbf{v} = \partial \mathbf{R}_\epsilon(-\boldsymbol{\epsilon}, \mathbf{v}). \quad (2.40)$$

Pour évaluer (2.38), il sera utile de dériver la forme simplifiée suivante :

$$\begin{aligned} & \partial \mathbf{R}_\epsilon \left( \begin{bmatrix} 0 \\ \epsilon_2 \\ 0 \end{bmatrix}, \begin{bmatrix} \mathbf{v}_1 \\ 0 \\ \mathbf{v}_3 \end{bmatrix} \right) \\ &= 2 \begin{bmatrix} 0 & \left(\bar{\eta} - \frac{\bar{\epsilon}_2^2}{\bar{\eta}}\right) \mathbf{v}_3 & 0 \\ -\bar{\eta} \mathbf{v}_3 & 0 & \bar{\eta} \mathbf{v}_1 \\ 0 & \left(\frac{\bar{\epsilon}_2^2}{\bar{\eta}} - \bar{\eta}\right) \mathbf{v}_1 & 0 \end{bmatrix} + 2\bar{\epsilon}_2 \begin{bmatrix} 0 & -2\mathbf{v}_1 & 0 \\ \mathbf{v}_1 & 0 & \mathbf{v}_3 \\ 0 & -2\mathbf{v}_3 & 0 \end{bmatrix}. \end{aligned} \quad (2.41)$$

Nous pouvons définir deux forces  $(f_d, f_\ell)$  qui agissent sur le drone à l'équilibre, exprimées dans le repère corps, et qui dépendent du vent  $\mathbf{w}$  et des deux entrées similaires des élévons  $\delta$ . Ces deux forces sont la traînée et la portance générées par l'écoulement de l'air sur l'aile. Elles résultent du développement de l'expression  $\mathbf{D}_f(\mathbf{u})\|\mathbf{v}_b\|\mathbf{v}_b$  provenant de (2.34b) avec  $\mathbf{D}_f(\mathbf{u})$  de (2.11) :

$$\begin{bmatrix} f_d \\ 0 \\ f_\ell \end{bmatrix} = -\mathbf{D}_f(\mathbf{u}_{eq})\|\mathbf{w}_{eq}\|\mathbf{R}_\theta^\top \mathbf{w}_{r,eq}. \quad (2.42)$$

Après calcul, cette expression coïncide avec celle de (2.48) donnée dans l'Algorithm 2.

À partir des deux forces  $(f_d, f_\ell)$  dans (2.42), il est possible de déterminer leurs dérivées partielles par rapport à la composante  $\bar{\epsilon}_2$  du quaternion, qui représente le tangage du drone. En utilisant (2.40), nous obtenons :

$$\begin{bmatrix} \frac{\partial f_d}{\partial \bar{\epsilon}_2} \\ 0 \\ \frac{\partial f_\ell}{\partial \bar{\epsilon}_2} \end{bmatrix} = -\mathbf{D}_f(\mathbf{u}_{eq})\|\mathbf{w}_{eq}\| \partial \mathbf{R}_\epsilon(-\epsilon, \mathbf{w}_{r,eq}) \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad (2.43)$$

qui après calcul en utilisant l'égalité  $\mathbf{D}_{f,eq} = \mathbf{D}_f(\mathbf{u}_{eq})$ , coïncide avec l'équation (2.48), donnée dans l'Algorithm 2.

En suivant des calculs similaires, la force  $f_m$  générée par les moteurs, liée à la traction des hélices et à la traînée générée par l'écoulement de l'air sur l'aile, et la force  $f_e$  générée par les élévons, liée à l'écoulement de l'air créé par les hélices, sont obtenues à partir de (2.9) et sont définies par :

$$\begin{bmatrix} f_m \\ 0 \\ f_e \end{bmatrix} = \mathbf{M}_f(\mathbf{u}_{eq}). \quad (2.44)$$

Cela coïncide, après calcul, avec les sélections de (2.49), données dans l'Algorithm 2.

En utilisant les définitions (2.39), (2.40), ainsi que les expressions (2.41), (2.42), (2.44), et leurs formes équivalentes indiquées dans (2.48), (2.49) données dans l'Al-

gorithme 2, nous pouvons finalement calculer à partir de (2.38) :

$$\begin{aligned} \mathbf{A}_{v\epsilon} = & \frac{1}{m} (\partial \mathbf{R}_\epsilon(\boldsymbol{\epsilon}, \mathbf{M}_f(\mathbf{u}_{eq})) - \partial \mathbf{R}_\epsilon(\boldsymbol{\epsilon}, \mathbf{D}_f(\mathbf{u}_{eq}) \|\mathbf{w}_{eq}\| \mathbf{R}_\theta^\top \mathbf{w}_{eq}) \\ & - \mathbf{R}_\theta \mathbf{D}_f(\mathbf{u}) \|\mathbf{w}_{r,eq}\| \partial \mathbf{R}_\epsilon(-\boldsymbol{\epsilon}, \mathbf{w}_{r,eq}) \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} ) \Big|_{eq}, \end{aligned}$$

qui fournit l'expression (2.51) dans l'Algorithme 2 après calculs exploitant également  $\mathbf{D}_{text{tf},eq} = \mathbf{D}_{text{tf}}(\mathbf{u}_{eq})$ .

Nous évaluons maintenant la matrice  $\mathbf{A}_{\epsilon\omega}$  de  $\mathbf{A}_w$ , et nous rappelons que, en raison des propriétés du produit de quaternion (voir, par exemple, [Hua 2013]),  $[\eta] \otimes [\omega_b] = [\begin{smallmatrix} -\epsilon^\top \\ \eta \mathbb{I}_3 + [\epsilon]_\times \end{smallmatrix}] \omega_b$ . À partir des deux termes inférieurs de la matrice du côté droit de cette dernière équation, en développant (2.34c) et en calculant  $\mathbf{A}_{\epsilon\omega} = \frac{\partial}{\partial \omega_b} \left( \frac{1}{2} \mathbf{q}_r \otimes [\omega_b] \right) \Big|_{eq}$ , nous obtenons les deux termes de l'expression (2.51) donnée dans l'Algorithme 2.

A propos de la matrice  $\mathbf{A}_{\omega\epsilon}$  de  $\mathbf{A}_w$ , qui doit être calculée à partir de (2.34d). Comme seul le dernier terme de la partie droite dépend de  $\boldsymbol{\epsilon}$  (par l'intermédiaire de  $\mathbf{q}_r$ ), nous obtenons :

$$\mathbf{A}_{\omega\epsilon} = \mathbf{J}^{-1} \mathbf{D}_m(\mathbf{u}_{eq}) \|\mathbf{w}_r\| \frac{\partial}{\partial \boldsymbol{\epsilon}} \left( \mathbf{R}^\top(\mathbf{q}_r)(\mathbf{v}_r - \mathbf{w}_r) \right) \Big|_{eq}. \quad (2.45)$$

Pour calculer l'expression explicite de (2.45), nous exploitons à nouveau (2.40) et (2.41), et utilisons l'expression de  $\mathbf{D}_m$  dans (2.12), ainsi que les identités  $\bar{\eta}^2 - \bar{\epsilon}_2^2 = \cos \theta$  et  $2\bar{\eta}\bar{\epsilon}_2 = \sin \theta$ , qui fournissent, après quelques simplifications, l'expression (2.52), donnée dans l'Algorithme 2.

Passons maintenant à la dérivation des entrées de la matrice  $\mathbf{G}_w$  dans (2.32), dont les composantes peuvent être dérivées de (2.34b) et (2.34d). En utilisant les quatre entrées de  $\mathbf{u}$  dans (2.7), et en se basant également sur la structure de  $\mathbf{M}_f$ ,  $\mathbf{D}_f$ , dans (2.9), (2.11), la forme explicite pour :

$$\mathbf{G}_v = \frac{1}{m} \mathbf{R}_\theta \frac{\partial}{\partial \mathbf{u}} \left( \mathbf{M}_f(\mathbf{u}) - \mathbf{D}_f(\mathbf{u}) \|\mathbf{w}_r\| \mathbf{w}_{eq}^b \right) \Big|_{eq}, \quad (2.46)$$

peut être calculée comme dans (2.53), après factorisations.

De même, sur la base des matrices  $\mathbf{M}_m$ ,  $\mathbf{D}_m$  dans (2.10), (2.12), nous pouvons calculer :

$$\mathbf{G}_\omega = \mathbf{J}^{-1} \frac{\partial}{\partial \mathbf{u}} \left( \mathbf{M}_m(\mathbf{u}) - \mathbf{D}_m(\mathbf{u}) \|\mathbf{w}_r\| \mathbf{w}_{eq}^b \right) \Big|_{eq} \quad (2.47)$$

comme dans (2.54), après factorisations.

Déterminons enfin l'expression de  $\mathbf{E}_v$  dans (2.32). Notons d'abord que nous pouvons écrire  $\|\mathbf{w}_r\|\mathbf{w}_r = \mathbf{w}_r\sqrt{\mathbf{w}_r^\top \mathbf{w}_r}$ , de sorte que

$$\frac{\partial}{\partial \mathbf{w}_r} \|\mathbf{w}_r\|\mathbf{w}_r = \|\mathbf{w}_r\| \mathbb{I}_3 + \frac{\mathbf{w}_r \mathbf{w}_r^\top}{\|\mathbf{w}_r\|} = \|\mathbf{w}_r\| \left( \mathbb{I}_3 + \frac{\mathbf{w}_r \mathbf{w}_r^\top}{\mathbf{w}_r^\top \mathbf{w}_r} \right).$$

À l'aide de (2.34b) et (2.34d) et de l'expression de  $\mathbf{w}_r$  dans (2.33) et en suivant des calculs similaires aux cas précédents, nous obtenons l'expression (2.55) (indiquée dans l'Algorithme 2), pour  $\mathbf{E}_v = -\frac{1}{m} \mathbf{R}_\theta \frac{\partial}{\partial \mathbf{w}_r} (\mathbf{D}_f(\mathbf{u}) \|\mathbf{w}_r\|\mathbf{w}_r) \Big|_{\text{eq}}$  et  $\mathbf{E}_w = -\mathbf{J}^{-1} \frac{\partial}{\partial \mathbf{w}_r} (\mathbf{D}_m(\mathbf{u}) \|\mathbf{w}_r\|\mathbf{w}_r) \Big|_{\text{eq}}$ , où nous rappelons que  $\mathbf{D}_{m,\text{eq}} = \mathbf{D}_m(\mathbf{u}_{\text{eq}})$ .  $\square$

## 2.5 Conclusion du Chapitre 2

Conclusion sur la modification du vecteur de commande

**Algorithme 2** Détermination des matrices de la linéarisation de (2.32)

**Entrées** : Vecteur de vent  $\mathbf{w}_{\text{eq}} = [w_x \ w_y \ w_z]^\top$  et

d'équilibre  $(\mathbf{u}_{\text{eq}}, \mathbf{x}_{\text{eq}})$  provenant de (2.17) et de l'Algorithmme 1.

**Sorties** : Matrices  $\mathbf{A}_w$ ,  $\mathbf{G}_w$ ,  $\mathbf{E}_w$  dans (2.32)

- 1: Sélectionner les paramètres  $\psi$ ,  $\theta$ ,  $\tau$ ,  $\delta$  de (2.17) à l'aide de l'Algorithmme 1 et de  $\bar{\epsilon}_2 = \sin \frac{\theta}{2}$ ,  $\bar{\eta} = \cos \frac{\theta}{2}$ .
- 2: Avec les valeurs de (2.24), (2.11), (2.12), définissons :

$$\begin{aligned}\mathbf{R}_\psi &:= \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{R}_\theta := \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, \\ \begin{bmatrix} w_{rx} \\ 0 \\ w_{rz} \end{bmatrix} &:= \mathbf{R}_\psi^\top \mathbf{w}_{\text{eq}}, \quad \begin{bmatrix} w_x^b \\ w_z^b \end{bmatrix} := \begin{bmatrix} w_{rx} \cos \theta - w_{rz} \sin \theta \\ w_{rz} \cos \theta + w_{rx} \sin \theta \end{bmatrix} \\ \left[ \mathbf{D}_{f,\text{eq}} \middle| \mathbf{D}_{m,\text{eq}} \right] &:= \frac{\rho S}{2} \begin{bmatrix} -C_d & 0 & C_d \xi_f \delta & 0 \\ 0 & 0 & 0 & \Delta_r C_\ell \xi_m \delta \\ -C_\ell \xi_f \delta & 0 & -C_\ell & 0 \\ 0 & 0 & 0 & 2 \Delta_r C_\ell \end{bmatrix}\end{aligned}$$

- 3: Définissons les forces de portance et de trainé ainsi que leurs dérivées par rapport à  $\epsilon_2$  (défini dans l'étape 1), comme

$$\begin{bmatrix} f_d & \frac{\partial f_d}{\partial \epsilon_2} \\ 0 & 0 \\ f_\ell & \frac{\partial f_\ell}{\partial \epsilon_2} \end{bmatrix} := -\|\mathbf{w}_{\text{eq}}\| \mathbf{D}_{f,\text{eq}} \begin{bmatrix} w_x^b \left(4\bar{\eta} - \frac{2\bar{\epsilon}_2^2}{\bar{\eta}}\right) w_{rz} - 8\bar{\epsilon}_2 w_{rx} \\ 0 \\ w_z^b \left(4\bar{\eta} - \frac{2\bar{\epsilon}_2^2}{\bar{\eta}}\right) w_{rx} - 8\bar{\epsilon}_2 w_{rz} \end{bmatrix}, \quad (2.48)$$

- 4: Définissons les forces des moteurs et des élevons comme

$$\begin{bmatrix} f_m \\ f_e \end{bmatrix} := \begin{bmatrix} \left(\frac{S_{\text{wet}} C_d}{2 S_p} - 2\right) \tau \\ -\frac{S_{\text{wet}} \tau \delta \xi_f C_\ell}{2 S_p} \end{bmatrix} \quad (2.49)$$

- 5: Sélectionnons les matrices  $\mathbf{A}_w$  dans (2.32) comme :

$$\mathbf{A}_{vv} = \frac{\|\mathbf{w}_{\text{eq}}\|}{m} \mathbf{R}_\theta \mathbf{D}_{f,\text{eq}} \mathbf{R}_\theta^\top \quad (2.50)$$

$$\begin{bmatrix} \mathbf{A}_{v\epsilon}^{1,2} \\ \mathbf{A}_{v\epsilon}^{2,1} \\ \mathbf{A}_{v\epsilon}^{2,3} \\ \mathbf{A}_{v\epsilon}^{3,2} \end{bmatrix} := \begin{bmatrix} 2\bar{\eta} - \frac{\bar{\epsilon}_2^2}{\bar{\eta}} & 4\bar{\epsilon}_2 & 2\bar{\epsilon}_2^2 - 1 & 2\bar{\epsilon}_2 \bar{\eta} \\ -2\bar{\eta} & -2\bar{\epsilon}_2 & 0 & 0 \\ 2\bar{\epsilon}_2 & -2\bar{\eta} & 0 & 0 \\ -4\bar{\epsilon}_2 & 2\bar{\eta} - \frac{\bar{\epsilon}_2^2}{\bar{\eta}} & -2\bar{\epsilon}_2 \bar{\eta} & 1 - 2\bar{\epsilon}_2^2 \end{bmatrix} \begin{bmatrix} f_e + f_\ell \\ f_m + f_d \\ \frac{\partial f_d}{\partial \epsilon_2} \\ \frac{\partial f_\ell}{\partial \epsilon_2} \end{bmatrix}$$

$$\mathbf{A}_{v\epsilon} = \frac{1}{m} \begin{bmatrix} 0 & \mathbf{A}_{v\epsilon}^{1,2} & 0 \\ \mathbf{A}_{v\epsilon}^{2,1} & 0 & \mathbf{A}_{v\epsilon}^{2,3} \\ 0 & \mathbf{A}_{v\epsilon}^{3,2} & 0 \end{bmatrix}, \quad \mathbf{A}_{\epsilon\omega} = \frac{\bar{\eta}}{2} \mathbb{I}_3 + \frac{\bar{\epsilon}_2}{2} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad (2.51)$$

$$\mathbf{A}_{\omega\epsilon} = \frac{\rho S C_\ell \Delta_r \|\mathbf{w}_{\text{eq}}\| (w_x^b - \xi_m \delta w_z^b)}{J_y \bar{\eta}} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.52)$$

6: Sélectionnons les matrices  $\mathbf{G}_w$  dans (2.32) comme :

$$\begin{aligned}\mathbf{G}_v &= \frac{1}{m} \mathbf{R}_\theta [\mathbf{G}_{v\tau} | \mathbf{G}_{v\delta}], \quad \mathbf{G}_{v\tau} := \begin{bmatrix} 1 - \frac{S_{\text{wet}} C_d}{4 S_p} \\ 0 \\ -\frac{S_{\text{wet}} C_\ell \xi_f \delta}{2 S_p} \end{bmatrix} [1]^\top \\ \mathbf{G}_{v\delta} &:= \begin{bmatrix} -\frac{1}{4} \rho S C_d \xi_f \|w_{\text{eq}}\| w_z^b \\ 0 \\ -\frac{S_{\text{wet}} C_\ell \xi_f \tau}{2 S_p} + \frac{1}{4} \rho S C_\ell \xi_f \|w_{\text{eq}}\| w_x^b \end{bmatrix} [1]^\top\end{aligned}\quad (2.53)$$

$$\begin{aligned}\mathbf{G}_\omega &= \mathbf{J}^{-1} [\mathbf{G}_{\omega\tau} \ \mathbf{G}_{\omega\delta}], \quad \mathbf{G}_{\omega\delta} := \frac{S_{\text{wet}} C_\ell \tau}{4 S_p} \begin{bmatrix} a_y \xi_f & -a_y \xi_f \\ \Delta_r \xi_m & \Delta_r \xi_m \\ 0 & 0 \end{bmatrix} + \\ &\quad + \frac{\rho S \|w_{\text{eq}}\| \xi_m}{4} \begin{bmatrix} a_y C_d w_x^b & -a_y C_d w_x^b \\ \Delta_r C_\ell w_x^b & \Delta_r C_\ell w_x^b \\ a_y C_\ell w_z^b & -a_y C_\ell w_z^b \end{bmatrix} \\ \mathbf{G}_{\omega\tau} &:= \begin{bmatrix} \frac{k_m}{k_f} + \frac{S_{\text{wet}}}{4 S_p} a_y \xi_f C_\ell \delta \\ 0 \\ p_y + \frac{S_{\text{wet}}}{4 S_p} a_y C_d \end{bmatrix} [1]^\top + \begin{bmatrix} 0 \\ \frac{S_{\text{wet}}}{4 S_p} \Delta_r \xi_m C_\ell \delta \\ 0 \end{bmatrix} [1]^\top\end{aligned}\quad (2.54)$$

7: Sélectionnons les matrices  $\mathbf{E}_w$  de (2.32) comme :

$$\begin{bmatrix} \mathbf{E}_v \\ \mathbf{E}_\omega \end{bmatrix} = - \begin{bmatrix} \mathbf{A}_{vv} \\ \mathbf{J} \|w_{\text{eq}}\| \mathbf{D}_{m,\text{eq}} \mathbf{R}_\theta^\top \end{bmatrix} \left( \mathbb{I}_3 + \frac{\mathbf{R}_\psi^\top w_{\text{eq}} w_{\text{eq}}^\top \mathbf{R}_\psi}{w_{\text{eq}}^\top w_{\text{eq}}} \right)\quad (2.55)$$

**Retourne :**  $\mathbf{A}_w, \mathbf{G}_w, \mathbf{E}_w$



## CHAPITRE 3

# Commande hybride

---

### Sommaire

<b>3.1</b>	<b>Motivation</b>	<b>47</b>
<b>3.2</b>	<b>Contrôleur par retour d'état non-linéaire</b>	<b>48</b>
<b>3.3</b>	<b>Contrôleur par retour d'état linéaire</b>	<b>49</b>
<b>3.4</b>	<b>Conception d'une commande locale-globale basée sur l'hystérésis</b>	<b>51</b>
<b>3.5</b>	<b>Conclusion du Chapitre 3</b>	<b>54</b>

---

### 3.1 Motivation

En prenant en compte les capacités d'un *tailsitter*, il est légitime de se poser la question du mode de vol utilisé pour rejoindre un point. Effectivement, le drone a la possibilité de se déplacer en stationnaire ou bien en vol d'avancement. Lors d'un déplacement en stationnaire, le drone est vertical donc il se retrouve fortement sujet aux perturbations. Il est donc nécessaire d'avoir une grande région d'attraction autour de la position d'équilibre pour assurer un rejet des perturbations et une stabilisation.

Nous avons donc proposé une stratégie de commande pour stabiliser le drone en position stationnaire. Cette stratégie repose sur une dynamique discrète permettant le passage d'une loi de commande non-linéaire présentée dans [Michieletto 2020] et qui fournit une grande région d'attraction et une seconde loi basée sur la dynamique linéarisée et fournissant une agressivité supérieure pour réaliser l'approche finale. Les deux contrôleurs sont réunis par un mécanisme hybride qui permet de conserver les performances en régime permanent de la conception linéarisée avec la grande région d'attraction garantie par la conception non linéaire. Notre solution est testée en simulant le modèle non linéaire complet.

Nous allons nous concentrer, dans cette partie, à la stabilisation stationnaire du drone. Ainsi nous nous appuyons sur la dynamique simplifiée décrite dans la section

2.1.2, avec la simplification  $\mathbf{w} = 0$  qui permet d'obtenir la dynamique simplifiée sans vent (2.14).

### 3.2 Contrôleur par retour d'état non-linéaire

Nous illustrons dans cette section une loi de contrôle dynamique non linéaire inspirée du résultat de [Michieletto 2020]. Pour que cette loi de contrôle non linéaire soit applicable, les matrices  $F$  et  $M$  mentionnées dans (2.14) doivent permettre de définir une direction dite de zéro moment  $\bar{\mathbf{u}} \in \mathbb{R}^4$  garantissant  $|F\bar{\mathbf{u}}| = 1$  et  $M\bar{\mathbf{u}} = 0$ , et la matrice inverse à droite  $M^r$  de  $M$  doit satisfaire  $MM^r = I$  et  $FM^r = 0$ .

Dans notre cas, il est immédiat de voir que la direction du moment zéro  $\bar{\mathbf{u}} = \frac{\sqrt{2}}{2a_f} [1 \ 1 \ 0 \ 0]^\top$  satisfait les conditions, alors que le fait que  $\text{rank}(F) = 2$  (donc que le noyau de  $F$  ( $\ker F$ ) soit de dimension 2) rend impossible l'obtention de la matrice inverse à droite  $M^r$  de  $M$  entièrement contenue dans  $\ker F$ .

Nous déterminons  $M^r$  en paramétrisant (de manière conservatrice) les pseudo-inverses à droites de  $M$  comme  $M^r := KM^\top(MKM^\top)^{-1}$ , où la matrice  $K \in \mathbb{R}^{4 \times 4}$  est symétrique et satisfait  $MKM^\top \geq I$  (pour assurer l'inversibilité). Avec cette paramétrisation, le but est de minimiser la norme de  $FM^r = FKM^\top(MKM^\top)^{-1}$ , ce qui est bien réalisé en minimisant la norme de  $FKM^\top$ , du fait que la contrainte sur  $MKM^\top \geq I$  garantisse que le facteur  $(MKM^\top)^{-1}$  ait une norme plus petite que 1. En effectuant un complément de Schur, cette minimisation est obtenue en résolvant le programme semi-défini suivant :

$$\min_{K, \kappa} \kappa, \text{ subject to : } MKM^\top \geq I, \begin{bmatrix} \kappa I & FKM^\top \\ MK^\top F^\top & \kappa I \end{bmatrix} \geq 0,$$

lequel minimise  $\kappa$  tout en assurant  $FKM^\top MK^\top F^\top \leq \kappa^2 I$ . En résolvant cette optimisation, on obtient, pour les matrices spécifiques considérées,

$$K = \begin{bmatrix} 0 & -737 & 171 & -171 \\ -737 & 0 & -171 & 171 \\ 171 & -171 & 1583.5 & -43.73 \\ -171 & 171 & -43.73 & 1583.5 \end{bmatrix}, M^r = \begin{bmatrix} 0 & 0 & -3.19 \\ 0 & 0 & 3.19 \\ -4.51 & -27.75 & -1.48 \\ 4.51 & -27.75 & 1.48 \end{bmatrix}$$

conduisant à  $\kappa = 39,7$ . Avec cette sélection basée sur l'optimalité, la conception dynamique non linéaire de [Michieletto 2020] peut être appliquée efficacement en obtenant des réponses qui sont presque impossibles à distinguer du cas entièrement découpé  $FM^r = 0$ . Il convient de noter qu'une approche similaire, négligeant essentiellement les termes supplémentaires agissant sur la dynamique de translation, est également suggérée dans l'étude [Hua 2013]. Sur la base du choix de  $M^r$  et de  $\bar{\mathbf{u}}$

décrit ci-dessus, en appliquant la loi de commande de [Michieletto 2020, eqn (19)], l'entrée  $\mathbf{u}$  devient :

$$\mathbf{u} = \mathbf{u}_{\text{nl}} := M^r \boldsymbol{\tau}_r + \bar{\mathbf{u}} \mathbf{f}, \quad (3.1)$$

où  $\boldsymbol{\tau}_r$  et  $\mathbf{f}$  sont fournis par le contrôleur proposé dans [Michieletto 2020].

La sélection de  $M^r$  basée sur l'optimalité peut être interprétée de manière intéressante lorsqu'on observe le produit  $M^r \boldsymbol{\tau}_r = M^r [\tau_{r,x} \ \tau_{r,y} \ \tau_{r,z}]^t op.$

Premièrement, pour obtenir un moment  $\tau_{r,z}$  autour de l'axe  $z_b$ , nous utilisons principalement l'action différentielle de la poussée ; deuxièmement, un moment  $\tau_{r,y}$  autour de l'axe  $y_b$  est généré par une utilisation symétrique des deux volets, avec une grande efficacité ; enfin, un moment  $\tau_{r,x}$  autour de l'axe  $x_b$  provient d'une utilisation différentielle des volets.

Enfin, par rapport à la solution proposée dans [Michieletto 2020], pour prendre partiellement en compte les effets de saturation énoncés dans la section 2.2.1, le bouclage décrit dans [Michieletto 2020] a été augmenté d'une stratégie de saturation d'erreur ne permettant jamais à l'erreur de position  $\mathbf{e}_p$  utilisée dans [Michieletto 2020, eqn. (22)] de dépasser la valeur maximale de 3 mètres. Les autres gains de réglage nécessaires à la solution de [Michieletto 2020] ont été sélectionnés en suivant une procédure de réglage des gains proportionnels et dérivés qui a conduit à  $k_{pp} = 0.5$ ,  $k_{pd} = 1.2$ ,  $k_{ap} = 0.08$ ,  $k_{ad} = 0.1$  et  $k_\Delta = 1$ .

La figure 3.1 montre la réponse du système en termes de positions, d'orientations (deux lignes du haut) et d'efforts des actionneurs (deux lignes du bas) lorsque le système part de la condition initiale  $\mathbf{x}(0) = [\mathbf{p}(0) \ \mathbf{v}(0) \ \mathbf{q}(0) \ \omega_b(0)]^\top = [0 \ 0 \ 0 \ 0 \ 0 \ 0.9140 \ 0.1134 \ -0.3728 \ 0.1134 \ 0 \ 0 \ 0]^T$  avec une position d'équilibre cible de  $\mathbf{p}_{\text{eq}} = [4 \ 5 \ 6]^\top$  et  $\mathbf{q}_{\text{eq}} = [\frac{\sqrt{2}}{2} \ 0 \ -\frac{\sqrt{2}}{2} \ 0]^\top$ . Une réponse adéquate peut être observée, laquelle reste assez éloignée des saturations des actionneurs (voir section 2.2.1). L'augmentation des gains peut accélérer la réponse, mais produit toutefois des oscillations d'attitude indésirables. Il est donc intéressant de combiner ce contrôleur non linéaire (qui fournit une grande région d'attraction) avec un contrôleur plus agressif, conçu sur la base de la dynamique linéarisée sans vent (2.29).

### 3.3 Contrôleur par retour d'état linéaire

Sur la base des observations de la section précédente et étant donné une position cible correspondant à un équilibre  $\mathbf{p}_{\text{eq}}$ ,  $\mathbf{q}_{\text{eq}}$  tel que caractérisé dans l'équation 2.16, nous concevons ici un contrôleur par retour d'état linéaire capable de produire une

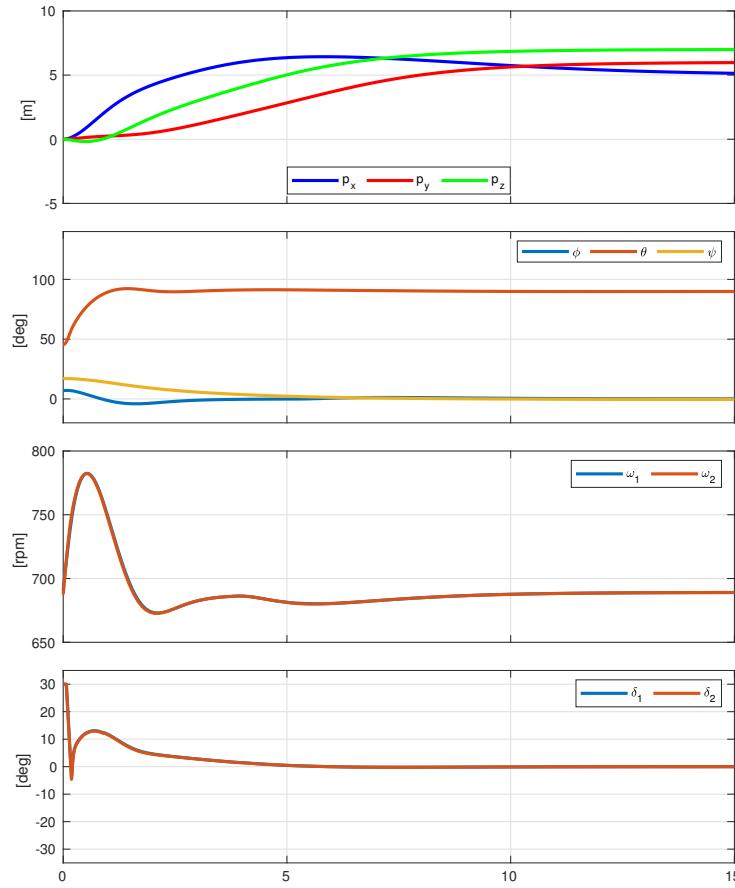


FIGURE 3.1 – Simulation de la loi de commande non-linéaire avec la dynamique de DarkO (2.1).

réponse plus agressive. Pour cela, nous nous concentrons sur la dynamique linéarisée (2.29) et proposons une loi de commande de la forme :

$$\mathbf{u}_{\text{lin}} := \mathbf{u}_{\text{eq}} - \mathbf{K}\tilde{\mathbf{x}}, \quad (3.2)$$

où  $\tilde{\mathbf{x}}$  a été introduit dans (2.29) et  $\mathbf{K} \in \mathbb{R}^{4 \times 12}$  est un gain de retour d'état qui peut être sélectionné, sur la base des matrices  $\mathbf{A}_0$  et  $\mathbf{G}_0$  apparaissant dans (2.29), de telle sorte que la boucle fermée du retour d'état  $A_{\text{cl}} := \mathbf{A}_0 - \mathbf{G}_0 \mathbf{K}$  soit exponentiellement stable.

Dans notre cas, nous avons utilisé une sélection basée sur la commande linéaire quadratique, LQR , associée aux matrices de pondération  $\mathbf{Q} = I_{12}$  et  $\mathbf{R} = I_4$ , qui donne une réponse en boucle fermée désirable. La conception LQR fournit également une matrice définie positive  $\mathbf{S} \in \mathbb{R}^{12 \times 12}$  (solution de l'équation algébrique

### **3.4. Conception d'une commande locale-globale basée sur l'hystérésis**

de Riccati) garantissant que  $\mathbf{A}_{\text{cl}}^\top \mathbf{S} + \mathbf{S} \mathbf{A}_{\text{textcl}} < 0$ . Il est donc possible d'utiliser  $\mathbf{S}$  pour former une fonction de Lyapunov. En particulier, il est bien connu, d'après le théorème d'approximation linéaire, que la fonction  $V(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^\top \mathbf{S} \tilde{\mathbf{x}}$  est également une fonction de Lyapunov certifiant la stabilité exponentielle locale de  $\mathbf{x}_{\text{eq}}$  pour la dynamique non-linéaire. Plus précisément, il existe un scalaire positif  $\bar{v} \in \mathbb{R}$  tel que, le long de la dynamique (2.1), nous avons :

$$V(\tilde{\mathbf{x}}) \leq \bar{v} \quad \Rightarrow \quad \dot{V}(\tilde{\mathbf{x}}) := \langle \nabla V(\tilde{\mathbf{x}}), \dot{\tilde{\mathbf{x}}} \rangle < 0, \quad (3.3)$$

pour tout  $\tilde{\mathbf{x}} \neq 0$ ; en d'autres termes, le sous-ensemble de  $V(\tilde{\mathbf{x}}) \leq \bar{v}$  est contenu dans le bassin d'attraction de l'équilibre  $\mathbf{x}_{\text{eq}}$ .

La détermination du plus grand scalaire  $\bar{v}$  assurant (3.3) est un problème complexe et des bornes inférieures conservatrices peuvent être déterminées en quantifiant l'effet des non-linéarités sur la dynamique. Puisque  $\dot{\tilde{\mathbf{x}}}$  est une fonction de  $\mathbf{x}$ , il est assez facile d'évaluer algébriquement  $\dot{V}(\tilde{\mathbf{x}})$  pour un grand nombre d'extractions aléatoires de la variable  $\tilde{\mathbf{x}}$ , afin d'obtenir une estimation probabiliste du plus grand scalaire  $\bar{v}$ . Des garanties rigoureuses sur ces sélections peuvent être obtenues en appliquant les résultats de [Tempo 2013], mais une évaluation de 10000 échantillons a confirmé que la valeur  $\bar{v} = 400$  est une sélection satisfaisante (3.3).

La figure 3.2 montre une simulation commençant à l'origine avec un drone vertical et des vitesses linéaires et angulaires initiales nulles. La position cible est  $\mathbf{p}_{\text{eq}} = [4, 5, 6]$  avec une stabilisation en vol stationnaire (drone vertical) avec  $\beta = 0$ . La ligne pointillée représente la position de la cible sur chaque axe. Le dernier graphique montre la décroissance exponentielle souhaitable de  $V$ . La figure 3.2 montre, à la fois, la simulation du modèle complet (continue) (2.1) et du modèle non linéaire simplifié (2.14) (en pointillé), ce qui met en évidence des différences dans la phase transitoire. Lorsqu'on fournit une position cible plus importante  $\mathbf{p}_{\text{eq}} = [8, 9, 10]$  (avec la même orientation), la condition initiale se situe en dehors du bassin d'attraction et une divergence apparaît, comme le montre la figure 3.3.

### **3.4 Conception d'une commande locale-globale basée sur l'hystérésis**

Considérant les stratégies locales-globales présentées dans [Goebel 2012, Ex. 1.7], similaires à la solution présentée dans [Andreetto 2016], nous utilisons un mécanisme hybride sélectionnant le contrôleur local agressif (3.2) (tant que l'état se

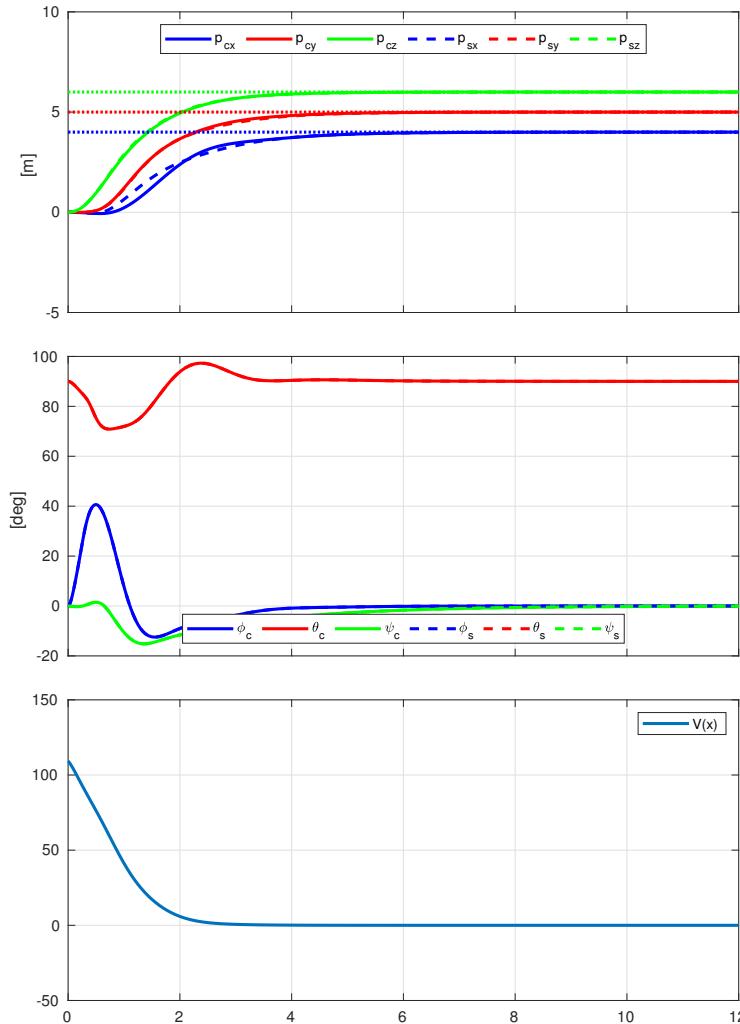


FIGURE 3.2 – Simulation du modèle complet (2.1) (ligne continue) et simplifié (2.14) (ligne pointillée) avec  $\mathbf{u} = \mathbf{u}_{\text{lin}}$  défini dans (3.2) et comme une condition initiale  $\tilde{\mathbf{x}}_0$  dans le bassin d'attraction.

trouve dans le bassin d'attraction de l'équilibre) ou le contrôleur non linéaire moins agressif (3.1), qui fournit une plus grande région d'attraction (et peut être appelé, par abus de langage, le "contrôleur global"). À cette fin, nous ajoutons à l'état du contrôleur une variable d'état logique  $\ell \in \{0, 1\}$ , qui régit le choix de l'entrée de contrôle entre (3.1) et (3.2) tel que

$$\mathbf{u} = \mathbf{u}_{\text{hyb}} := \ell \mathbf{u}_{\text{nl}} + (1 - \ell) \mathbf{u}_{\text{lin}}, \quad (3.4)$$

### 3.4. Conception d'une commande locale-globale basée sur l'hystérésis

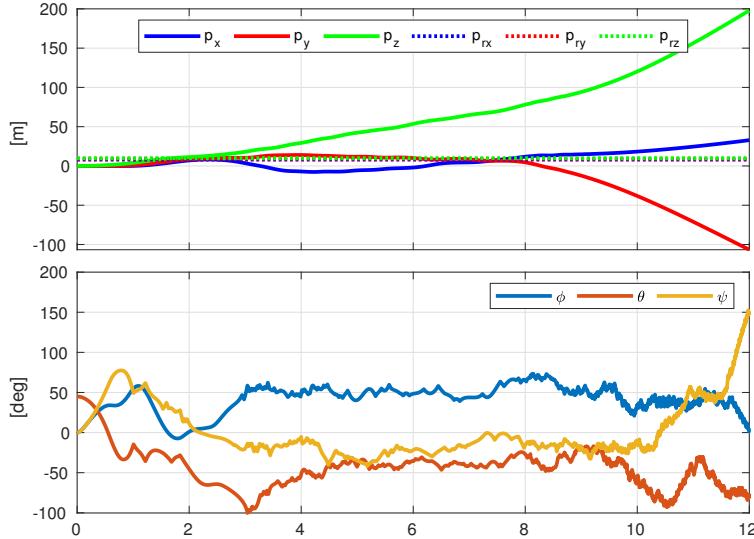


FIGURE 3.3 – Simulation divergente du modèle complet (2.1) avec  $\mathbf{u} = \mathbf{u}_{\text{lin}}$  défini dans (3.2) et une condition initiale  $\tilde{\mathbf{x}}_0$  en dehors du bassin d'attraction.

Nous nous assurons, grâce à la dynamique hybride, que  $\ell$  ne puisse prendre que des valeurs dans  $\{0, 1\}$ . Sa dynamique est définie par :

$$\begin{cases} \dot{\ell} = 0, & \chi \in \mathcal{C} \\ \ell^+ = 1 - \ell, & \chi \in \mathcal{D} \end{cases}$$

où  $\chi = [\mathbf{p}, \mathbf{v}, \mathbf{q}, \boldsymbol{\omega}, l]$  est l'état complet de la boucle fermée et  $\mathcal{C}$  et  $\mathcal{D}$  sont, respectivement, les ensembles continus et discrets, définis par

$$\begin{aligned} \mathcal{C} &:= \mathcal{C}_0 \cup \mathcal{C}_1, \quad \mathcal{D} := \mathcal{D}_0 \cup \mathcal{D}_1, \\ \mathcal{C}_0 &:= \{\chi \in \mathbb{R}^{14} : V(\tilde{\mathbf{x}}) \leq \bar{v} \text{ and } \ell = 0\} \\ \mathcal{C}_1 &:= \{\chi \in \mathbb{R}^{14} : V(\tilde{\mathbf{x}}) \geq \underline{v} \text{ and } \ell = 1\} \\ \mathcal{D}_0 &:= \{\chi \in \mathbb{R}^{14} : V(\tilde{\mathbf{x}}) \geq \bar{v} \text{ and } \ell = 0\} \\ \mathcal{D}_1 &:= \{\chi \in \mathbb{R}^{14} : V(\tilde{\mathbf{x}}) \leq \underline{v} \text{ and } \ell = 1\} \end{aligned}$$

où  $V(\tilde{\mathbf{x}}) := \tilde{\mathbf{x}}^\top S \tilde{\mathbf{x}}$  a été défini dans la section précédente,  $\bar{v} = 400$  a été déterminé pour satisfaire (3.3) et  $\underline{v}$  est toute constante positive satisfaisant  $\underline{v} < \bar{v}$  (un choix plus grand de  $\underline{v}$  augmente la marge d'hystérésis, mais retarde le changement de loi de commande). Dans notre cas, nous choisissons  $\underline{v} = 350$ .

Le résultat suivant est une conséquence immédiate des résultats de [Goebel 2012,

Ex. 1.7] et des propriétés de nos modèles linéaires et non linéaires.

**Proposition 1.** *Avec l'action du bouclage hybride (3.4), la boucle fermée présente, quand elle utilise le contrôleur linéaire (3.2), le même bassin d'attraction que celui associé au contrôleur non linéaire (3.1).*

Nous avons réalisé plusieurs simulations de la boucle fermée à l'aide de la *toolbox* Matlab [Sanfelice 2013]. Les simulations sont effectuées avec le modèle complet du drone (2.1), comprenant tous les effets aérodynamiques non linéaires. Un exemple de simulation est présenté dans la Figure 3.4, où nous initialisons le drone à l'origine avec une orientation nulle, sauf pour l'angle de tangage fixé à 45 degrés. L'orientation de la cible est en configuration de vol stationnaire vertical et la position de la cible est assignée à  $\mathbf{p}_{\text{eq}} = [50, 25, 12.5]$ .

Nous observons que sur la période  $t$  dans  $[0, 38]$ , le drone présente une convergence élégante mais lente vers la position cible souhaitée, en utilisant le contrôleur global ( $\ell = 1$ ). Dès lors, l'état discret  $\ell$  entre dans l'ensemble  $\mathcal{D}_1$  et le contrôleur local plus agressif est activé jusqu'à la convergence vers l'équilibre souhaité.

Pour obtenir des simulations réalistes, les mesures sont affectées par le bruit des capteurs. La robustesse intrinsèque de la rétroaction hybride, établie dans le [Goebel 2012, Chapitre 7], est confirmée par le maintien des performances, malgré le bruit de mesure.

### 3.5 Conclusion du Chapitre 3

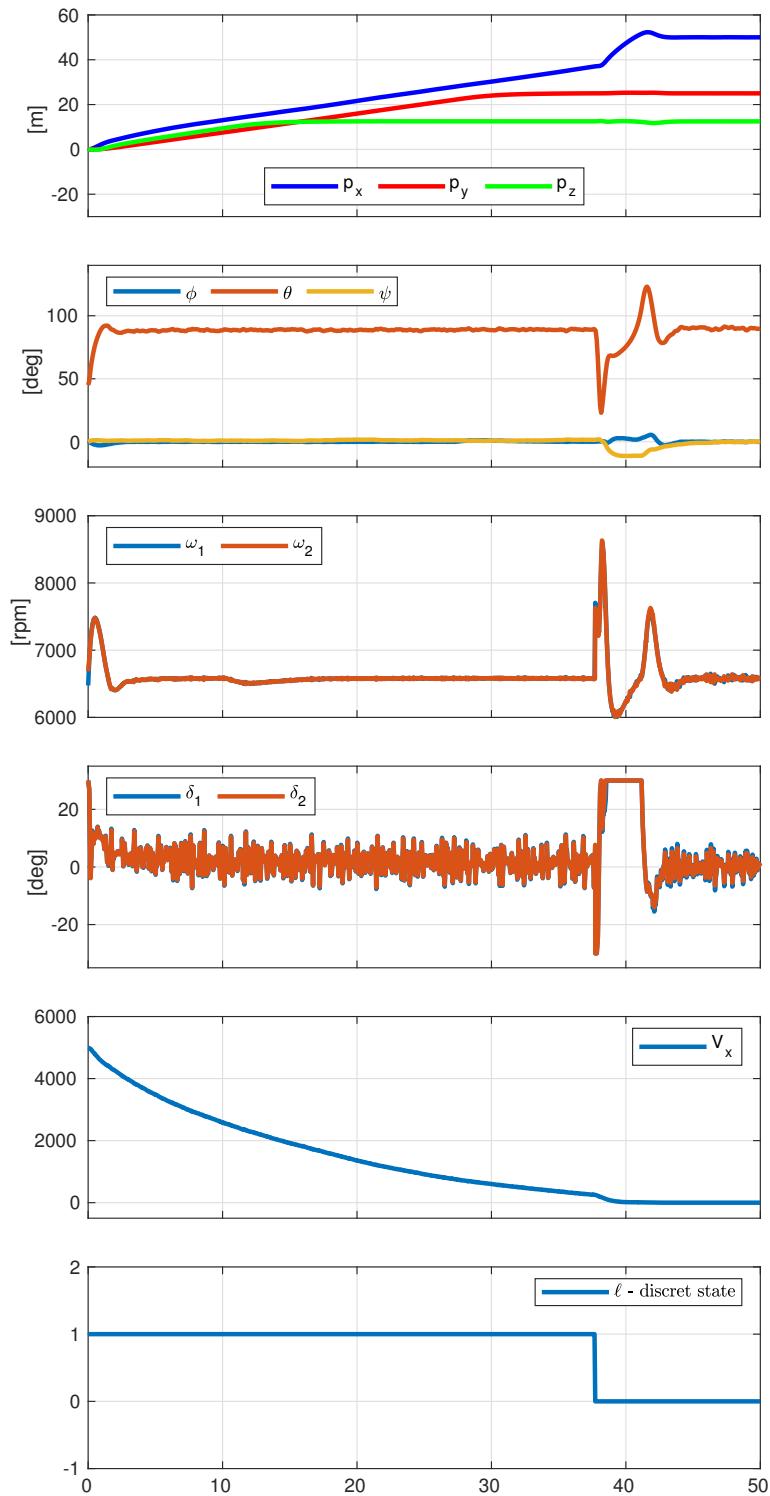


FIGURE 3.4 – Simulation en boucle fermée avec le contrôleur hybride (3.4).



## CHAPITRE 4

# Commande longitudinale d'une maquette de DarkO

---

## Sommaire

---

4.1	Motivation	57
4.2	Présentation de la maquette expérimentale	58
4.2.1	Description physique, capteurs et actionnements	58
4.2.2	Simulation des mouvements du drone	61
4.3	Contrôle linéaire, proportionnel-intégral à 3 DOF	62
4.3.1	Description du schéma de contrôle	62
4.3.2	Optimisation $H_\infty$	64
4.4	Résultats	65
4.5	Conclusion du Chapitre 4	67

---

## 4.1 Motivation

Des simulations en boucle fermée avec le contrôleur (3.4) développé dans 3.4 montrent qu'en présence d'un vent horizontal constant dans le plan  $(x_{[b]}, z_{[b]})$ , le drone modifie son angle de tangage. Ce comportement a également été observé lors d'essais en soufflerie avec le dispositif expérimental [Olszanecki Barth 2020]. Intuitivement, une réduction de l'angle d'attaque entraîne une diminution de la surface exposée au vent, de manière à réduire la force de traînée, ce qui a une forte incidence sur la position. Dans le même temps, le flux d'air dû au vent constant génère une portance, compensée par une réduction de la poussée de l'hélice et une réduction conséquente de la consommation du drone. L'objectif de la maquette décrite ici est d'évaluer expérimentalement l'effet du vent sur le dispositif DarkO.

Explication de la  
non mesure du ve

## 4.2 Présentation de la maquette expérimentale

### 4.2.1 Description physique, capteurs et actionnements

Le prototype développé comprend des pièces imprimées en 3D, en Onyx et PLA (acide polylactique, un polyester thermoplastique). Le drone est spécialement conçu pour réaliser des expériences devant une soufflerie avec un comportement semblable à celui de DarkO en raison de leur forme similaire (voir Fig. 4.2). La partie centrale, qui contient l'avionique embarquée (pilote automatique, GPS, etc.) dans DarkO, a été remplacée ici par un joint tournant à un degré de liberté (voir Fig. 4.1). Les ailes sont les mêmes que celles du DarkO, avec les contrôleurs électroniques de vitesse (ESC), régissant la vitesse du moteur *brushless*, placés dans les ailes.

Comme décrit dans la section 4.1, nous souhaitons représenter et étudier le degré de liberté de l'axe  $y_b$  du drone DarkO. Le tube principal en carbone reliant les deux ailes est utilisé comme axe de rotation. Ce tube est fixé sur deux roulements espacés de 28.5 mm afin d'obtenir une fixation solide de l'ensemble. Cet axe de rotation est équipé d'un codeur optique rotatif en quadrature pour mesurer précisément l'orientation de l'appareil. L'avantage de ce capteur est qu'il ne produit pas de couple résistant sur l'axe de rotation. Ce codeur offre 4000 impulsions par tour, ce qui donne une résolution de  $0.09^\circ/pulse$ .

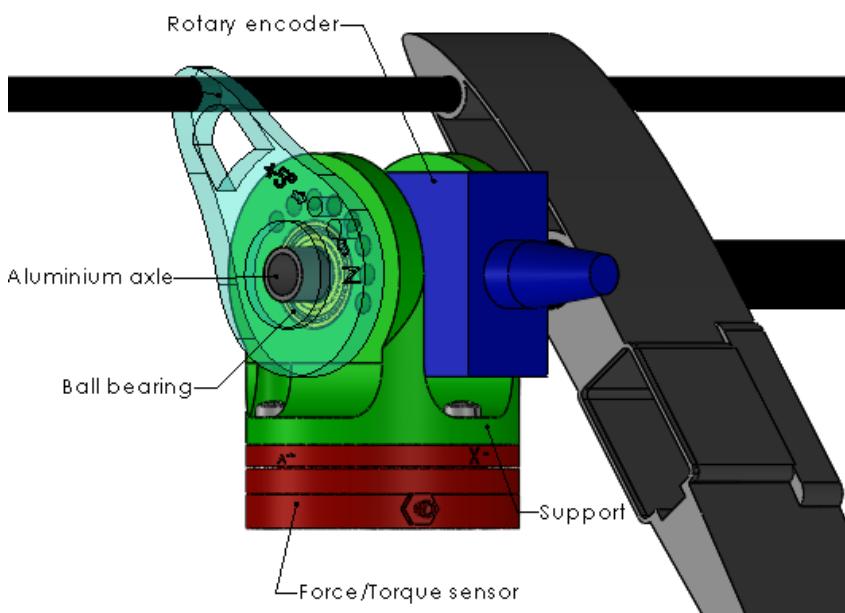


FIGURE 4.1 – Montage à un degré de liberté.

Comme le montre la figure 4.1, l'indexer et le support sont percés de manière à ce que la rotation puisse être bloquée à des positions connues ( $0^\circ$ ,  $90^\circ$ , etc.) par une vis sur l'indexer qui s'insère dans les trous du support. Le verrouillage de l'appareil permet une initialisation correcte de l'encodeur incrémental. Le verrouillage permet également de placer l'appareil dans des positions exactes spécifiques afin d'identifier les coefficients aérodynamiques.

Le mécanisme est également équipé d'un capteur de forces et de moments à 6 degrés de liberté (DOF), qui permet de mesurer la force exercée sur le dispositif expérimental par le support. Le banc d'essai expérimental est également équipé d'un fil chaud pour mesurer la vitesse de l'air affectant la maquette.

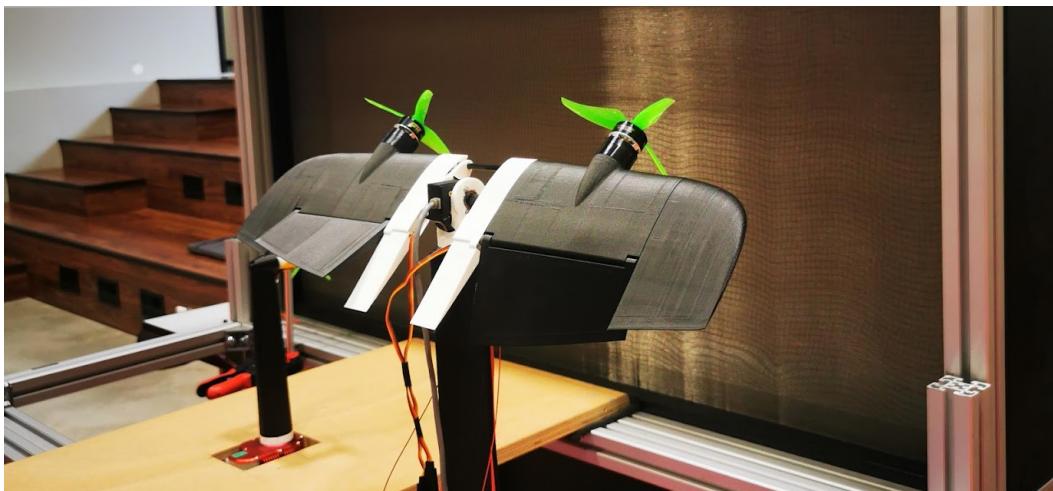


FIGURE 4.2 – Modèle de DarkO à un seul degré de liberté devant le WindShape.

La photo de la figure 4.2 montre le dispositif expérimental dans son environnement de test. Le drone est placé devant une soufflerie ouverte, appelée WindShape, qui génère un vent horizontal compris entre  $2$  et  $16 \text{ m s}^{-1}$ . Ainsi, lors de nos tests, nous considérons que la composante verticale du vent est nulle. Le drone est placé au centre du WindShape, dans la zone d'écoulement la plus laminaire, tandis que le capteur à fil chaud est placé aussi près que possible du drone.

La géométrie du dispositif expérimental permet de placer les câbles d'alimentation et de signal près du centre de rotation afin de minimiser leurs effets de friction sur la structure. Malgré cela, le système de rotation interfère inévitablement avec le drone, en créant des forces parasites, notamment de la traînée. La surface projetée de l'articulation étant faible par rapport à la surface de l'aile, la traînée générée par ce support est faible par rapport à la traînée de l'aile et des hélices, et peut donc être négligée.

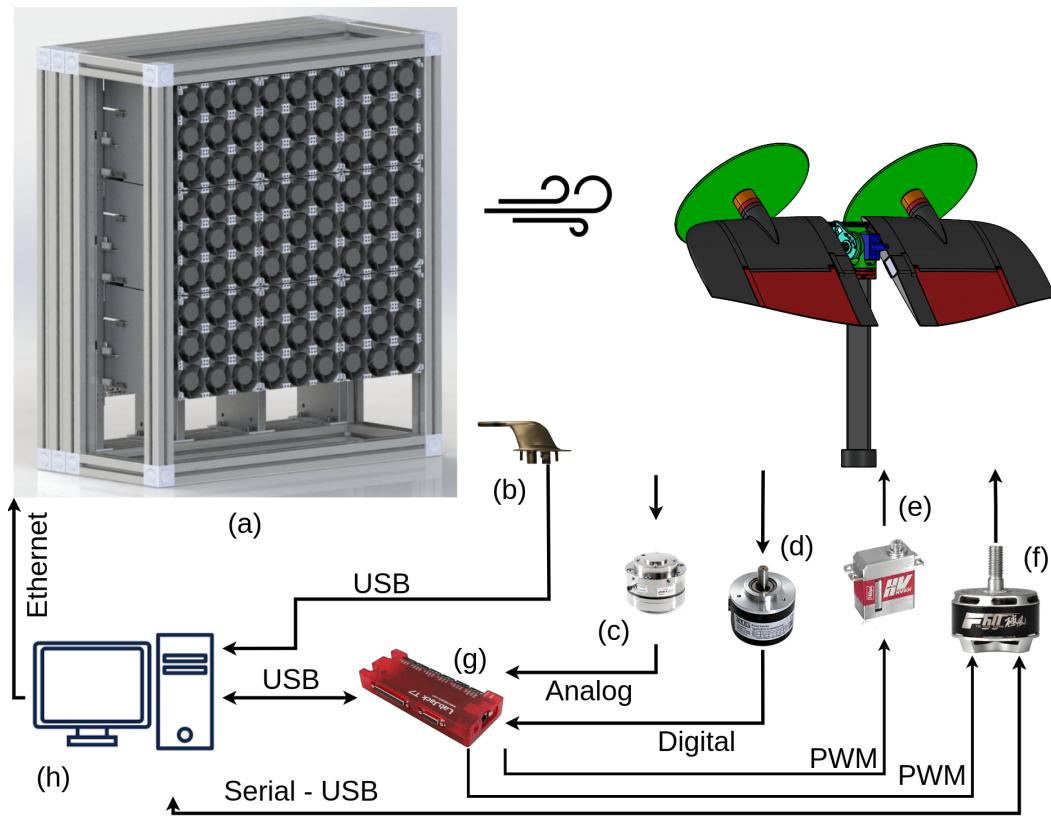


FIGURE 4.3 – Architecture d'essai en vol virtuel : WindShape (a) ; capteur de vitesses (b) ; capteur de forces et moments (c) ; encodeur rotatif (d) ; servomoteur (e) ; moteur *brushless* + ESC (f) ; LabJack (g) ; ordinateur de contrôle (h)

Un diagramme schématique des composants du dispositif expérimental et de leur interconnexion est présenté à la Fig. 4.3.

Les moteurs (f) sont alimentés par une batterie externe de 12v 20Ah et les servomoteurs (e) sont alimentés par du 5 V, via un module d'acquisition LabJack T7 [LabJack 2001, ] (g). Le module LabJack (g) concentre la plupart des signaux des capteurs et des actionneurs : six entrées analogiques pour le capteur de force/couple (c), deux entrées numériques en quadrature pour l'encodeur rotatif (d), une entrée analogique (ou une liaison série selon le capteur) pour le capteur de vitesse (b), deux sorties numériques PWM (Pulse Width Modulation) pour les moteurs (f) et deux sorties numériques PWM pour les servomoteurs (e).

Les elevons sont commandés par des servomoteurs qui ne fournissent pas de signal de mesure de la position. Nous utilisons donc le point de consigne, en supposant que l'actionneur est parfait. Cela est raisonnable en raison de la saturation logicielle

imposée à l'entrée des élevons et du dimensionnement correct des servomoteurs par rapport aux forces impliquées. Le LabJack (g) possède une interface de programmation d'application (API), permettant une connexion avec un ordinateur. Nous avons développé un code Python qui communique avec le LabJack afin de récupérer les valeurs des capteurs, de calculer la commande à appliquer aux servomoteurs selon le schéma de contrôle présenté ci-dessous et de générer les signaux de sortie pour les servomoteurs. Les données collectées par le LabJack sont enregistrées afin d'être utilisées pour le post-traitement et de générer le graphique présenté dans la section 4.4.

Pour générer le vent, nous utilisons un dispositif WindShape, lequel dispose également d'une API lui permettant d'être contrôlé via un réseau Ethernet. Le code Python développé peut assigner la vitesse du vent générée par le WindShape et donc agir sur le modèle. Il est ainsi possible de tester un ensemble de configurations de vols stationnaires et leurs transitoires associés dans la même campagne d'essais, sans aucune action sur le modèle.

### 4.2.2 Simulation des mouvements du drone

Le prototype étant relié à un support fixe, il n'est pas possible de reproduire expérimentalement le mouvement de translation. Nous avons donc inclus une simulation logicielle du mouvement en intégrant les mesures de force disponibles au niveau de la fixation. La vitesse de translation (respectivement la position) du drone est obtenue par intégration simple (respectivement double) des données mesurées par le capteur de force. Par souci de simplicité, nous négligeons l'influence aérodynamique de la vitesse (simulée) sur l'aile. En particulier, à partir des équations (2.1a) et (2.1b), nous obtenons le modèle simplifié suivant :

$$\dot{\mathbf{v}} = \mathbf{g} + \frac{1}{m} \left( R(\mathbf{q})(F\mathbf{u} + D_f(\delta)R^\top(\mathbf{q})\|\mathbf{w}\|\mathbf{w}) \right) \quad (4.1a)$$

$$= \mathbf{g} + \frac{1}{m} \mathbf{F}_{meas}, \quad (4.1b)$$

où  $\mathbf{F}_{meas}$  représente les forces mesurées par le capteur dans le repère inertiel corrigé du biais. Pour calibrer la correction du biais, lors de l'initialisation, les forces mesurées sont moyennées sur 6000 échantillons, le modèle étant bloqué dans une position stable (angle de tangage à  $0^\circ$ , c'est-à-dire orientation verticale). Pour éliminer le biais de la force mesurée à chaque mesure, nous soustrayons l'effet de la gravité sur le modèle de la mesure. Une masse artificielle  $m$  est attribuée à la dynamique du logiciel dans la boucle conformément à (4.1b), ce qui permet de tester

plusieurs configurations afin de mieux apprécier l'influence de la masse du drone sur d'éventuels événements transitoires de saturation. Cela permet d'étudier des scénarios impliquant la masse non négligeable de la batterie, qui n'est pas présente dans notre modèle. Bien que cette manipulation soit aisée, elle ne représente pas parfaitement la réalité car nous ne tenons pas compte de la répartition des masses dans le drone et donc des modifications de l'inertie. La vitesse et la position transitoire du drone sont ensuite obtenues par intégration numérique simple et double de l'accélération comme dans (4.1), en utilisant une intégration numérique trapézoïdale.

### 4.3 Contrôle linéaire, proportionnel-intégral à 3 DOF

Dans la section 3.4, nous avons proposé un bouclage proportionnel stabilisant une position de vol stationnaire en l'absence de vent (sans perturbation). Nous proposons ici une extension incluant une action intégrale, adaptée au fonctionnement avec une perturbation non mesurée représentée par un vent constant. L'objectif est de stabiliser le drone à la position de référence, en rejetant une perturbation de vent constant inconnue.

#### 4.3.1 Description du schéma de contrôle

Nous expérimentons la situation avec le vent agissant uniquement le long de l'axe  $x_i$ , avec le drone orienté vers le vent, c'est-à-dire avec des angles de roulis et de lacet nuls. Dans cette configuration, le vent n'agit que sur la vitesse linéaire le long des axes  $x_b$  et  $z_b$ , et ne génère qu'un moment autour de l'axe  $y_b$ . Un examen attentif de la commande et des matrices d'entrée des perturbations  $F$ ,  $M$  dans (2.15) suggère une architecture de commande efficace pour rejeter une perturbation constante. En effet, les élévons et les hélices peuvent être utilisés symétriquement pour générer respectivement un moment autour de l'axe  $y_b$  et une force le long de l'axe  $x_b$ , compensant ainsi l'effet de la perturbation. Néanmoins, il reste une force le long de l'axe  $z_b$  à compenser et une action intégrale peut converger asymptotiquement vers la force désirée, même avec une perturbation du vent non mesurée  $w$ . Nous pouvons ainsi stabiliser le drone à une position de vol stationnaire, différente de l'équilibre sans vent. La solution de contrôle exploite le degré de liberté de l'angle de tangage pour compenser l'effet du vent.

améliorer les équa-  
ns

Le contrôleur proposé, représenté sur la figure 4.4, correspond à

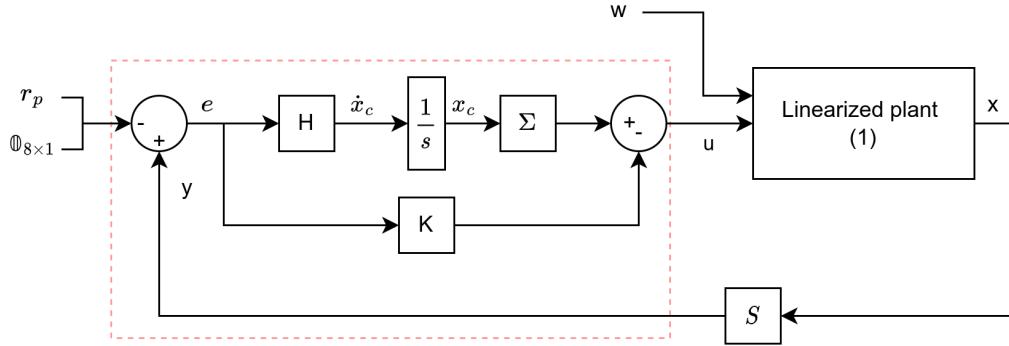


FIGURE 4.4 – Schéma de commande linéaire, proportionnel-intégral.

$$\dot{x}_c = H(y - \begin{bmatrix} r_p \\ 0_{8 \times 1} \end{bmatrix}), \quad (4.2)$$

$$y = Sx, \quad (4.3)$$

$$u = \Sigma x_c + K(y - \begin{bmatrix} r_p \\ 0_{8 \times 1} \end{bmatrix}), \quad (4.4)$$

$$S = \begin{bmatrix} I_7 & 0_{7 \times 5} \\ 0_{4 \times 8} & I_4 \end{bmatrix}, \quad (4.5)$$

$$\Sigma = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}^\top, \quad (4.6)$$

où  $x_c \in \mathbb{R}^2$  est l'état de l'intégrateur ;  $r_p \in \mathbb{R}^3$  est la référence constante comprenant une position cible pour le mouvement de translation ;  $S$  est une matrice de sélection de sortie, qui supprime la composante de l'angle de tangage de la sortie mesurée (n'affectant que la linéarisation par quaternion) pour former  $y$  ;  $\Sigma$  est une matrice d'allocation d'entrée qui permet d'affecter la première composante de l'état de l'intégrateur à la commande du moteur et la seconde composante à la commande de la gouverne de profondeur.  $K$ ,  $H$  sont des gains constants à sélectionner pour que la matrice linéaire de la boucle fermée  $A_{cl}$  caractérisant la boucle fermée linéaire soit Hurwitz, afin d'assurer la stabilisation avec la dynamique linéarisée liée au scénario sans vent (2.14).

$$A_{cl} = \begin{bmatrix} A & 0_{12 \times 2} \\ HS & 0_{2 \times 2} \end{bmatrix} - \begin{bmatrix} G \\ 0_{2 \times 4} \end{bmatrix} \left( K \begin{bmatrix} S & 0_{11 \times 2} \end{bmatrix} - \begin{bmatrix} 0_{4 \times 12} & \Sigma \end{bmatrix} \right), \quad (4.7)$$

De manière synthétique, la matrice (4.7) décrit la boucle fermée illustrée à la Fig. 4.4 : un retour de sortie avec 11 sorties, comprenant les trois positions, les trois vitesses linéaires, deux des trois angles ( $\epsilon_1$  et  $\epsilon_3$ ) et les trois vitesses angulaires. Cette structure peut être considérée comme une solution proportionnelle-intégrale MIMO résultant d'une observation attentive de la dynamique linéarisée du drone, ce qui permet un nombre minimal d'intégrateurs intégrés dans le contrôleur. Ce contrôle devrait permettre de rejeter les perturbations constantes tout en ayant une robustesse satisfaisante. Le gain  $K$  correspond au terme proportionnel et le gain  $H$  pondère le terme intégral, induisant une convergence vers la cible. La matrice d'allocation  $\Sigma$  conduit à une utilisation symétrique des hélices et des ailerons. Il faut alors ajuster  $K$  et  $H$  pour obtenir un compromis satisfaisant entre robustesse et rejet des perturbations. Nous mettons en œuvre une synthèse multiobjectifs basée sur une méthode d'optimisation  $H_\infty$ , décrite ci-après.

difier la com-  
nde pour expli-  
er la commande à  
DOF

### 4.3.2 Optimisation $H_\infty$

Pour effectuer une sélection robuste de  $K$  et  $H$ , nous caractérisons d'abord plusieurs fonctions de transfert dans la figure 4.4.

La sortie de mesure  $y$  est utilisée pour la rétroaction, l'entrée  $u$  est la somme de l'entrée intégrale  $\Sigma x_c$  et de l'action proportionnelle  $Ke$ . La sortie  $z$  correspond aux signaux de performance de sortie à contrôler ( $e, w, u, y, r_p$ ). Grâce aux fonctions de pondération  $W = \text{diag}(W_1, \dots, W_4)$ , la conception de  $H$  et  $K$  vise à rejeter une perturbation ou un échelon à basse fréquence  $w$  agissant sur  $y$ . L'objectif de la conception est d'amener  $y$  à zéro, malgré la perturbation à basse fréquence sur  $w$ .

Plus précisément, les constantes de pondération sont réglées comme suit

$$W_1 = 0.5, \quad W_2 = 0.5, \quad W_3 = 0.8, \quad W_4 = 0.5. \quad (4.8)$$

$$\min_C \begin{array}{l} \left\| W_1 T_{r \rightarrow \epsilon}(P, C) \right\| \\ \left\| W_2 T_{d \rightarrow u}(P, C) \right\| \\ \left\| W_3 T_{r \rightarrow u}(P, C) \right\| \\ \left\| W_4 T_{w \rightarrow y}(P, C) \right\|_\infty \end{array}, \text{ sous condition que} \\ C \in \mathbb{R}^{11 \times 4} \text{ stabilise } P \text{ en interne,} \quad (4.9)$$

Nous avons résolu (4.9) en utilisant le logiciel **Systune** [Apkarian 2006]. Basé sur l'optimisation non lisse, **Systune** traite plusieurs scénarios non convexes, tels que l'architecture de contrôle structurée où nous optimisons les matrices de gain  $K$ ,  $H$ . L'algorithme d'optimisation renvoie la sélection optimisée de suivante :

$$\begin{bmatrix} H \\ K \end{bmatrix} = \begin{bmatrix} -1.902 & 0 & 7.201 & -9.043 & 0 & 33.244 & 0 & 0 & 0 & 4.696 & 0 \\ 0.425 & 0 & -1.620 & 2.024 & 0 & -7.480 & 0 & 0 & 0 & -1.045 & 0 \\ 0.035 & 0 & -0.728 & -1.853 & 0 & -4.445 & 0 & 0 & 0 & -0.323 & 0 \\ 0.035 & 0 & -0.728 & -1.853 & 0 & -4.445 & 0 & 0 & 0 & -0.323 & 0 \\ 0.217 & 0 & -0.164 & 1.074 & 0 & -0.527 & 0 & 0 & 0 & -0.773 & 0 \\ 0.217 & 0 & -0.164 & 1.074 & 0 & -0.527 & 0 & 0 & 0 & -0.773 & 0 \end{bmatrix}. \quad (4.10)$$

Nous obtenons une abscisse spectrale en boucle fermée pour  $A_{cl}$  dans (4.7) valant  $\alpha = -0.2381$ .

## 4.4 Résultats

La figure 4.5 montre le résultat de la boucle fermée de la figure 4.4 avec la sélection des gains (4.10) et une valeur constante et croissante par morceaux du vent horizontal  $w$  (dernier graphique). Malgré quelques oscillations, le drone maintient sa position en dépit du vent, en inclinant convenablement l'angle de tangage. Les oscillations expérimentales sont absentes de nos simulations, ce qui suggère la présence de phénomènes non modélisés. Nous observons également un comportement qui pourrait être important pour de futures recherches : le drone semble se stabiliser plus facilement le long de l'axe vertical que le long de l'axe horizontal.

Comme prévu, lorsque le vent augmente, l'angle d'inclinaison diminue, ce qui modifie la poussée nécessaire et la déflexion des élévons. En effet, le vent génère de la portance sur les ailes, ce qui compense l'effet de la gravité, et donc la poussée nécessaire devient plus faible. Pour chaque valeur de  $w$ , le modèle converge vers un équilibre, dont la caractérisation mathématique précise est détaillée dans 2.3.2. Il est donc nécessaire d'étendre la robustesse du contrôleur en effectuant une optimisation multimodèle du contrôleur (décrise dans 5.2.3). Il est également possible de supprimer les contraintes sur la structure du contrôleur afin de lui donner plus de

degrés de liberté dans l'optimisation.

La vidéo de la maquette avec les résultats expérimentaux est disponible via le lien [https://youtu.be/ce4\\_FUzeVzI](https://youtu.be/ce4_FUzeVzI).

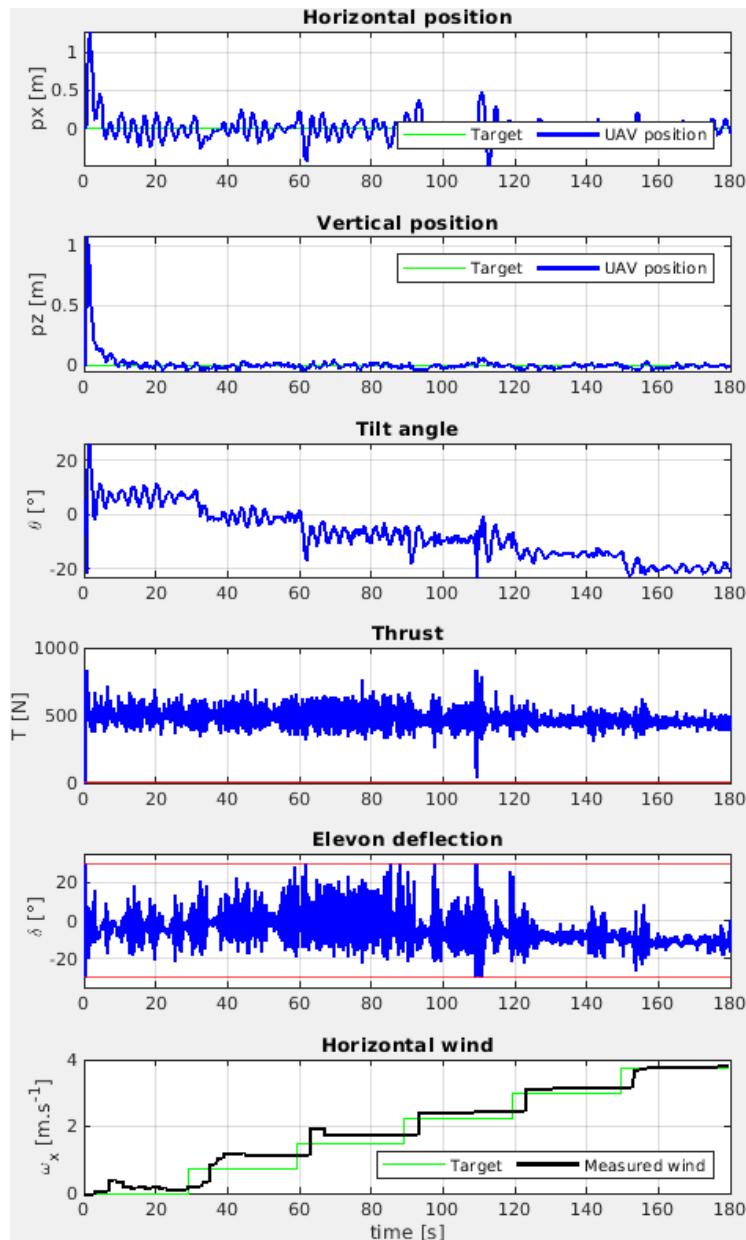


FIGURE 4.5 – Résultats expérimentaux.

## 4.5 Conclusion du Chapitre 4

Nous avons décrit une maquette permettant de simuler le vol d'un drone convertible, DarkO, en soufflerie. L'objectif de cette maquette est de tester le système de contrôle sur une représentation fidèle de la dynamique longitudinale, tout en simulant la dynamique de translation. Nous avons également présenté un contrôleur linéaire de sortie basé sur une architecture proportionnelle-intégrale pour la stabilisation du vol stationnaire dans des conditions de vent constant. Les gains du contrôleur ont été obtenus à l'aide d'une optimisation non convexe. Les résultats des essais expérimentaux montrent qu'il est possible de stabiliser l'équilibre en vol stationnaire dans la plage de vitesse du vent testée. Toutefois, d'autres architectures de contrôle devraient être étudiées à l'avenir pour traiter les oscillations indésirables. Ces résultats indiquent qu'il est nécessaire de tester cette architecture de commande sur un modèle complet à six degrés de liberté, dans des conditions similaires de vent.



## CHAPITRE 5

# Commande proportionnelle intégrale de DarkO

---

## Sommaire

<b>5.1</b>	<b>Motivation</b>	<b>69</b>
<b>5.2</b>	<b>Schéma de commande linéaire proportionnelle intégrale</b>	<b>69</b>
5.2.1	Description du schéma de contrôle	69
5.2.2	Contrôleur optimisé sous contraintes $H_\infty$ , cas sans vent	73
5.2.3	Contrôleur optimisé sous contrainte $H_\infty$ , cas multimodèles	75
<b>5.3</b>	<b>Vol expérimental en soufflerie ouverte</b>	<b>81</b>
<b>5.4</b>	<b>Conclusion du Chapitre 5</b>	<b>84</b>

---

## 5.1 Motivation

Nous avons observé expérimentalement dans le chapitre 4 qu'un drone *tailsitter* pouvait se stabiliser en stationnaire face à du vent, vers les points d'équilibre décrits dans la section 2.3.2, grâce à une architecture de commande linaire. Cette dernière est basée sur un retour de sortie, proportionnel intégral, libérant deux degrés de liberté sur l'orientation du drone

## 5.2 Schéma de commande linéaire proportionnelle intégrale

### 5.2.1 Description du schéma de contrôle

Une inspection minutieuse des matrices de commande et d'entrée des perturbations  $\mathbf{G}_w$  et  $\mathbf{E}_w$  du modèle (2.32) (voir la sortie de l'Algorithm 2) suggère une architecture de contrôle efficace pour rejeter une perturbation de vent constante  $w$ . En effet, les ailerons et les hélices peuvent être utilisés symétriquement pour

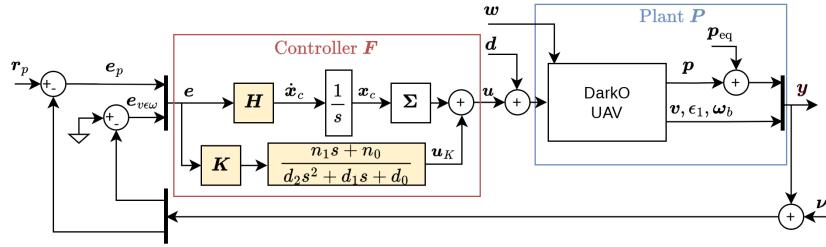


FIGURE 5.1 – Schéma de commande intégrale avec la perturbation de vent  $w$ , la perturbation du système à l'entrée  $d$  et à la sortie  $v$ .

générer respectivement un moment autour de l'axe  $y_{[b]}$ , vérifiant l'équation (2.27), et une force le long de l'axe  $x_{[b]}$ , vérifiant l'équation (2.25), compensant ainsi l'effet de la perturbation. Néanmoins, il reste une force le long de l'axe  $z_{[b]}$  à compenser en vérifiant l'équation (2.26). Une action intégrale peut converger asymptotiquement vers la force désirée, même avec une perturbation du vent non mesurée ( $w$ ). Nous pouvons donc stabiliser le drone à l'équilibre en vol stationnaire tel que caractérisé dans le Théorème 1. Comme nous ne mesurons pas le vent  $w$ , les valeurs de  $\psi$  et  $\theta$  dans l'Algorithme 1 sont inconnues. Le contrôleur proposé, illustré à la Fig. 5.1, utilise l'action intégrale pour faire converger ces angles vers leur valeur d'équilibre. Le bouclage utilise les variables d'erreur suivantes :

$$\mathbf{e}_p = \mathbf{r}_p - \mathbf{p}, \quad \mathbf{e}_{v\epsilon\omega} = - \begin{bmatrix} \mathbb{I}_3 & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 2} & \mathbf{0}_3 \\ \mathbf{0}_{1 \times 3} & 1 & \mathbf{0}_{1 \times 2} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_3 & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 2} & \mathbb{I}_3 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{v}} \\ \tilde{\boldsymbol{\epsilon}} \\ \tilde{\boldsymbol{\omega}}_b \end{bmatrix}. \quad (5.1)$$

Elles doivent converger vers zéro dans le cas de vol stationnaire et pour n'importe quelle position de référence constante  $\mathbf{r}_p \in \mathbb{R}^3$ . Notons que  $\mathbf{r}_p$  est l'entrée de référence du schéma de contrôle.

Les variables d'erreur dans (5.1) peuvent être représentées comme dans le schéma de la Fig. 5.1 en définissant la sortie  $\mathbf{y} \in \mathbb{R}^{10}$  de la dynamique du système linéarisée (2.32), ayant le vecteur d'état incrémental  $\tilde{\mathbf{x}} \in \mathbb{R}^{10 \times 1}$  défini ci-dessous :

$$\mathbf{y} = \mathbf{C} \tilde{\mathbf{x}} + \begin{bmatrix} \mathbf{p}_{eq} \\ \mathbf{0}_{7 \times 1} \end{bmatrix}, \quad \mathbf{C} := \begin{bmatrix} \mathbb{I}_6 & \mathbf{0}_{6 \times 1} & \mathbf{0}_{6 \times 2} & \mathbf{0}_{6 \times 3} \\ \mathbf{0}_{1 \times 6} & 1 & \mathbf{0}_{1 \times 2} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 6} & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 2} & \mathbb{I}_3 \end{bmatrix}, \quad (5.2)$$

où la matrice de sortie  $\mathbf{C} \in \mathbb{R}^{10 \times 12}$  enlève la composante  $\tilde{\boldsymbol{\epsilon}}_2$  et  $\tilde{\boldsymbol{\epsilon}}_3$  du vecteur d'état  $\tilde{\mathbf{x}}$ .

Comme le montre la Figure 5.1, les équations de la dynamique du contrôleur sont basées sur l'erreur  $\mathbf{e}$  suivante :

$$\mathbf{e} = [\mathbf{e}_p^\top \ \mathbf{e}_{v\omega}^\top]^\top, \quad \dot{\mathbf{x}}_c = \mathbf{H}\mathbf{e}, \quad \mathbf{u} = \Sigma \mathbf{x}_c + \mathbf{u}_K, \quad (5.3a)$$

$$\Sigma := \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}^\top, \quad \mathbf{u}_K = \frac{n_1 s + n_0}{d_2 s^2 + d_1 s + d_0} \mathbf{K} \mathbf{e}, \quad (5.3b)$$

où  $\mathbf{x}_c \in \mathbb{R}^2$  est l'état intégral,  $\Sigma$  est une matrice d'allocation des entrées qui permet d'affecter la première composante de l'état de l'intégrateur à l'action des hélices et la deuxième composante à l'action des elevons. Les scalaires  $n_1$ ,  $n_0$ ,  $d_2$ ,  $d_1$ ,  $d_0$  sont respectivement les coefficients du numérateur et du dénominateur d'un filtre utilisé pour éviter une transmission directe entrée-sortie qui amplifierait le bruit de mesure à haute fréquence. Ce filtre induit un contrôleur strictement propre, pour une robustesse accrue aux incertitudes additives. Nous définissons le contrôleur  $\mathbf{F}$  ayant pour dimensions  $4 \times 10$  avec la matrice de transfert  $\mathbf{F}(s) = T_{\mathbf{e} \rightarrow \mathbf{u}}(s)$  telle que décrite dans (5.3) et l'interconnexion détaillée dans la Figure 5.1. Le système  $\mathbf{P}$  de dimensions  $10 \times 4$  représente la dynamique linéarisée de DarkO. La sortie du système  $\mathbf{y} \in \mathbb{R}^{10 \times 1}$  est utilisée comme entrée du contrôleur  $\mathbf{F}$ .

Compte tenu des symétries des actionneurs du drone, nous avons contraint la structure de la matrice  $\mathbf{K}$  dans (5.3), associée à l'action proportionnelle du contrôleur, afin d'utiliser les actionneurs selon leur action physique. Ainsi,  $\mathbf{K}$  prend la forme :

$$\mathbf{K}_{\text{struct}} = \begin{bmatrix} k_1 & -k_2 & k_3 & k_4 & -k_5 & k_6 & -k_7 & k_8 & k_9 & -k_{10} \\ k_1 & k_2 & k_3 & k_4 & k_5 & k_6 & k_7 & -k_8 & -k_9 & k_{10} \\ -k_{11} & -k_{12} & k_{13} & -k_{14} & -k_{15} & -k_{16} & k_{17} & -k_{18} & k_{19} & -k_{20} \\ -k_{11} & k_{12} & k_{13} & -k_{14} & k_{15} & k_{16} & -k_{17} & k_{18} & k_{19} & k_{20} \end{bmatrix}. \quad (5.4)$$

La structure précédente s'explique par le comportement physique du drone, une erreur de position sur l'axe  $z_{[i]}$  du repère inertiel NED (voir Fig. 2.2) entraîne une utilisation symétrique des deux hélices, ce qui génère une force le long de l'axe  $x_{[b]}$  du drone. L'utilisation symétrique des deux moteurs se traduit par le même signe dans les coefficients  $k_3$  et  $k_6$  des colonnes 3 et 6 de  $\mathbf{K}$ , qui corresponde respectivement aux erreurs de position et de vitesse sur l'axe  $z_{[i]}$ . De même, une erreur de position ou de vitesse le long de l'axe latéral du drone  $y_{[b]}$  sera compensée par une utilisation antisymétrique des moteurs, comme le montrent les coefficients  $k_2$  et  $k_5$  et leurs signes opposés dans les colonnes 2 et 5 de  $\mathbf{K}$ . Une erreur de vitesse angulaire autour de l'axe  $x_{[b]}$  doit être compensée par une utilisation antisymétrique des elevons, comme le montre le coefficient  $k_{18}$  de signe opposé dans la colonne 8 de  $\mathbf{K}$ . Des arguments parallèles expliquent les coefficients restants de la matrice  $\mathbf{K}$  dans (5.4). Toutes ces explications ne sont valables que dans un voisinage de l'équilibre,

où le drone se trouve être à la verticale. On comprend aisément que dans d'autres configurations, ces contraintes d'actionnement ne sont plus valides. Un avantage de la structure de (5.4) est la réduction du nombre de variables à optimiser, de 40 à 20 gains scalaires, cela se traduisant par une diminution du temps nécessaire à l'optimisation.

La boucle fermée illustrée à la Figure 5.1 est un retour de sortie à 10 éléments, comprenant les trois positions, les trois vitesses linéaires, l'un des trois angles d'attitude ( $\epsilon_1$ ) et les trois vitesses angulaires. Cette structure peut être considérée comme un bouclage proportionnel-intégral MIMO. Les paramètres à régler dans le contrôleur  $\mathbf{F}$  (5.3) sont le gain proportionnel  $\mathbf{K} \in \mathbb{R}^{4 \times 10}$  dans (5.4), le gain intégral  $\mathbf{H} \in \mathbb{R}^{2 \times 10}$  et les paramètres du filtre  $n_1, n_0, d_2, d_1, d_0$ , comme coloré en jaune sur la Fig. 5.1. Une méthode de réglage appropriée doit garantir une réjection adéquate des perturbations et une robustesse satisfaisante aux dynamiques non modélisées. Ces deux objectifs conduisent à un compromis car le rejet des perturbations nécessite un réglage agressif tandis que les propriétés de robustesse sont assurées par une stratégie d'atténuation des hautes fréquences.

er le paragraphe précédent

Nous examinons ensuite deux méthodes de réglage basées sur l'optimisation. La première est issue des idées proposées dans 4.3, qui ne nécessitait pas la dynamique linéarisée des Théorèmes 1 et 2, et est résumée dans la section 5.2.2. Il s'agit d'une synthèse multiobjectifs avec des contraintes  $H_\infty$  basée sur le modèle de vent nul discuté dans la section 2.3.1 et 2.4.1 et détaillée dans 4.3.

Nous montrerons que cette première méthode ne permet pas de stabiliser le drone dans certaines plages de vent, en raison de la méconnaissance de la dynamique caractérisée par les Théorèmes 1 et 2. La deuxième méthode de réglage, présentée à la section 5.2.3, est une synthèse itérative multiobjectifs avec des contraintes  $H_\infty$ , basée sur un ensemble de modèles associés à différentes conditions de vent et dérivés des Théorèmes 1 et 2, par le biais des Algorithmes 1 et 2.

Dans notre validation numérique, présentée dans les sections 5.2.2 et 5.2.3 (voir en particulier les Fig. 5.2 et Fig. 5.4), un bruit de mesure est ajouté à la sortie pour produire des résultats numériques semblable aux expérimentaux. Les écarts types des niveaux de bruit adoptés sont indiqués dans la Table 5.1.

En plus de présenter les résultats de la simulation du bouclage linéaire de la Fig. 5.1 avec le modèle linéarisé (2.32), dans les sections 5.2.2 et 5.2.3, nous simulons également la boucle fermée en remplaçant le modèle linéarisée  $\mathbf{P}$  par le modèle non linéaire (2.1), comprenant la dynamique réelle du drone.

Lorsque l'on remplace le modèle linéarisé par la dynamique non linéaire (2.1), dont l'état est  $\mathbf{x} = (\mathbf{p}, \mathbf{v}, \mathbf{q}, \boldsymbol{\omega}_b) \in \mathbb{R}^{13}$ , nous remplaçons la sortie linéaire  $\mathbf{y}$  avec la

Measurement	Value	Units
$\mathbf{p}$	$2.5 \times 10^{-4}$	m
$\tilde{\mathbf{v}}$	$1.2 \times 10^{-3}$	$\text{m s}^{-1}$
$\tilde{\epsilon}$	$4.7 \times 10^{-4}$	
$\tilde{\omega}_b$	$2.7 \times 10^{-3}$	$\text{rad s}^{-1}$

TABLE 5.1 – Écart-type du bruit pour la modélisation des capteurs en simulation.

version non linéaire suivante :

$$\mathbf{y}_{\text{NL}} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \epsilon_1 \\ \tilde{\omega}_b \end{bmatrix} = \begin{bmatrix} \mathbb{I}_6 & \mathbf{0}_{6 \times 1} & \mathbf{0}_{6 \times 1} & \mathbf{0}_{6 \times 2} & \mathbf{0}_3 \\ \mathbf{0}_{1 \times 3} & 0 & 1 & \mathbf{0}_{1 \times 2} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_3 & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 2} & \mathbb{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{R}_\psi^\top \mathbf{p} \\ \mathbf{R}_\psi^\top \mathbf{v} \\ \mathbf{q}_{\text{eq}\psi}^{-1} \otimes \mathbf{q} \\ \tilde{\omega}_b \end{bmatrix}. \quad (5.5)$$

Dans les sections suivantes, nous notons la marge de module d'une matrice de transfert  $s \mapsto T_{v \rightarrow z}$  as  $\Delta_m(T_{v \rightarrow z}) = \min_{\omega \in R} \sigma_{\min}(T_{v \rightarrow z}(j\omega))$ .

### 5.2.2 Contrôleur optimisé sous contraintes $H_\infty$ , cas sans vent

Pour régler le contrôleur dans le cas sans vent, nous utilisons le modèle linéaire du système décrit dans la section 2.4.1,  $\mathbf{P}(s) = T_{u \rightarrow y}(s)$ , obtenue à partir des équations (2.29) et (5.2) tel que :

$$\mathbf{P}(s) = \mathbf{C}(s\mathbb{I}_{12} - \mathbf{A}_0)^{-1} \mathbf{G}_0.$$

En lien avec la Figure 5.1, nous introduisons des matrices de transfert qui correspondent aux objectifs de robustesse : la fonction de sensibilité en sortie définie par  $T_{\nu \rightarrow e} = (\mathbb{I}_{10} + \mathbf{P}\mathbf{F})^{-1}$  de dimensions  $10 \times 10$ , telle que  $\|T_{\nu \rightarrow e}\|_\infty = \Delta_m(T_{\nu \rightarrow e})^{-1}$  et la fonction de sensibilité en entrée  $T_{d \rightarrow u} = (\mathbb{I}_4 + \mathbf{F}\mathbf{P})^{-1}$  de dimensions  $4 \times 4$ , définie par  $\|T_{d \rightarrow u}\|_\infty = \Delta_m(T_{d \rightarrow u})^{-1}$ . Par conséquent, la minimisation de la norme  $H_\infty$  de  $T_{\nu \rightarrow e}$  ou de  $T_{d \rightarrow u}$  correspond à l'augmentation des marges de module en entrée et en sortie. Étant donné que le système  $\mathbf{P}$  est MIMO, nous accordons de l'importance aux fonctions de sensibilité en entrée et en sortie qui ne coïncident pas, car  $\mathbf{P}$  et  $\mathbf{F}$  ne commute pas.

Nous définissons aussi la matrice de transfert  $T_{\nu \rightarrow u}$  de dimensions  $4 \times 10$  liée à l'impact du bruit de mesure  $\nu$  sur la commande  $\mathbf{u}$ , et  $T_{d \rightarrow y}$  de dimensions  $10 \times 4$  représentant l'impact de la perturbation en entrée  $\mathbf{d}$  sur la sortie du système  $\mathbf{y}$ .

Nous résolvons le même problème que dans notre travail précédent 4.9 en utilisant le logiciel **Systune** [Apkarian 2006], mais nous utilisons le diagramme de

contrôle présenté dans la section 5.2.1 qui comprend un filtre sur l'action proportionnelle et un nombre différent de sorties. Nous incluons également dans le système  $\mathcal{P}$  la dynamique des actionneurs linéaires discutée dans la section 2.2.1.

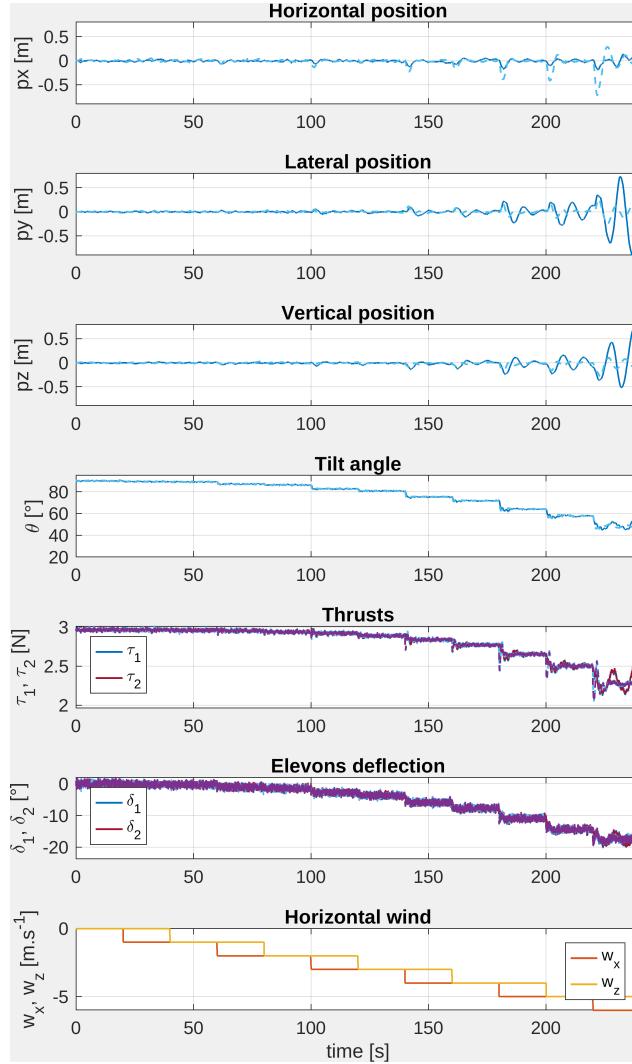


FIGURE 5.2 – Simulation du modèle non linéaire (2.1) (ligne continue) et du modèle linéarisé (2.32) (ligne pointillée) avec des incrémentations de vent constants croissants et le contrôleur basé sur l'optimisation sans vent de la section 5.2.2.

Une augmentation successive de l'intensité horizontale et verticale du vent (allant de zéro à  $-6 \text{ m s}^{-1}$ ) est appliquée, comme le montre le tracé inférieur de la Fig. 5.2. Les couples de vents sélectionnés ( $w_x, w_z$ ) sont représentées par des points rouges sur les surfaces de la figure 2.7, où l'on peut voir que l'équilibre ( $\mathbf{u}_{\text{eq}}, \mathbf{x}_{\text{eq}}$ ) est atteint sans que les actionneurs ne soient saturés. Nous ne nous intéressons qu'à la

partie négative de la vitesse verticale du vent car elle est la plus limitante. En effet, le drone est soulevé par le vent vertical ascendant (dont le signe est négatif dans le cadre de la NED), nécessitant moins de traction sur les hélices pour compenser la gravité. Les moteurs génèrent moins de flux d'air sur les élévons, ce qui réduit leur efficacité, conduit à la saturation et déstabilise le drone. L'objectif du système de contrôle est de maintenir le drone en position de vol stationnaire (définie comme  $\mathbf{r}_p = [0, 0, 0]^\top$ ), malgré l'augmentation du vent horizontal et vertical  $w_x$  et  $w_z$ .

La Figure 5.2 présente à la fois des simulations linéaires avec la dynamique linéarisée (2.32) (en traits pointillés) et des simulations non linéaires avec le modèle non-linéaire (2.1) (en traits pleins). Les simulations linéaires et non linéaires montrent systématiquement que le contrôleur fonctionne bien à faible vitesse de vent (effectivement, le réglage est effectué sur la base du modèle de vent nul). Cependant, lorsque la vitesse du vent  $w_x$  et  $w_z$  dépasse  $-5 \text{ m s}^{-1}$ , la position de vol stationnaire devient instable et le drone oscille et diverge. Les angles d'inclinaison  $\theta$  sont utilisés pour représenter l'attitude afin de donner un meilleur aperçu du comportement du véhicule, mais la simulation de la dynamique non linéaire (2.1) est effectuée avec un quaternion unitaire. L'instabilité observée dans les résultats de la simulation de la figure 5.2 confirme les instabilités expérimentales rapportées dans la section 4.4 où nous avons utilisé cette même méthode de réglage, et confirme l'importance des Théorèmes 1 et 2 dans la Section 2.1, pour un réglage approprié des gains du contrôleur, ce qui est effectué dans la section suivante.

### 5.2.3 Contrôleur optimisé sous contrainte $H_\infty$ , cas multimodèles

Les résultats de simulation obtenus avec la méthode de réglage sans vent (voir la Figure 5.2) ainsi que les instabilités expérimentales observées dans la section 4.4 confirment la nécessité d'une procédure de réglage du gain du contrôleur exploitant les linéarisations paramétrées en fonction du vent des Théorèmes 1 et 2. En nous concentrant à nouveau sur le schéma de contrôle de la Figure 5.1, nous considérons maintenant explicitement l'effet du vent (linéarisé) sur le système (2.32) avec la sortie (5.2) et avec les sélections de l'algorithme 2 comme défini ci-dessous :

$$\begin{aligned} \mathbf{P}_w(s) &= \begin{bmatrix} \mathbf{P}_u(s; w) & \mathbf{P}_w(s; w) \end{bmatrix} \\ &:= \mathbf{C}(s\mathbb{I}_{12} - \mathbf{A}_w)^{-1} \begin{bmatrix} \mathbf{G}_w & \mathbf{E}_w \end{bmatrix}, \end{aligned} \quad (5.6)$$

où l'entrée est la concaténation de l'entrée de commande  $\mathbf{u}$  et de l'entrée de perturbation de vent  $\mathbf{w}$ . Comme le modèle dépend de la vitesse du vent  $\mathbf{w}$ , nous

introduisons une nouvelle matrice de transfert  $T_{w \rightarrow y}$  ayant des dimensions de  $10 \times 3$ , qui correspond à la matrice de transfert entre l'entrée du vent  $\mathbf{w}$  et la sortie du système  $\mathbf{y}$ , quantifiant l'effet de la perturbation du vent sur la boucle de commande du drone.

Avec l'ensemble des matrices de transfert définies dans la section 5.2.2 et la nouvelle matrice de transfert  $T_{w \rightarrow y}$ , nous utilisons l'approche proposée dans [Apkarian 2006, Apkarian 2013] qui utilise des techniques d'optimisation non lisses pour traiter les problèmes de bouclage non convexes, appelée "Systune". Ainsi nous pouvons régler notre architecture de contrôle structurée, où nous optimisons les matrices de gain  $\mathbf{K}$ ,  $\mathbf{H}$  et les paramètres de filtrage  $n_1, n_0, d_2, d_1, d_0$  (en jaune sur la Figure 5.1). Comme indiqué dans [Apkarian 2013, eq. (2)], nous résolvons le problème d'optimisation multiobjectifs, en exploitant l'implémentation Matlab bien expliquée dans [Apkarian 2013, §3].

En particulier, sur la base d'un ensemble  $\mathcal{W}$  comprenant une collection finie de couple  $(w_x, w_z)$ , avec  $w_x \in [0, 8] \text{ m s}^{-1}$  et  $w_z \in [-4, 4] \text{ m s}^{-1}$ , nous considérons l'ensemble des systèmes linéarisés qui en résulte de (5.6) et nous résolvons l'optimisation convexe suivante, où les scalaires  $W_1, W_2, W_3, W_4$  et  $W_5$  sont des facteurs de pondération à ajuster pour obtenir un compromis satisfaisant entre la robustesse (associée à  $W_2, W_3$  et  $W_4$ ) et la performance (associée à  $W_1$  et  $W_5$ )

$$\gamma^* = \min_{\mathbf{F}} \max_{w \in \mathcal{W}} \left| \begin{array}{l} \|W_1 T_{v \rightarrow e}(\mathbf{P}_w, \mathbf{F})\|_\infty \\ \|W_2 T_{d \rightarrow u}(\mathbf{P}_w, \mathbf{F})\|_\infty \\ \|W_3 T_{v \rightarrow u}(\mathbf{P}_w, \mathbf{F})\|_\infty \\ \|W_4 T_{d \rightarrow y}(\mathbf{P}_w, \mathbf{F})\|_\infty \\ \|W_5 T_{w \rightarrow y}(\mathbf{P}_w, \mathbf{F})\|_\infty \end{array} \right|, \text{ sous condition que} \quad (5.7)$$

$\mathbf{F}$  stabilise en interne  $\mathcal{F}_\ell(\mathbf{P}_w, \mathbf{F}), \forall w \in \mathcal{W}$ ,

où  $\mathcal{F}_\ell(\mathbf{P}_w, \mathbf{F})$  désigne l'interconnexion de bouclage linéaire de la Figure 5.1 pour une valeur spécifique de  $w$  (ceci est cohérent avec la notation classique de contrôle robuste [Apkarian 2006, Apkarian 2013]). Nous notons que, par rapport à [Apkarian 2013, eq. (2)], nous ne spécifions que des contraintes *soft* et non des contraintes *hard*.

Le problème d'optimisation (5.7) devient de plus en plus lourd d'un point de vue informatique, à mesure que nous augmentons la cardinalité de l'ensemble des conditions de vent considérées dans  $\mathcal{W}$ . En fait, une approche de force brute incluant un maillage fin de points dans  $\mathcal{W}$  conduit à une optimisation difficile à calculer. Au

---

**Algorithme 3** Réglage itératif et multimodèle des gains du contrôleur.

**Entrées :**  $\mathbf{A}_w, \mathbf{G}_w, \mathbf{E}_w$  les matrices de sortie de l'algorithme 2 et les scalaires de pondération positifs  $W_1-W_5$

**Sorties :**  $\mathbf{K}, \mathbf{H}$  et les gains du filtre

- 1: (Initialisation) Initialiser  $\mathcal{W}$  comme un maillage comprenant toutes les paires  $w_x \in \{0, -4, -8\}$  et  $w_z \in \{-4, 0, 4\}$
- 2: (Synthèse) Résoudre l'optimisation (5.7) avec le logiciel **Systune**
- 3: (Analyse) Définir un maillage de validation  $\mathcal{W}_v$  en discrétilisant l'intervalle  $(w_x, w_y) \in [0, 8] \times [-4, 4]$  avec un pas de discrétilisation de 1. En utilisant le contrôleur  $\mathbf{F}$  obtenu à l'étape précédente et pour chaque  $w_v \in \mathcal{W}_v$ , nous calculons :

$$\gamma_v = \begin{vmatrix} \|W_1 T_{v \rightarrow e}(\mathbf{P}_{w_v}, \mathbf{F})\|_\infty \\ \|W_2 T_{d \rightarrow u}(\mathbf{P}_{w_v}, \mathbf{F})\|_\infty \\ \|W_3 T_{v \rightarrow u}(\mathbf{P}_{w_v}, \mathbf{F})\|_\infty \\ \|W_4 T_{d \rightarrow y}(\mathbf{P}_{w_v}, \mathbf{F})\|_\infty \\ \|W_5 T_{w \rightarrow y}(\mathbf{P}_{w_v}, \mathbf{F})\|_\infty \end{vmatrix}, \quad (5.8)$$

et augmenter  $\mathcal{W}$  avec le point correspondant si  $\gamma_v > 1$  ou si  $\gamma_v$  est indéfini (à savoir si  $\mathbf{F}$  n'est pas stabilisant).

- 4: (Conclusion) Si  $\mathcal{W}$  n'a pas été augmenté à l'étape précédente, passer à l'étape 5, sinon passer à l'étape 2.
  - 5: **Retourne :**  $\mathbf{K}, \mathbf{H}$  et les paramètres du filtre  $n_1, n_0, d_2, d_1, d_0$
-

Pondération	$W_1$	$W_2$	$W_3$	$W_4$	$W_5$
Values	18	16	11	26	5

TABLE 5.2 – Valeurs des scalaires de pondération positifs  $W_1$ – $W_5$  utilisés dans l'exécution de l'Algorithme 3.

lieu de cela, nous suivons ici la procédure itérative décrite dans l'algorithme 3, où  $\mathcal{W}$  est initialement sélectionné comme un maillage grossier comprenant  $3 \times 3 = 9$  points (étape 1); une étape de synthèse (étape 2) est ensuite suivie de manière répétée par une étape d'analyse (simple du point de vue du calcul) (étape 3) où le contrôleur  $\mathbf{F}$  est fixé. L'étape 3 identifie les points de violation en utilisant un maillage de validation plus fin  $\mathcal{W}_v$  et les ajoute à l'ensemble d'optimisation  $\mathcal{W}_v$ . L'algorithme se termine après quelques itérations, lorsque aucun point du maillage de validation ne viole les contraintes.

L'exécution de l'algorithme 3 pour les modèles de DarkO issus des théorèmes 1 et 2 avec la sélection des scalaires de pondération positifs  $W_1$ – $W_5$  indiqués dans la Table 5.2, a donné la sélection suivante après 2 itérations :

$$\begin{aligned} \left[ \mathbf{K}^\top \mid \mathbf{H}^\top \right] &= \left[ \begin{array}{cccc|cc} -3.86 & -3.86 & 0.79 & 0.79 & 0.02 & 0.48 \\ 1.43 & -1.43 & 1.71 & -1.71 & -0.47 & -1.63 \\ 4.06 & 4.06 & -2.07 & -2.07 & -0.45 & 0.52 \\ -6.86 & -6.86 & -11.60 & -11.60 & -0.14 & 1.40 \\ -10.75 & 10.75 & -1.89 & 1.89 & 3.35 & 5.69 \\ 27.20 & 27.20 & -4.29 & 4.29 & -1.84 & 3.79 \\ -12.32 & 12.32 & -3.46 & 3.46 & 3.72 & 6.81 \\ -5.84 & 5.84 & -2.29 & 2.29 & 1.58 & 3.13 \\ -5.19 & 5.19 & 5.79 & 5.79 & 2.86 & -1.54 \\ -6.52 & 6.52 & 0.08 & -0.08 & 0.08 & 2.82 \end{array} \right], \\ \left[ \begin{array}{c|c} n_1 & n_0 \\ \hline d_2 & d_1 \\ \hline d_0 \end{array} \right] &= \left[ \begin{array}{c|c} -429 & -389 \\ \hline 1 & 6475 \\ \hline 4905 & \end{array} \right], \end{aligned} \tag{5.9}$$

Après la première itération de l'algorithme 3 et qu'un contrôleur candidat  $\mathbf{F}$  ait été évalué à l'étape 2, nous pouvons tracer la Figure 5.3. Elle montre en bleu les diagrammes de bode des valeurs singulières maximales de  $T_{\nu \rightarrow e}$ ,  $T_{d \rightarrow u}$ ,  $T_{\nu \rightarrow u}$ ,  $T_{d \rightarrow y}$ , et  $T_{w \rightarrow y}$  (associées à la valeur de  $\gamma_v$ ) reportée dans (5.8) à l'étape d'analyse 3, à comparer à l'inverse des cinq poids  $W_1$ – $W_5$ , représentés par les lignes horizontales vertes. Les diagrammes en rouge correspondent aux points qui violent les contraintes et qui sont ajoutés à l'ensemble  $\mathcal{W}$  pour l'itération suivante. Les quelques diagrammes en magenta, en revanche, correspondent aux 9 points considérés dans  $\mathcal{W}$  pour la première itération de l'étape de synthèse 2. Les diagrammes rouges de la Fig. 5.3 illustrent clairement que l'algorithme itératif parvient à détecter les valeurs critiques de la vitesse du vent ( $w_x, w_z$ ) à ajouter à l'ensemble d'optimisation  $\mathcal{W}$ .

Les valeurs singulières de la fonction de sensibilité de la sortie et de l'entrée (respectivement  $T_{r \rightarrow e}$  et  $T_{d \rightarrow u}$ ) sont représentées sur la Fig. 5.3 ligne supérieure. Le graphique de la troisième ligne représente la valeur singulière du transfert entre la perturbation du vent  $w$  et la sortie du drone  $y$ . La valeur singulière qui tangente la contrainte est celle de la condition de vent la plus élevée du modèle de synthèse,  $(w_x, w_z) = (-8, -4) \text{ ms}^{-1}$ .

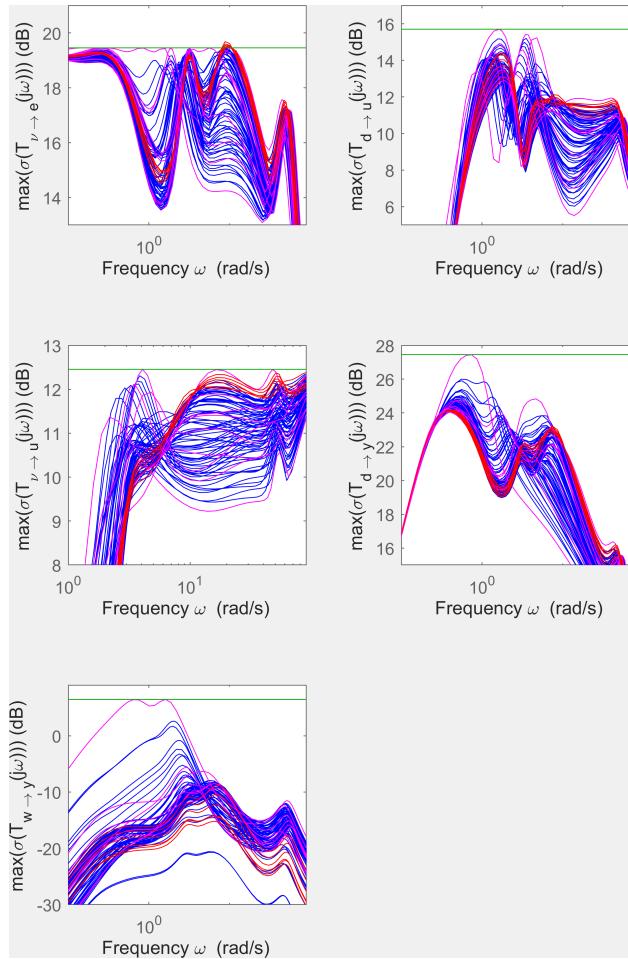


FIGURE 5.3 – Diagrammes des valeurs singulières des fonctions de transfert dans (5.8) à la première itération de l'algorithme 3.

Avec le réglage indiqué dans (5.9), tel qu'il est obtenu avec l'Algorithme 3, nous présentons dans la Figure 5.4 des résultats de simulation similaire à ceux déjà présentés dans la Figure. 5.2 pour la méthode de réglage sans vent discutée dans la Section 5.2.2. Une fois de plus, nous simulons à la fois le modèle non linéaire (2.1) (lignes pleines) et le modèle linéarisé (2.32) (ligne en pointillés). Par rapport à la Fig. 5.2, les simulations de la Figure 5.4 montrent que le réglage du contrôleur basé

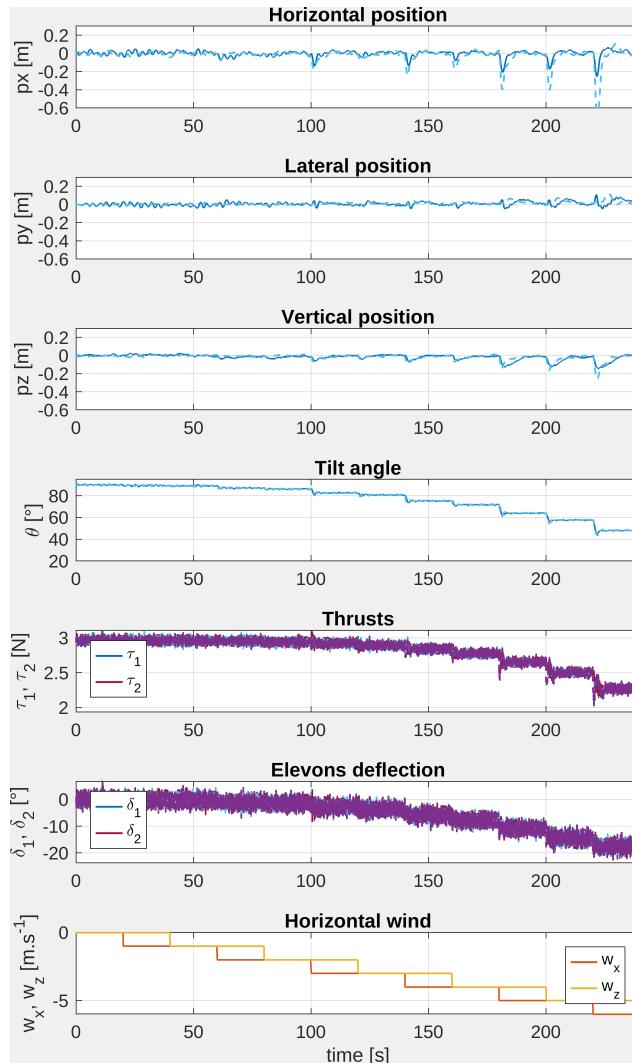


FIGURE 5.4 – Simulation du modèle non linéaire (2.1) (ligne continue) et du modèle linéarisé (2.32) (ligne pointillée) avec des incrémentations de vent constantes croissantes, le contrôleur étant réglé à l'aide de l'optimisation multimodèle de l'algorithme 3 dans la section 5.2.3.

sur les Théorèmes 1 et 2 résout les problèmes d'instabilité et parvient à stabiliser le vol stationnaire dans tous les scénarios de vent considérés. Nous notons également que la Figure. 5.4 montre une action plus agressive. En effet l'entrée de contrôle  $u$  (à la fois la poussée et les déflections) est plus affectée par le bruit de mesure. L'efficacité du schéma de contrôle réglé sur la base de l'Algorithme 3 est également confirmée par les résultats expérimentaux présentés dans la section suivante.

### 5.3 Vol expérimental en soufflerie ouverte

Les vols expérimentaux de DarkO se sont déroulés dans un espace dédié (voir Fig. 5.5) avec un système de localisation Optitrack basé sur une convention NED selon la Figure 2.2. Nous avons utilisé un générateur de vent à veine ouverte pour obtenir des incrément de vent que nous avons mesurés à l'aide d'une sonde à fil chaud (la barre verticale dans la Fig. 5.5). Bien que cette information sur le vent



FIGURE 5.5 – Vol expérimental de DarkO devant la soufflerie ouverte.

soit enregistrée à bord du drone pour synchroniser les données, nous n'utilisons pas cette mesure dans la loi de commande. La fréquence de mesure de cette sonde de vent n'est que de 0,5 Hz, de sorte que nous n'avons qu'une mesure toutes les deux secondes. L'estimation de l'état est effectuée à l'aide d'un système de navigation inertiel pour fusionner les données de l'unité de mesure inertiel (IMU) et du système de localisation Optitrack afin d'obtenir une estimation précise de la sortie  $y$  dans la Fig. 5.1. Cependant, la vitesse angulaire du drone  $\omega_{textb}$  est mesurée sur la base du gyromètre de l'IMU, qui fournit des mesures bruitées. Nous avons donc ajouté un filtre passe-bas *Butterworth* de second ordre avec une fréquence de coupure de 20 Hz pour lisser la sortie  $\omega_{textb}$ . Le filtre de *Butterworth* est pris en compte dans la dynamique linéarisée lors de l'optimisation des gains du contrôleur en suivant l'Algorithme 3.

Nous avons également utilisé les ESC similaires à ceux utilisés lors de l'identification montrée dans la Figure 2.3 pour l'actionnement des hélices. Les deux ESC ont été flashés avec le code open-source AM32 (voir Annexe A.2). L'avantage de ce *firmware*, par rapport au code commercial, est qu'il exploite un retour PID de bas niveau de la vitesse de rotation du moteur, qui est calculée à la même vitesse que la commutation de phase du moteur. Nous avons adapté le code de la boucle de vitesse dans le *firmware*, en suivant l'approche de [Franchi 2017], avec un algorithme de biais et de gain adaptatif (ABAG) . De cette manière, nous compensons les effets de

Comparaison linéaire

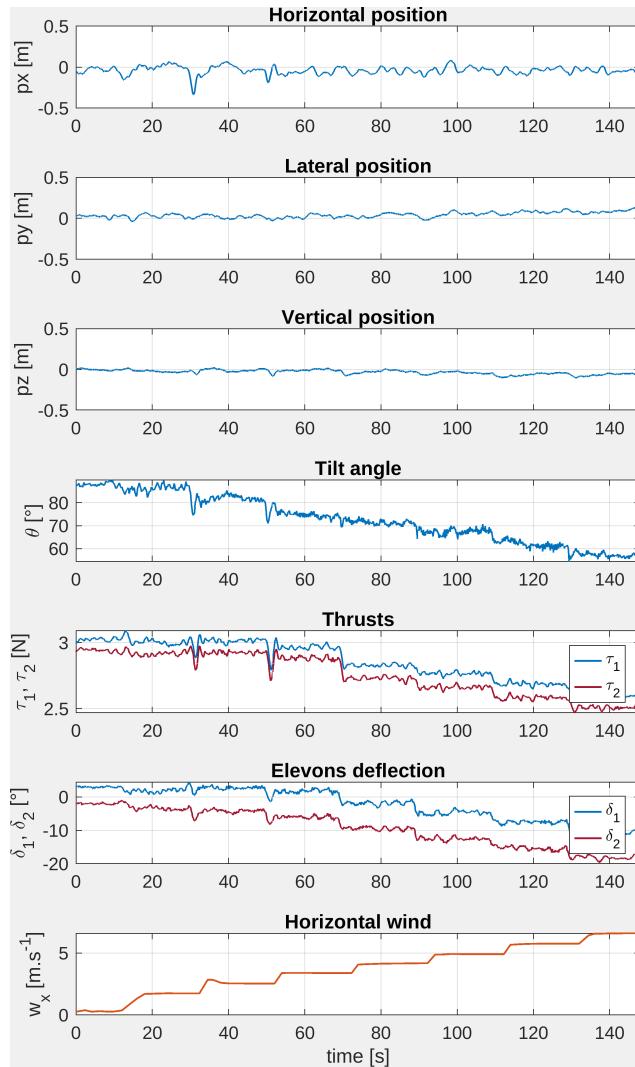


FIGURE 5.6 – Expérience du drone DarkO devant la soufflerie avec des incrémentations de vent constants croissants (graphique du bas).

décharge de la batterie et obtenons un suivi précis de la vitesse commandée. Avant cette modification, l'action intégrale de la rétroaction stabilisatrice de la Fig. 5.1 compensait la perte de vitesse du moteur causée par la réduction de la tension de la batterie pendant le vol. Cette compensation intégrale a été indirectement générée par la perte d'altitude du drone causée par la réduction de la traction. Les avantages de la solution ABAG sont une grande réactivité et une grande adaptabilité, puisque les dimensions de l'hélice peuvent être modifiées sans qu'il soit nécessaire de modifier les gains d'actionnement.

Nous avons réalisé une expérience de vol au cours de laquelle DarkO a été

mis manuellement en mode de vol stationnaire stabilisé devant la soufflerie, puis nous avons activé la loi de contrôle de l’Algorithme 3. Comme le drone devait être stabilisé à au moins 30 cm de la soufflerie, un pilotage manuel a été réalisé pour éviter tout dépassement qui pourrait endommager la soufflerie. Une fois que DarkO était suffisamment proche du point de consigne  $r_p$  de la Fig. 5.1, nous avons activé le contrôleur proposé, obtenant les résultats de la Fig. 5.6. Au cours de la phase d’expérimentation, comme le montre le graphique inférieur de la Fig. 5.6, nous avons augmenté progressivement la vitesse du vent, en attendant 20 secondes entre chaque évolution de vent jusqu’à une vitesse finale de  $7 \text{ m s}^{-1}$ .

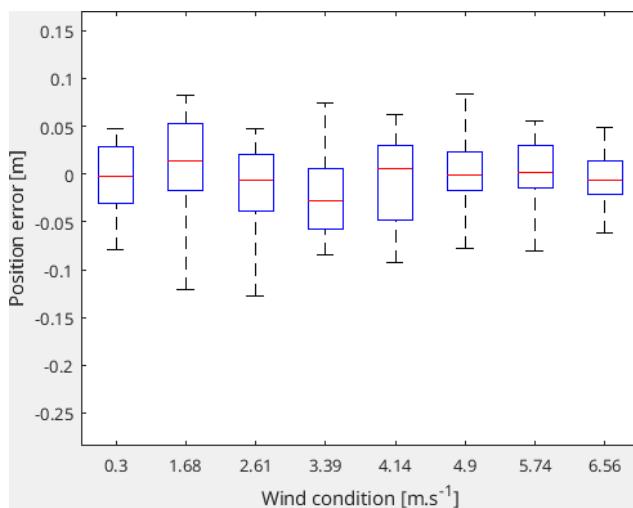


FIGURE 5.7 – Visualisation statistique des performances de vol stationnaire.

Les figures 5.6 et 5.7 montrent que le drone maintient sa position malgré l’augmentation de la vitesse du vent. Nous pouvons noter quelques points importants, en accord avec les simulations : la traction du moteur diminue lorsque la vitesse du vent augmente. Le schéma de contrôle profite de la portance générée par le vent pour soutenir le drone, de sorte que moins d’énergie est nécessaire pour stabiliser la position en vol stationnaire. Le drone maintient son angle d’inclinaison à une valeur inconnue a priori pour la loi de commande et qui découle naturellement de l’action intégrale. Cette valeur est atteinte asymptotiquement et converge vers la valeur requise de  $\theta$ . Pour stabiliser la position, le drone utilise les élévons pour annuler le moment de tangage généré par la forme de l’aile, soumise à un vent horizontal, sans atteindre les limites de saturation. On note également une légère asymétrie de l’efficacité des actionneurs, qui est efficacement compensée par l’action proportionnelle du schéma de contrôle.

## **5.4 Conclusion du Chapitre 5**

## CHAPITRE 6

# Modélisation et commande d'un drone à aile libre rotation libre

---

## Sommaire

<b>6.1</b>	<b>Motivation</b>	<b>85</b>
<b>6.2</b>	<b>Design et modélisation d'un drone : Colibri</b>	<b>85</b>
<b>6.3</b>	<b>State estimation</b>	<b>90</b>
6.3.1	Placement des capteurs	90
6.3.2	Estimation de la vitesse angulaire	92
6.3.3	Estimation de l'état de l'aile	94
6.3.4	Estimation de l'orientation du fuselage	95
<b>6.4</b>	<b>Inversion non linéaire incrémentale de la dynamique du drone</b>	<b>95</b>
<b>6.5</b>	<b>Expérimentations</b>	<b>97</b>
<b>6.6</b>	<b>Vol avec un contrôleur unifié</b>	<b>97</b>
<b>6.7</b>	<b>Commande Udwadia-Kalaba</b>	<b>97</b>
<b>6.8</b>	<b>Vols expérimentaux</b>	<b>97</b>
<b>6.9</b>	<b>Conclusion du Chapitre 6</b>	<b>97</b>

---

## 6.1 Motivation

## 6.2 Design et modélisation d'un drone : Colibri

Le drone Colibri est dérivé d'un drone *tailsitter* qui génère de la portance pendant le vol d'avancement. Cette aile possède plusieurs actionneurs : quatre moteurs  $u_i$ ,  $i = 1, 2, 3, 4$  et deux élevons  $\delta_l$  et  $\delta_r$ . Nous pouvons définir le vecteur de contrôle  $u_W$  de l'aile d'après la Figure 6.1 comme  $u_W = [u_1 \ u_2 \ u_3 \ u_4 \ \delta_l \ \delta_r]^\top$ . Un fuselage relié par un pivot est fixé au centre aérodynamique de l'aile. Ce fuselage supporte

Explication sur l'intérêt d'avoir un fuselage, capteur, charge utile

l'autopilote, la batterie, un moteur et un empennage pour le maintenir horizontal. Dans la Figure 6.1, toutes les surfaces de contrôle aérodynamiques sont représentées en rose et les hélices en vert. Trois repères sont attachés au drone. (I) est un repère inertiel NED lié à la surface de la terre, (W) est un repère attaché à l'aile du drone et (F) est un repère attaché au fuselage du drone.

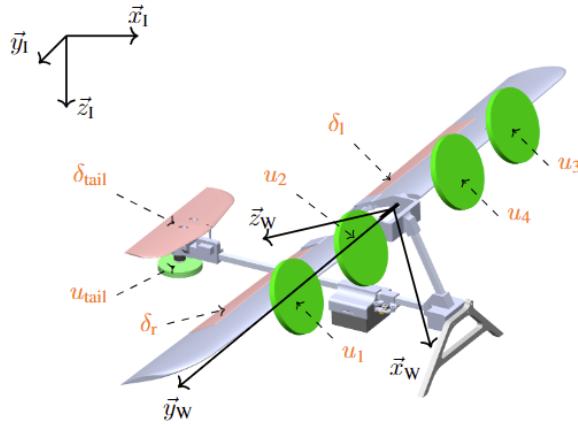


FIGURE 6.1 – Les repères, inertiel (I) et aile (W), attaché à l'architecture de Colibri.

Certaines des dimensions caractéristiques sont indiquées dans la Table 6.1. Notons que les moteurs sont positionnés symétriquement sur l'aile, ce qui signifie que la position peut être décrite en se concentrant sur un seul côté.

Paramètres	Valeur	Unités
$m_W$ (masse de l'aile)	0.53	kg
$m_F$ (masse du fuselage)	1.17	kg
$J_W = diag(J_x^W, J_y^W, J_z^W)$	diag(0.1677, 0.0052, 0.1634)	$kg\ m^2$
$J_F = diag(J_x^F, J_y^F, J_z^F)$	diag(0.0191, 0.0161, 0.0343)	$kg\ m^2$
$k_f$ (coefficients de poussée de l'hélice)	1.7800e-8	kg m
$d_{MO_W}$	[0.383, 0, -0.167] <sup>T</sup>	m
$d_{GO_W}$	[0.052, 0, -0.171] <sup>T</sup>	m

TABLE 6.1 – Paramètres numériques du modèle Colibri.

La modélisation est basée sur les résultats de [Udwadia 2006, Section 2.15]. L'algorithme de calcul des matrices  $M$ ,  $A$ ,  $Q$  et  $B$  se trouve dans [Udwadia 2012], qui nous fournit les équations de mouvement d'un système multicorps contraint :

$$\ddot{x} = \hat{M}^\dagger \begin{bmatrix} Q \\ B \end{bmatrix} = \begin{bmatrix} (I - A^\dagger A)M \\ A \end{bmatrix}^\dagger \begin{bmatrix} Q \\ B \end{bmatrix} \quad (6.1)$$

dont l'expression est valide tant que  $\hat{M}$  a un rang complet et où  $A, M, Q$  et  $B$  sont décrits par la suite.

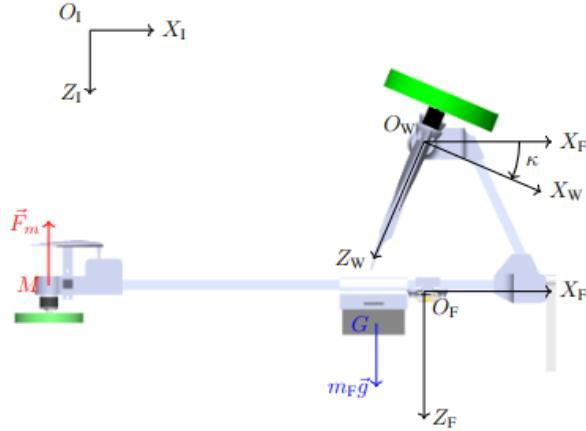


FIGURE 6.2 – Repère inertiel (I), du fuselage (F) et de l'aile (W) et forces agissant sur le drone Colibri.

Nous utiliserons les quaternions  $q = [\eta \epsilon^\top]^\top \in \mathbb{S}^3 := \{q \in \mathbb{R}^4 : |q| = 1\}$  pour représenter les orientations des deux corps. La matrice de rotation qui en résulte  $R(q) \in SO(3) := \{R \in \mathbb{R}^{3 \times 3} : R^\top R = I, \det(R) = 1\}$  est définie de manière unique comme  $R(q) := I + 2\eta[\epsilon]_\times + 2[\epsilon]^2 = [R_1 \ R_2 \ R_3]$ .

D'après les Figures 6.1 et 6.2, nous définissons les vecteurs  $p_F = \overrightarrow{O_I O_F}$ ,  $p_W = \overrightarrow{O_I O_W}$ ,  $d_{FW} = \overrightarrow{O_F O_W}$  satisfaisant  $d_{FW} = p_W - p_F$  and  $d_{MO_W} = \overrightarrow{M O_W}$ ,  $d_{GO_W} = \overrightarrow{G O_W}$ .

Le vecteur d'état global est  $(x, v) \in \mathbb{R}^{28}$  avec  $x = (p_W, q_W, p_F, q_F) \in \mathbb{R}^{14}$  et  $v = (v_W, \dot{q}_W, v_F, \dot{q}_F) = (\dot{p}_W, \dot{q}_W, \dot{p}_F, \dot{q}_F) = \dot{x} \in \mathbb{R}^{14}$ , où  $v_W = \dot{p}_W \in \mathbb{R}^3$  représente la vitesse linéaire de l'aile dans le repère inertiel,  $\dot{q}_W \in \mathbb{R}^4$  est la dérivée du quaternion,  $q_W \in \mathbb{R}^4$  représentant l'orientation de l'aile,  $v_F = p_F \in \mathbb{R}^3$  est la vitesse linéaire du fuselage dans le repère inertiel et  $\dot{q}_F \in \mathbb{R}^4$  est la dérivée du quaternion,  $q_F \in \mathbb{R}^4$  représentant l'orientation du fuselage. Nous observons que le vecteur d'état n'est pas minimal. Il convient de noter que la vitesse angulaire  $\omega \in \mathbb{R}^3$  peut être obtenue à partir de la dérivée du quaternion  $\dot{q}$  en utilisant l'équation [Udwadia 2012, équation (2.7)] rappelée ici :

$$\omega = H(q)\dot{q}$$

où  $H(q) \in \mathbb{R}^{3 \times 4}$  est une matrice définie par  $H(q) = 2[-\epsilon \quad \eta I_3 - [\epsilon]_\times]$ . Pour dériver les équations du mouvement, rappelons que  $R_i(q) \in \mathbb{R}^3, i = 1, 2, 3$  sont

Minimalité du vecteur

les trois colonnes d'une matrice de rotation associée au quaternion  $q$ , définissent les matrices  $L_i^W(q_W) = \frac{\partial R_i}{\partial q}(q_W) \in \mathbb{R}^{3 \times 4}$ ,  $L_i^F(q_F) = \frac{\partial R_i}{\partial q}(q_F) \in \mathbb{R}^{3 \times 4}$  et  $L_{O_F^W} = \sum_{i=1}^3 d_{FW}(i)L_i^F(q_F)$ ,  $i \in 1, 2, 3$ , où  $d_{FW}(i)$  désigne la  $i$ -ème composante du vecteur  $d_{FW} = p_W - p_F$ . Comme  $O_W$  est situé au centre de rotation de l'aile, la distance  $d_{FW}$  est une constante, puisque  $O_W$  et  $O_F$  appartiennent au même solide (le fuselage).

On en déduit, avec homogénéité,  $\dot{L}_{O_F^W} = \sum_{i=1}^3 d_{FW}(i)L_i^F(\dot{q}_F)$ . Avec ces définitions, les matrices de (6.1) sont

$$M = \begin{bmatrix} m_W I_3 & \mathbb{0}_{3 \times 4} & \mathbb{0}_3 & \mathbb{0}_{3 \times 4} \\ \mathbb{0}_{4 \times 3} & H_W^\top J_W H_W & \mathbb{0}_{4 \times 3} & \mathbb{0}_4 \\ \mathbb{0}_3 & \mathbb{0}_{3 \times 4} & m_F I_3 & \mathbb{0}_{3 \times 4} \\ \mathbb{0}_{4 \times 3} & \mathbb{0}_4 & \mathbb{0}_{4 \times 3} & H_F^\top J_F H_F \end{bmatrix} \in \mathbb{R}^{14 \times 14}, \quad (6.2)$$

où nous avons noté  $H_W = H(q_W)$ ,  $H_F = H(q_F)$ , et

$$Q = \begin{bmatrix} m_W g e_3 + R(q_W) F_b \\ -2 \dot{H}_W^\top J_W \dot{H}_W \dot{q}_W + H_W^\top M_W \\ m_F g e_3 + R(q_F) F_F \\ -2 \dot{H}_F^\top J_F \dot{H}_F \dot{q}_F + H_F^\top M_F \end{bmatrix} \in \mathbb{R}^{14}, \quad (6.3)$$

où  $\dot{H}_W$  désigne  $H(\dot{q}_W)$ , coïncidant avec la dérivée temporelle de  $H(q_W)$  et  $\dot{H}_F$  désigne  $H(\dot{q}_F)$ , coïncidant avec la dérivée temporelle de  $H(q_F)$ . De plus,  $F_b$  et  $M_b$  représentent respectivement toutes les forces et tous les moments agissant sur l'aile. Les expressions de  $M$  et  $Q$  sont tirées de [Lustosa 2019, équations (45) et (57)] où est développée la  $\phi$ -théorie, paramétrage qui permet de soustraire les angles classiques d'incidence et de dérapage et d'éviter la singularité du vol stationnaire.

Enfin,  $F_F = F_m$  et  $M_F$  représentent respectivement l'ensemble des forces et moments non gravitationnels agissant sur le fuselage exprimé dans le repère  $O_W$ . En particulier,  $F_m = -k_f u_{tail}^2$  est la force générée par le moteur situé à l'arrière du fuselage et  $u_{tail}$  est la vitesse de rotation du moteur telle que :

$$M_F = m_F g e_3 \times d_{GO_W} + F_m \times d_{MO_W}, \quad (6.4)$$

où  $d_{MO_W}$  est la distance entre l'emplacement du moteur et le centre de rotation et  $d_{GO_W}$  est la distance entre l'emplacement du centre de gravité du fuselage et le centre de rotation.

L'ensemble des contraintes, associées à la non-minimalité de l'état  $(x, v)$  et à la

liaison pivot entre les deux corps, est donné par :

$$\begin{cases} \varphi_1 := q_W^\top q_W - 1 = 0 \\ \varphi_2 := q_F^\top q_F - 1 = 0 \\ \varphi_3 := R_2(q_W)^\top R_3(q_F) = 0 \\ \varphi_4 := R_2(q_W)^\top R_1(q_F) = 0 \\ \varphi_5 := p_F + d_{FA} + p_W = 0 \end{cases} \quad (6.5)$$

Les deux premières contraintes imposent la norme unitaire des quaternions  $q_F$  et  $q_W$ . Les troisième et quatrième contraintes sont liées à une contrainte pivot, c'est-à-dire que l'orthogonalité de deux vecteurs est imposée. La dernière est une contrainte de position, de sorte que le point du centre de rotation de l'aile coïncide avec le point défini dans le fuselage. Cette contrainte est basée sur une fermeture géométrique tridimensionnelle. Il est plus commode d'exprimer l'ensemble des contraintes sous la forme d'un système dynamique stable convergeant vers zéro, et nous convertissons donc chacune des contraintes sous la forme suivante :

$$\ddot{\varphi}_i + \delta_1 \dot{\varphi}_i + \delta_2 \varphi_i = 0, i \in \{1, 2, 3, 4, 5\}, \quad (6.6)$$

avec les sélections  $(\delta_1, \delta_2) = (0.5, 8)$  étant les coefficients d'un polynôme stable, de sorte que, quelle que soit la sélection  $\varphi_i(0) = 0$ , nous avons  $\lim_{t \rightarrow \infty} \varphi_i(t) = 0$ . En différenciant deux fois les contraintes (6.5) et en les factorisant sous la forme  $A(x, \dot{x})\ddot{x} = B(x, \dot{x})$ , nous obtenons l'expression de  $A(x, \dot{x})$  reportée dans l'équation (??) et de  $B(x, \dot{x})$  reportée dans l'équation (6.8).

$$A = \begin{bmatrix} 0_{1 \times 3} & q_W^\top & 0_{1 \times 3} & 0_{1 \times 4} \\ 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & q_F^\top \\ 0_{1 \times 3} & R_3(q_F)^\top L_2^W(q_W) & 0_{1 \times 3} & R_2(q_W)^\top L_3^F(q_F) \\ 0_{1 \times 3} & R_1(q_F)^\top L_2^W(q_W) & 0_{1 \times 3} & R_2(q_W)^\top L_3^F(q_F) \\ \mathbb{I}_3 & L_{O_F^W} & -\mathbb{I}_3 & 0_{3 \times 4} \end{bmatrix} \quad (6.7)$$

$$B = \begin{bmatrix} -\delta_1 q_W^\top \dot{q}_W - \frac{\delta_2}{2} (q_W^\top q_W - 1) - \dot{q}_W^\top \dot{q}_W \\ -\delta_1 q_F^\top \dot{q}_F - \frac{\delta_2}{2} (q_F^\top q_F - 1) - \dot{q}_F^\top \dot{q}_F \\ -R_3(q_F)^\top \dot{L}_2^W \dot{q}_W - R_2(q_W)^\top \dot{L}_3^F \dot{q}_F - 2\dot{q}_W^\top L_2^W^\top L_3^F \dot{q}_F - \delta_1(R_3(q_F)^\top L_2^W \dot{q}_W + R_2(q_W)^\top L_3^F \dot{q}_F) - \delta_2 \varphi_3 \\ -R_1(q_F)^\top \dot{L}_2^W \dot{q}_W - R_2(q_W)^\top \dot{L}_1^F \dot{q}_F - 2\dot{q}_W^\top L_2^W^\top L_1^F \dot{q}_F - \delta_1(R_1(q_F)^\top L_2^W \dot{q}_W + R_2(q_W)^\top L_1^F \dot{q}_F) - \delta_2 \varphi_4 \\ \dot{L}_{O_F^W} \dot{q}_F - \delta_1(v_W + \dot{L}_{O_F^W} \dot{q}_W - v_F) - \delta_1 \varphi_5 \end{bmatrix} \quad (6.8)$$

La simulation d'un drone reste complexe, car il est naturellement instable. Nous avons choisi d'utiliser la loi de contrôle proposée dans 5.2.3 pour stabiliser le système. Ce contrôle stabilise l'aile. Une autre loi de contrôle basée sur une rétroaction proportionnelle-dérivée stabilise le fuselage pour le maintenir horizontal. Les résultats de la simulation en boucle fermée sont présentés à la Figure 6.3.

En considérant les degrés de liberté de la liaison pivot, le couplage entre les deux corps est clairement visible sur les trois graphiques du bas. En effet, les angles de roulis et de lacet ( $\phi_F, \psi_F$ ) et ( $\phi_W, \psi_W$ ) du fuselage et de l'aile coïncident parfaitement, alors que les angles de tangage ( $\theta_F, \theta_W$ ) sont différents.

### 6.3 State estimation

Pour stabiliser ce système de drone à deux corps, il est nécessaire de connaître la position et l'orientation des deux corps. Grâce à la liaison pivot entre l'aile et le fuselage, la différence entre l'orientation de l'aile et l'orientation du fuselage est simplement une rotation autour de l'axe de tangage de l'aile. Les deux autres orientations (roulis et lacet) coïncident. La position du centre de gravité du fuselage peut être déduite de la position du centre de gravité de l'aile et de l'angle entre le fuselage et l'aile. Cet angle est mesuré par un codeur rotatif en quadrature (CUI Devices AMT22, codeurs absous, 12 bits, SPI), qui renvoie une mesure angulaire quantifiée avec un pas de  $0.09^\circ$ . Compte tenu de cette mesure angulaire, nous examinons ci-dessous l'estimation des informations relatives à la vitesse, afin de reconstruire l'état du drone.

#### 6.3.1 Placement des capteurs

Une première question concerne l'emplacement des capteurs : l'IMU (accéléromètre, gyroscope et magnétomètre) peut être installé sur le fuselage ou sur l'aile. L'installation de l'IMU sur l'aile permet d'effectuer les mesures directement dans le

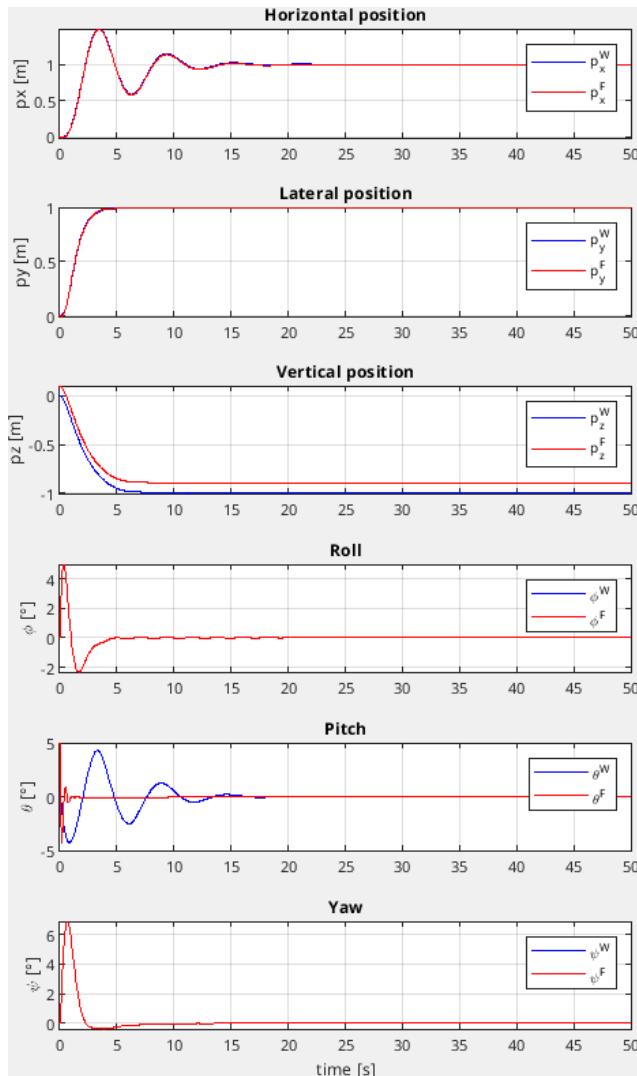


FIGURE 6.3 – Simulation de la position et de l'orientation du drone multicorps Colibri en boucle fermée avec un contrôleur à double boucle.

référentiel souhaité, mais les mesures sont plus bruitées car l'IMU est attaché à la structure supportant les moteurs. Compte tenu de la taille des ailes, leur flexibilité peut générer des résonances et perturber les mesures.

L'installation de l'IMU sur le fuselage réduit les vibrations, mais implique que les mesures soient transformées dans le référentiel de l'aile. La transformation correspondante peut être calculée à partir de la mesure de l'encodeur rotatif, qui fournit l'angle entre l'aile et le fuselage, ainsi qu'à partir des mesures prises avec le logiciel de CAO, qui fournissent des informations précises sur les distances entre les référentiels de l'aile et du fuselage. Notre choix final est de fixer l'IMU au fuselage.

Une autre considération est que la carte de pilotage automatique, qui a déjà une IMU intégrée, est également supposée être connectée à la charge utile et à d'autres capteurs fixés sur fuselage. Il s'agit donc de limiter le nombre de câbles au point de pivot pour les commandes des actionneurs et l'alimentation électrique.

### 6.3.2 Estimation de la vitesse angulaire

Comme expliqué ci-dessus, nous pouvons mesurer l'angle  $\kappa \in \mathbb{R}$  entre l'aile et le fuselage à l'aide de l'encodeur rotatif. Ensuite, pour estimer la vitesse angulaire, nous utilisons un observateur grand gain proposé dans [Nicosia 1990] (voir également [Chitour 2002] pour l'utilisation d'observateurs grand gain pour estimer les dérivées temporelles). Cette méthode est préférable à une dérivée par différence fine, car les informations quantifiées générées par l'encodeur rotatif peuvent donner lieu à du bruit numérique dans les valeurs estimées de vitesse angulaire.

Désignons par  $\kappa \in \mathbb{R}$  l'angle mesuré, par  $\omega_\kappa := \dot{\kappa} \in \mathbb{R}$  sa dérivée, à estimer, et par  $\xi = [\kappa, \omega_\kappa]^\top \in \mathbb{R}^2$  leur juxtaposition en un seul vecteur. Notons également  $\hat{\xi}$  l'estimation de  $\xi$  définie par :

$$\hat{\xi} = [\hat{\kappa}, \hat{\omega}_\kappa]^\top \in \mathbb{R}^2.$$

Suivant [Nicosia 1990], la dynamique de l'estimateur est donnée par :

$$\dot{\hat{\xi}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \hat{\xi} + \begin{bmatrix} k_p \\ \frac{\epsilon_\kappa}{k_v} \\ \frac{k_v}{\epsilon_\kappa^2} \end{bmatrix} (\kappa - \hat{\kappa}), \quad (6.9)$$

où  $\kappa$  est la mesure angulaire récupérée du capteur,  $k_p$  et  $k_v$  sont deux gains scalaires positifs tels que l'équation caractéristique  $s^2 + k_v s + k_p = 0$  ait des racines à partie réelle négative. Pour notre estimateur, nous avons choisi  $k_p = 1$  et  $k_v = 1,3$  de manière à obtenir un facteur d'amortissement  $\zeta = 0,65$  conduisant à une réponse légèrement sous-amortie comme compromis approprié entre un temps de montée rapide et une réponse légèrement oscillatoire. Le gain  $\epsilon_\kappa$  peut être ajusté de manière pratique afin d'obtenir un compromis entre l'action de lissage (obtenue en augmentant  $\epsilon_\kappa$ ) et la réduction du retard temporel de l'estimateur (obtenue en réduisant  $\epsilon_\kappa$ ). En outre, l'action de lissage de l'approche proposée atténue l'effet de la quantification de la mesure angulaire. Nous avons choisi  $\epsilon_\kappa = 0,05$  pour nos expériences. La Figure 6.4 montre les résultats expérimentaux obtenus après la mise en œuvre du filtre grand gain (6.9) dans le cas d'un vol générant des oscillations angulaires de grande amplitude.

Nous avons effectué une dérivation par différence finie (en vert) en post-traitement pour comparer les résultats. En raison de la nature quantifiée de l'encodage rotatif, nous observons que la vitesse angulaire obtenue par différence finie est très bruitée. On constate que le filtre grand gain permet d'estimer la vitesse angulaire avec plus de précision (en rouge), bien qu'avec un léger retard. Grâce à l'ajout d'une IMU supplémentaire sur l'aile lors d'un essai en vol, il est possible de comparer l'estimation de la vitesse avec les mesures du gyroscope de l'aile (MPU9250), visibles sur le graphique du bas de la Figure 6.4 (trace bleue). On constate que les mesures du gyroscope sont quelque peu bruitées, notamment en raison des vibrations générées par les moteurs.

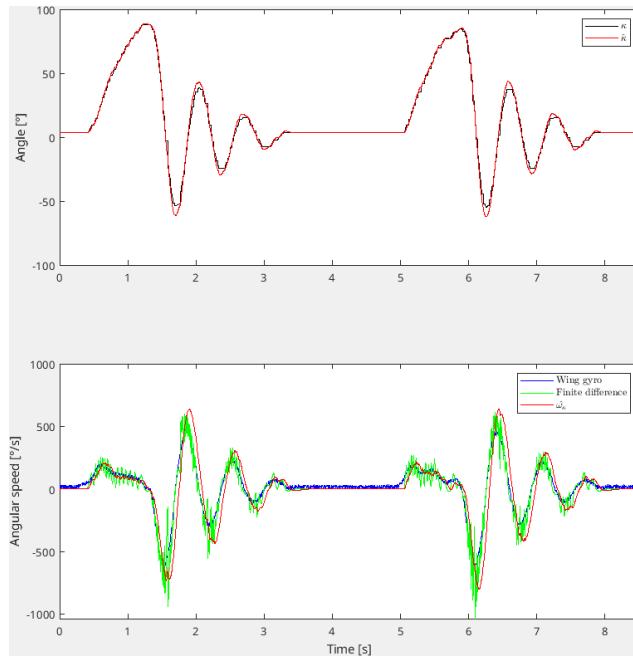


FIGURE 6.4 – Mesure de l'angle (noir, graphique du haut), mesure de la vitesse angulaire avec le gyroscope de l'aile (bleu, graphique du bas), estimation de la vitesse angulaire par différence finie (vert, graphique du bas) et estimation avec le filtre grand gain (courbes rouges).

Afin d'effectuer la transformation nécessaire entre les repères, nous définissons le quaternion  $q_{\hat{\kappa}} \in \mathbb{S}^3$  suivant :

$$q_{\hat{\kappa}} = \left[ \cos\left(\frac{\hat{\kappa}}{2}\right) \ 0 \ \sin\left(\frac{\hat{\kappa}}{2}\right) \ 0 \right]^T \quad (6.10)$$

Ce n'est pas très lisible comme figure

### 6.3.3 Estimation de l'état de l'aile

Sur la base de l'angle estimé  $\hat{\kappa}$  et de la vitesse angulaire estimée  $\hat{\omega}_\kappa$ , il est possible de transformer les mesures réalisées dans le repère du fuselage vers le repère de l'aile. Tous les capteurs sont installés sur la carte de l'autopilote, qui est elle-même fixée au fuselage. Cependant, nous voulons utiliser INDI pour stabiliser l'aile. Cette loi de contrôle nécessite donc des informations sur l'état du drone dans repère de l'aile, où toutes les forces sont appliquées (aérodynamiques et de traction). Deux solutions viables sont alors possibles : effectuer l'estimation d'état dans le repère du fuselage et faire pivoter l'estimation, en utilisant l'estimation de l'angle  $\hat{\kappa}$ , ou faire pivoter les mesures brutes à l'avance pour les exprimer dans le repère l'aile, et ensuite effectuer l'estimation d'état dans ce dernier. Compte tenu de l'architecture actuelle du logiciel du système Paparazzi (voir Annexe A.1.4), il est impossible d'avoir deux structures conjointes d'estimation de l'état, et il est donc complexe de mettre en œuvre la première solution, dans laquelle le contrôleur récupère directement l'estimation de l'état actuel. C'est pourquoi nous avons choisi d'estimer l'état de l'aile à partir des données mesurées sur le fuselage. À cette fin, nous détaillons ci-dessous les transformations pour les trois capteurs : gyroscope, accéléromètre et magnétomètre.

Pour les mesures de la vitesse angulaire basées sur le gyroscope, nous pouvons calculer la vitesse angulaire de l'aile exprimée dans le repère de l'aile :

$$\omega_W = R(q_{\hat{\kappa}}) \left( \omega_{gyro}^F + \begin{bmatrix} 0 \\ \hat{\omega}_\kappa \\ 0 \end{bmatrix} \right) \quad (6.11)$$

où  $\omega_{gyro}^F$  est la vitesse angulaire mesurée par le gyroscope sur le fuselage, exprimée dans le repère du fuselage,  $\hat{\omega}_\kappa$  est la vitesse angulaire estimée de l'aile par rapport au fuselage, conformément à (6.9), et  $q_{\hat{\kappa}}$  est le quaternion défini dans (6.10). L'expression (6.11) est similaire à une composition de vitesses angulaires et à un changement du repère.

Pour la mesure de l'accélération avec l'accéléromètre, nous pouvons utiliser la relation de l'équation (6.12), qui est obtenue à partir du théorème de transport du taux de variation [Brizard 2004], où nous trouvons le terme d'accélération d'Euler  $\omega_F \times d_{AF}$  et le terme d'accélération centripète  $\omega_F \times (\omega_F \times d_{AF})$ . L'accélération de Coriolis  $2\omega_F \times \frac{d(d_{FW})}{dt} \Big|_{O_F}$  et le taux d'accélération  $\frac{d^2(d_{FW})}{d^2t} \Big|_{O_F}$  sont nuls car  $d_{FW}$  est constant.

$$a_W = R(q_{\hat{\kappa}}) \left( a_{acc}^F + \dot{\omega}_{gyro}^F \times d_{FW} + \omega_{gyro}^F \times (\omega_{gyro}^F \times d_{FW}) \right) \quad (6.12)$$

où  $a_{acc}^F \in \mathbb{R}^3$  est l'accélération mesurée par l'accéléromètre sur le fuselage, exprimée dans le repère du fuselage et  $\omega_{gyro}^F$  est la vitesse angulaire du fuselage, identique à l'équation (6.11). L'accélération angulaire  $\dot{\omega}_{gyro}^F$  dans (6.12) est calculée par différence finie.

Pour les mesures du magnétomètre, nous avons

$$E_W = R(q_{\hat{\kappa}}) E_{mag} \quad (6.13)$$

où  $E_{mag} \in \mathbb{R}^3$  est la sortie du magnétomètre, exprimée dans le repère du fuselage et  $E_W \in \mathbb{R}^3$  est la mesure exprimée dans le repère de l'aile.

Pour obtenir une estimation de l'état de l'aile, nous utilisons un algorithme de fusion des mesures des capteurs : le filtre de Kalman étendu (EKF) qui fournit une estimation des états suivants :  $p_W$ ,  $v_W$ ,  $q_W$  à partir de mesures transformées dans le repère de l'aile  $\omega_W$  (équation (6.11)),  $a_W$  (équation (6.12)),  $E_W$  (équation (6.13)) et les données du système de vision externe, qui fournissent une mesure précise de la position  $p_{text{W}}$  et de la vitesse  $v_{text{W}}$  du drone dans le repère inertiel (I).

### 6.3.4 Estimation de l'orientation du fuselage

Pour déterminer l'orientation du fuselage, nous pouvons effectuer une composition entre le quaternion représentant l'orientation de l'aile  $q_{text{W}}$  résultat de l'EKF et le quaternion construit à partir de la mesure filtrée de l'encodeur rotatif  $q_{\hat{\kappa}}$  dans (6.10),

$$q_F = q_W \otimes q_{\hat{\kappa}} \quad (6.14)$$

où l'opérateur  $\otimes$  désigne le produit hamiltonien. La connaissance de  $q_F$  est nécessaire pour maintenir le fuselage parfaitement horizontal.

## 6.4 Inversion non linéaire incrémentale de la dynamique du drone

Détails

La théorie de l'inversion dynamique non linéaire incrémentale (INDI) utilisée dans le contexte des micro-drones est présentée dans [Smeur 2016]. Nous utilisons

la notation proposée dans [Smeur 2020]. L'hypothèse centrale sous-jacente est que le principe de séparation des échelles de temps s'applique à la dynamique de l'actionneur et à la dynamique des forces et des moments aérodynamiques. Le signal de commande peut alors être calculé de manière incrémentale en utilisant la matrice d'efficacité de l'actionneur  $G$ .

$$u_W = u_W + G^\dagger (\nu - \begin{bmatrix} \dot{\omega}_W \\ T_W \end{bmatrix}) \quad (6.15)$$

où  $\dot{\omega}_W \in \mathbb{R}^3$  est l'accélération angulaire obtenue par différence finie à partir de l'équation (6.11),  $T_W \in \mathbb{R}$  est la poussée actuelle,  $\nu$  est défini dans [Smeur 2020, équation (4)] et  $G$  est la matrice d'efficacité du contrôle, défini par :

$$\begin{bmatrix} \partial\phi \\ \partial\theta \\ \partial\psi \\ \partial T \end{bmatrix} = Gu_f = \begin{bmatrix} -7.5 & -15 & 7.5 & 15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 15 & 15 \\ 0 & 0 & 0 & 0 & 4 & -4 \\ -0.6 & -0.6 & -0.6 & -0.6 & 0 & 0 \end{bmatrix} u_f$$

Cette sélection de la matrice d'efficacité a été déterminée pour les vols stationnaires, mais il est nécessaire d'effectuer une étude différente pour le vol vers l'avant.

Pour stabiliser le fuselage, nous utilisons un bouclage proportionnel-dérivé de l'angle  $\theta_F$  formé entre le fuselage et l'horizontale, que nous voulons maintenir à zéro. Cet angle est obtenu en convertissant le quaternion  $q_F$  de l'équation (6.14) en un angle d'Euler en suivant la convention d'Euler 'ZYX'. Le bouclage fournit la commande  $u_{tail}$  pour la vitesse angulaire du moteur générant la force  $F_m$  (voir Figure 6.2), suivant l'expression :

$$u_{tail} = u_{eq} + k_p \theta_F + k_d \dot{\theta}_F,$$

où  $u_{eq}$  est la commande, à l'équilibre, du moteur pour maintenir le fuselage horizontal en l'absence de perturbation et  $k_p$ ,  $k_d$  sont des gains scalaires ajustables. La valeur  $u_{eq}$  a été obtenue en appliquant le théorème des moments au fuselage au point  $O_W$ . En effet, les deux moments qui agissent sur le fuselage sont le couple dû à la force de poussée du moteur et le couple dû à la position du centre de gravité du fuselage.

plier les mo-  
nts

Les gains  $k_p$  et  $k_d$  ont été ajustés en vol pour assurer un comportement satisfaisant. On obtient  $\dot{\theta}_F$  à partir de  $\omega_{gyro}^F = [\dot{\phi}_F \ \dot{\theta}_F \ \dot{\psi}_F]^\top$ .

## 6.5 Expérimentations

Un prototype a été mis au point, comme le montre la Figure 6.5. La Figure 6.6 présente une sélection des résultats expérimentaux en environnement contrôlé.

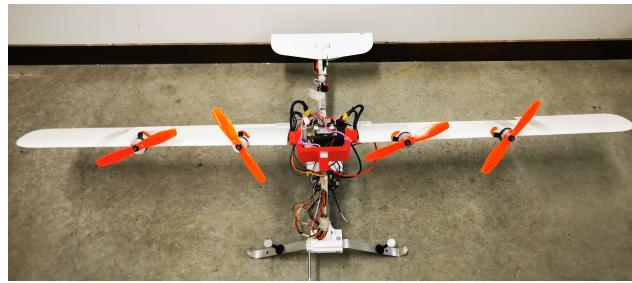


FIGURE 6.5 – Prototype : Colibri.

Sur la Figure 6.6, de 0 s à 8 s, le drone est au sol. De 8 s à 16 s, le drone décolle pour atteindre une hauteur de 2 mètres visible sur le troisième graphique. Cette hauteur est atteinte après un dépassement de 10 %. Le drone est maintenu dans cette position pendant 54 s. Des oscillations d'incidence sont observées dans le cinquième et le dernier graphique, générant des oscillations dans la position horizontale du drone. Ce phénomène est dû au couplage entre les deux corps, qui n'est pas correctement stabilisé. À partir de 70 s, le drone commence à se diriger vers le point  $p_c = [3 \ 0.9 \ -1.5]^\top$  et  $\psi_c = 90^\circ$ .

## 6.6 Vol avec un contrôleur unifié

## 6.7 Commande Udwadia-Kalaba

Mesure du vent, sonde 5 trous

## 6.8 Vols expérimentaux

## 6.9 Conclusion du Chapitre 6

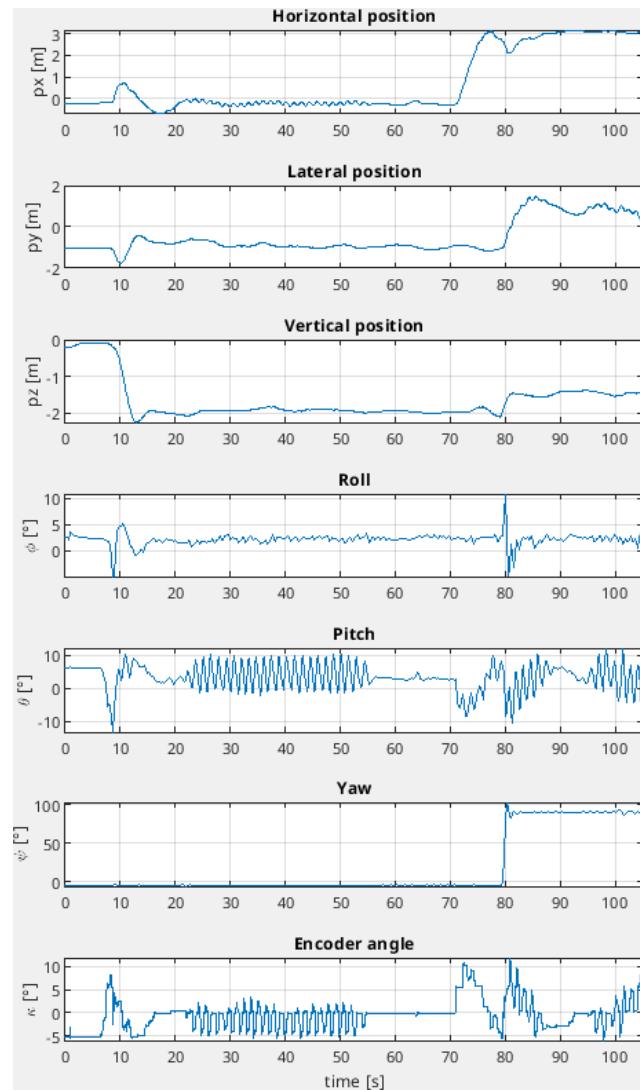


FIGURE 6.6 – Position et orientation de l'aile dans les six premiers graphiques et mesure de l'angle entre l'aile et le fuselage sur le dernier graphique lors d'un vol réel.

# **Conclusion**

Répondre aux objectifs de l'introduction qui engendre les ouvertures



## ANNEXE A

# Annexe technique sur les drones

---

## A.1 Système de drone : Paparazzi

Un drone est composé de plusieurs pièces assemblées entre elles pour former la structure sur laquelle sont fixés des actionneurs, un autopilote et une charge utile (colis, caméra, capteur, etc.). L'élément central est l'autopilote qui assure la communication entre tous les éléments. Nous pouvons décomposer l'autopilote en deux parties : la partie matérielle et la partie logicielle.

La partie matérielle est constituée d'un circuit imprimé (PCB) sur lequel des composants sont installés pour assurer les tâches relatives au vol. Ainsi, nous pouvons détailler les capteurs embarqués et le microcontrôleur avec l'ensemble de ses ports de communication A.1.1 et A.1.2. Et la partie logicielle qui se scinde en deux éléments qui sont : le segment sol et le logiciel embarqué A.1.4.

### A.1.1 Les capteurs d'un autopilote

Un autopilote comporte généralement un accéléromètre, un gyroscope, un magnétomètre et un baromètre.

**L'accéléromètre** à trois axes permet de mesurer l'ensemble des forces appliquées sur le véhicule à l'exception du poids. Il est possible d'obtenir la position du drone par double intégration de la mesure de l'accéléromètre. Toutefois, il convient de souligner que la position dérive rapidement en raison des bruits de mesure.

**Le gyroscope** à trois axes permet de mesurer les vitesses de rotation du véhicule. Il est possible d'obtenir l'orientation du drone par intégration de la mesure du gyroscope. Toutefois, comme précédemment, l'orientation dérive rapidement en raison des bruits de mesure.

**Le magnétomètre** à trois axes indique la direction du nord magnétique. Il permet de se diriger par rapport à une référence connue. Le principal inconvénient de

ce capteur est sa perturbation par les masses magnétiques environnantes, ainsi que par les champs magnétiques parasites induits par la proximité des moteurs électriques par exemple. Il est donc difficile de les utiliser à l'intérieur d'un bâtiment. L'influence magnétique de l'engin porteur et les perturbations dues à d'éventuels moteurs électriques peuvent être éliminées en qualifiant, de manière statique, les erreurs dues aux masses métalliques du véhicule et aux moteurs électriques (en fonction des tensions et courants d'alimentation).

**Le baromètre** est un capteur d'altitude basée sur la mesure de la pression atmosphérique.

Il est courant de retrouver plusieurs capteurs dans un même boitier, que l'on nomme centrale inertie (Inertial Measurement Units, IMU), . Ces dernières sont composées au minimum d'un accéléromètre 3-axes et d'un gyroscope 3-axes, mais il est courant de les trouver avec un magnétomètre 3-axes.

**Le GPS** est monté en extérieur de l'autopilote. Ce système de géo-positionnement par satellite (*Global Positioning System, GPS*) permet d'obtenir un positionnement absolu du drone.

### A.1.2 Le microcontrôleur d'un autopilote

Le microcontrôleur (Microcontroller Unit, MCU) est la pièce maîtresse de l'autopilote en ce qu'elle permet d'effectuer l'ensemble des traitements nécessaires à la conduite du vol.

De plus il possède plusieurs ports de communication pour récupérer les données de capteur ou envoyer des ordres aux actionneurs.

### A.1.3 Évolutions

Les nombreux progrès dans les systèmes d'estimation état permettent de connaître précisément l'orientation et la position des drones pour assurer la stabilisation, le guidage et la navigation. Les progrès sont liés à l'amélioration continue des capteurs, notamment des centrales inertielles constituées d'un accéléromètre, d'un gyroscope et d'un magnétomètre.

La Table A.1 montre l'évolution des vitesses des microcontrôleurs (Microcontroller Unit, MCU) embarqués sur les autopilotes et de la réduction du bruit des capteurs inertIELS.

Type	Date	MCU	Vitesse	Capteur	Bruit RMS
Apogee	2013	STM32F4	168 MHz	MPU-9150	Gyro : 0.06 dps Accel : 4 mg
Chimera	2016	STM32F7	216 MHz	MPU-9250	Gyro : 0.1 dps Accel : 8 mg
Tawaki 1	2019	STM32F7	216 MHz	ICM-20600	Gyro : 0.04 dps Accel : 1 mg
Tawaki 2	2023	STM32H7	480 MHz	ICM-42688-P	Gyro : 0.028 dps Accel : 0.70 mg

TABLE A.1 – Évolution des autopilotes paparazzi sur dix ans.

Sur une période de dix ans, nous pouvons observer que les microcontrôleurs ont doublé leur vitesse d'exécution, que les fabricants ont divisé par deux le bruit moyen sur les gyroscopes et par quatre le bruit moyen des accéléromètres. Ces évolutions continues permettent une amélioration de l'estimation du drone utilisée pour la stabilisation. Il en résulte une stabilité accrue et de nouvelles possibilités pour la commande des drones.

#### A.1.4 Les logiciels d'un autopilote

Tout le fonctionnement d'un drone repose sur le logiciel qui permet de le faire voler. Il se décompose en deux catégories : la partie sol et la partie embarquée.

#### A.1.5 Le segment sol

#### A.1.6 Le logiciel embarqué

Le logiciel embarqué  
fusion de donnée estimation d'état

## A.2 AM32

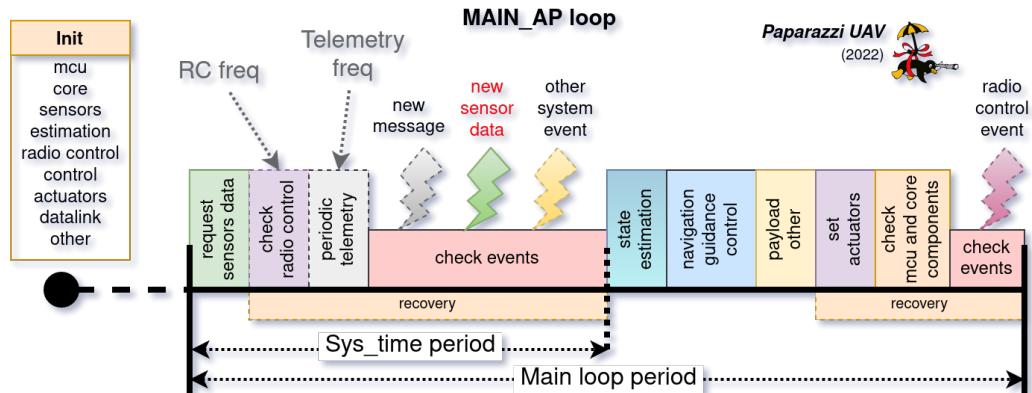


FIGURE A.1 – Todo.

# Bibliographie

- [Andreetto 2016] Andreetto, M., Fontanelli, D. et Zaccarian, L. *Quasi time-optimal hybrid trajectory tracking of an n-dimensional saturated double integrator.* Dans IEEE Conference on Control Applications, pages 550–555, 2016. (Cité en page 51.)
- [Apkarian 2006] Apkarian, P. et Noll, D. *Nonsmooth  $H_\infty$  Synthesis.* IEEE Transactions on Automatic Control, vol. 51, no. 1, pages 71–86, 2006. [En ligne]. Disponible : <http://dx.doi.org/10.1109/TAC.2005.860290>. (Cité en pages 65, 73 et 76.)
- [Apkarian 2013] Apkarian, P. *Tuning controllers against multiple design requirements.* Dans 2013 American Control Conference, pages 3888–3893, 2013. (Cité en page 76.)
- [Aref 2018] Aref, P., Ghoreyshi, M., Jirasek, A., Satchell, M. J. et Bergeon, K. *Computational Study of Propeller-Wing Aerodynamic Interaction.* Aerospace, vol. 5, no. 3, 2018. [En ligne]. Disponible : <http://dx.doi.org/10.3390/aerospace5030079>. (Cité en page 3.)
- [Avy 2023] Avy. 2023. [En ligne]. Disponible : <https://avy.eu/>. (Cité en pages vii et 9.)
- [Axtén 2021] Axtén, R. et Johnson, E. *Vtol freewing testbed vehicle in hover and forward flight.* Dans AIAA Scitech 2021 Forum, AIAA Scitech 2021 Forum, pages 1–19. American Institute of Aeronautics and Astronautics Inc, AIAA, 2021. Publisher Copyright : © 2021, American Institute of Aeronautics and Astronautics Inc, AIAA. All rights reserved. ; AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2021 ; Conference date : 11-01-2021 Through 15-01-2021. (Cité en page 11.)
- [Axtén 2023] Axtén, R. M., Khamvilai, T. et Johnson, E. N. *VTOL Freewing Design and Adaptive Controller Development.* Dans AIAA SCITECH 2023 Forum, 2023. (Cité en page 11.)
- [Brizard 2004] Brizard, A. J. *Motion in a non-inertial frame.* Saint Michael's College, Colchester, VT, 2004. (Cité en page 94.)
- [Chiappinelli 2018] Chiappinelli, R. et Nahon, M. *Modeling and Control of a Tail-sitter UAV.* Dans 2018 International Conference on Unmanned Aircraft Systems (ICUAS), pages 400–409, 2018. (Cité en page 3.)

- [Chitour 2002] Chitour, Y. *Time-varying high-gain observers for numerical differentiation.* IEEE Transactions on Automatic Control, vol. 47, no. 9, pages 1565–1569, 2002. [En ligne]. Disponible : <http://dx.doi.org/10.1109/TAC.2002.802740>. (Cité en page 92.)
- [Droandi 2015] Droandi, G., Zanotti, A., Gibertini, G., Grassi, D. et Campanardi, G. *Experimental investigation of the rotor-wing aerodynamic interaction in a tiltwing aircraft in hover.* The Aeronautical Journal, vol. 119, no. 1215, page 591–612, 2015. [En ligne]. Disponible : <http://dx.doi.org/10.1017/S0001924000010708>. (Cité en page 3.)
- [Ducard 2014] Ducard, G. et Hua, M.-D. *Modeling of an unmanned hybrid aerial vehicle.* Dans 2014 IEEE Conference on Control Applications (CCA), pages 1011–1016, 2014. (Cité en page 13.)
- [Ducard 2021] Ducard, G. J. J. et Allenspach, M. *Review of designs and flight control techniques of hybrid and convertible VTOL UAVs.* Aerospace Science and Technology, vol. 118, page 107035, November 2021. [En ligne]. Disponible : <http://dx.doi.org/10.1016/j.ast.2021.107035>. (Cité en page 8.)
- [Dufour 2024] Dufour. 2024. [En ligne]. Disponible : <https://www.dufour.aero/aero2>. (Cité en pages vii et 11.)
- [Escareno 2006] Escareno, J., Salazar, S. et Lozano, R. *Modelling and Control of a Convertible VTOL Aircraft.* Dans Proceedings of the 45th IEEE Conference on Decision and Control, pages 69–74, 2006. (Cité en page 3.)
- [Escareno 2007] Escareno, J., Stone, R., Sanchez, A. et Lozano, R. *Modeling and control strategy for the transition of a convertible tail-sitter UAV.* Dans European Control Conference, pages 3385–3390, 2007. (Cité en page 3.)
- [Fernandez 2023a] Fernandez, L. F., Bronz, M., Bartoli, N. et Lefebvre, T. *Assessment of Methods for Propeller Performance Calculation at High Incidence Angles.* Dans AIAA SCITECH 2023 Forum, 2023. (Cité en page 3.)
- [Fernandez 2023b] Fernandez, L. F. T., Bronz, M., Bartoli, N. et Lefebvre, T. *Development of a Mission-Tailored Tail-Sitter MAV.* Unmanned systems, vol. 12, no. 03, pages 611–625, November 2023. [En ligne]. Disponible : <https://hal.science/hal-04612206>. (Cité en pages vii et 10.)
- [Flores 2014] Flores, G., Lugo, I. et Lozano, R. *6-DOF hovering controller design of the Quad Tiltrotor aircraft : Simulations and experiments.* Dans 53rd IEEE Conference on Decision and Control, pages 6123–6128, 2014. (Cité en pages vii et 10.)

- [Franchi 2017] Franchi, A. et Mallet, A. *Adaptive Closed-loop Speed Control of BLDC Motors with Applications to Multi-rotor Aerial Vehicles*. Dans IEEE International Conference on Robotics and Automation, Singapour, Singapore, May 2017. (Cité en page 81.)
- [Goebel 2012] Goebel, R., Sanfelice, R. G. et Teel, A. R. Hybrid dynamical systems : Modeling, stability, and robustness. Princeton University Press, New Jersey, 2012. (Cité en pages 51 et 54.)
- [Guerrero 2009] Guerrero, J. A., Lozano, R., Romero, G., Lara-Alabajares, D. et Wong, K. C. *Robust control design based on sliding mode control for hover flight of a mini tail-sitter Unmanned Aerial Vehicle*. Dans 2009 35th Annual Conference of IEEE Industrial Electronics, pages 2342–2347, 2009. (Cité en page 3.)
- [Hua 2013] Hua, M.-D., Hamel, T., Morin, P. et Samson, C. *Introduction to feedback control of underactuated VTOL vehicles : A review of basic control design ideas and principles*. IEEE Control Systems, vol. 33, pages 61–75, February 2013. [En ligne]. Disponible : <http://dx.doi.org/10.1109/MCS.2012.2225931>. (Cité en pages 21, 22, 42 et 48.)
- [Jardin 2007] Jardin, M. et Mueller, E. *Optimized Measurements of UAV Mass Moment of Inertia with a Bifilar Pendulum*. Dans AIAA Guidance, Navigation and Control Conference and Exhibit, 2007. (Cité en page 26.)
- [Ke 2017a] Ke, Y. et Chen, B. M. *Full envelope dynamics modeling and simulation for tail-sitter hybrid UAVs*. Dans 2017 36th Chinese Control Conference (CCC), pages 2242–2247, 2017. (Cité en page 3.)
- [Ke 2017b] Ke, Y., Wang, K., Gong, K., Lai, S. et Chen, B. M. *Model based robust forward transition control for tail-sitter hybrid unmanned aerial vehicles*. Dans 2017 13th IEEE International Conference on Control and Automation (ICCA), pages 828–833, 2017. (Cité en page 2.)
- [LabJack 2001] LabJack. *T7*. 2001. [En ligne]. Disponible : <https://labjack.com/products/t7>. Accessed : 2022-09-19. (Cité en page 60.)
- [Lustosa 2015] Lustosa, L., Defaÿ, F. et Moschetta, J. *Longitudinal study of a tilt-body vehicle : modeling, control and stability analysis*. Dans Proc. of International Conference on Unmanned Aircraft Systems, pages 816–824, Denver, Colorado, US, June 2015. (Cité en page 2.)
- [Lustosa 2017] Lustosa, L. R. *La Phi-théorie : une approche pour la conception de lois de commande de vol des véhicules convertibles*. PhD thesis, Toulouse,

- ISAE, 2017. Thèse de doctorat dirigée par Moschetta, Jean-Marc et Defaÿ, François Automatique, Dynamique des fluides Toulouse, ISAE 2017. (Cité en page 13.)
- [Lustosa 2019] Lustosa, L. R., Defaÿ, F. et Moschetta, J.-M. *Global Singularity-Free Aerodynamic Model for Algorithmic Flight Control of Tail Sitters*. Journal of Guidance, Control, and Dynamics, vol. 42, no. 2, pages 303–316, February 2019. [En ligne]. Disponible : <http://dx.doi.org/10.2514/1.G003374>. (Cité en pages 3, 4, 13, 20, 21, 22, 23, 24 et 88.)
- [Michieletto 2020] Michieletto, G., Cenedese, A., Zaccarian, L. et Franchi, A. *Hierarchical nonlinear control for multi-rotor asymptotic stabilization based on zero-moment direction*. Automatica, vol. 117, page 108991, 2020. (Cité en pages 47, 48 et 49.)
- [Misra 2022] Misra, A., Jayachandran, S., Kenche, S., Katoch, A., Suresh, A., Gundabattini, E., Selvaraj, S. K. et Legesse, A. A. *A Review on Vertical Take-Off and Landing (VTOL) Tilt-Rotor and Tilt Wing Unmanned Aerial Vehicles (UAVs)*. Journal of Engineering, vol. 2022, no. 1, page 1803638, 2022. [En ligne]. Disponible : <https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/1803638>. (Cité en page 8.)
- [Nicosia 1990] Nicosia, S., Tornambe, A. et Valigi, P. *Experimental results in state estimation of industrial robots*. Dans 29th IEEE Conference on Decision and Control, pages 360–365 vol.1, 1990. (Cité en page 92.)
- [Olszanecki Barth 2020] Olszanecki Barth, J. M., Condomines, J.-P., Bronz, M., Moschetta, J.-M., Join, C. et Fliess, M. *Model-free control algorithms for micro air vehicles with transitioning flight capabilities*. International Journal of Micro Air Vehicles, vol. 12, pages 1–22, April 2020. [En ligne]. Disponible : <http://dx.doi.org/10.1177/1756829320914264>. (Cité en pages 4, 21, 24, 26 et 57.)
- [Ostermann 2012] Ostermann, T., Holsten, J., Dobrev, Y. et Moormann, D. *Control concept of a tiltwing uav during low speed manoeuvring*. Dans Proceeding of the 28th International Congress of the Aeronautical Sciences : ICAS Brisbane, Australia, volume 1190, 2012. (Cité en pages vii et 11.)
- [Pfimlin 2006] Pfimlin, J.-M. *Commande d'un minidrone à hélice carénée : de la stabilisation dans le vent à la navigation autonome*. Theses, SUPAERO, November 2006. (Cité en pages vii et 10.)
- [Rohr 2021] Rohr, D., Studiger, M., Stastny, T., Lawrance, N. R. J. et Siegwart, R. *Nonlinear Model Predictive Velocity Control of a VTOL Tiltwing UAV*.

- IEEE Robotics and Automation Letters, vol. 6, no. 3, pages 5776–5783, 2021.  
[En ligne]. Disponible : <http://dx.doi.org/10.1109/LRA.2021.3084888>. (Cité en page 13.)
- [Saeed 2018] Saeed, A. S., Younes, A. B., Cai, C. et Cai, G. *A survey of hybrid Unmanned Aerial Vehicles*. Progress in Aerospace Sciences, vol. 98, pages 91–105, April 2018. [En ligne]. Disponible : <http://dx.doi.org/10.1016/j.paerosci.2018.03.007>. (Cité en page 8.)
- [Sanfelice 2013] Sanfelice, R., Copp, D. et Nanez, P. *A toolbox for simulation of hybrid systems in Matlab/Simulink : Hybrid Equations (HyEQ) Toolbox*, 2013. (Cité en page 54.)
- [Sansou 2022a] Sansou, F. *Commande hybride d'un drone convertible pour des déplacements sous optimaux*. Master Thesis. ENAC Toulouse, 2022. [En ligne]. Disponible : <http://dx.doi.org/10.48550/ARXIV.2203.15387>. (Cité en pages 4, 21, 23, 24 et 26.)
- [Sansou 2022b] Sansou, F., Demourant, F., Hattenberger, G., Loquen, T. et Zaccarian, L. *Open wind tunnel experiments of the DarkO tail-sitter longitudinal stabilization with constant wind*. IFAC-PapersOnLine, vol. 55, no. 22, pages 1–6, 2022. [En ligne]. Disponible : <http://dx.doi.org/https://doi.org/10.1016/j.ifacol.2023.03.001>. 22nd IFAC Symposium on Automatic Control in Aerospace ACA 2022. (Cité en page 4.)
- [Sansou 2022c] Sansou, F. et Zaccarian, L. *On local-global hysteresis-based hovering stabilization of the DarkO convertible UAV*. Dans 2022 European Control Conference (ECC), pages 40–45, 2022. (Cité en page 4.)
- [Sansou 2024a] Sansou, F., Demourant, F., Hattenberger, G., Loquen, T. et Zaccarian, L. *Hovering stabilization of the DarkO tail-sitter drone with constant wind*. IEEE Transactions on Control Systems Technology, 2024. (Cité en page 4.)
- [Sansou 2024b] Sansou, F., Hattenberger, G., Zaccarian, L., Demourant, F. et Loquen, T. *Modelling and Hovering Stabilisation of a Free-Rotating Wing UAV*. Dans 2024 International Conference on Unmanned Aircraft Systems (ICUAS), pages 779–785, La Canée (Crete), Greece, June 2024. IEEE. (Cité en page 4.)
- [Simmons 2022] Simmons, B. M. et Murphy, P. C. *Aero-Propulsive Modeling for Tilt-Wing, Distributed Propulsion Aircraft Using Wind Tunnel Data*. Journal of Aircraft, vol. 59, no. 5, pages 1162–1178, 2022. (Cité en page 3.)

- [Smeur 2016] Smeur, E. J. J., Chu, Q. et de Croon, G. C. H. E. *Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles*. Journal of Guidance, Control, and Dynamics, vol. 39, no. 3, pages 450–461, 2016. [En ligne]. Disponible : <https://doi.org/10.2514/1.G001490>. (Cité en page 95.)
- [Smeur 2020] Smeur, E. J. J., Bronz, M. et de Croon, G. C. H. E. *Incremental Control and Guidance of Hybrid Aircraft Applied to a Tailsitter Unmanned Air Vehicle*. Journal of Guidance, Control, and Dynamics, vol. 43, no. 2, pages 274–287, 2020. [En ligne]. Disponible : <https://doi.org/10.2514/1.G004520>. (Cité en pages vii, 10 et 96.)
- [Stone 2002] Stone, H. et Wong, K. C. *Preliminary Design of a Tandem-Wing Tail-Sitter UAV Using Multi-Disciplinary Design Optimisation*. Dans International Aerospace Congress, 2002. (Cité en page 2.)
- [Stone 2008] Stone, R. H., Anderson, P., Hutchison, C., Tsai, A., Gibbens, P. et Wong, K. C. *Flight Testing of the T-Wing Tail-Sitter Unmanned Air Vehicle*. Journal of Aircraft, vol. 45, no. 2, pages 673–685, 2008. [En ligne]. Disponible : <http://dx.doi.org/10.2514/1.32750>. (Cité en page 2.)
- [Tal 2022] Tal, E. et Karaman, S. *Global Incremental Flight Control for Agile Maneuvering of a Tailsitter Flying Wing*. arXiv preprint arXiv :2207.13218, 2022. (Cité en page 3.)
- [Tempo 2013] Tempo, R., Calafiore, G. et Dabbene, F. Randomized algorithms for analysis and control of uncertain systems : with applications. Springer, 2013. (Cité en page 51.)
- [Tregouet 2015] Tregouet, J.-F., Arzelier, D., Peaucelle, D., Pittet, C. et Zaccarian, L. *Reaction Wheels Desaturation Using Magnetorquers and Static Input Allocation*. IEEE Transactions on Control Systems Technology, vol. 23, no. 2, pages 525–539, February 2015. [En ligne]. Disponible : <http://dx.doi.org/10.1109/TCST.2014.2326037>. (Cité en page 39.)
- [Udwadia 2006] Udwadia, F. E. et Phohomsiri, P. *Explicit Equations of Motion for Constrained Mechanical Systems with Singular Mass Matrices and Applications to Multi-Body Dynamics*. Proceedings : Mathematical, Physical and Engineering Sciences, vol. 462, no. 2071, pages 2097–2117, 2006. [En ligne]. Disponible : <http://www.jstor.org/stable/20208995>. (Cité en page 86.)
- [Udwadia 2012] Udwadia, F. E. et Schutte, A. D. *A unified approach to rigid body rotational dynamics and control*. Proceedings : Mathematical, Physical and

- Engineering Sciences, vol. 468, no. 2138, pages 395–414, 2012. [En ligne]. Disponible : <http://www.jstor.org/stable/41345881>. (Cité en pages 86 et 87.)
- [Welstead 2012] Welstead, J. et Crouse, G. *A Segmented Freewing Concept for UAS Gust Alleviation in Adverse Environments*. Dans AIAA Infotech@Aerospace Conference, 2012. (Cité en page 11.)
- [Zhang 2017] Zhang, S., Fei, Q., Liang, J. et Geng, Q. *Modeling and control for longitudinal attitude of a twin-rotor tail-sitter unmanned aerial vehicle*. Dans 2017 13th IEEE International Conference on Control and Automation (ICCA), pages 816–821, 2017. (Cité en page 3.)



---

**Résumé :** resume

**Mots clés :** mots, clefs

---

---

**Abstract :** abstrat

**Keywords :** key, words

---

