

---

---

# Commande d'un drone de type tilt-rotor

---

---

*Étudiant :*

Idriss CHELIKH

Matis PERAUDEAU

*Numéros étudiant :*

3704957

3672030

Professeur encadrant : Pascal Morin

Juillet 2020

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>État de l’art</b>	<b>3</b>
2.1	Quadcoptère classique . . . . .	3
2.2	Modélisation d’un quadcoptère . . . . .	4
2.3	Quadcoptère tilt-rotor . . . . .	7
<b>3</b>	<b>Étude d’un simulateur de quadcoptère existant</b>	<b>9</b>
3.1	Étude du schéma de commande . . . . .	9
3.2	Résultats et discussion . . . . .	11
<b>4</b>	<b>Développement d’un simulateur pour quadcoptère</b>	<b>12</b>
4.1	Quadcoptère sans tilt . . . . .	12
4.2	Modélisation d’un quadcoptère avec tilt rotors . . . . .	17
4.3	Stratégie de détermination des valeurs de tilt . . . . .	18
4.3.1	Résultats sur le tilt en tangage . . . . .	19
4.4	Stratégie de détermination des tilts avec lacet . . . . .	19
<b>5</b>	<b>Conclusion</b>	<b>21</b>
<b>6</b>	<b>Bibliographie</b>	<b>22</b>

# 1 Introduction

Les quadcoptères sont depuis quelques années l'un des modèles de drone le plus couramment utilisé dans le domaine civil comme militaire. Ils sont simples à produire, facile à contrôler et peuvent se stabiliser à une altitude désirée. Ceci les rends parfaits pour la surveillance ou l'exploration de zones dangereuses voir sinistrées dans le cadre d'opérations militaires. Dans le cadre du civil ils sont utilisés pour filmer depuis le ciel ou simplement pour le loisir. Récemment de nouvelles architectures de quadcoptères sont apparues, notamment le modèle "tilt-rotor" qui consiste à diriger le drone en inclinant ses rotors. Cette forme plus complexe de quadcoptère offre de nouvelles possibilités de commande et de contrôle de vol. Les figures ci-dessous présentent un quadcoptère classique à gauche et un quadcoptère tilt-rotor à droite, tout deux du domaine civil et disponible sur le marché. Il est possible d'observer sur le drone tilt-rotor que les hélices s'inclinent vers l'avant ou l'arrière du drone tandis que les hélices du quadcoptère classique sont fixes.



FIGURE 1 – Drone quadcoptère classique



FIGURE 2 – Drone quadcoptère tilt-rotor

L'objectif de cette étude est d'étudier les quadcoptères tilt-rotor et de développer des stratégies de commande pour son comportement en vol. Afin d'atteindre cet objectif, l'étude réalise un état de l'art des quadcoptères classiques et tilt-rotor en couvrant la partie mathématique de la modélisation de ces drones. Dans un second temps, il est proposé d'étudier un simulateur de quadcoptère existant pour avoir une idée des possibilités à explorer et des stratégies mises en place pour la modélisation d'un quadcoptère. Enfin, l'étude présente la création d'un simulateur pour le quadcoptère tilt-rotor et les stratégies de commande utilisées pour contrôler le vol du drone.

## 2 État de l'art

### 2.1 Quadcoptère classique

Les différents types de drones sont classés en deux catégories, à savoir les drones à voilure tournante ou multirotor proches des hélicoptères, et les drones à voilure fixe qui s'apparentent à des avions. Un drone à voilure tournante peut être généralement composé de trois, quatre, six ou huit hélices. Ils sont excellents pour le vol stationnaire mais bien moins rapide que les drones à voilure fixe. Cette étude se concentre sur des drones de type quadcoptère, c'est à dire des drones à voilure tournante équipés de quatre moteurs à hélice afin de voler. Le modèle de quadcoptère est le drone multirotor le plus courant sur le marché de par sa structure simple et efficace. Les quatre rotors sont placés autour de l'appareil, pointant vers le haut. En les faisant tourner à vitesse constante, il est alors simple de placer le drone en vol stationnaire et en contrôlant les vitesses angulaires de chaque moteur, il est possible de contrôler sa trajectoire. Ainsi contrairement aux hélicoptères, les hélices d'un quadcoptère sont fixes puisque l'orientation du drone dépend de la vitesse de rotation de ces dernières. A l'opposé, les hélices d'un hélicoptère sont plus complexes puisque le pilote doit faire varier l'angle d'incidence entre les palmes et le vent relatif. En comparaison, cette différence permet aux quadcoptères d'être plus fiables, faciles à produire et moins chers.

Les quadcoptères peuvent être assemblés selon deux configurations différentes, la configuration "Plus" ou la configuration "Croix", selon la position des rotors. Si les rotors sont orientés de  $0, \pi/2, \pi, 3\pi/2$  il s'agit d'une configuration "Plus". En revanche s'ils sont orientés de  $\pi/4, 3\pi/4, 5\pi/4, 7\pi/4$  c'est une configuration "Croix" qui est retenue.

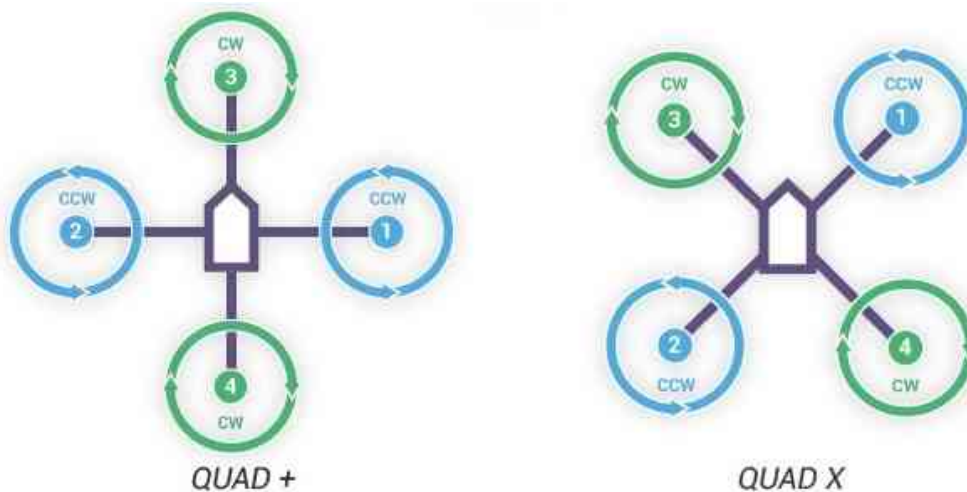


FIGURE 3 – Configuration "Plus" à gauche et "Croix" à droite [11]

Selon sa position, un rotor pourra tourner dans le sens horaire ou anti-horaire. Deux hélices adjacentes ont obligatoirement un sens de rotation contraire, créant un système de counter-rotating propellers, ou CRP. Un tel système entre deux hélices permet un meilleur équilibre des forces et donc une meilleure stabilité de l'appareil en vol. Ainsi, la configuration "Croix" permet au drone d'avoir 2 systèmes CRP sur ses axes de roulis et de tangage tandis que la configuration "Plus" n'en offre qu'un seul par axe. La distribution des forces des rotors est meilleure en configuration "Croix", offrant donc plus de fiabilité

et de stabilité à cette configuration [5]. De plus, ce type de drone est très souvent équipé d'une caméra pour réaliser l'observation d'une zone ou permettre un pilotage manuel de l'appareil. Il est bien plus pratique d'installer une caméra pointant vers l'avant sur la configuration "Croix" car sa vue sera dégagée. Cette configuration est meilleure en tout point que la configuration "Plus" et c'est celle qui est la plus utilisée sur les quadcoptères. Le seul avantage de la configuration "Plus" est que ses équations de modélisation sont légèrement plus simples. Dans cette étude, la configuration "Croix" sera celle retenue pour l'état de l'art et la modélisation.

## 2.2 Modélisation d'un quadcoptère

Pour pouvoir étudier un quadcoptère, il est nécessaire de mettre en équation son attitude de vol, sans l'effet tilt dans un premier temps. Soit les quatre rotors  $M1$ ,  $M2$ ,  $M3$  et  $M4$ , générant respectivement une force  $F1$ ,  $F2$ ,  $F3$  et  $F4$ . L'équilibre entre ces forces et la force gravitationnelle liée à la masse du drone permet à ce dernier de voler. Soit les axes  $XB$ ,  $YB$  et  $ZB$  les angles de roulis, tangage et lacet permettant de repérer le drone dans l'espace et les trois angles d'Euler qui leur sont associés ( $\phi, \theta, \psi$ ). Ces paramètres sont regroupés dans le schéma de la Figure 4.

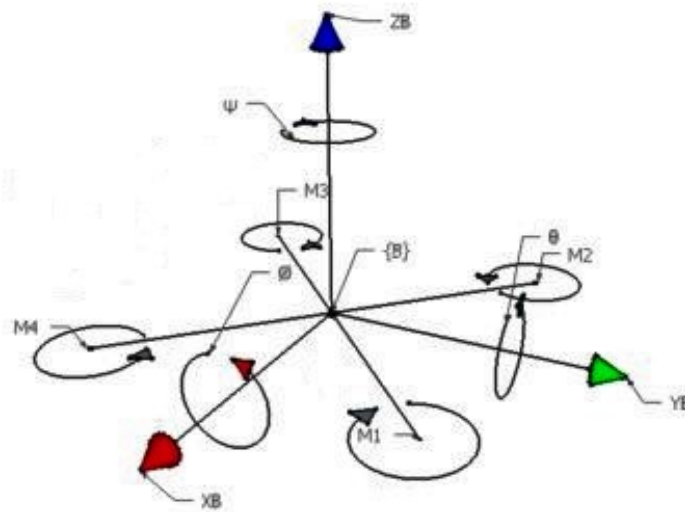


FIGURE 4 – Représentation schématique d'un quadcoptère [1]

Ainsi pour faire avancer le drone vers l'avant, l'appareil doit se pencher vers l'avant c'est à dire créer un moment  $\tau_y$  selon l'axe de tangage. En rendant la vitesse angulaire des moteurs avant inférieure à celle des moteurs arrière, le drone sera en mesure d'avancer tout droit. Le même principe s'applique pour un déplacement latéral en créant un moment  $\tau_x$  autour de l'axe de roulis. Pour faire tourner le quadcoptère autour de l'axe de lacet, il faudra alors utiliser le fait que les moteurs ne tournent pas dans le même sens de rotation. Les moteurs  $M1$  et  $M3$  tournent dans le sens horaire tandis que les moteurs  $M2$  et  $M4$  tournent dans le sens anti-horaire. Si les vitesses de rotation des moteurs de sens horaire sont supérieures aux vitesses de rotation des moteurs anti-horaire, le quadcoptère tournera autour de l'axe de lacet dans le sens horaire, et inversement pour le sens anti-horaire.

La force de poussée de chaque moteur est donnée en utilisant la Théorie de Froude [4] :

$$F_i = C_T \rho A_{r_i} r_i^2 \Omega_i^2 \quad (1)$$

Avec  $\Omega_i$  la vitesse de rotation des hélices,  $C_T$  la constante de poussée issue de la géométrie et du profil du rotor,  $\rho$  la densité de l'air,  $A_{r_i}$  la surface du rotor et  $r_i$  son rayon. Ce type de drone ne vole pas à des altitudes assez importantes pour considérer  $\rho$  comme une variable, la densité de l'air sera donc traitée comme une constante. En introduisant  $k_t$  comme une constante positive regroupant les constantes de l'équation (1) la force de poussée de chaque moteur se résume à :

$$F_i = k_t \Omega_i^2 \quad (2)$$

La force de réaction induit par la traînée du rotor se modélise de la même façon avec une constante  $k_d$  dépendant de la constante de traînée  $C_d$ ,  $A_{r_i}$ ,  $r_i$  et  $\rho$  :

$$D_i = k_d \Omega_i^2 \quad (3)$$

La force de poussée totale du drone est obtenue selon la somme des quatre moteurs.

$$F_{total} = \sum_{i=1}^4 F_i \quad (4)$$

La matrice de rotation du quadcoptère en "X" est donnée selon l'équation suivante, où S et C sont respectivement les fonctions sinus et cosinus [1].

$$R = \begin{pmatrix} C\phi C\theta & C\theta S\phi S\psi - C\psi S\phi & S\phi S\psi + C\phi C\psi S\theta \\ C\theta S\phi & C\phi C\psi + S\phi S\psi S\theta & C\theta C\psi S\phi - C\phi S\psi \\ -S\theta & C\theta S\psi & C\theta C\psi \end{pmatrix} \quad (5)$$

Le modèle des mouvements de translation du drone peuvent être dérivé de la seconde loi de Newton  $F = ma$ . En effet, avec  $\ddot{x}$ ,  $\ddot{y}$  et  $\ddot{z}$ , respectivement les accélérations selon  $x$ ,  $y$  et  $z$ , il vient :

$$m \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = R \begin{pmatrix} 0 \\ 0 \\ F_{total} \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} \quad (6)$$

Les moments sont données dans les équations suivantes, avec  $L$  la distance entre un rotor et le centre de masse du drone et  $I_{xx}$ ,  $I_{yy}$ ,  $I_{zz}$  respectivement les inerties de roulis, tangage et lacet, découlant de l'effet gyroscopique subit par le drone.

$$\tau_x = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{pmatrix}^T L \begin{pmatrix} \cos(\frac{\pi}{4}) \\ \cos(\frac{3\pi}{4}) \\ \cos(\frac{5\pi}{4}) \\ \cos(\frac{7\pi}{4}) \end{pmatrix}^T + \dot{\theta}\dot{\psi}(I_{yy} - I_{zz}) \quad (7)$$

$$\tau_y = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{pmatrix}^T L \begin{pmatrix} \sin(\frac{\pi}{4}) \\ \sin(\frac{3\pi}{4}) \\ \sin(\frac{5\pi}{4}) \\ \sin(\frac{7\pi}{4}) \end{pmatrix}^T + \dot{\phi}\dot{\psi}(I_{zz} - I_{xx}) \quad (8)$$

$$\tau_z = \frac{2C_d}{C_T} L(-F_1 + F_2 - F_3 + F_4) + \dot{\theta} \dot{\phi} (I_{xx} - I_{yy}) \quad (9)$$

L'effet gyroscopique peut être considéré comme négligeable car les perturbations qu'il entraîne sur le drone sont insignifiantes. Ainsi (7), (8), (9) peuvent être réécrites sous la forme :

$$\tau_x = k_t L \sum_{i=1}^4 \Omega_i^2 \cos\left(\frac{\pi}{2}(i-1) + \frac{\pi}{4}\right) \quad (10)$$

$$\tau_y = k_t L \sum_{i=1}^4 \Omega_i^2 \sin\left(\frac{\pi}{2}(i-1) + \frac{\pi}{4}\right) \quad (11)$$

$$\tau_z = k_d L \sum_{i=1}^4 (-1)^i \Omega_i^2 \quad (12)$$

Depuis ces équations, il est possible d'obtenir 4 entrées de commande pour le quadcoptère, soit :

$$\begin{cases} u_1 = F_{total} \\ u_2 = \tau_x \\ u_3 = \tau_y \\ u_4 = \tau_z \end{cases} \quad (13)$$

On obtient donc la relation liant couples/forces et vitesses des rotors :

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} k_t & k_t & k_t & k_t \\ k_t L & k_t L & -k_t L & -k_t L \\ -k_t L & k_t L & k_t L & -k_t L \\ -k_d & k_d & -k_d & k_d \end{pmatrix} \begin{pmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{pmatrix} \quad (14)$$

Ainsi il vient les accélérations selon les 6 coordonnées du quadcoptère.

$$\begin{cases} \ddot{x} = \frac{(s\phi s\psi + c\phi c\psi s\theta)u_1}{m} \\ \ddot{y} = \frac{(c\phi c\psi c\theta + c\phi s\psi)u_1}{m} \\ \ddot{z} = \frac{(c\theta s\psi)u_1}{m} \\ \ddot{\phi} = \frac{u_2}{I_{xx}} \\ \ddot{\theta} = \frac{u_3}{I_{yy}} \\ \ddot{\psi} = \frac{u_4}{I_{zz}} \end{cases} \quad (15)$$

En intégrant deux fois, il vient les coordonnées du drone selon les 6 coordonnées. Ces équations permettent de modéliser l'attitude d'un quadcoptère classique, néanmoins elles ne suffisent pas à modéliser le problème sur lequel se penche cette étude. En effet, l'objectif est d'étudier le modèle tilt des quadcoptères, qui s'appuie sur la modélisation du schéma classique mais qui reste à compléter.

## 2.3 Quadcoptère tilt-rotor

La technologie de tilt-rotor consistant à changer l'angle des rotors du quadcoptère s'est massivement développée ces dernières années mais elle n'est pas si récente. Le modèle de quadcoptère utilisant une technologie de type tilt-rotor le plus ancien remonte aux années 1960. Il s'agit du Curtiss X-19, un avion militaire pour le transport de passagers, composé de 4 rotors à 3 hélices capable de s'incliner jusqu'à  $90^\circ$  [12].



FIGURE 5 – Modèle du Curtiss X-19 [12]

Il s'agit ici d'un avion massif capable de transporter près de 550kg, bien loin donc du modèle de drone sur lequel se penche cette étude. Cependant avec l'apparition des drones miniatures et l'utilisation du système de quadcoptère pour sa maniabilité et ses capacités de décollage et atterrissage de façon vertical, cette technologie de tilt des rotors s'est vu apparaître sur ces petits engins également.

Un frein majeur au système classique de quadcoptère abordé dans les parties précédentes est qu'il est possible de contrôler uniquement 4 commandes indépendantes, soit les 4 vitesses des rotors du drone. Il n'est ainsi pas possible de contrôler les six positions et angles du quadcoptères de façons indépendantes puisque le système est sous-actionné. En effet, afin d'avancer selon  $x$  en conservant la même hauteur, les moteurs avant ralentissent tandis que les moteurs arrières accélèrent, permettant au drone de contrôler sa position en  $x$  mais faisant pencher le quadcoptère vers l'avant. Ainsi un changement selon  $x$  entraîne un changement selon l'angle de tangage  $\theta$ . L'avantage de pouvoir tilt les rotors deux à deux selon l'axe de tangage permet d'avoir 6 commandes indépendantes, permettant au système de ne plus être sous-actionné [3]. Dans le même exemple que précédemment, en faisant avancer le drone selon l'axe  $x$ , avec un système de tilt il est possible de maintenir l'assiette du quadcoptère plate. Ainsi, seulement l'axe  $x$  se retrouve modifier tandis que les angles de roulis, tangage et lacet restent constants.



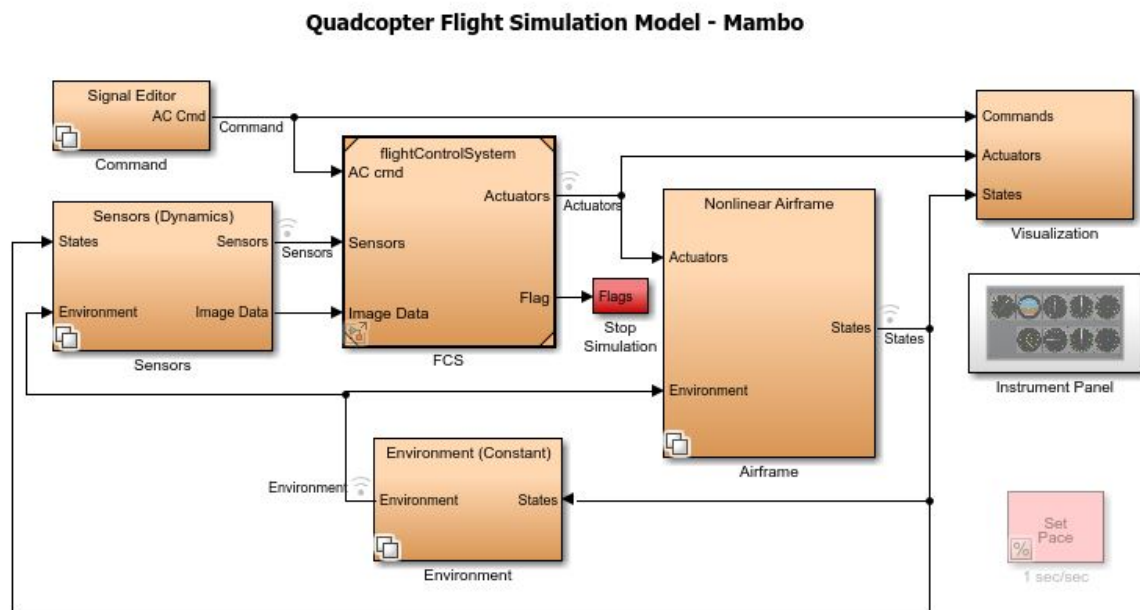
Les quadcoptère tilt-rotor se divisent en deux catégories, ceux capable de tilt leur rotors deux à deux en les faisant fonctionner par pair et ceux pouvant tilt chaque rotor individuellement à des angles différents dans des directions différentes. Cette étude se penche sur le premier cas, avec un quadcoptère pouvant tilt ses rotors latéraux par pair d'un angle  $\alpha_1$  et  $\alpha_2$ . Cela permet en principe d'avoir les 6 commandes indépendantes évoquées précédemment, sans avoir à changer drastiquement les équations dynamiques du quadcoptère classique couverte dans la dernière partie. L'objectif principal des quadcoptères tilt-rotor est de réaliser la transition entre le décollage vertical du drone et son déplacement horizontal. Les modèles de quadcoptère en tilt-rotor sont certes plus complexes que les quadcoptères classiques, ils offrent cependant plus de possibilité de contrôle, notamment un contrôle indépendant des coordonnées du drone.

Ce document propose dans un premier temps une étude d'un simulateur de quadcoptère classique existant pour avoir une meilleure compréhension des méthodes de commande de ce genre d'appareil. Dans un second temps, cette étude se penche sur la réalisation d'un simulateur plus simple pour le quadcoptère tilt-rotor, adaptée à la recherche de commande pour contrôler les vitesses des moteurs et les angles de tilt dans le but de déplacer le drone tout en gardant une assiette plate.

## 3 Étude d'un simulateur de quadcoptère existant

### 3.1 Étude du schéma de commande

Dans un premier temps, cette étude se propose d'étudier un modèle de simulateur pour quadcoptère existant. Matlab et Simulink propose une bibliothèque appelée Aerospace Blockset adaptée à la simulation et à la modélisation de véhicules aéronautiques. Aerospace Blockset propose un exemple de simulateur de quadcoptère en "X", qui constitue un excellent exemple de simulateur pour drone classique, avant de passer à un simulateur incluant l'effet tilt. Le schéma de commande proposé par ce simulateur est présenté par la figure ci-dessous.



Copyright 2013-2019 The MathWorks, Inc.

FIGURE 6 – Schéma de commande de l'exemple d'Aerospace Blockset

C'est un schéma de commande en boucle fermée avec retour d'état. Il s'articule autour de six blocs principaux :

- le modèle mathématique
- les capteurs
- les paramètres liés à l'environnement du drone
- le système de vol
- les signaux d'entrée
- la sortie d'affichage

Il existe également deux blocs secondaires, flags qui gèrent les conditions responsables de l'arrêt de la simulation, c'est à dire en cas de crash, et Set Pace permettant de choisir la vitesse de simulation, ralenti, temps réel ou accéléré.

Le premier bloc est celui des signaux d'entrées, permettant de régler les coordonnées  $x$ ,  $y$ ,  $z$ ,  $\phi$ ,  $\theta$ ,  $\psi$  voulues par l'utilisateur. Pour cela il est possible d'utiliser un éditeur de signal

de Simulink, brancher un joystick, utiliser des données sauvegardées ou un tableur. Cela permet de créer les commandes d'entrée du quadcoptère, utilisées par le contrôleur de vol et l'afficheur.

Le bloc responsable de l'environnement dans lequel évolue le drone peut être constant ou variable. Il permet d'estimer des éléments tels que la température, la pression de l'air, sa densité, qui seront repris par les capteurs. Ces éléments peuvent être paramétrés comme fixes ou variables dans le cas où le drone atteint une altitude suffisante pour considérer comme non négligeables les changements des paramètres extérieurs.

L'exemple utilise des capteurs qui sont une centrale à inertie, un sonar pour calculer l'altitude et une caméra pour l'estimation du flux optique. Les paramètres nécessaires à ces capteurs sont obtenus par le bloc environnement et par un retour d'état du bloc du modèle mathématique, calculant les coordonnées du drone aux instants  $t$  de la simulation. Ils permettent de fournir des informations au bloc de système de vol.

Ce bloc de système de vol est le plus complexe du schéma de commande et découle du travail de Sertac Karaman et Fabian Riether, professeurs au MIT. Il utilise les commandes d'entrée du drone et les informations des capteurs pour retourner les commandes moteurs et les flags en cas de crash ou perte de contrôle. Il agit donc comme un contrôleur et un estimateur. Il utilise des filtres de Kalman pour déduire sa position et sa vitesse. L'exemple étudié utilise également un contrôleur PID (proportional integral derivative) pour le contrôle de l'angle de roulis et tangage, un contrôleur PD (proportional-derivative) pour le contrôle de l'angle de lacet et un dernier contrôleur PD pour le contrôle des coordonnées en  $x$ ,  $y$  et  $z$  [2]. C'est ce bloc qui est responsable des vitesses des moteurs au cours de la simulation. En d'autre terme, il est essentiel à la détermination des commandes  $u_{1,2,3,4}$  évoqués dans l'équation (13)

Le bloc du modèle est celui permettant d'estimer les coordonnées du drone à un instant  $t$ . Il a donc vocation à se rapprocher le plus possible de la réalité. L'exemple étudié donne la possibilité d'utiliser un modèle linéaire ou non-linéaire. C'est souvent le modèle non-linéaire qui est favorisé pour la simulation car il offre une meilleure fidélité. Le modèle mathématique utilise les commandes moteurs issues du simulateur de vol et les forces extérieures liées au bloc environnement puis utilise les équations de modélisations pour retourner l'état du drone.

Enfin le bloc de sortie est responsable de l'affichage du drone. Ce dernier peut se faire sous la forme d'un affichage en trois dimensions à l'aide de l'outil Simulink3D. Il existe d'autres options d'affichage qui ne seront pas discutées dans ce document. L'objectif est de charger un environnement 3D et un modèle de drone. Les coordonnées du drone étant connues, il est possible d'afficher sa position dans l'espace. Les vitesses des moteurs sont connues également et permettent d'activer en temps réel le joint des hélices dans le modèle pour les faire tourner.

## 3.2 Résultats et discussion

Une fois le fonctionnement du simulateur assimilé, il est temps de se pencher sur les résultats qu'il produit. Soit un exemple simple, où le drone part du sol, décolle jusqu'à atteindre une altitude désirée et s'y stabilise. Il s'agit donc d'une action sur  $z$  en gardant les autres coordonnées constantes pendant toute la simulation. La simulation du drone ayant atteint sa position finale est affichée ci-dessous.



FIGURE 7 – Affichage du drone de l'exemple d'Aerospace Blockset

Le simulateur permet d'afficher le quadcoptère par un modèle très réaliste dans un environnement désiré et d'observer son attitude de vol pendant la simulation. Cependant, aucun affichage des coordonnées au cours du temps n'est donné, ce qui permettrait d'établir le temps de stabilisation du drone ou la précision de la commande en mesurant le dépassement. Ce simulateur est donc parfait pour charger un modèle de drone classique en "X" existant et étudier son comportement. En revanche il n'est pas adapté à cette étude qui cherche à étudier des stratégies de commande. De plus le schéma de commande étant bien trop compliqué, l'adapter pour un drone en tilt-rotor semble impossible. L'architecture générale du schéma de commande, bien que trop chargée, reste intéressante et pourra inspirer un autre simulateur plus simple. Le passage par les capteurs et les variables d'environnement n'est pas pertinent dans cette étude. Ainsi, ce simulateur ne sera pas utilisé pour développer un modèle de tilt-rotor mais servira de première étape vers un simulateur plus simple et compact.

## 4 Développement d'un simulateur pour quadcoptère

Cette partie présente le développement d'un simulateur dont la logique est similaire à celui étudié dans la partie précédente, avec un schéma de commande bien plus simple. Par exemple les impacts de l'environnement tel que la densité de l'air seront négligés et il portera sur des petits déplacements.

### 4.1 Quadcoptère sans tilt

La partie modélisation mécanique est couverte dans la section 2.2 et sera réutilisée pour ce simulateur. L'objectif est d'avoir en entrée du contrôleur un signal désiré sur  $x$ ,  $y$ ,  $z$  et l'angle de lacet  $\psi$  et d'être capable en sortie d'avoir les six coordonnées du quadcoptère. Cela permet d'étudier les déplacements du drone étant donné qu'en comparant les coordonnées de sortie et d'entrée, il est possible de déterminer la précision de réponse du drone. Par ailleurs, en ayant ses coordonnées dans le temps, la modélisation du drone dans un espace 3D devient accessible. Afin d'estimer la trajectoire du quadcoptère, le contrôleur estime à un instant  $t$  les coordonnées de l'objet qu'il étudie, et réutilise ces coordonnées pour calculer les coordonnées de l'objet à l'instant  $t + \delta t$ . Il est nécessaire d'utiliser un système à retour d'état, ainsi il ne faut pas confondre les coordonnées désirées et les coordonnées réelles de l'objet. Le schéma de commande utilisé par le contrôleur est donné ci-dessous.

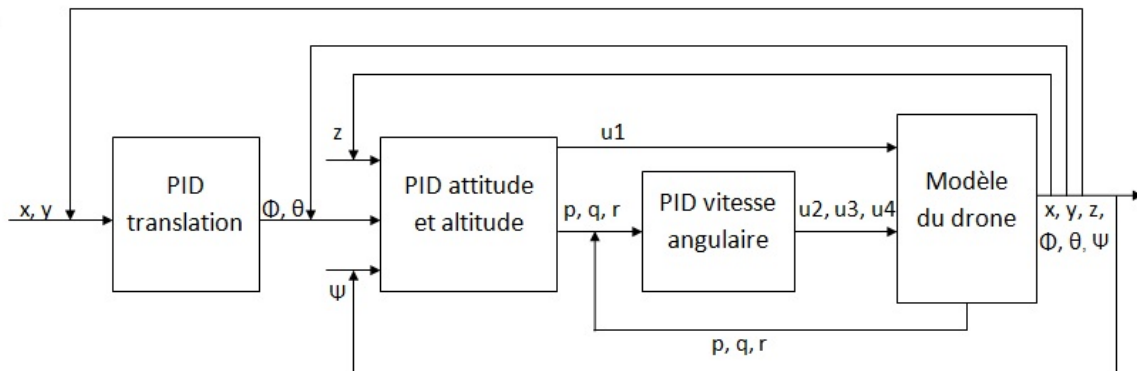


FIGURE 8 – Schéma de commande utilisé sans tilt

Les coordonnées  $x$ ,  $y$ ,  $z$  et  $\psi$  sont données par l'utilisateur en utilisant une fonction *step*. L'utilisateur choisit ainsi la position des coordonnées que le drone doit atteindre. Dans un premier temps, le contrôleur utilise un régulateur PID pour déterminer les  $\phi$  et  $\theta$  désirés, soit  $\phi_{des}$  et  $\theta_{des}$ . En utilisant les coordonnées  $x_{des}$  et les coordonnées  $x$  du drone par retour d'état, le contrôleur obtient  $\phi_{des}$ . De la même façon avec les  $y_{des}$  et  $y$  par retour d'état, il vient  $\theta_{des}$ . Le contrôleur PID utilise une architecture classique d'une structure PID parallèle agissant sur l'erreur.

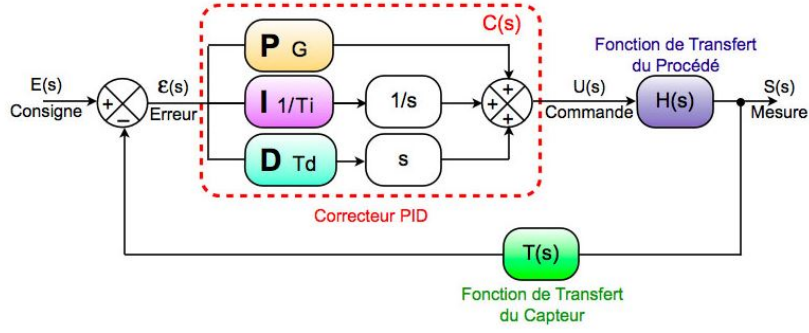


FIGURE 9 – Structure théorique d'un PID sur l'erreur [9]

Ci-dessous est donné la structure du PID utilisé pour déterminer  $\theta_{des}$ . Par comparaison, le PID utilisé dans le contrôleur étudié est identique au PID théorique. Le PID avec  $y$  et  $\phi$  est d'une structure similaire. Les gains de commande pour ce PID et tout ceux utilisés sont reportés dans un fichier Matlab.

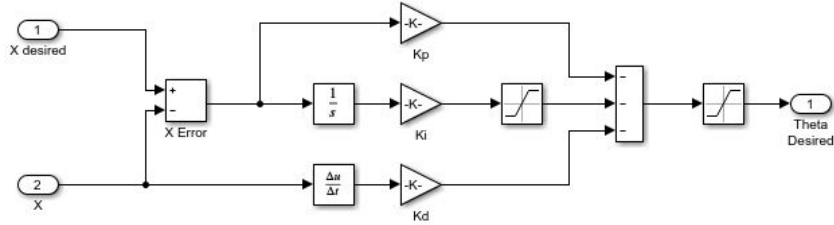


FIGURE 10 – Structure du PID pour  $\theta$  désiré

Une fois toutes les coordonnées désirées obtenues, il faut passer au contrôle d'altitude et d'attitude. Pour l'altitude du drone il s'agit de la poussée totale des moteurs, soit la commande  $u_1$  abordée en section 2.2. Pour les commandes  $u_2$ ,  $u_3$  et  $u_4$  soit les commandes de roulis, tangage et lacet il est nécessaire d'exprimer l'attitude du drone selon ses vitesses angulaires  $p$ ,  $q$  et  $r$ . Le schéma ci-dessous explique le lien entre les angles d'Euler et les vitesses angulaires.

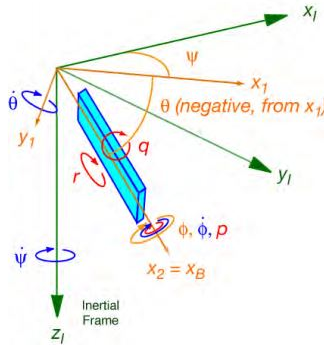


FIGURE 11 – Lien entre angles d'Euler et vitesses angulaires [10]

Les angles d'Euler et les vitesses angulaires sont reliés par [6] :

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (16)$$

Dans le simulateur étudié, les  $p_{des}$ ,  $q_{des}$  et  $r_{des}$  sont obtenus par le contrôleur d'attitude. Pour obtenir  $p_{des}$ , il faut utiliser un contrôleur P sur l'erreur entre  $\phi$  et  $\phi_{des}$ . De la même façon pour  $q_{des}$  avec  $\theta$  et  $\theta_{des}$  ainsi que pour  $r_{des}$  avec  $\psi$  et  $\psi_{des}$ .

Enfin il vient le dernier contrôleur, celui sur les vitesses angulaires. De façon similaire aux autres contrôleurs les  $p$  et  $p_{des}$  passent par un contrôleur PID pour sortir la commande  $u_2$ . La commande  $u_3$  est issue de  $q_{des}$  et  $q$  ainsi que  $u_4$  est issue de  $r_{des}$  et  $r$ . Les 4 entrées de commandes obtenues, il est possible de passer au modèle du système et déterminer toutes les coordonnées du quadcoptère. Comme explicité dans les équations (15), le modèle détermine  $\ddot{\phi}$ ,  $\ddot{\theta}$  et  $\ddot{\psi}$  qu'il faut intégrer deux fois pour avoir  $\phi$ ,  $\theta$  et  $\psi$ . De cela, sont déduits les accélérations  $\ddot{x}$ ,  $\ddot{y}$  et  $\ddot{z}$  qu'il faut intégrer deux fois pour avoir  $x$ ,  $y$  et  $z$ . Enfin, d'après (16)  $p$ ,  $q$  et  $r$  deviennent eux aussi accessibles. Une fois les coordonnées obtenues, il est possible de simuler le quadcoptère.

La simulation du quadcoptère se fait dans un espace 3D. Une fonction Matlab qui affiche le quadcoptère dans l'espace 3D est appelée toutes les trois itérations et affiche le drone, permettant un suivi de la trajectoire du drone par l'utilisateur. Dans un exemple où l'utilisateur demande un déplacement en  $x$  au drone et une augmentation d'altitude en  $z$ , en gardant  $y$  et  $\psi$  nuls, l'affichage est le suivant.

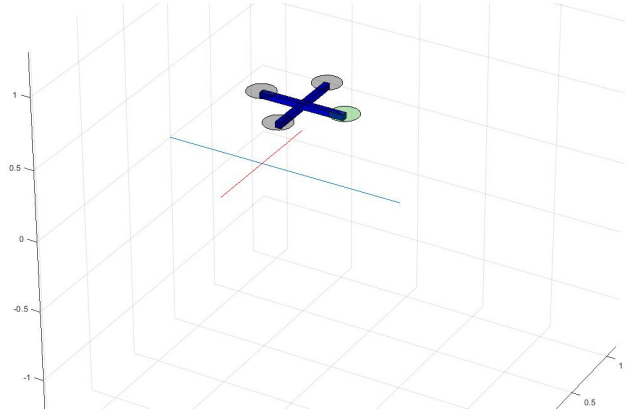


FIGURE 12 – Affichage d'un drone sans tilt dans l'espace 3D

Il est possible d'observer que le drone fait un déplacement vers l'avant en gagnant en altitude comparé à son point d'origine marqué par le croisement des droites rouges et bleues. Les coordonnées du drone en fonction du temps sont représentées ci-dessous avec en bleu les coordonnées réelles et en rouge les coordonnées parfaites, désirées par l'utilisateur.

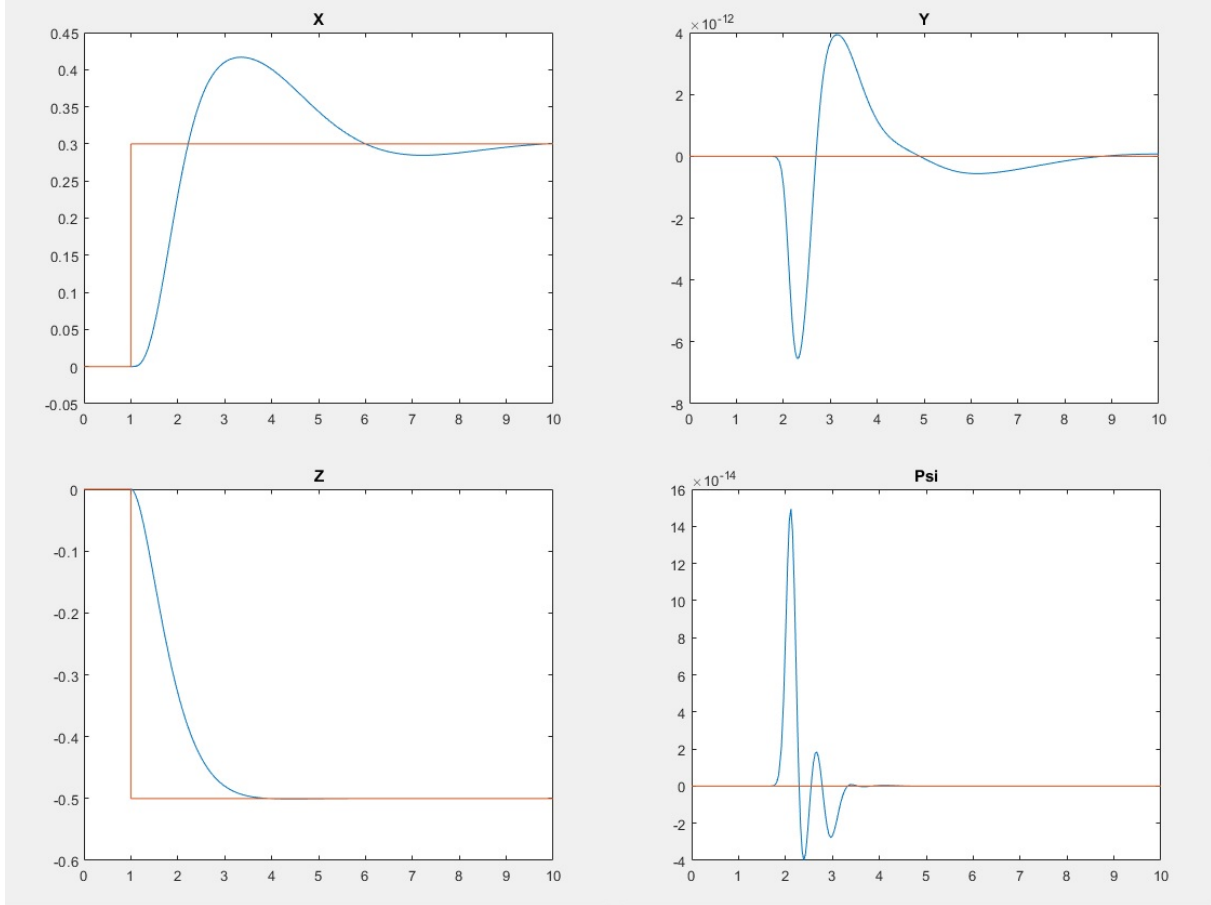


FIGURE 13 – Coordonnées du quadcoptère sans tilt dans le temps selon un déplacement  $x$  et  $z$

Il faut remarquer dans un premier temps que le contrôle sur l'altitude est très précis puisqu'il n'y a aucun dépassement en  $z$  et que le système est à équilibre rapidement. Les perturbations sur  $y$  et  $\psi$  sont de l'ordre de  $10^{-12}$  et  $10^{-14}$ , donc négligeables. Cependant la stabilisation en  $x$  est assez imprécise, à l'image du grand dépassement et du temps de mise en équilibre de 10 secondes, quasiment égale au temps de la simulation. Cela illustre justement le problème des quadcoptères classiques qui en avançant selon  $x$  par exemple génère un changement sur l'angle de tangage  $\theta$ , ce qui empêche un équilibrage rapide et précis une fois que le drone a atteint les coordonnées voulues. L'utilisation de rotors pouvant tilt permet une convergence plus rapide sur  $x$  puisque que l'assiette est maintenue plate.



Il est également possible de modifier l'angle de lacet du drone et d'étudier la précision de la trajectoire du drone sur ce dernier. Ce qu'il est donc intéressant d'observer est l'évolution de  $\psi$  au cours du temps, pour voir si le modèle de quadcoptère classique peut tourner sur lui-même avec précision. Les résultats pour une rotation de  $\frac{\pi}{4}$  autour de l'axe de lacet sont représentés ci-dessous.

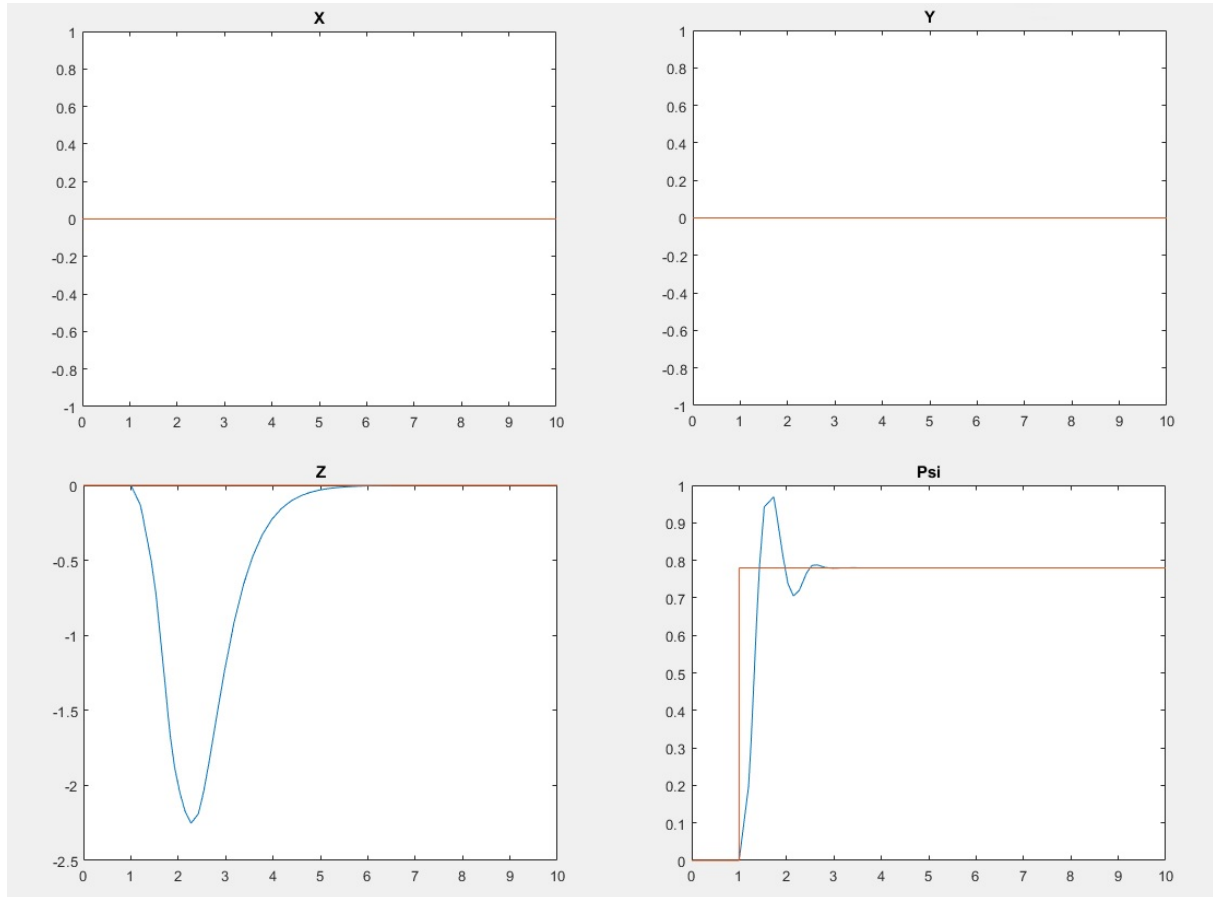


FIGURE 14 – Coordonnées du quadcoptère sans tilt dans le temps selon un déplacement  $\psi$

Il apparaît que les coordonnées  $x$  et  $y$  sont parfaitement conservées mais que le drone subit un changement d'altitude proportionnellement important au changement de l'angle de lacet. Cela s'explique par le fait que les quadcoptères classiques ne peuvent jouer que sur la puissance de leurs moteurs pour changer l'angle de lacet, pouvant influencer sur la poussée totale et donc sur l'altitude du quadcoptère. L'angle  $\psi$  montre également un dépassement d'environ 0.2 rad soit 11,4 deg. Pour un déplacement initial de  $\frac{\pi}{4}$ , cela représente un dépassement de l'angle de lacet d'un facteur de 0,25. Le temps de stabilisation de l'angle en question est lui de 2 secondes. Ce sont des conséquences du sous-actionnement des quadcoptères de ce type, que le système de tilt-rotor cherche à résoudre.

## 4.2 Modélisation d'un quadcoptère avec tilt rotors

De manière analogue au quadcoptère classique à rotor fixes. Il faut partir de la matrice de rotation des angles d'euler reliant rotations roulis/tangage/lacet au repère fixe avec la matrice suivante :

$${}^B R_G = \begin{pmatrix} C\theta S\psi & C\theta S\psi & -S\theta \\ -C\phi S\psi + S\phi S\theta C\psi & C\phi C\psi + S\phi S\theta S\psi & S\psi C\theta \\ S\phi S\psi + C\phi S\theta S\psi & -S\phi C\psi + C\phi S\theta C\psi & C\phi C\theta \end{pmatrix} \quad (17)$$

Dans le cas du système étudié, il y a 2 angles de tilting  $\alpha_1$  et  $\alpha_2$ . Ces deux angles sont par conséquent couplés aux vitesses de rotation des rotors et permettent de déduire d'une relation analogue à celle pour le système classique en (32)

$$V = \Gamma U \quad (18)$$

Où la matrice  $\Gamma$  correspond à la matrice d'allocation des forces/couples et le vecteur  $V$ , vecteur de commande virtuel défini de la manière suivante :

$$V = \begin{pmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} \quad (19)$$

$$T = k[(\Omega_1^2 + \Omega_2^2)C1 + (\Omega_3^2 + \Omega_4^2)C2] \quad (20)$$

$$\Gamma = \begin{pmatrix} 0 & k & 0 & k & 0 & k & 0 & k \\ -b & kL & b & kL & -b & -kL & b & -kL \\ 0 & -kL & 0 & kL & 0 & kL & 0 & -kL \\ -kL & -b & -kL & b & kL & -b & kL & b \end{pmatrix} \quad (21)$$

Cependant, le vecteur  $U$  des vitesses des rotors dépend également des angles de tilt comme suivant :

$$U = \begin{pmatrix} \Omega_1^2 S1 \\ \Omega_1^2 C1 \\ \Omega_2^2 S1 \\ \Omega_2^2 C1 \\ \Omega_3^2 S2 \\ \Omega_3^2 C2 \\ \Omega_4^2 S2 \\ \Omega_4^2 C2 \end{pmatrix} \quad (22)$$

Afin de pouvoir séparer les angles de tilt des vitesses  $\Omega_i$  des rotors, une linéarisation du système sous hypothèse de faibles angles est nécessaire.

Les relations entre forces et grandeurs contrôlées explicitées, on en déduit, respectivement, la dynamique du système en translation et rotation avec les équations suivantes :

$$a = \frac{1}{m} \left( {}^B R_G \begin{pmatrix} kS1(\Omega_1^2 + \Omega_2^2) + kS2(\Omega_3^2 + \Omega_4^2) \\ 0 \\ kC1(\Omega_1^2 + \Omega_2^2) + kC2(\Omega_3^2 + \Omega_4^2) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} \right) \quad (23)$$

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = I^{-1} \begin{pmatrix} kLC1[\Omega_1^2 + \Omega_2^2] + bS1[\Omega_2^2 - \Omega_1^2] + bS1[\Omega_4^2 - \Omega_3^2] - kLC1[\Omega_3^2 + \Omega_4^2] \\ kL[-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2] \\ -kLS1[\Omega_1^2 + \Omega_2^2] + bC1[\Omega_2^2 - \Omega_1^2] + kLS2[\Omega_3^2 + \Omega_4^2] + bC2[\Omega_4^2 - \Omega_3^2] \end{pmatrix} \quad (24)$$

### 4.3 Stratégie de détermination des valeurs de tilt

Une fois la dynamique du modèle de tilt-rotor établie il faut l'intégrer au contrôleur. Dans un premier temps, les angles de tilt  $\alpha_1$  et  $\alpha_2$  sont généralisés sous un angle de tilt général  $\alpha$ . Son expression dépend de  $q$ ,  $x$  et  $\theta$ , soit toutes les coordonnées influant avec un déplacement en  $x$ , l'orientation des angles de tilt de cette étude. Le tilt corrige une composante selon  $x$  et une selon  $\theta$  d'où l'existence de  $\alpha_x$  et  $\alpha_\theta$  [7] :

$$\alpha_x = k_{px}(x_{des} - x) + k_{dx}(\dot{x}_{des} - \dot{x}) \quad (25)$$

$$\alpha_\theta = k_{p\theta}(\theta_{des} - \theta) + k_{d\theta}(\dot{\theta}_{des} - \dot{\theta}) \quad (26)$$

Les coefficients  $k_{px}$   $k_{dx}$  sont les gains utilisés dans le PID sur  $x$  et les coefficients  $k_{p\theta}$  et  $k_{d\theta}$  sont les gains utilisés pour le PID sur  $\theta$ . Enfin l'angle de tilt obtenu est la somme de ces deux  $\alpha$  :

$$\alpha = \alpha_x + \alpha_\theta \quad (27)$$

Il faut par la suite être capable d'extraire  $\alpha_1$  et  $\alpha_2$  pour que le drone réalise le déplacement qui lui est commandé. Si le quadcoptère doit réaliser un déplacement dans lequel l'angle de lacet  $\psi$  reste constant, par exemple se déplacer tout droit vers l'avant, les angles de tilt seront en phase tel que  $\alpha_1 = \alpha_2$ . Dans un cas différent, les angles de tilt doivent être dissociable.

Le contrôleur requiert un nouveau schéma de commande dérivé du précédent, donné sur la figure 8, permettant de contrôler le tilt. Le schéma de commande suivant est proposé :

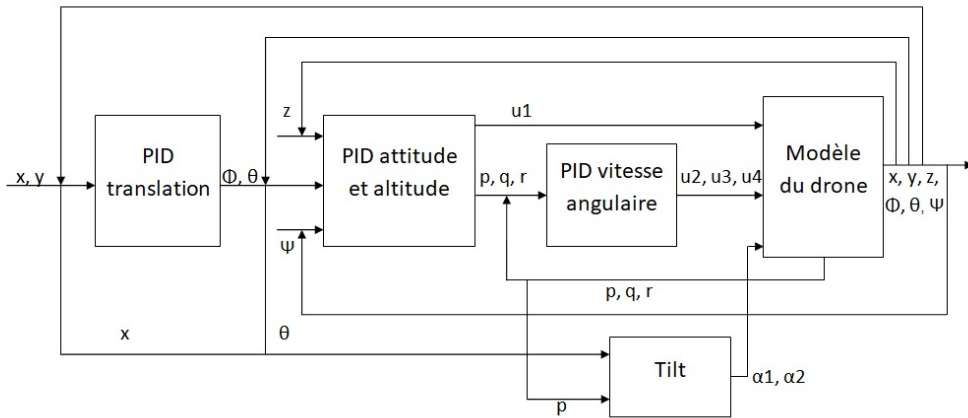


FIGURE 15 – Schéma de commande utilisé avec tilt

Ainsi le sous-bloc tilt récupère  $x$ ,  $\theta$  et  $p$  évoqué précédemment et envoie les angles de tilt  $\alpha_1$  et  $\alpha_2$  dans le modèle où ils impacteront les calculs de vitesses de moteurs, d'accélération angulaire soit  $\ddot{\phi}$ ,  $\ddot{\theta}$  et  $\ddot{\psi}$  ainsi que les accélérations de translation soit  $\ddot{x}$ ,  $\ddot{y}$  et  $\ddot{z}$ . En reprenant le premier exemple présenté dans la partie précédente sur les figures 12 et 13, soit un déplacement en  $x$  et  $z$ , il est possible d'observer l'impact que les angles de tilt devraient avoir. Il est obligatoire qu'ils soient en phase et leur évolution est présenté ci-dessous.

#### 4.3.1 Résultats sur le tilt en tangage

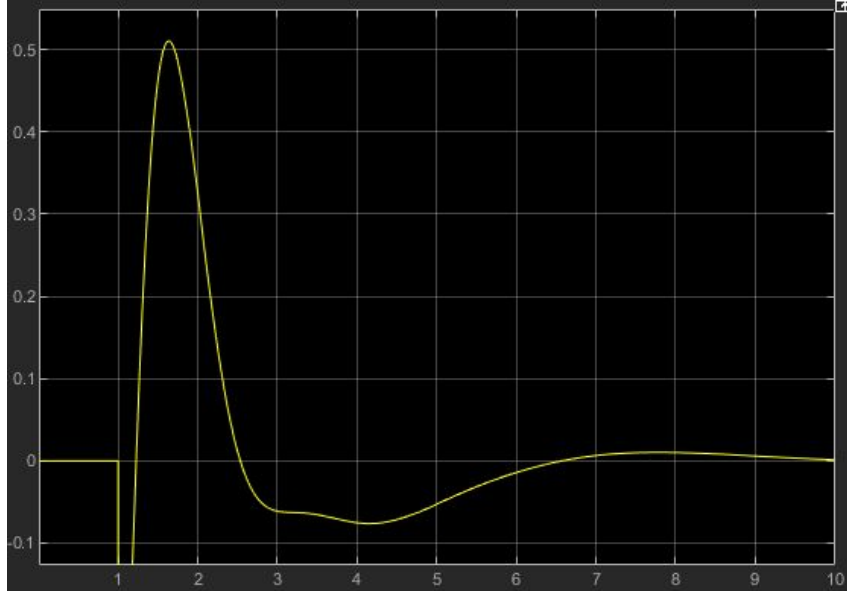


FIGURE 16 – Évolution de l'angle de tilt  $\alpha$  au cours d'un déplacement

Cet exemple s'effectue avec un déplacement selon  $x$  de 40cm, et un gain d'altitude selon  $z$  de 50 cm. Le contrôleur propose un tilt de 0.5 rad soit une rotation de  $\frac{\pi}{6}$ , pendant 3 secondes et se stabilise en retournant à sa position d'origine. Ces résultats semblent réalistes et valident le contrôle du tilt pour les déplacements en  $x$ .

#### 4.4 Stratégie de détermination des tilts avec lacet

Dans le cas d'un système où les tilts sont utilisés pour produire également un couple en lacet, en linéarisant l'équation donné en (14), on en déduit la relation suivante :

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} k_t & k_t & k_t & k_t & 0 \\ k_t L & k_t L & -k_t L & -k_t L & 0 \\ -k_t L & k_t L & k_t L & -k_t L & 0 \\ -k_d & k_d & -k_d & k_d & k_t L \end{pmatrix} \begin{pmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \\ (\Omega_3^2 + \Omega_4^2)\alpha_2 - (\Omega_1^2 + \Omega_2^2)\alpha_1 \end{pmatrix} \quad (28)$$

La matrice d'allocation n'est plus carrée. Pour résoudre ce système et ainsi déduire des vitesses de rotors (et du décalage des tilt pour engendrer un couple important en lacet), il est nécessaire de passer par sa matrice pseudo-inverse définie par :

$$\Gamma^\# = \Gamma^t(\Gamma^t\Gamma)^{-1} \quad (29)$$

$$\begin{pmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \\ (\Omega_3^2 + \Omega_4^2)\alpha_2 - (\Omega_1^2 + \Omega_2^2)\alpha_1 \end{pmatrix} = \Gamma^\# \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} \quad (30)$$

Avec ce système, le décalage de tilt entre  $\alpha_1$  et  $\alpha_2$  est découplé des vitesses des rotors déterminés par les 4 relations précédentes du système. L'utilisation des tilting est une stratégie intéressante du fait de la possibilité de produire un mouvement en translation suivant  $x$  tout en maintenant un tangage faible voire nul. Aussi il faut donc définir un angle de tilting moyen permettant de reproduire le tangage désiré pour un drone classique. Ceci impose la relation suivante avec  $\theta$ , angle de tangage apparent :

$$\theta = \frac{\alpha_1 + \alpha_2}{2} \quad (31)$$

Pour déterminer les angles de tilt On en déduit donc le système d'équations suivant, avec A le résultat obtenu par l'équation (30) :

$$\begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 0.5 & 0.5 \\ -(\Omega_1^2 + \Omega_2^2) & (\Omega_3^2 + \Omega_4^2) \end{pmatrix}^{-1} \begin{pmatrix} \phi \\ A \end{pmatrix} \quad (32)$$

## 5 Conclusion

Cette étude réalise un état de l'art sur les quadcoptères en configuration "X" en expliquant les avantages d'une telle configuration mais également ses limites. La modélisation de ce type de drone est également réalisée et sera utilisée dans la création d'un simulateur de vol. Pour palier à ces limites, la technologie des tilt-rotor est abordée dans l'état de l'art comme une solution possible pour avoir un meilleur contrôle en vol des quadcoptères. La stratégie de tilt qui est proposée est d'avoir une rotation deux à deux sur les deux moteurs latéraux avec donc deux angle de tilt, un à droite et un à gauche dans le sens de l'axe  $x$ .

Le document propose également une étude d'un simulateur existant pour quadcoptère, très complet et instructif mais trop complexe pour être utilisé en implémentant une stratégie de tilt.

Enfin, la création d'un contrôleur plus simple est détaillée, en commençant par un modèle de quadcoptère classique sans le tilt, en utilisant la modélisation de l'état de l'art. La stratégie de commande et le schéma de commande sont passés en revue et les résultats obtenus sont présentés. Le drone est alors capable de se déplacer à des coordonnées choisies et de s'y stabiliser, en retournant l'évolution de ses coordonnées dans le temps afin d'étudier la fiabilité de la stratégie de commande et la rapidité du temps de réponse. Par la suite, la stratégie de tilt-rotor est abordée en redéfinissant les équations de dynamique et la modélisation du modèle. La nouvelle stratégie de commande est détaillée en incorporant les tilts, et les premiers résultats sur les angles de tilt sont analysés.

## 6 Bibliographie

### Références

- [1] Ni'am Tamami, E. Pitowarno et I Gede Puja Astawa , "Proportional derivative active force control for "X" configuration quadcopter", *Mechatronics, Electrical Power, and Vehicular Technology* 05, 2014, pp 67-74.
- [2] "Aerospace Blockset<sup>TM</sup> User's Guide", MathWorks R2020a, 2020, pp 879-885.
- [3] Alireza Nemati, "Designing, Modelling and Control of a Tilting Rotor Quadcopter", Cincinnati University, 2017.
- [4] Robert Mahony, Vijay Kumar et Peter Corke, "Multirotor Aerial Vehicles, Modeling, Estimation and Control of Quadrotor", *IEEE ROBOTICS & AUTOMATION MAGAZINE*, SEPTEMBER 2012, pp 20-32.
- [5] Gupte et al. "A survey of quadrotor Unmanned Aerial Vehicles." *Southeastcon, 2012 Proceedings of IEEE. IEEE*, 2012, pp. 1-6.
- [6] Grewal, Mohinder S., Lawrence R. Weill, and Angus P. Andrews, "Global positioning systems, inertial navigation, and integration", John Wiley & Sons, 2007.
- [7] Sujit Rajappa, Markus Ryll, Heinrich H. Bühlhoff, Antonio Franchi, "Modeling, Control and Design Optimization for a Fully-actuated Hexarotor Aerial Vehicle with Tilted Propellers", *IEEE ICRA 2015*, May 2015, Seattle, Washington, USA, United States.
- [8] Kumar, Rumit and Nemati, Alireza and Kumar, Manish and Sharma, Rajnikant and Cohen, Kelly and Cazaurang, Franck, "Tilting-Rotor Quadcopter for Aggressive Flight Maneuvers Using Differential Flatness Based Flight Controller", *Proceedings of the ASME Dynamic Systems and Control Conference*, October 11-13, 2017, Tysons, Virginia, USA.
- [9] Prouvost Patrick, "Automatique contrôle et régulation : cours et exercices corrigés", Dunod, impr. 2010.
- [10] Robert F. Stengel, "Flight Dynamics", Princeton University Press, October 2004.
- [11] <https://ardupilot.org/copter/docs/dji-f330-flamewheel.html>
- [12] <https://www.kampfgruppe144.com/phpBB3/viewtopic.php?f=39&t=4982>
- [13] Victor Manuel Aboytes Reséndiz, Edgar Rivas Araiza, "System Identification of a Quad-rotor in X Configuration from Experimental Data", 2016, Universidad Autónoma de Queretaro, Queretaro, Mexico.