

VINCENT MYRAND-LAPIERRE

**Transitions autonomes entre les vols non
stationnaire et stationnaire d'un véhicule aérien
miniature à ailes fixes**

Mémoire présenté
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de maîtrise en génie électrique
pour l'obtention du grade de Maître ès sciences (M.Sc.)

Faculté des sciences et de génie
UNIVERSITÉ LAVAL
QUÉBEC

2010

Résumé

Les véhicules aériens miniatures à ailes fixes (MiniAV) sont de petits avions avec une envergure d'ailes de moins de 1000 mm et pesant moins de 500 grammes. Grâce aux avancées qui ont été réalisées ces dernières années dans la miniaturisation des autopilotes et dans la propulsion électrique, il est maintenant possible de construire des MiniAVs qui peuvent être utilisés à des fins de reconnaissance en milieu restreint ou hostile. Pour réaliser cet objectif, le MiniAV doit être capable d'effectuer, de façon semi-autonome, des vols non stationnaires et stationnaires et être capable de réaliser des transitions autonomes entre ces modes. Ce mémoire décrit le développement d'une stratégie de contrôle pour permettre à un MiniAV de réaliser des transitions autonomes entre les modes de vol non stationnaire et stationnaire. Il est possible de diviser l'enveloppe de vol d'un MiniAV en 4 modes distincts : le mode non stationnaire, le mode non stationnaire vers stationnaire (L2H), le mode stationnaire et le mode stationnaire vers non stationnaire (H2L). Les structures des modèles pour les modes non stationnaires et stationnaires sont basées sur la linéarisation d'un modèle de MiniAV à corps rigide ayant 6 degrés de liberté. Les contrôleurs de ces deux principaux modes de vol sont présentés. Le mode L2H est géré par le contrôleur du mode non stationnaire, tandis que le mode H2L est géré par le contrôleur du mode stationnaire. Une approche systématique, appuyée par un superviseur basé sur la logique, est développée pour gérer les transitions entre les modes. La performance du superviseur est démontrée à travers des vols expérimentaux sur un banc de test. Il est montré que la stratégie proposée est capable de mieux performer que les méthodes rencontrées dans la littérature utilisant des plateformes similaires.

Abstract

Fixed-wing Mini Aerial Vehicules (MiniAV) are small airplanes with wingspans less than 1000 mm and weights less than 500 grams. With the advances in the miniaturization of autopilots and the improvement of electrical propulsion systems in recent years, it may be possible to build MiniAVs that can be used for reconnaissance in restricted or hostile areas. To achieve this objective, the MiniAV must be able to perform semi-autonomous level flight and hovering and be able to autonomously transition between these two modes. This MSc thesis describes the development of the control strategy needed to allow a fixed-wing MiniAV to autonomously transition between level and hovering flight. There are four flight modes during a mission : the level-flight mode, the level-flight to hovering mode (L2H), the hovering mode, and the hovering to level-flight mode (H2L). The model structures for both level-flight and hovering modes are based on the linearization of a six-degree-of-freedom rigid body model of a fixed-wing mini-aerial vehicle. Controllers for these two flying modes are presented. The L2H mode is managed by the level-flight controller, whereas the H2L is managed by the hovering controller. A systematic approach based on a logic-based switching supervisor is developed to handle the transition between modes. The performance of the switching supervisor is demonstrated through flight testing with a mini-aerial vehicle testbed. It was found that the proposed strategy is able to perform better than methods encountered in the literature who used similar platform.

Remerciements

En premier lieu, je voudrais remercier tout ceux qui m'ont assisté dans mes recherches : mon directeur de maîtrise, M. André Desbiens, mes co-directeurs, M. Eric Gagnon (RDDC Valcartier) et M. Éric Poulin. Ils ont toujours su prendre le temps de répondre patiemment à mes questions et de reviser mes ébauches d'articles scientifiques. Je voudrais souligner aussi le support logistique et scientifique de M. Franklin Wong au RDDC Valcartier et remercier M. Pierre-Richard Bilodeau qui m'a conseillé à plusieurs reprises et qui m'a aidé à réaliser certains essais. Enfin, je veux remercier le Fonds québécois de la recherche sur la nature et les technologies (FQRNT) et le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) pour leur support financier durant mes études graduées.

Évidemment, je veux aussi remercier mes parents qui m'ont supporté, financièrement et moralement, pendant ces (trop ?) longues années universitaires. Aussi, un merci très particulier à ma conjointe Marianne pour sa présence quotidienne durant ces derniers 20 mois. Finalement, je voudrais remercier mes souliers de course qui m'ont permis de m'entraîner et courir 3 marathons durant cette maîtrise.

Table des matières

Résumé	ii
Abstract	iii
Remerciements	iv
Table des matières	v
Table des figures	viii
Table des symboles	xii
1 Introduction	1
1.1 Revue de la littérature	3
1.2 Objectifs du projet	7
2 La plateforme et sa représentation mathématique	10
2.1 Définition des variables et repères	11
2.1.1 Repères	11
2.1.2 Vitesses, positions, accélérations et angles	12
2.1.3 Variables manipulées	14
2.1.4 Représentation en automatique	14
2.2 Angles d'Euler	15
2.3 Quaternions	17
2.3.1 Multiplication	18
2.3.2 Transformation des angles d'Euler vers les quaternions	22
2.3.3 Quaternion d'erreur de l'algorithme de Green [17, 18]	23
2.3.4 Transformation des quaternions vers les angles d'Euler	25
2.4 Équations auxiliaires	28
2.5 Équations de mouvement d'un corps rigide	29
2.5.1 Équations de translation dans le repère avion	29
2.5.2 Équations de rotation dans le repère avion	30
2.6 Équations de navigation	30

2.7	Présentation de la plateforme	31
2.8	Conclusion	32
3	Commande pour les vols non stationnaire et stationnaire	34
3.1	Vol non stationnaire	35
3.1.1	Linéarisation des équations 6-DOF	35
3.1.2	Fonctions de transfert de l'avion en mode non-stationnaire	38
3.1.3	Identification	43
3.1.4	Conception des autopilotes	54
3.1.5	Simulateur non-linéaire du NRC-CNRC	71
3.2	Vol stationnaire	78
3.3	Conclusion	81
4	Stratégies de transition	82
4.1	Équations de forces aérodynamiques	82
4.2	Analyse comparative des deux principaux modes de vol	84
4.2.1	Analyse comparative des fonctions de transfert	84
4.2.2	Utilisation des deux modes principaux pour une approche systématique	86
4.3	Superviseur basé sur la logique des états	87
4.3.1	Principe de la commutation sans à-coup	93
4.3.2	Modes semi-autonomes	94
4.3.3	Contrôle manuel des variables manipulées	94
4.3.4	Commutation des contrôleurs des différents modes de l'enveloppe de vol	96
4.4	Transition du vol non stationnaire vers le vol stationnaire (<i>L2H</i>)	99
4.4.1	Étape 1 - Consignes	100
4.4.2	Étape 2 - Commutation du contrôleur <i>hover</i>	101
4.4.3	Étape 3 - Contrer l'élan vers l'avant	101
4.4.4	Étape 4 - Contrôle de l'altitude	102
4.5	Transition du vol stationnaire vers le vol non stationnaire (<i>H2L</i>)	106
4.5.1	Étape 1 - Consignes	108
4.5.2	Étape 2 - Commutation du contrôleur <i>level</i>	108
4.6	Conclusion	109
5	Résultats expérimentaux en vol	110
5.1	Description du banc d'essais	110
5.1.1	Flatana	113
5.2	Description de la structure des algorithmes sous xPC-Target	116
5.2.1	Présentation des essais	120
5.3	Essais de transition du vol non stationnaire vers le vol stationnaire (L2H)	122

5.3.1	Angle de décrochage θ_{stall}	122
5.3.2	Variables latérales	124
5.3.3	Contrôle de l'altitude en boucle ouverte	127
5.3.4	Contrôle de l'altitude en boucle fermée avec le système de vision .	130
5.3.5	Contrôle de l'altitude en boucle fermée avec estimation de l'altitude	136
5.4	Essais de transition du vol stationnaire vers le vol non stationnaire (H2L)	142
5.4.1	δ_f constante	142
5.4.2	Augmentation de δ_f	145
5.5	Manoeuvres de transitions multiples	151
5.5.1	Manoeuvre : vol non stationnaire \rightarrow vol stationnaire \rightarrow vol non stationnaire	151
5.5.2	Manoeuvre : vol stationnaire \rightarrow vol non stationnaire \rightarrow vol sta- tionnaire	156
5.6	Conclusion	161
Conclusion		162
Bibliographie		165
A Identification		169
A.1	Identification et validation	169
A.2	Identification du NRC-CNRC	184
B Système de vision		186
C Articles		188
C.1	Article soumis pour l' <i>American Control Conference 2010</i>	188

Table des figures

1.1	États possibles d'un vol d'un MiniAV durant une mission	2
1.2	Application militaire : mission de reconnaissance (image tirée de elbit-systems.com)	3
1.3	Application civile : surveillance policière (image tirée de latimes.com) . .	3
1.4	Vol de transition de la plateforme de Green and Oh (image tirée de [18]).	4
1.5	Vol de transition de la plateforme de Frank et al. (image tirée de [12]) . .	4
1.6	Plateforme de Stone (image tirée de [37])	5
1.7	Vol de transition de la plateforme de Stone (image tirée de [1]).	6
2.1	Définition des axes et variables	11
2.2	Définition des repères	12
2.3	Définition des angles aérodynamiques	13
2.4	Variables manipulées	14
2.5	Diagramme fonctionnel du système	14
2.6	Séquence de rotation ψ, θ, ψ (les axes x, y, z représentent $X_E, Y_E, -Z_E$ et X représente $X_B, Y_B, -Z_B$). Image tirée de [41].	17
2.7	Exemple de rotation d'un vecteur Euclidien avec un quaternion	20
2.8	Flatana de la compagnie <i>Great Planes</i>	32
2.9	Caractéristiques typiques des composants du Flatana et placement par rapport au centre de gravité	33
3.1	Diagramme fonctionnel du système découplé longitudinal	37
3.2	Diagramme fonctionnel du système découplé latéral	38
3.3	Avion effectuant un virage	43
3.4	Processus d'identification	44
3.5	Manoeuvres d'excitation du δ_e	44
3.6	Manoeuvres d'excitation des δ_a	45
3.7	Manoeuvres d'excitation du δ_r	45
3.8	Gymnase utilisé pour réaliser les essais	45
3.9	Manoeuvres δ_e	47
3.10	Manoeuvres δ_a	47
3.11	Manoeuvres δ_r	48

3.12	Identification et validation de $\frac{q(s)}{\delta_e(s)}$	52
3.13	Diagramme fonctionnel d'un contrôleur longitudinal	55
3.14	Diagramme fonctionnel d'un contrôleur anti-windup	55
3.15	Approximation de $\frac{\theta(s)}{\delta_e(s)}$	56
3.16	Analyse fréquentielle - Réponse en fréquences de G_{θ}	58
3.17	Sortie du système θ en asservissement	58
3.18	Diagramme fonctionnel d'un contrôleur latéral	59
3.19	Diagramme fonctionnel de la première boucle interne	60
3.20	Fonctions de transfert vues par P_p et P_r	61
3.21	Analyse fréquentielle - Réponse en fréquences de $G_p(s)$ et $G_r(s)$	62
3.22	Sorties du système p et r en asservissement	63
3.23	Diagramme fonctionnel de la deuxième boucle interne	64
3.24	Approximation de ϕ suite à un changement de consigne de p	65
3.25	Analyse fréquentielle - Réponse en fréquences de G_{ϕ}	65
3.26	Simulateur linéaire - changement de consigne ϕ de 1 degré - variables latérales	66
3.27	Approximation de ψ suite à un changement de consigne de ϕ	67
3.28	Analyse fréquentielle - Réponse en fréquences de G_{θ}	68
3.29	Simulateur linéaire - changement de consigne (sans filtre) ψ de 1 degré - variables latérales	69
3.30	Simulateur linéaire - changement de consigne (avec filtre) ψ de 1 degré - variables latérales	70
3.31	Diagramme fonctionnel d'un contrôleur longitudinal pour maintenir une vitesse u et une altitude z constantes	73
3.32	Simulateur non-linéaire NRC - changement de consigne ϕ de 10 degrés - variables longitudinales	74
3.33	Simulateur non-linéaire NRC - changement de consigne ϕ de 10 degrés - variables latérales	75
3.34	Simulateur non-linéaire NRC - changement de consigne (sans filtre) ψ de 10 degrés - variables latérales	76
3.35	Simulateur non-linéaire NRC - changement de consigne (avec filtre) ψ de 10 degrés - variables latérales	77
3.36	Diagramme fonctionnel du calcul de l'attitude du MiniAV en mode stationnaire	79
3.37	Diagramme fonctionnel générique d'un contrôleur monovariable en mode stationnaire	80
3.38	Diagramme fonctionnel du contrôleur de l'axe X_B en mode stationnaire	80
3.39	Diagramme fonctionnel du contrôleur de l'axe Y_B en mode stationnaire	80
3.40	Diagramme fonctionnel du contrôleur de l'axe Z_B en mode stationnaire	80
3.41	Diagramme de blocs du contrôleur de l'altitude en mode stationnaire	81

4.1	Équations de forces - plan longitudinal	83
4.2	Analyse comparative des variables manipulées	85
4.3	Diagramme de la machine d'état du superviseur	88
4.4	Diagramme de la machine d'état du superviseur : exemple	90
4.5	Diagramme fonctionnel du processus de commutation sans à-coup	93
4.6	Diagramme fonctionnel du contrôle manuel des consignes	94
4.7	Diagramme fonctionnel du contrôle manuel des variables manipulées	94
4.8	Diagramme fonctionnel des différents modes de la commutation du contrô- leur longitudinal	96
4.9	Diagramme fonctionnel des différents modes de la commutation du contrô- leur latéral	96
4.10	Diagramme fonctionnel des différents modes et de la commutation du contrôleur du mode stationnaire	97
4.11	Diagramme fonctionnel des différents modes de la commutation du contrô- leur de l'altitude	97
4.12	Diagramme de la machine d'état du superviseur avec le principe de com- mutation, selon la valeur de t	98
4.13	Défis de la transition L2H	99
4.14	Défis de la transition H2L	107
5.1	Rampe pour le décollage	111
5.2	Banc d'essais	112
5.3	Flatana	114
5.4	Exemples de modifications apportées au Flatana	115
5.5	Blocs Simulink des deux structures de contrôleur ainsi que du superviseur	118
5.6	Superviseur basé sur la logique	119
5.7	Schéma des essais	121
5.8	Essai LEVEL \rightarrow L2H : variables longitudinales	123
5.9	Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) : variables latérales	126
5.10	Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle ouverte) : variables lon- gitudinales et latérales	129
5.11	Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) : variables longitudinales et latérales	132
5.12	Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) : position ($x - y$) et vitesse V_t	133
5.13	Images capturées d'une vidéo : transition L2H, vue d'en dessous de la rampe de lancement	134
5.14	Images capturées d'une vidéo : transition L2H, vue de côté	135

5.15	Reconstruction de la vitesse versus les données du système de vision : θ , u , v et w	137
5.16	Reconstruction de la position versus les données du système de vision : θ , x , y et z	138
5.17	Comparaison de l'attitude en utilisant les données du IMU versus celles du système de vision	138
5.18	Essai LEVEL \rightarrow L2H \rightarrow HOVER (z boucle fermée, reconstruction de z et commutation du système de vision) : variables longitudinales et latérales	141
5.19	Essai HOVER \rightarrow H2L (δ_t constante) \rightarrow LEVEL : variables longitudinales et latérales	144
5.20	Essai HOVER \rightarrow H2L ($\delta_t + 10\%$) \rightarrow LEVEL : variables longitudinales et latérales	147
5.21	Essai HOVER \rightarrow H2L ($\delta_t + 10\%$) \rightarrow LEVEL : position ($x - y$) et vitesse V_t	148
5.22	Images capturées d'une vidéo : transition H2L, vue d'en dessous de la rampe de lancement	149
5.23	Images capturées d'une vidéo : transition L2H, vue de côté	150
5.24	Essai LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow Navigation \rightarrow H2L \rightarrow LEVEL : variables longitudinales et latérales	153
5.25	Essai LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow Navigation \rightarrow H2L \rightarrow LEVEL : position ($x - y$) et vitesse V_t	154
5.26	Capture d'images d'une manoeuvre :LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow Navigation \rightarrow H2L \rightarrow LEVEL	155
5.27	Essai HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER : variables longitudinales et latérales	158
5.28	Essai HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER : position ($x - y$) et vitesse V_t	159
5.29	Capture d'images d'une manoeuvre HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER	160
A.1	Identification et validation de $\frac{\theta(s)}{\delta_e(s)}$	170
A.2	Identification et validation de $\frac{p(s)}{\delta_a(s)}$	173
A.3	Identification et validation de $\frac{p(s)}{\delta_r(s)}$	175
A.4	Identification et validation de $\frac{r(s)}{\delta_a(s)}$	177
A.5	Identification et validation de $\frac{r(s)}{\delta_r(s)}$	179
A.6	Identification et validation de $\frac{\phi(s)}{\delta_a(s)}$	181
A.7	Identification et validation de $\frac{\phi(s)}{\delta_r(s)}$	183
B.1	Tracking tools	187

Table des symboles

α	–	Angle d'attaque [<i>deg</i>]
β	–	Angle de dérive (<i>sideslip</i>) [<i>deg</i>]
$\delta_a, \delta_e, \delta_r$	–	Surfaces de contrôle (ailerons, gouverne de profondeur, gouverne de direction) [<i>deg</i>]
δ_f	–	Surface de propulsion [%]
γ	–	Pente [<i>deg</i>]
ε	–	Erreur sur la consigne
Δ	–	Petites perturbations
ϕ, θ, ψ	–	Positions angulaires (roulis, tangage, lacet) [<i>deg</i>]
a_x, a_y, a_z	–	Accélérations mesurées au centre de gravité [<i>m/s²</i>]
f_x, f_y, f_z	–	Forces aérodynamiques produites par le mouvement de l'avion [<i>N</i>]
g	–	Constante gravitationnelle 9.81 [<i>m/s²</i>]
k	–	Indice de temps discret
m	–	Masse du corps [<i>kg</i>]
l, m, n	–	Moments aérodynamiques totaux
op	–	Point d'opération
p, q, r	–	Vitesses angulaires [<i>deg/s</i>]
s	–	Opérateur de Laplace
q_0, q_1, q_2, q_3	–	Composants des quaternions
q_c	–	Quaternion de consigne
q_m	–	Quaternion de mesure
q_e	–	Quaternion d'erreur
u, v, w	–	Vitesses en translation [<i>m/s</i>]

x	–	État du système
y	–	Sortie du système
x, y, z	–	Positions de l'avion dans le repère NED
B	–	Matrice de transformation
C_L, C_D	–	Coefficients sans dimension de force (<i>portance et traînage</i>)
D	–	Vecteur de force de traînée [N]
G	–	Vecteur de force gravitationnelle [N]
G_c	–	Fonction de transfert d'un régulateur
G_p	–	Fonction de transfert d'un procédé
H	–	Fonction de transfert d'un procédé en boucle fermée
I	–	Matrice identité
J_x, J_y, J_z	–	Moment d'inertie du corps dans le référentiel de l'avion X_B, Y_B, Z_B [N]
J_{xy}	–	Produit d'inertie du corps par rapport aux axes X_B, Z_B
L	–	Vecteur de force de portance [N]
T	–	Vecteur de force de poussée [N]
V_T	–	Vitesse relative du corps par rapport à l'air [m/s]
X_B, Y_B, Z_B	–	Axes de référence de l'avion (<i>body axis</i>)
X_E, Y_E, Z_E	–	Axes de référence de la terre (<i>NED</i>)
ARX	–	Structure basée sur un modèle auto régressif qui inclut des entrées (<i>Auto Regressive model with eXternal inputs</i>)
AW	–	Contrôleur intégrant la saturation des actionneurs <i>anti-windup</i>
$ARMAX$	–	Structure d'un modèle auto régressif qui inclut une fonction de transfert avec une moyenne ajustable sur le bruit blanc (<i>Auto Regressive moving average with eXternal inputs</i>)
BJ	–	Structure <i>Box Jenkins</i>
CG	–	Centre de gravité
GPS	–	Géo-positionnement par satellite <i>Global Positioning System</i>
$Hover$	–	Mode stationnaire
HPF	–	Filtre passe-haut (<i>High Pass Filter</i>)
$H2L$	–	Transition du mode stationnaire vers le mode non stationnaire
IMU	–	Unité de mesure inertielle (<i>Inertial Measurement Unit</i>)
INS	–	Système de navigation inertielle (<i>Strap-down inertial navigation system</i>)
LAT	–	Mouvement latéral-directionnel du mode non stationnaire
$Level$	–	Mode non stationnaire
$LONG$	–	Mouvement longitudinal du mode non stationnaire

- LOOP* – Laboratoire d’Observation et Optimisation des Procédés - Université Laval
- L2H* – Transition du mode non stationnaire vers le mode stationnaire
- NRC – CNRC* – Conseil national de recherches Canada
- NED* – Référentiel inertiel de la terre (*North-East-Down*)
- MiniAV* – Véhicule aérien miniature
- MISO* – Système à entrées multiples et à une sortie (*Multi Input Single Output*)
- OE* – Structure basée sur l’erreur de sortie (*Output error*)
- P* – Régulateur Proportionnel
- PD* – Régulateur Proportionnel Dérivé
- PI* – Régulateur Proportionnel Intégral
- PID* – Régulateur Proportionnel Intégral Dérivé
- RDDC* – Recherche et développement pour la défense Canada
- TF* – Fonction de transfert
- trim* – Valeur d’équilibre des variables manipulées
- UAV* – Drone (*Unmanned Aerial Vehicle*)
- UDP* – Protocole de datagramme utilisateur (*User Datagram Protocol*)
- USB* – Bus de séries universelles (*Universal Serial Bus*)
- SISO* – Système à une entrée et à une sortie (*Single Input Single Output*)

Chapitre 1

Introduction

Les véhicules aériens miniatures (MiniAVs) à ailes fixes sont des avions de petite taille (envergure d'ailes de moins de 1000 millimètres) et de faible poids (moins de 500 grammes). Grâce aux avancées technologiques qui ont été réalisées ces dernières années, notamment dans le domaine de la miniaturisation des capteurs, les MiniAVs peuvent être utilisés à des fins de reconnaissance en milieu restreint ou hostile. Le haut niveau de performance des MiniAVs permet des modes opérationnels novateurs tels que le vol sans pilote à l'intérieur d'un édifice ou dans un milieu urbain. Pour ce faire, le véhicule doit être en mesure d'effectuer, de façon semi-autonome, des vols stationnaires et non stationnaires, et des transitions autonomes entre ces deux principaux modes.

Le mode non stationnaire est le mode classique du vol en palier (horizontal). Le mode stationnaire consiste à faire voler l'avion sur place, à la verticale, comme un hélicoptère. La figure 1.1 présente les différents états possibles de vol d'un véhicule capable de réaliser les deux modes de vols. Selon les besoins, le véhicule peut décoller ou atterrir à l'horizontal ou en mode stationnaire. Les transitions sont donc des trajectoires nécessaires pour passer entre les deux modes de vols.

Les plateformes de type MiniAV peuvent réaliser toutes sortes de missions. Du côté militaire (figure 1.2), on peut utiliser ces plateformes pour réaliser des missions de reconnaissance en milieu urbain ou hostile. Du côté des applications civiles (figure 1.3), les forces policières peuvent, en temps réel, faire de la surveillance de foule ou de la patrouille frontalière. Aussi, il est possible d'utiliser des MiniAVs pour faire de l'inspection de structures urbaines, comme des ponts ou des tours. Enfin, on peut réaliser des missions de sauvetage où des catastrophes naturelles, chimiques, biologiques ou nucléaires auraient eu lieu.

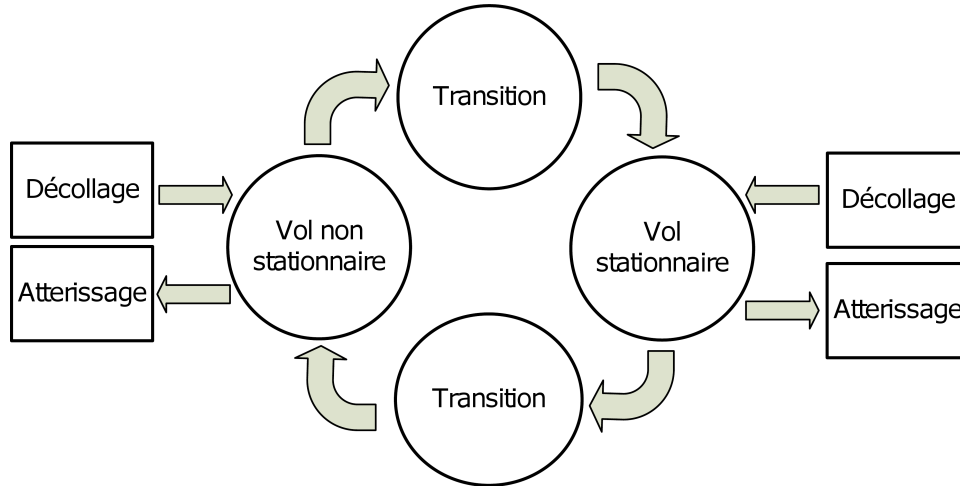


FIGURE 1.1 – États possibles d'un vol d'un MiniAV durant une mission

Ce projet de recherche s'inscrit dans la première phase d'un projet du RDDC Valcartier qui consiste à développer une plateforme, facilement portable par une personne, capable de réaliser de telles missions. Le choix du type de plateforme est donc très important. Un avion à ailes fixes, contrairement aux hélicoptères, n'est pas conçu pour réaliser des vols stationnaires et des décollages/atterrissages à la verticale. L'avantage d'utiliser un avion à ailes fixes réside donc dans son endurance en vol en palier. En effet, en utilisant le comportement aérodynamique naturel d'une telle plateforme (la force ascendante créée par l'écoulement de l'air), il est possible de se déplacer rapidement à l'horizontal, et ce, en consommant relativement peu d'énergie. De plus, on peut retrouver facilement dans la littérature des stratégies de contrôle de ce mode. Pour les plateformes conçues pour le vol stationnaire, comme les hélicoptères, les avions à voilure tournante (*rotary-wing*) ou les réacteurs à soufflante canalisée (*ducted-fan*), la force ascendante, qui permet le déplacement en palier à l'horizontal, provient principalement de la propulsion électrique ou au gaz. Ainsi, ces plateformes sont plus gourmandes en énergie que les avions à ailes fixes, mais elle sont plus manoeuvrables dans les milieux restreints.

La première phase du projet du RDDC Valcartier est de démontrer qu'il est possible de contrôler et stabiliser un avion à ailes fixes dans les différents modes de vol. Pour ce faire, il est nécessaire de développer des autopilotes semi-autonomes en mode de vol non stationnaire et en mode stationnaire. Ainsi, un utilisateur peut diriger le MiniAV à l'aide de commandes simples, sans se soucier de la problématique de la stabilisation. Il faut aussi développer des transitions autonomes entre les deux modes de vol principaux.

Quelques groupes de recherche ont travaillé sur des projets semblables à celui de RDDC Valcartier. Les principaux travaux ont été réalisés par William Green (Drexel



FIGURE 1.2 – Application militaire : mission de reconnaissance (image tirée de elbit-systems.com)



FIGURE 1.3 – Application civile : surveillance policière (image tirée de latimes.com)

Universiy, Philadelphie), par un groupe de recherche du *Massachusetts Institute of Technology* (Boston, Massachusetts), par Eric N. Johnson (*Georgia Institute of Technology*, Atlanta, Georgia), par Rolland Stone (*University of Sydney*, Australie) et par Nathan B. Knoebel (Brigham Young University, Provo, Utah). Les résultats de recherche sont présentés dans la section 1.1.

1.1 Revue de la littérature

Green and Oh [17] et Green [18] ont développé une plateforme semblable (poids, grandeur et capteur) à celle utilisée dans le cadre de ce projet. En effet, ils ont développé un avion à ailes fixes capable d'effectuer des vols stationnaires semi-autonomes ainsi que



FIGURE 1.4 – Vol de transition de la plateforme de Green and Oh (image tirée de [18]).

des transitions du vol en palier vers le vol stationnaire (figure 1.4). Les transitions se font uniquement du vol en palier vers le vol stationnaire. Il est à noter que le vol en palier doit être contrôlé de façon manuelle par un pilote d'expérience. De plus, lors de la transition, le pilote expérimenté doit contrôler manuellement la force de poussée du moteur. Enfin, quand le véhicule entre en mode stationnaire, il n'y a aucun contrôle en boucle fermée de l'angle de roulis. Aucun modèle aérodynamique de la plateforme a été identifié. Des régulateurs PD (proportionnel dérivé), qui ont été trouvés de façon itérative, sont utilisés pour stabiliser la plateforme en mode stationnaire ainsi que pour réaliser les transitions. Une centrale inertielle (IMU), le *3DM – GX1* de Microstrain, est utilisée pour générer l'attitude. Le circuit de contrôle consiste en un microcontrôleur (PIC16F87) qui communique avec le IMU via le protocole de communication *RS232*. Les quaternions sont utilisés pour représenter l'attitude de l'avion et pour calculer les efforts de commandes à appliquer aux variables manipulées. La conception du vol de transition est conçue pour que l'avion tourne sur un rayon d'au plus 1 m, et ce, en un temps de réponse de 2 secondes (figure 1.4). En fait, ceci permet à l'avion de voler à 5 *m/s* vers un obstacle et de l'éviter si celui-ci est détecté à au moins 1 m. Les vols ont été effectués à l'intérieur et à l'extérieur. Il est à noter que Green [18] a été le premier à réaliser des manoeuvres de transition pour une plateforme de type MiniAV.



FIGURE 1.5 – Vol de transition de la plateforme de Frank et al. (image tirée de [12])

Frank et al. [12], du Massachusetts Institute of Technology, ont aussi développé une plateforme expérimentale similaire, mais l'attitude et la position de l'avion sont uniquement obtenues par un système de caméra numérique (figure 1.5). Évidemment, ce type

de capteurs n'est pas utilisable dans un contexte de mission réelle. Les vols sont effectués à l'intérieur d'un laboratoire. Pour naviguer, les quaternions sont utilisés comme une représentation globale de l'attitude. Le modèle de l'avion est obtenu mathématiquement avec les équations de la dynamique et les différents coefficients aérodynamiques de forces et de moments. Les transitions sont réalisées dans les deux sens et des résultats satisfaisants ont été obtenus. La transition du mode en palier vers le mode stationnaire est réalisée en 6 secondes (contrôle de tous les axes autour de leurs consignes) avec une oscillation en altitude de 1.4 m. La transition du mode stationnaire vers le mode en palier est réalisée en moins de 1.5 secondes avec une chute en altitude d'environ 50 cm. Cette chute en altitude est due à une diminution du régime du moteur pour aider le véhicule à basculer vers l'avant. Un contrôleur gère le vol en palier et un autre contrôleur gère le mode stationnaire. Ainsi, lors des transitions, il y a commutation entre les deux principaux contrôleurs.

Johnson et al.[23, 24] ont aussi développé un avion à ailes fixes capable de voler dans les deux principaux modes de vol. L'avion, un *GTEdge UAS*, est propulsé à la gasoline, a 8.75 pieds d'envergure d'ailes et est long de 7.8 pieds. Cette plateforme est beaucoup plus grande que celle utilisée pour le projet RDDC Valcartier et les capteurs sont aussi beaucoup plus performants (autopilote avec un filtre de Kalman, GPS et capteur de pression). Un réseau de neurone adaptatif en temps réel a été développé afin qu'un contrôleur adaptatif soit capable de contrôler l'avion dans l'entière enveloppe de vol. Le système de contrôle de la plateforme, basé sur l'inversion de la dynamique et sur l'observation des états, est donc très complexe. Pour le contrôle de l'altitude, un capteur de pression est utilisé. Les vols sont réalisés à l'extérieur et les transitions entre les deux modes sont réalisées sur une période de 5 à 9 secondes, et ce, sur une distance horizontale de plus de 100 mètres et avec une variation en altitude d'environ 30 mètres.



FIGURE 1.6 – Plateforme de Stone (image tirée de [37])

Stone et al. [36] ont identifié une base de données aérodynamiques d'un *tail-sitter*, qui est une plateforme à décollage et atterrissage verticaux (figure 1.6). Ils utilisent un IMU avec un filtre de Kalman, un capteur de pression statique et un GPS. Ce véhicule est différent du point de vue de la forme et de la grosseur, mais la recherche sur les différentes stratégies de transition qui ont été réalisées et étudiées sont très pertinentes pour ce projet. Pour représenter l'attitude du *tail-sitter*, 3 représentations sont utilisées : les angles d'Euler normaux sont utilisés pour le mode en palier, les angles d'Euler verticaux sont utilisés pour le mode stationnaire et les quaternions sont utilisés lors des transitions entre les modes. Le système ne possède pas un contrôleur unique pour l'entière de l'enveloppe de vol. L'environnement Simulink et le *Mathworks Stateflow Toolbox* de Matlab sont utilisés pour développer et tester les différentes stratégies de vol. Cet outil permet de représenter les logiques des modes de vol comme une machine à états finis, avec les transitions entre les modes qui sont présentées graphiquement. Aussi, le *toolbox* génère automatiquement le code de la logique de contrôle qui est utilisé.



FIGURE 1.7 – Vol de transition de la plateforme de Stone (image tirée de [1]).

Stone et Clarke [37] ont aussi proposé une méthodologie pour optimiser les trajectoires des manoeuvres de transition. Les objectifs de l'optimisation des manoeuvres sont les suivants : les transitions doivent s'effectuer le plus rapidement possible pour remplir les missions du plan de vol, le véhicule doit gagner le moins de hauteur possible durant les manoeuvres de transition pour minimiser les conflits possibles dans l'espace aérien et pour éviter de se faire repérer, le véhicule doit éviter les angles d'attaque α trop agressifs dans des environnements incertains et, finalement, la plateforme doit éviter les situations de décrochage. En se basant sur les travaux de Stone, Anderson and Stone [1] ont développé une stratégie de commande prédictive. À chaque période d'échantillonnage, un modèle linéaire est créé, à partir du modèle non-linéaire, pour calculer les variables manipulées. Par conséquent, un contrôleur unique est utilisé pour le système en entier. Des essais en vol de Stone et al. [38] ont montré qu'il y avait un gain en altitude de 40-m pour les transitions du mode en palier vers le mode stationnaire et une perte de 5-m en altitude pour les transitions du mode stationnaire vers le mode en palier.

Knoebel [27] et Osborne [31] ont aussi développé des stratégies de transition pour un *tail-sitter*. Une approximation des coefficients aérodynamiques est utilisée comme modèle de l'avion et les capteurs à bord du véhicule sont semblables à ceux utilisés par l'équipe de Stone. Les quaternions sont utilisés pour représenter l'attitude. En utilisant des trajectoires pré-établies en 2 dimensions (trajectoires de hauteur h et de vitesse u), Osborne [31] a testé trois types de contrôleur : un contrôleur linéaire simple, un contrôleur adaptatif à retour d'état linéarisant (*feedback linearization*) et un contrôleur adaptatif. Les résultats d'essais en vol [31] sont similaires en grandeur à ceux de Stone et al. [38].

1.2 Objectifs du projet

M. Pierre-Richard Bilodeau est le premier étudiant gradué du LOOP (Laboratoire d'Observation et Optimisation des Procédés - Université Laval) à avoir participé au projet en mai 2007. Il a travaillé sur l'identification des structures des modèles et des contrôleurs en mode stationnaire [2, 3]. De plus, il a participé au processus itératif du choix de la plateforme. Enfin, il a aidé au développement du banc d'essais et de la structure de base des algorithmes de contrôle sous Matlab/Simulink. Les travaux réalisés dans cette thèse sont la suite logique des travaux débutés par M. Pierre-Richard Bilodeau, soit :

- Identifier, autour d'un point d'opération de vol, des modèles linéaires représentant les principaux comportements aérodynamiques du MiniAV en mode non stationnaire
- Concevoir et intégrer au système de commande des autopilotes capables de maintenir un vol nominal stable en palier
- Développer une méthodologie permettant de réaliser des manoeuvres de transition dans les deux sens
- Intégrer au système de commande un superviseur, basé sur la logique des états du système, capable de gérer l'ensemble de l'enveloppe de vol
- Réaliser des essais sur un banc de test.

Les objectifs de ce mémoire sont donc de proposer une méthode systématique pour réaliser des manoeuvres de transition dans les deux sens pour un véhicule aérien sans pilote. En utilisant des modèles linéaires simples représentant les principaux comportements aérodynamiques des deux principaux modes de vol et en utilisant des contrôleurs simples (de type PID), il sera démontré qu'il est possible de réaliser ce type de manoeuvres. Ainsi, on évite un processus ardu d'identifier un modèle non-linéaire complet

et de réaliser la conception d'une loi de commande non-linéaire capable de gérer l'ensemble de l'enveloppe de vol. Enfin, une structure de commutation simple entre les contrôleurs est aussi proposée via un superviseur basé sur la logique des états du système.

Les principales contributions de ce mémoire ont servi à l'élaboration d'un article scientifique qui a été accepté à l'*American Control Conference 2010*. Cet article [30], qui se retrouve à l'annexe C, propose une revue de littérature, une description du système et du superviseur, une présentation des contrôleurs et des structures des deux principaux modes de vol, ainsi que des résultats expérimentaux de vol pour les deux transitions.

Ce mémoire est divisé de la façon suivante :

- Dans le chapitre 2, les principaux axes, variables et équations qui décrivent le comportement d'un avion sont présentés. De plus, une description détaillée de la représentation de l'attitude des angles d'Euler et des quaternions est présentée et des exemples sont détaillés. Enfin, la plateforme utilisée dans le cadre du projet est décrite.
- Le chapitre 3 présente les deux modes de vol principaux. Tout d'abord, le principe de la linéarisation des équations à 6 degrés de liberté d'un corps rigide en vol de palier est présenté. Les différentes équations sont découplées et les principales fonctions de transfert en mode longitudinal et latéral sont approximées. Les étapes du processus d'identification (planification et réalisation de l'expérience, traitement des données et choix des structures) sont décrites. La conception des autopilotes, basée sur des structures connues dans la littérature, est présentée et des résultats en simulation sont illustrés. Enfin, le mode stationnaire (structures et contrôleurs) est brièvement introduit.
- Le chapitre 4 décrit la méthodologie systématique qui a été développée pour réaliser les transitions. Tout d'abord, une analyse comparative entre les deux principaux modes de vol est décrite. Le superviseur basé sur la logique, qui s'occupe du moment de commutation, est ensuite présenté. Par la suite, les stratégies des modes de vol L2H (transition du mode non stationnaire vers le mode stationnaire) et H2L (transition du mode stationnaire vers le mode non stationnaire) sont détaillées.
- Enfin, le chapitre 5 présente différents essais en vol qui ont été réalisés sur un banc d'essais à RDDC Valcartier. Certains de ces essais peuvent être visionnés sur le

CD-ROM inclus avec ce mémoire.

Chapitre 2

La plateforme et sa représentation mathématique

Ce chapitre présente la plateforme et la représentation mathématique qui lui est associée. Tout d'abord, les principaux repères (axes de référence) et les principales variables (vitesses, positions, accélérations, angles et variables manipulées) qui sont utilisés pour contrôler l'avion sont introduits. Par la suite, une description détaillée de la représentation de l'orientation de l'avion (l'attitude) selon le principe des angles d'Euler et des quaternions est présentée. Les angles d'Euler, qui sont utilisés pour représenter l'attitude en mode non stationnaire, sont une séquence de 3 rotations successives d'axes et sont une façon intuitive de représenter l'orientation de l'avion. Les quaternions, qui sont utilisés pour l'attitude en mode stationnaire, sont utiles pour représenter l'orientation d'un véhicule qui doit effectuer de larges manoeuvres angulaires. Les principes de base de la manipulation mathématique des quaternions sont décrits et des exemples sont présentés afin de comparer les différents algorithmes qui peuvent être utilisés pour faire le contrôle de l'orientation en mode stationnaire. En fin de chapitre, les 12 équations classiques qui expliquent le comportement de l'avion sont présentés : les équations auxiliaires (taux de variation de l'orientation en fonction des vitesses angulaires du corps), les équations de translation (ou de force) dans le repère avion, les équations de rotation (ou de moment) dans le repère avion et les équations de navigation. Finalement, la plateforme utilisée dans le cadre du projet est décrite.

2.1 Définition des variables et repères

Cette section définit les repères et les différentes variables du système. La figure 2.1 illustre ces définitions.

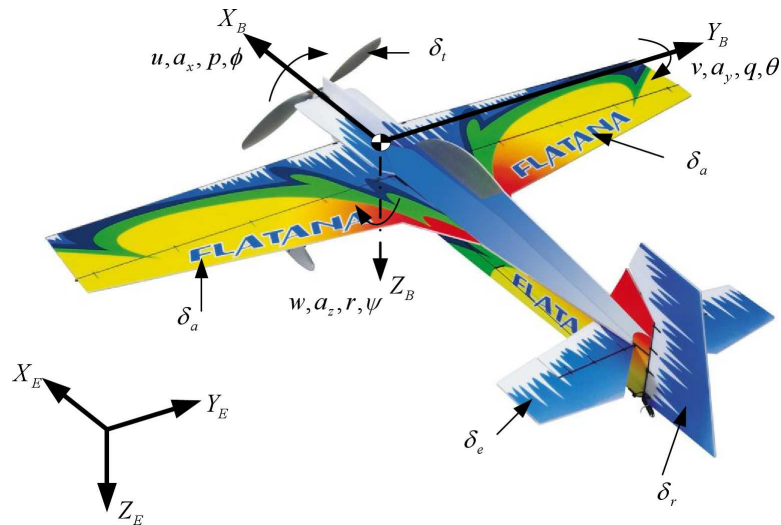


FIGURE 2.1 – Définition des axes et variables

2.1.1 Repères

Tout d'abord, on utilise le repère fixe *NED* (*north-east-down*) comme repère de base. On considère que la terre est plate. L'axe X_E pointe vers le Nord, l'axe Y_E pointe vers l'Est et l'axe Z_E pointe vers le sol. Dans la littérature, on peut aussi rencontrer la notation p_N, p_E, d . Les axes de référence de l'avion (*body axis*) ont comme origine le centre de gravité de l'avion où l'axe X_B pointe vers le nez (axe longitudinal), l'axe Y_B pointe vers l'aile droite (axe latéral) et l'axe Z_B pointe vers le ventre de l'avion (axe directionnel). On considère que ces axes sont fixes et que le plan X_B-Z_B est un plan de symétrie. La figure 2.2 présente ces repères.

Deux autres types d'axes existent. Il s'agit de l'axe de stabilité X_S (*stability axis*) et de l'axe de vent X_W (*wind axis*). Ces deux axes sont aussi appelés *axes aérodynamiques*. L'axe de stabilité représente une condition de vol stable, c'est-à-dire un axe de référence, à partir duquel une manoeuvre est effectuée ou lorsqu'une perturbation survient. L'axe X_S est obtenu grâce à une rotation de l'axe du corps Y_B d'un angle α_e . Cet angle d'attaque, α_e , est l'angle où l'avion est nominalement stable en vol. L'axe de vent

X_W est obtenu, quant à lui, grâce à une rotation de β de l'axe des Z_B pour ainsi aligner l'axe X_W avec le vent relatif. Ainsi, l'axe X_W est toujours tangent à la trajectoire du vent relatif (X_W n'est pas fixe). La figure 2.2 présente aussi ces axes.

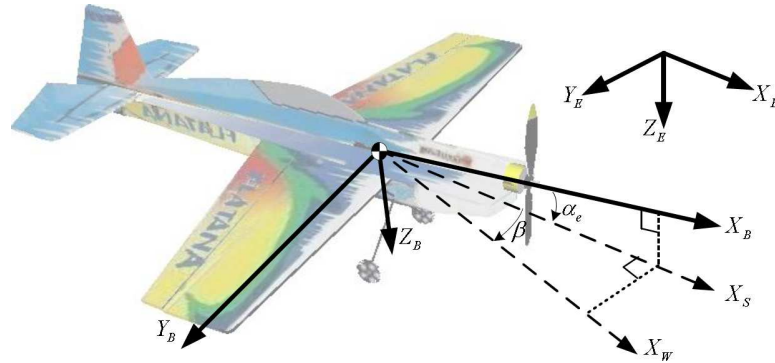


FIGURE 2.2 – Définition des repères

2.1.2 Vitesses, positions, accélérations et angles

Pour toutes les définitions, on utilise la règle de la main droite pour trouver le sens positif (le pouce vers la direction positive de l'axe).

Pour exprimer la vitesse angulaire dans le repère de l'avion, on utilise p, q, r , où :

- p [deg/s], selon l'axe X_B , est le taux de roulis (*roll rate*)
- q [deg/s], selon l'axe Y_B , est le taux de tangage (*pitch rate*)
- r [deg/s], selon l'axe Z_B , est le taux de lacet (*yaw rate*)

Pour exprimer la position angulaire dans le repère de l'avion (nommée aussi attitude), les principales méthodes qui sont utilisées dans l'aviation sont les angles d'Euler, les cosinus de direction et les quaternions. Cependant, les deux premières méthodes peuvent occasionner des singularités à certaines orientations critiques de l'avion (ex : les angles d'Euler et un angle de tangage de $\pm\pi/2$). La section 2.2 illustrera ce propos. Pour éviter ces désagréments, on utilise les quaternions. Ainsi, pour représenter l'attitude d'un avion, on utilise principalement les angles d'Euler ϕ, θ, ψ , où :

- ϕ [deg], selon l'axe X_B , est l'angle de roulis (*roll*)
- θ [deg], selon l'axe Y_B , est l'angle de tangage (*pitch*)
- ψ [deg], selon l'axe Z_B , est l'angle de lacet (*yaw*)

L'attitude de référence correspond à $\phi = 0$ deg, $\theta = 0$ deg et $\psi = 0$ deg lorsque le repère avion est aligné avec le repère NED. La section 2.2 présente en détail la représentation de l'attitude avec les angles d'Euler.

Pour exprimer l'accélération dans le repère de l'avion, on utilise a_x, a_y, a_z :

- $a_x [m/s^2]$ est selon l'axe X_B
- $a_y [m/s^2]$ est selon l'axe Y_B
- $a_z [m/s^2]$ est selon l'axe Z_B

Pour exprimer la vitesse en translation du centre de masse dans le repère de l'avion, on utilise u, v, w :

- $u [m/s]$ est selon l'axe X_B
- $v [m/s]$ est selon l'axe Y_B
- $w [m/s]$ est selon l'axe Z_B

Les angles aérodynamiques sont des angles très utiles pour définir les différents coefficients de force et de moment et certains différents axes. Ces angles, α (angle d'attaque) et β (angle de dérapage (*side-slip*)) sont en fonction de la vitesse de translation :

$$\alpha = \arctan\left(\frac{w}{u}\right) \quad (2.1)$$

$$\beta = \arcsin\left(\frac{v}{V_T}\right) \quad (2.2)$$

avec

$$V_T = \sqrt{u^2 + v^2 + w^2} \quad (2.3)$$

où $V_T [m/s]$ est la vitesse relative du corps par rapport à l'air. Ainsi :

$$u = V_T \cos \alpha \cos \beta \quad (2.4)$$

$$v = V_T \sin \beta \quad (2.5)$$

$$w = V_T \cos \beta \sin \alpha \quad (2.6)$$

La figure 2.3 présente les angles aérodynamiques. Aussi, on trouve que $\theta = \alpha + \gamma$ où γ est la pente (*path angle*). En montée, $\gamma > 0$ et, en descente, $\gamma < 0$.

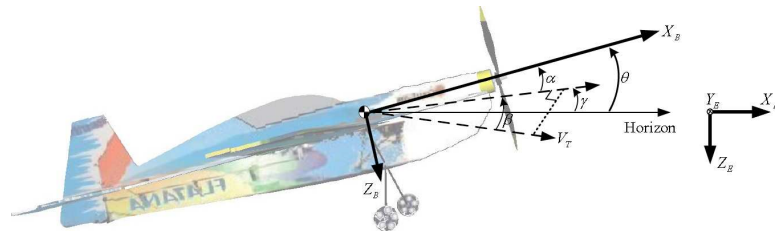


FIGURE 2.3 – Définition des angles aérodynamiques

2.1.3 Variables manipulées

Il y a quatre variables manipulées (figure 2.4) : les ailerons (*aileron*, δ_a), la gouverne de profondeur (*elevator*, δ_e), la gouverne de direction (*rudder*, δ_r) et la manette des gaz (*throttle*, δ_t). On les appelle aussi δ_a , δ_e et δ_r les surfaces de contrôle, les surfaces aérodynamiques, les gouvernes ou les volets primaires.

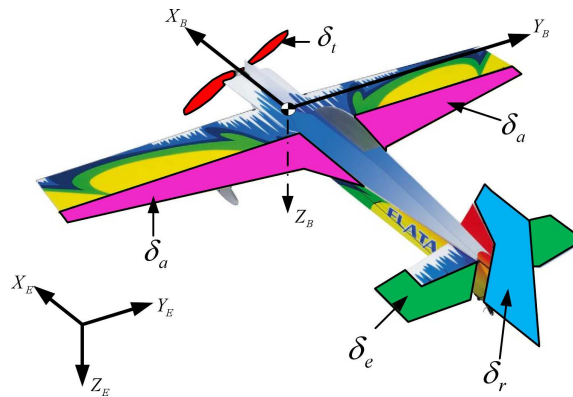


FIGURE 2.4 – Variables manipulées

Les ailerons δ_a agissent principalement sur X_B , la gouverne de profondeur δ_e sur Y_B et la gouverne de direction δ_r sur l'axe Z_B . Selon le mode de vol du système, les variables manipulées n'ont pas le même effet sur le système. La section 4.2 illustrera ce propos.

2.1.4 Représentation en automatique

Dans le domaine de l'automatique, l'avion est vu comme un système multivariable. La figure 2.5 présente un diagramme fonctionnel du système. Dans le cadre du projet, on considère nulles les perturbations atmosphériques.

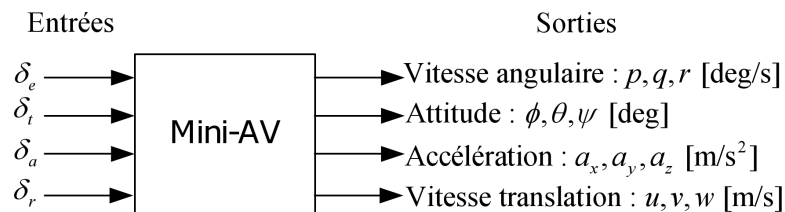


FIGURE 2.5 – Diagramme fonctionnel du système

2.2 Angles d'Euler

L'orientation d'un système de coordonnées cartésiennes par rapport à un autre peut toujours être décrite par 3 rotations successives. Les angles de rotation sont appelés *angles d'Euler*. Ces angles peuvent être exprimés de différentes façons, et ce, selon leur utilisation. Dans le domaine de l'aérospatial, les rotations sont effectuées dans un ordre spécifique selon les 3 axes cartésiens. En fait, on effectue trois rotations de plan.

Par exemple, supposons que les composants d'un vecteur $u^a = [x_a, y_a, z_a]^T$ sont connus dans le repère a . On veut connaître ces composants dans un repère b , c'est-à-dire $u^b = [x_b, y_b, z_b]^T$. Les repères a et b possèdent la même composante en z . Il existe un angle μ entre les plans $x_a - y_a$ et $x_b - y_b$. Ainsi, on trouve :

$$u^b = C_{b/a} u^a \quad (2.7)$$

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \begin{bmatrix} \cos \mu & \sin \mu & 0 \\ -\sin \mu & \cos \mu & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} \quad (2.8)$$

où $C_{b/a}$ est la matrice de rotation entre les plans $x_a - y_a$ et $x_b - y_b$. Les 0 et le 1 dans la matrice de rotation correspondent aux éléments qui ne changent pas lors de la rotation. Évidemment, la matrice de rotation est différente pour les 2 autres rotations de plan.

On utilise les angles d'Euler en aviation, car ceux-ci sont intuitifs et simples à utiliser. On utilise souvent la séquence de rotation $\psi \rightarrow \theta \rightarrow \phi$ (lacet, tangage, roulis). Ce type de notation est notamment utilisé pour définir l'attitude d'un avion lors de vols en palier. On utilise la règle de la main droite pour trouver le sens positif de la rotation (le pouce vers la direction positive de l'axe). Ainsi, à partir d'un axe de référence x, y, z , on effectue :

1. Rotation vers la droite par rapport à l'axe positif z (ψ positif)
2. Rotation vers la droite par rapport à l'axe y (θ positif)
3. Rotation vers la droite par rapport à l'axe x (ϕ positif)

Le système de référence NED et les axes de référence de l'avion sont alignés lorsque le devant de l'avion (axe X_B) pointe vers le Nord et que le côté droit (axe Y_B) de l'avion pointe vers l'Est. Les axes Z_E et Z_B sont donc colinéaires. Par exemple, une rotation en utilisant la règle de la main droite sur l'axe Z_B correspond à un lacet positif. On obtient donc, en effectuant les trois rotations successives :

$$u^b = C_{b/r} u^r \quad (2.9)$$

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r \\ y_r \\ z_r \end{bmatrix} \quad (2.10)$$

La matrice B_{Euler} ($C_{b/r}$) est la transformation complète du système de référence de la terre u^r vers le repère de l'avion u^b :

$$B_{Euler} = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ (-\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi) & (\cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi) & \sin \phi \cos \theta \\ (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) & (-\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi) & \cos \phi \cos \theta \end{bmatrix} \quad (2.11)$$

On remarque que la matrice B_{Euler} est orthogonale. Donc $B_{Euler}^{-1} = B_{Euler}^T$. Ainsi, pour passer du repère de l'avion u^b vers le repère de la terre u^r :

$$u^r = B_{Euler}^T u^b \quad (2.12)$$

La matrice B_{Euler} est aussi appelée matrice d'orientation, matrice de rotation ou matrice des cosinus de direction (direction cosine matrix, *DCM*). C'est d'ailleurs sous cette forme que le IMU calcule les orientations de l'avion. Il est évidemment possible d'obtenir les angles d'Euler directement :

$$\phi = \arctan \left(\frac{B_{Euler23}}{B_{Euler33}} \right) \quad (2.13)$$

$$\theta = \arcsin(-B_{Euler13}) \quad (2.14)$$

$$\psi = \arctan \left(\frac{B_{Euler12}}{B_{Euler11}} \right) \quad (2.15)$$

La figure 2.6 illustre cette séquence de rotation. Il est à noter que le principe de base de rotation des angles d'Euler ne garantit pas une séquence unique pour chaque orientation. Pour illustrer ce propos, on peut imaginer un avion qui effectue une boucle verticale (*loop*), la tête du pilote étant considérée comme le capteur d'inertie. Il est possible de représenter l'attitude de la trajectoire comme étant un intervalle continu de $-\pi < \theta < \pi$ et des angles de roulis et de lacet nuls. Cette notation n'étant pas pratique pour obtenir une représentation unique, il est possible de restreindre l'attitude en tangage à $\pm\pi/2$. De ce fait, jusqu'à $\pi/2$, l'attitude en ϕ et ψ est nulle. Cependant, dès que le véhicule dépasse $\pi/2$, l'angle de ϕ et de ψ change abruptement par π , comme si le cap de l'avion changeait de direction et que l'avion volait à l'envers. L'avion poursuit sa boucle et l'angle de θ diminue jusqu'à 0 degré lorsque l'avion est au plus haut de sa boucle. Lorsque l'avion atteint un angle de $-\pi/2$ (le nez de l'avion pointant vers le sol), les angles de ϕ et de ψ redeviennent 0 degré. Les angles d'Euler sont donc uniques, sauf dans le cas où l'angle de θ est exactement $\pm\pi/2$. Ainsi, $(\psi, \theta, \phi) = (\psi + \alpha, \pm\pi/2, -\alpha)$ donne la même orientation quelle que soit la valeur de α .

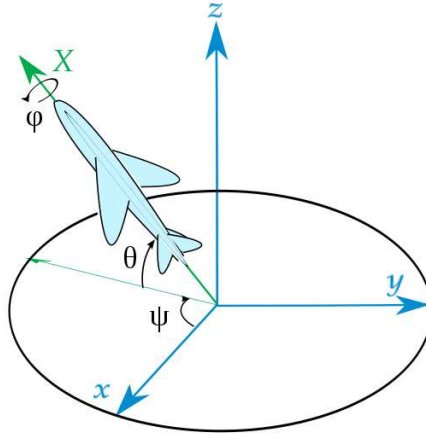


FIGURE 2.6 – Séquence de rotation ψ, θ, ψ (les axes x, y, z représentent $X_E, Y_E, -Z_E$ et X représente $X_B, Y_B, -Z_B$). Image tirée de [41].

2.3 Quaternions

À l'origine, les quaternions étaient utilisés pour généraliser les nombres complexes d'un espace en 2 dimensions en un espace en 3 dimensions. Les propriétés des quaternions permettent aussi d'effectuer la rotation d'un vecteur d'un espace euclidien de la même façon que les matrices de rotation. Les quaternions peuvent donc être utiles pour représenter l'orientation d'un véhicule qui doit effectuer de larges manoeuvres angulaires. Les quaternions consistent en 4 paramètres. Ils sont définis par :

$$q_0 = \cos(\Theta/2) \quad (2.16)$$

$$q_1 = e_x \sin(\Theta/2) i \quad (2.17)$$

$$q_2 = e_y \sin(\Theta/2) j \quad (2.18)$$

$$q_3 = e_z \sin(\Theta/2) k \quad (2.19)$$

où les 3 termes (q_1, q_2, q_3) représentent la partie vectorielle (i, j, k) du quaternion, q_0 est la partie scalaire, e_x, e_y, e_z représentent un vecteur unitaire qui est l'axe de rotation et Θ est l'angle de rotation appliqué à cet axe. Donc, la partie vectorielle du quaternion est utilisée pour définir la rotation de l'axe et la partie réelle sert à définir l'angle de rotation. Le quaternion a la forme suivante :

$$q = q_0 + q_1 i + q_2 j + q_3 k \quad (2.20)$$

De plus, un quaternion est un vecteur unitaire qui doit satisfaire la condition suivante :

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (2.21)$$

Cette formulation garantit qu'il existe un quaternion unique pour chaque valeur de $\Theta/2$ dans ± 180 degrés. Enfin, pour trouver l'inverse d'un quaternion, on n'a qu'à changer le signe des éléments vectoriels q_1 , q_2 et q_3 , ce qui donne :

$$q^* = q_0 + (-q_1)i + (-q_2)j + (-q_3)k \quad (2.22)$$

Le quaternion q^* est aussi appelé le quaternion *conjugué*. Il est à noter que $q_0 + q_1i + q_2j + q_3k = -q_0 - q_1i - q_2j - q_3k$. Par convention, on manipule les quaternions en utilisant le scalaire q_0 comme étant positif.

Exemple 1

Par exemple, le centre de gravité (CG) d'un MiniAV se dirige vers le Nord (son cap, dans le repère NED, est 0), son angle θ est de 30 degrés et son angle de roulis est nul. Par rapport à son repère $(\phi, \theta, \psi = 0)$, l'avion effectue une rotation dans l'axe Y_B de l'avion ($(e_x, e_y, e_z) = (0, 1, 0)$), et ce, d'un angle de θ de 30 degrés ($\frac{\pi}{6}$). Ainsi, l'attitude en quaternion est :

$$q_0 = \cos(30/2) = 0.966 \quad (2.23)$$

$$q_1 = 0 * \sin(30/2) = 0i \quad (2.24)$$

$$q_2 = 1 * \sin(30/2) = 0.258j \quad (2.25)$$

$$q_3 = 0 * \sin(30/2) = 0k \quad (2.26)$$

2.3.1 Multiplication

L'opérateur \otimes est utilisé pour définir la multiplication de deux quaternions (ne pas confondre avec le produit vectoriel ou scalaire). Il est à noter que la multiplication de deux quaternions n'est pas commutative. Le tableau 2.1 illustre les propriétés de la multiplication des coefficients pour expliquer la non-commutativité.

\times	i	j	k
i	-1	k	$-j$
j	$-k$	-1	i
k	j	$-i$	-1

TABLE 2.1 – Tableau de multiplication des coefficients entre eux

Par exemple, on remarque que $i^2 = j^2 = k^2 = ijk = -1$. Aussi, $ij = k$, mais $ji = -k$. Ainsi, en utilisant les propriétés du produit des coefficients, la multiplication entre deux

quaternions q_A et q_B est définie par :

$$\begin{aligned}
q_C = q_A \otimes q_B &= (q_{A0} + q_{A1}i + q_{A2}j + q_{A3}k) \otimes (q_{B0} + q_{B1}i + q_{B2}j + q_{B3}k) \\
&= q_{A0}q_{B0} - q_{A1}q_{B1} - q_{A2}q_{B2} - q_{A3}q_{B3} + \\
&\quad (q_{A1}q_{B0} + q_{A0}q_{B1} - q_{A3}q_{B2} + q_{A2}q_{B3})i + \\
&\quad (q_{A2}q_{B0} + q_{A3}q_{B1} + q_{A0}q_{B2} - q_{A1}q_{B3})j + \\
&\quad (q_{A3}q_{B0} - q_{A2}q_{B1} + q_{A1}q_{B2} + q_{A0}q_{B3})k
\end{aligned} \tag{2.27}$$

On peut utiliser la représentation de façon matricielle :

$$\begin{bmatrix} q_{C0} \\ q_{C1} \\ q_{C2} \\ q_{C3} \end{bmatrix} = \begin{bmatrix} q_{A0} & -q_{A1} & -q_{A2} & -q_{A3} \\ q_{A1} & q_{A0} & -q_{A3} & q_{A2} \\ q_{A2} & q_{A3} & q_{A0} & -q_{A1} \\ q_{A3} & -q_{A2} & q_{A1} & q_{A0} \end{bmatrix} \begin{bmatrix} q_{B0} \\ q_{B1} \\ q_{B2} \\ q_{B3} \end{bmatrix} \tag{2.28}$$

Pour illustrer le propos, on peut utiliser les quaternions pour effectuer la rotation d'un vecteur d'un espace Euclidien.

Tout d'abord, pour s'assurer de la compatibilité entre la multiplication d'un vecteur d'un espace Euclidien avec un quaternion, le vecteur Euclidien $\mathbf{u}_a = [x_a, y_a, z_a]^T$ est écrit sous la forme d'un quaternion avec la partie scalaire nulle :

$$q_{u_a} = \begin{bmatrix} 0 \\ x_a i \\ y_a j \\ z_a k \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{u}_a^r \end{bmatrix} \tag{2.29}$$

où \mathbf{u}_a^r représente les termes de la partie vectorielle du quaternion. Par conséquent, le résultat de la rotation d'un vecteur Euclidien par un quaternion $q_{b/a}$ doit être un quaternion q_{u_b} avec une partie scalaire nulle. La transformation doit aussi être réversible en utilisant le quaternion inverse et les propriétés de grandeur de l'espace Euclidien doivent être préservées. La transformation $q_{u_b} = q_{b/a} \otimes q_{u_a}$, comme le principe de transformation de l'équation 2.7, ne satisfait pas la première condition. Donc, on considère la transformation suivante :

$$q_{u_b} = q_{b/a}^{-1} \otimes q_{u_a} \otimes q_{b/a} \quad \text{ou} \quad q_{u_b} = q_{b/a} \otimes q_{u_a} \otimes q_{b/a}^{-1} \tag{2.30}$$

Si q_0 représente la partie scalaire du quaternion de rotation $q_{b/a}$ et \mathbf{q} représente les termes de la partie vectorielle, on trouve, après de savantes manipulations (Stevens & Lewis [35]) :

$$q_{u_b} = \begin{bmatrix} 0 \\ ((\mathbf{q} \cdot \mathbf{u}_a) \mathbf{q} + q_0(q_0 \mathbf{u}_a - \mathbf{q} \times \mathbf{u}_a) + (q_0 \mathbf{u}_a - \mathbf{q} \times \mathbf{u}_a) \times \mathbf{q})^r \end{bmatrix} \tag{2.31}$$

où \cdot est le produit scalaire et \times est le produit vectoriel. Toutes les conditions sont maintenant respectées.

Par exemple, le vecteur $u_a = (i + j + k)$ subit une transformation par le quaternion $q_{b/a} = (1 + j)/\sqrt{2}$. Ici, le vecteur u_a a subi une rotation de 90 degrés (dans le sens anti-horaire) par rapport à l'axe j . En utilisant l'équation 2.31, on trouve le vecteur $u_b = -i + j + k$. La figure 2.7 illustre cette transformation.

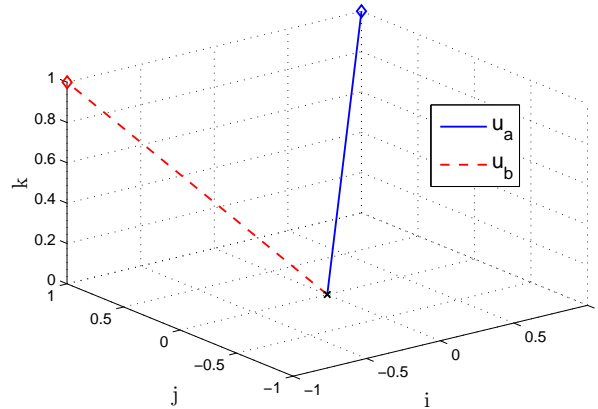


FIGURE 2.7 – Exemple de rotation d'un vecteur Euclidien avec un quaternion

Donc, la motivation derrière la multiplication est simple : celle-ci permet de définir la rotation d'un objet d'une orientation à une autre. Par exemple, si on utilise le quaternion q_A pour représenter l'orientation actuelle d'un corps et le quaternion q_{AB} pour représenter la rotation qui est nécessaire pour amener A vers l'orientation B , la nouvelle position, q_B , est calculée comme étant :

$$q_B = q_A \otimes q_{AB} \quad (2.32)$$

Dans le cadre du projet, le contrôle de l'attitude requiert la connaissance de l'orientation actuelle de l'avion, la consigne (une orientation désirée) et la différence d'orientation (erreur) entre les deux attitudes. On peut donc définir les 3 quaternions principaux :

- q_c : le quaternion de consigne (*commanded quaternion*) représente l'orientation voulue de l'avion (i.e. mode stationnaire, mode non stationnaire, une attitude de transition)
- q_m : le quaternion mesuré (*measured quaternion*) représente l'orientation actuelle de l'avion. Ce quaternion est obtenu à l'aide d'un capteur.
- q_e : le quaternion d'erreur (*error quaternion*) représente la rotation requise pour amener l'avion à la position désirée à partir de sa position actuelle.

En utilisant le même principe de l'équation 2.32, on a :

$$q_c = q_m \otimes q_e \quad (2.33)$$

Il faut trouver la solution de l'équation 2.33 pour q_e , car c'est ce quaternion d'erreur qui détermine l'effort de commande à appliquer. On trouve :

$$q_e = q_m^{-1} \otimes q_c \quad (2.34)$$

Par conséquent, l'équation 2.34 devient :

$$q_e = q_m^* \otimes q_c \quad (2.35)$$

Cette dernière équation représente le quaternion d'erreur de l'avion. Ainsi, les erreurs angulaires par rapport à chacun des axes peuvent être déduites de q_e . Une loi de contrôle peut alors modifier les variables manipulées pour contrôler l'orientation.

Exemple 2

Le MiniAV, dont le quaternion d'attitude a été défini dans le dernier exemple, doit effectuer une transition en mode stationnaire. Le quaternion de consigne en mode stationnaire est défini par une rotation de 90 degrés ($\frac{\pi}{2}$) de l'axe Y_B ($(e_x, e_y, e_z) = (0,1,0)$). Il y a une erreur de θ de 60 degrés. Le quaternion de consigne q_c est :

$$q_{0c} = \cos(90/2) = 0.707 \quad (2.36)$$

$$q_{1c} = 0 * \sin(90/2) = 0i \quad (2.37)$$

$$q_{2c} = 1 * \sin(90/2) = 0.7071j \quad (2.38)$$

$$q_{3c} = 0 * \sin(90/2) = 0k \quad (2.39)$$

En utilisant l'équation 2.35, on a :

$$q_e = q_m^* \otimes q_c \quad (2.40)$$

$$= (0.966 - 0i - 0.258j - 0k) \otimes (0.707 + 0i + 0.7071j + 0k) \quad (2.41)$$

$$= (0.866 + 0i + 0.5j + 0k) \quad (2.42)$$

On remarque que les termes i et k du quaternion d'erreur sont nuls, ce qui signifie que l'erreur de rotation est dans l'axe Y_B . Sachant que $\cos(\Theta/2) = 0.866$, on trouve $\Theta = 60$.

2.3.2 Transformation des angles d'Euler vers les quaternions

Un exemple simple où une rotation était appliquée à l'axe Y_B a été illustré dans la section 2.3. Évidemment, l'attitude d'un avion est rarement définie par la rotation sur un axe. Ainsi, pour des transformations multiples, on a :

$$q_{c/a} = q_{b/a} \otimes q_{c/b} \quad (2.43)$$

Si on suppose la séquence ψ, θ, ϕ , on obtient :

$$v^b = q_{roll}^{-1} \otimes q_{pitch}^{-1} \otimes q_{yaw}^{-1} \otimes v^r \otimes q_{roll} \otimes q_{pitch} \otimes q_{yaw} \quad (2.44)$$

avec :

$$q_{yaw} = \begin{bmatrix} \cos(\psi/2) \\ 0 \\ 0 \\ \sin(\psi/2) \end{bmatrix} \quad q_{pitch} = \begin{bmatrix} \cos(\theta/2) \\ 0 \\ \sin(\theta/2) \\ 0 \end{bmatrix} \quad q_{roll} = \begin{bmatrix} \cos(\phi/2) \\ \sin(\phi/2) \\ 0 \\ 0 \end{bmatrix} \quad (2.45)$$

où $v^r = 1 + 0i + 0j + 0k$ est le vecteur quaternion de référence des axes de l'avion ($\phi = 0, \theta = 0, \psi = 0, \Theta/2 = 0$) et v^b est le quaternion dans le repère avion. Ainsi, en utilisant les propriétés de multiplication des coefficients, on obtient :

$$\begin{aligned} q_0 &= \pm (\cos \phi/2 \cos \theta/2 \cos \psi/2 + \sin \phi/2 \sin \theta/2 \sin \psi/2) \\ q_1 &= \pm (\sin \phi/2 \cos \theta/2 \cos \psi/2 - \cos \phi/2 \sin \theta/2 \sin \psi/2) \\ q_2 &= \pm (\cos \phi/2 \sin \theta/2 \cos \psi/2 + \sin \phi/2 \cos \theta/2 \sin \psi/2) \\ q_3 &= \pm (\cos \phi/2 \cos \theta/2 \sin \psi/2 - \sin \phi/2 \sin \theta/2 \cos \psi/2) \end{aligned} \quad (2.46)$$

Le signe \pm signifie que cette transformation n'est pas affectée par le choix des angles et des signes. De manière générale, la rotation par un quaternion $q = q_0 + q_1i + q_2j + q_3k$ donne :

$$B_{quat} = \begin{bmatrix} (q_0^2 + q_1^2 - q_2^2 - q_3^2) & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & (q_0^2 - q_1^2 + q_2^2 - q_3^2) & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & (q_0^2 - q_1^2 - q_2^2 + q_3^2) \end{bmatrix} \quad (2.47)$$

Cette matrice B_{quat} transforme le repère NED vers celui de l'avion en quaternions. La matrice B_{quat} est aussi orthogonale, donc $B_{quat}^{-1} = B_{quat}^T$. Ainsi, on passe du repère de l'avion u^b vers le repère de la terre u^r avec les quaternions :

$$u^r = B_{quat}^T u^b \quad (2.48)$$

Exemple 3

Il est possible d'utiliser la propriété de l'équation 2.44 pour créer un quaternion de consigne dont l'axe Z_B du MiniAV pointe vers une direction précise. Par exemple, en mode stationnaire, on veut que le ventre d'un MiniAV pointe vers la direction 30^0 . Si on utilise la séquence de rotation ψ, θ, ϕ , on peut définir le quaternion de consigne en faisant une rotation de θ de 90 degrés, suivie d'une rotation de ϕ de -30 degrés. Ainsi, le quaternion de consigne q_c est :

$$q_{0c} = \cos(-30/2) \cos(90/2) + \sin(-30/2) \sin(90/2) = 0.6830 \quad (2.49)$$

$$q_{1c} = \sin(-30/2) \cos(90/2) - \cos(-30/2) \sin(90/2) = -0.1830i \quad (2.50)$$

$$q_{2c} = \cos(-30/2) \sin(90/2) + \sin(-30/2) \cos(90/2) = 0.6830j \quad (2.51)$$

$$q_{3c} = \cos(-30/2) \cos(90/2) - \sin(-30/2) \sin(90/2) = 0.1830k \quad (2.52)$$

2.3.3 Quaternion d'erreur de l'algorithme de Green [17, 18]

Pour réaliser la transition du vol non stationnaire vers le vol stationnaire, Green [17, 18] a développé un algorithme qui est utilisé pour initialiser la transition. L'objectif de cet algorithme est de générer un quaternion de consigne qui est découplé de la direction initiale. Ainsi, peu importe la direction vers laquelle l'avion se dirige ou vers le ventre de l'avion pointe, l'algorithme génère des quaternions d'erreur dans les axes Y_B et Z_B dans le but de garder l'avion à la verticale.

Lorsque la transition du vol non-stationnaire vers le mode stationnaire est initialisée, il y a une erreur angulaire entre l'axe X_B de l'avion et un vecteur vertical du repère NED . En assumant des conditions parfaites en vol non-stationnaire (i.e. $\phi = 0$, $\theta = 0$ et $-180 \leq \psi \leq 180$ degrés), cette erreur sera de 90 degrés en tangage. Un des objectifs de la transition est de, entre autres, faire pointer le ventre de l'avion en mode stationnaire vers la direction où l'avion se dirigeait en vol horizontal.

Pour y arriver, on génère un quaternion q_Δ qui représente la rotation nécessaire pour passer de l'orientation initiale à la bonne orientation verticale. En utilisant le quaternion q_Δ et le quaternion mesuré q_m , un quaternion de consigne q'_c représentant l'avion à la verticale, le ventre dans la direction où l'avion se dirigeait, peut être calculé. Un quaternion d'erreur peut donc être trouvé à chaque itération, ce qui générera une manoeuvre pure de tangage lors de la transition.

Assumons que la transition du vol non-stationnaire vers le vol stationnaire est ini-

tialisée. À l'aide du dernier quaternion mesuré q_m , la première étape consiste à calculer le quaternion q_Δ . Pour ce faire, on définit le vecteur de référence vertical V_{z_r} dans le repère NED :

$$V_{z_r} = 0i + 0j - 1k \quad (2.53)$$

On définit l'axe de l'avion V_{x_b} dans son propre repère,

$$V_{x_b} = 1i + 0j + 0k \quad (2.54)$$

L'étape suivante consiste à transformer le vecteur vertical V_{z_r} dans le référentiel de l'avion en utilisant le quaternion de mesure :

$$V_{z_b} = q_m^* \otimes V_{z_r} \otimes q_m \quad (2.55)$$

où V_{z_b} représente le vecteur V_{z_r} transformé dans le repère avion. Il est à noter que pour effectuer cette opération, on ajoute comme composant scalaire 0 à V_{z_r} (pour respecter la multiplication des quaternions) et on enlève la partie scalaire résultante de V_{z_b} .

Ensuite, on utilise les propriétés vectorielles pour trouver un axe de rotation orthogonal et l'axe entre les deux vecteurs. Le produit vectoriel est utilisé pour trouver l'axe de rotation, ou tout simplement, l'axe qui est orthogonal à ces deux vecteurs :

$$V_{rot} = V_{x_b} \times V_{z_b} \quad (2.56)$$

Ensuite, l'angle entre les axes V_{x_b} et V_{z_b} peut être trouvé en utilisant le produit scalaire :

$$\gamma = \cos^{-1}(V_{x_b} \cdot V_{z_b}) \quad (2.57)$$

L'angle γ et l'axe V_{rot} représente la rotation que doit effectuer l'avion pour que son axe V_{x_b} coïncide avec le vecteur vertical V_{z_b} . On peut convertir ces deux informations dans un quaternion, q_Δ :

$$q_{\Delta_1} = \cos(\gamma/2) \quad (2.58)$$

$$q_{\Delta_2} = V_{rot_1} \sin(\gamma/2) \quad (2.59)$$

$$q_{\Delta_3} = V_{rot_2} \sin(\gamma/2) \quad (2.60)$$

$$q_{\Delta_4} = V_{rot_3} \sin(\gamma/2) \quad (2.61)$$

Ce nouveau quaternion, q_Δ , est alors utilisé, avec le premier quaternion de mesure q_m , pour calculer le nouveau quaternion de consigne,

$$q'_c = q_m \otimes q_\Delta \quad (2.62)$$

Ce quaternion est donc utilisé comme le quaternion de consigne pour toutes les itérations. Le quaternion d'erreur de Green est donc :

$$q'_e = q_m^* \otimes q'_c \quad (2.63)$$

2.3.4 Transformation des quaternions vers les angles d'Euler

On peut trouver la valeur des angles d'Euler à partir des quaternions, en utilisant les équations 2.13 à 2.15 avec la matrice B_{quat} . Ainsi, on obtient :

$$\phi = \arccos \left(\frac{1 - 2(q_1^2 + q_2^2)}{\sqrt{1 - 4(q_0q_2 - q_1q_3)^2}} \right) = \arcsin \left(\frac{2(q_0q_1 + q_2q_3)}{\sqrt{1 - 4(q_0q_2 - q_1q_3)^2}} \right) \quad (2.64)$$

$$\theta = \arccos \left(\sqrt{1 - 4(q_0q_2 - q_1q_3)^2} \right) = \arcsin(2(q_0q_1 + q_2q_3)) \quad (2.65)$$

$$\psi = \arccos \left(\frac{1 - 2(q_2^2 + q_3^2)}{\sqrt{1 - 4(q_0q_2 - q_1q_3)^2}} \right) = \arcsin \left(\frac{2(q_1q_2 + q_0q_3)}{\sqrt{1 - 4(q_0q_2 - q_1q_3)^2}} \right) \quad (2.66)$$

Ainsi, les erreurs calculées par rapport à la consigne en mode stationnaire sont tout simplement des erreurs angulaires, sur chaque axe, par rapport à une attitude de consigne. On peut remarquer que si $\sqrt{1 - 4(q_0q_2 - q_1q_3)^2} = 0$, on retrouve une singularité. Cette possibilité survient si l'erreur en θ est de $\pm\pi/2$.

Il existe une façon d'approximer ces valeurs. Green [17, 18] utilise une approximation pour calculer les erreurs angulaires par rapport à la consigne en mode stationnaire. Pour ce faire, on estime les erreurs angulaires en extrayant l'angle de rotation de chaque composante vectorielle du quaternion :

$$\phi \approx \frac{2 \arccos(q_0) q_1}{\sqrt{q_1^2 + q_2^2 + q_3^2}} \quad (2.67)$$

$$\theta \approx \frac{2 \arccos(q_0) q_2}{\sqrt{q_1^2 + q_2^2 + q_3^2}} \quad (2.68)$$

$$\psi \approx \frac{2 \arccos(q_0) q_3}{\sqrt{q_1^2 + q_2^2 + q_3^2}} \quad (2.69)$$

Cette approximation est utile lorsque l'effort de calcul en ligne est limité.

Exemple 4

Par exemple, le CG d'un MiniAV est en transition du mode non stationnaire vers le mode stationnaire. Son angle de θ est de 70 degrés et son angle de roulis est de -15 degrés. En utilisant la matrice de tranformation de l'équation 2.46, l'attitude en

quaternion est :

$$q_{0m} = 0.8121$$

$$q_{1m} = -0.1069i$$

$$q_{2m} = 0.5687j$$

$$q_{3m} = 0.0749k$$

Le quaternion de consigne du mode stationnaire est $q_c = (0.707 + 0i + 0.7071j + 0k)$ (le ventre de l'avion pointant vers le nord). En utilisant l'équation 2.35, on a :

$$q_e = q_m^* \otimes q_c \quad (2.70)$$

$$= (0.8121 + 0.1069i - 0.5687j - 0.0338k) \otimes (0.707 + 0i + 0.7071j + 0k) \quad (2.71)$$

$$= (0.8586 + -0.1130i + 0.4957j - 0.0749k) \quad (2.72)$$

Ainsi, en utilisant les équations 2.64 à 2.66, on obtient les erreurs angulaires suivantes :

$$\phi = 15.9153 \quad (2.73)$$

$$\theta = 19.2910 \quad (2.74)$$

$$\psi = 5.3815 \quad (2.75)$$

En utilisant les approximations des équations 2.67 à 2.69, on obtient :

$$\phi = 14.8470 \quad (2.76)$$

$$\theta = 19.8852 \quad (2.77)$$

$$\psi = 2.6179 \quad (2.78)$$

On remarque que l'approximation des équations 2.67 à 2.69 donne des résultats similaires aux équations 2.64 à 2.66. Si on utilise le quaternion d'erreur de l'algorithme de Green (équation 2.63) avec les équations 2.64 à 2.66, on obtient les erreurs angulaires suivantes :

$$\phi = 0.9153 \quad (2.79)$$

$$\theta = 19.2910 \quad (2.80)$$

$$\psi = 5.3815 \quad (2.81)$$

On remarque que l'algorithme de Green donne les mêmes erreurs angulaires θ et ψ qu'avec le quaternion d'erreur de l'équation 2.35 en utilisant les équations 2.64 à 2.66 pour obtenir les erreurs angulaires. Cet algorithme suppose que la direction dans laquelle l'avion se dirige au début de la manoeuvre est la consigne de direction, d'où la faible erreur en ϕ . Cet algorithme génère un quaternion de consigne à chaque itération, peu importe sa direction.

Finalement, en utilisant l'approche des quaternions d'erreur, on remarque qu'il existe une erreur dans les 3 axes, même si l'attitude du MiniAV est représentée par seulement 2 rotations d'axes par rapport au repère NED. L'exemple suivant illustrera ce propos.

Exemple 5

Le CG d'un MiniAV est en transition du vol non stationnaire vers le mode stationnaire. Son angle de θ est de 50 degrés, sa direction est de $\psi = 30$ degrés et son angle de roulis est de -15 degrés. En utilisant la matrice de tranformation de l'équation 2.46, l'attitude en quaternion est :

$$\begin{aligned} q_{0m} &= 0.8537 \\ q_{1m} &= -0.2227i \\ q_{2m} &= 0.3741j \\ q_{3m} &= 0.2858k \end{aligned}$$

Le quaternion de consigne du mode stationnaire est $q_c = 0.6830 - 0.1830i + 0.6830j + 0.1830k$ (le ventre de l'avion pointant vers la direction 30 degrés). En utilisant l'équation 2.35, on a le quaternion d'erreur suivant :

$$q_e = 0.9317 + 0.1227i + 0.3391j - 0.0446k \quad (2.82)$$

Ainsi, en utilisant les équations 2.64 à 2.66, on obtient les erreurs angulaires suivantes :

$$\phi = 19.2790 \quad (2.83)$$

$$\theta = 38.3808 \quad (2.84)$$

$$\psi = 12.2529 \quad (2.85)$$

Intuitivement, on se serait attendu à ce que l'on retrouve seulement une erreur en tangage (40 degrés) ainsi qu'en roulis (15 degrés). Cependant, en utilisant le quaternion d'erreur, il existe des erreurs angulaires dans les 3 axes de l'avion. Dans le présent cas, on remarque une erreur non négligeable en lacet. Donc, lorsque l'erreur en tangage θ est grande par rapport à la consigne à la verticale, l'utilisation de cette stratégie fait en sorte que les erreurs en roulis et en lacet ne sont pas celles que l'on s'attend (une erreur en tangage de 40 degrés et en roulis de 15 degrés). Dans le cas de cet exemple, on voit qu'à $\theta = 50$ degrés et avec $\phi = -15$ degrés, la rotation en roulis crée une erreur en lacet par rapport à la consigne θ à la verticale. Les erreurs angulaires calculées sont différentes, car celles-ci sont des déviations calculées dans un plan cartésien en trois dimensions. La représentation des angles d'Euler, qu'ils soient horizontaux ou verticaux, est une séquence de rotation d'angles. La discontinuité qui apparaît est donc seulement due au changement de la représentation de l'attitude. Il n'existe pas de réelle discontinuité de l'attitude au niveau des angles d'Euler.

Si on utilise le quaternion d'erreur de l'algorithme de Green (équation 2.63) avec les

équations 2.64 à 2.66, on obtient les erreurs angulaires suivantes :

$$\phi = 4.2790 \quad (2.86)$$

$$\theta = 38.3808 \quad (2.87)$$

$$\psi = 12.2529 \quad (2.88)$$

On observe le même résultat qu'avec l'exemple précédent, soit une petite erreur en ϕ et des erreurs θ et ψ identiques au quaternion d'erreur de l'équation 2.35.

2.4 Équations auxiliaires

Les équations auxiliaires représentent le taux de variation des angles d'orientation en fonction des vitesses angulaires du corps de l'avion. En utilisant les angles d'Euler [35], on obtient :

$$\dot{\phi} = p + \tan \theta (q \sin \phi + r \cos \phi) \quad (2.89)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (2.90)$$

$$\dot{\psi} = (q \sin \phi + r \cos \phi) / \cos \theta \quad (2.91)$$

On peut aussi exprimer ces équations de cette façon :

$$p = \dot{\phi} - \dot{\psi} \sin \theta \quad (2.92)$$

$$q = \dot{\theta} \cos \phi + \dot{\psi} \cos \theta \sin \phi \quad (2.93)$$

$$r = \dot{\psi} \cos \theta \cos \phi - \dot{\theta} \sin \phi \quad (2.94)$$

On remarque ici que les équations 2.89 et 2.91 s'approchent de l'infini lorsque θ s'approche de $\pm\pi/2$. On peut remédier à ce problème en exprimant l'attitude avec les quaternions. Après quelques savantes manipulations [35], on trouve alors :

$$\dot{q}_0 = \frac{1}{2}(-pq_1 - qq_2 - rq_3) \quad (2.95)$$

$$\dot{q}_1 = \frac{1}{2}(pq_0 + rq_2 - qq_3) \quad (2.96)$$

$$\dot{q}_2 = \frac{1}{2}(qq_0 - rq_1 + pq_3) \quad (2.97)$$

$$\dot{q}_3 = \frac{1}{2}(rq_0 + qq_1 - pq_2) \quad (2.98)$$

2.5 Équations de mouvement d'un corps rigide

Les équations générales du mouvement d'un avion peuvent être décrites par la deuxième loi de Newton :

$$F = m\dot{V} \quad (2.99)$$

$$M = J\dot{\omega} \quad (2.100)$$

où F est la force appliquée, m est la masse du corps rigide, V est la vitesse en translation, M est le moment appliqué par rapport au centre de gravité, J est la matrice d'inertie et ω est la vitesse angulaire. Ces équations, qui sont sous forme de vecteurs, décrivent respectivement le mouvement de translation du centre de gravité et le mouvement de rotation par rapport au centre de gravité. Il y a donc un total de 6 équations, ce qui donne les 6 degrés de liberté de mouvement de l'avion.

2.5.1 Équations de translation dans le repère avion

Les équations de translation (ou de force), selon le vecteur d'état $[u, v, w]^T$ sont [35] :

$$\dot{u} = rv - qw + g_x + f_x/m \quad (2.101)$$

$$\dot{v} = -ru + pw + g_y + f_y/m \quad (2.102)$$

$$\dot{w} = qu - pv + g_z + f_z/m \quad (2.103)$$

où f_x, f_y et f_z sont les forces appliquées au corps de l'avion, m est la masse totale de l'avion et $[g_x, g_y, g_z]^T$ est la force de gravité appliquée dans le repère de l'avion. Ce vecteur est en fait :

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = B \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (2.104)$$

où B est une matrice de transformation qui transforme le vecteur de gravité NED vers le repère de l'avion et g ($9,81 \text{ m/s}^2$) est la constante gravitationnelle. En utilisant les angles d'Euler pour représenter l'attitude, on obtient :

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = B_{Euler} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} g(-\sin\theta) \\ g(\sin\phi\cos\theta) \\ g(\cos\phi\cos\theta) \end{bmatrix} \quad (2.105)$$

Ce qui donne :

$$\dot{u} = rv - qw + g(-\sin\theta) + f_x/m \quad (2.106)$$

$$\dot{v} = -ru + pw + g(\sin\phi \cos\theta) + f_y/m \quad (2.107)$$

$$\dot{w} = qu - pv + g(\cos\phi \cos\theta) + f_z/m \quad (2.108)$$

Avec les quaternions, on obtient :

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = B_{quat} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} 2g(q_1q_3 - q_0q_2) \\ 2g(q_2q_3 + q_0q_1) \\ g(q_0^2 - q_1^2 - q_2^2 + q_3^2) \end{bmatrix} \quad (2.109)$$

Ce qui donne :

$$\dot{u} = rv - qw + 2g(q_1q_3 - q_0q_2) + f_x/m \quad (2.110)$$

$$\dot{v} = -ru + pw + 2g(q_2q_3 + q_0q_1) + f_y/m \quad (2.111)$$

$$\dot{w} = qu - pv + g(q_0^2 - q_1^2 - q_2^2 + q_3^2) + f_z/m \quad (2.112)$$

2.5.2 Équations de rotation dans le repère avion

Les équations de rotation (ou de moment), qui ne sont pas en fonction des quaternions ou des angles d'Euler, sont exprimées de la façon suivante [35] :

$$\Gamma \dot{p} = J_{xz} [J_x - J_y + J_z] pq - [J_z (J_z - J_y) + J_{xz}^2] qr + J_z l + J_{xz} n \quad (2.113)$$

$$\dot{q} = (J_z - J_x) pr - J_{xz} (p^2 - r^2) + m \quad (2.114)$$

$$\Gamma \dot{r} = [(J_x - J_y) J_x + J_{xz}^2] pq - J_{xz} [J_x - J_y + J_z] qr + J_{xz} l + J_x n \quad (2.115)$$

$$\Gamma = J_x J_y - J_{xz}^2 \quad (2.116)$$

où l, m, n sont des moments appliqués au repère de l'avion X_B, Y_B, Z_B , J_x, J_y, J_z sont les inerties du repère X_B, Y_B, Z_B de l'avion et J_{xz} est le produit d'inertie entre X_B et Z_B .

2.6 Équations de navigation

Ces équations permettent au véhicule se s'orienter par rapport au repère NED . On utilise alors les matrices de transformation inversées. En utilisant les angles d'Euler pour représenter l'attitude, ces équations sont :

$$\begin{bmatrix} \dot{X}_E \\ \dot{Y}_E \\ \dot{Z}_E \end{bmatrix} = B_{Euler}^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.117)$$

$$\begin{aligned}\dot{X}_E &= u \cos \theta \cos \psi + v (-\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi) \\ &\quad + w (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi)\end{aligned}\quad (2.118)$$

$$\begin{aligned}\dot{Y}_E &= u \cos \theta \sin \psi + v (\cos \phi \cos \psi + \sin \phi \sin \theta \cos \psi) \\ &\quad + w (-\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi)\end{aligned}\quad (2.119)$$

$$\dot{Z}_E = -u \sin \theta + v \sin \phi \cos \theta + w \cos \phi \cos \theta \quad (2.120)$$

Pour exprimer ces équations avec les quaternions :

$$\begin{bmatrix} \dot{X}_E \\ \dot{Y}_E \\ \dot{Z}_E \end{bmatrix} = B_{quat}^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.121)$$

Finalement, on trouve :

$$\dot{X}_E = u (q_0^2 + q_1^2 - q_2^2 + q_3^2) + 2v (q_1 q_2 - q_0 q_3) + 2w (q_1 q_3 + q_0 q_2) \quad (2.122)$$

$$\dot{Y}_E = 2u (q_1 q_2 + q_0 q_3) + v (q_0^2 - q_1^2 + q_2^2 - q_3^2) + 2w (q_2 q_3 - q_0 q_1) \quad (2.123)$$

$$\dot{Z}_E = 2u (q_1 q_3 - q_0 q_2) + 2v (q_2 q_3 + q_0 q_1) + w (q_0^2 + q_1^2 + q_2^2 + q_3^2) \quad (2.124)$$

2.7 Présentation de la plateforme

Le MiniAV est un avion à ailes fixes fabriqué en carton-mousse. Il s'agit du *Flatana* de la compagnie *Great Planes* (figure 2.8). L'avion a une envergure d'ailes de 910 mm et il pèse 475 grammes avec tout l'équipement. Le système de propulsion consiste en un moteur sans balais (*brushless*), le RIMFIRE 22M-1000. Celui-ci est couplé à un contrôleur de vitesse électronique de 25A et une hélice en plastique de 10x3.8. Les surfaces aérodynamiques sont contrôlées par un receveur standard 72 MHz et des micros-servos.

L'attitude de la plateforme est générée par un IMU sans fils (unité de mesure d'inertie), le Microstrain 3DM-GX2. Cet IMU calcule et transmet l'orientation sous forme d'une matrice d'orientation à 100 Hz. La figure 2.9 présente les principaux composants de l'avion, leur poids, ainsi que leur position par rapport au centre de gravité. Il est très important que le poids soit bien réparti autour du centre de gravité naturel.

Un tube de Pitot peut aussi être installé sur l'avion. Il s'agit du *FDR PRO* de *Eagle Tree Systems*. On peut uniquement enregistrer les données en vol. On peut mesurer la vitesse V_T (précision de ± 1 pied/s) ainsi que l'altitude relative (initialisation au sol, précision de ± 1 pied). Finalement, un système de vision par caméra (OptiTrack par

FIGURE 2.8 – Flatana de la compagnie *Great Planes*

NaturalPoint) est utilisé pour obtenir l'attitude en quaternions ainsi que la position en mètre dans le plan $X_V - Y_V - Z_V$ du système de vision. Tout comme le repère NED $X_E - Y_E - Z_E$, on peut considérer l'origine du repère du système de vision comme étant arbitraire. Ainsi, les axes Z_E et Z_V sont colinéaires et il existe un angle μ entre les plan $X_E - Y_E$ et $X_V - Y_V$.

2.8 Conclusion

Dans ce chapitre, la plateforme et la représentation mathématique qui décrit son comportement ont été présentées. Les principaux repères et les principales variables utiles au contrôle ont été introduits. Par la suite, une description détaillée de la représentation de l'orientation de l'avion selon le principe des angles d'Euler et des quaternions a été présentée. Les principes de base des angles d'Euler et des quaternions ont été décrits et des exemples ont été présentés. Enfin, les 12 équations classiques qui expliquent le comportement d'un avion ont été énumérés (équations auxiliaires, équations de translation dans le repère de l'avion, équations de rotation dans le repère de l'avion, équations de navigation). La plateforme utilisée dans le cadre du projet a aussi été décrite. Ces définitions du système sont donc une première étape dans le processus du développement d'une stratégie de commande pour les vols non stationnaires et stationnaires. Le prochain chapitre décrit ce propos.

Sous-système	Item	Poids (g)	Placement		
			axe XB (mm)	axe YB (mm)	Axe ZB (mm)
Capteurs	Microstrain IMU + antenne	34	-45	0	-20
	Régulateur 5V	5	80	15	-20
	Batt 7.4V, 300mAh	24	75	-17	25
	Poids capteurs total	63			
Avion	Avions vide + boules réflex.	237			
	Rimfire 22M-1000 motor	45	140	0	0
	Phoenix 25A ESC	22	45	-17	25
	GP 10x3.5	10	145	0	0
	Batt, 11.1V, 910 mAh	65	-15	0	20
	Poids avion total	379			
Poids total		442			
C.G. (ideal)			-45	0	-20

FIGURE 2.9 – Caractéristiques typiques des composants du Flatana et placement par rapport au centre de gravité

Chapitre 3

Commande pour les vols non stationnaire et stationnaire

Ce chapitre présente la méthodologie qui est utilisée pour développer les lois de commande pour les vols non stationnaires et stationnaires. Tout d'abord, en utilisant les 12 équations non-linéaires du comportement d'un avion présentées dans le chapitre 2, une linéarisation est utilisée pour approximer le comportement de l'avion autour d'un point d'opération. On obtient alors deux catégories de mouvement : les mouvements longitudinaux et les mouvements latéraux. Ainsi, les principaux comportements aérodynamiques et leurs caractéristiques, sous forme de fonctions de transfert, sont présentées et détaillées. Cette connaissance linéarisée du comportement du système permet ensuite de réaliser l'identification. Les principales étapes pour réaliser l'identification (planification, réalisation, traitements des données et choix des structures) sont décrites et des exemples de données recueillies et traitées sont présentés. À partir de ces modèles identifiés, il est alors possible de concevoir les autopilotes. Ceux-ci sont basés sur des structures bien connues dans la littérature. Une méthode fréquentielle est utilisée pour le réglage des contrôleurs et des résultats en simulation sont présentés pour valider le résultat. Finalement, un résumé de la stratégie pour la commande du vol stationnaire réalisé par Bilodeau et al. [2, 3] est présenté.

3.1 Vol non stationnaire

3.1.1 Linéarisation des équations 6-DOF

Comme il a été montré dans le chapitre 2, il existe 12 équations pour expliquer le comportement d'un avion. Les lois de la mécanique et de l'aérodynamique produisent un modèle non-linéaire. Il est donc difficile d'obtenir un modèle complet de l'avion dans son enveloppe de vol. La linéarisation permet donc d'approximer le comportement du système non-linéaire autour d'un point d'opération (une condition de vol). La linéarisation, à partir des équations à 6 degrés de liberté (référentiel NED et axes du corps fixes) qui est présentée est tirée de Etkin et Reid [9].

Tout d'abord, on considère l'avion rigide et qu'il possède des axes de référence fixes au centre de gravité de sorte que $X_B - Z_B$ est un plan de symétrie. On dit alors que les axes de référence sont les axes de stabilisation. Les équations sont linéarisées selon le principe des petites perturbations (*small disturbances*). Les perturbations se produisent quand les ailes de l'avion sont en position nominale et que l'avion est en condition d'équilibre. Les variables sont substituées par leurs points d'opération et additionnées d'une perturbation. Par exemple, $X_E = X_{Eop} + \Delta X_E$. Dans des conditions de vol stable non stationnaire, on peut affirmer que les points d'opération suivants sont nuls :

$$\phi_{op} \equiv 0, \psi_{op} \equiv 0, v_{op} \equiv 0, w_{op} \equiv 0, p_{op} \equiv 0, q_{op} \equiv 0, r_{op} \equiv 0, \quad (3.1)$$

Les autres valeurs sont supposées non nulles. Aussi, il est nécessaire d'utiliser les approximations des petits angles. Par exemple, $\cos(\Delta\phi) \approx 1$ et $\sin(\Delta\psi) \approx \Delta\psi$. On néglige aussi le produit des variables de perturbation.

Considérons la linéarisation de la première équation des équations de navigation (équation 2.118). On définit \dot{x} le vecteur d'état et la fonction f_1 qui est une des 12 équations qui définit le mouvement d'un avion :

$$\dot{x} = f_1(X_E, Y_E, Z_E, \phi, \theta, \psi, u, v, w, p, q, r, \delta a, \delta e, \delta r, \delta t) \quad (3.2)$$

$$= u \cos \theta \cos \psi + v (-\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi) \\ + w (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \quad (3.3)$$

Pour linéariser cette équation, on utilise la première approximation de la série de Taylor :

$$\Delta \dot{x} = \frac{\partial f_1}{\partial x} \Delta x \quad (3.4)$$

$$\begin{aligned} &= \left. \frac{\partial f_1}{\partial \phi} \right|_{op} \Delta \phi + \left. \frac{\partial f_1}{\partial \theta} \right|_{op} \Delta \theta + \left. \frac{\partial f_1}{\partial \psi} \right|_{op} \Delta \psi + \left. \frac{\partial f_1}{\partial u} \right|_{op} \Delta u + \left. \frac{\partial f_1}{\partial v} \right|_{op} \Delta v \\ &\quad + \left. \frac{\partial f_1}{\partial w} \right|_{op} \Delta w \end{aligned} \quad (3.5)$$

$$= -u_{op} \sin \theta_{op} \Delta \theta + \cos \theta_{op} \Delta u + \sin \theta_{op} \Delta w \quad (3.6)$$

En utilisant les approximations des petits angles et en remplaçant les variables par leurs points d'opération additionnés d'une perturbation, on peut réécrire l'équation 3.6 :

$$\dot{X}_{Eop} + \Delta \dot{X}_E \approx -u_{op} \sin \theta_{op} \Delta \theta + (u_{op} + \Delta u) \cos \theta_{op} + \sin \theta_{op} \Delta w \quad (3.7)$$

En mettant les valeurs de perturbation Δ à 0, on obtient les valeurs nominales. En effet :

$$\dot{X}_{Eop} = u_{op} \cos \theta_{op} \quad (3.8)$$

En remplaçant l'équation 3.8 dans 3.7, on obtient l'équation finale linéarisée :

$$\Delta \dot{X}_E = \cos \theta_{op} \Delta u - u_{op} \sin \theta_{op} \Delta \theta + \sin \theta_{op} \Delta w \quad (3.9)$$

On effectue les mêmes manipulations pour les autres équations de navigation, de cinématique, de moment et de force. Plusieurs hypothèses de simplification sur les conditions de vols sont formulées et les équations sont réarrangées. Par conséquent, la dynamique du système ne dépend pas de X_E , Y_E , Z_E ni de ψ . La linéarisation des équations permet aussi de découpler ces équations en deux catégories : les mouvements longitudinaux et les mouvements latéraux-directionnels. Il est à noter que la notion de découplage n'est valide que si le plan X_B, Z_B est un plan de symétrie. On peut aussi simplifier ces équations linéarisées en utilisant les coefficients de dérivée de stabilité.

Mouvements longitudinaux

Ce sont des mouvements dans l'axe Y_B (mouvements avant-arrière, haut-bas). Les quantités utiles sont u , w , q , θ , δ_e et δ_t . Les vecteurs d'état et de contrôle longitudinaux sont :

$$x_{LONG} = \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta \theta \end{bmatrix} \quad \text{et} \quad u_{LONG} = \begin{bmatrix} \Delta \delta_e \\ \Delta \delta_t \end{bmatrix}$$

Les équations linéarisées longitudinales prennent donc la forme suivante :

$$\dot{x}_{LONG} = A_{LONG}x_{LONG} + B_{LONG}u_{LONG} \quad (3.10)$$

$$\begin{bmatrix} \Delta \dot{u} \\ \Delta \dot{w} \\ \Delta \dot{q} \\ \Delta \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{m}X_u & \frac{1}{m}X_w & 0 & -g \cos \theta_{op} \\ \frac{Z_u}{m-Z_{\dot{w}}} & \frac{Z_w}{m-Z_{\dot{w}}} & \frac{Z_q + mu_{op}}{m-Z_{\dot{w}}} & -\frac{mg \sin \theta_{op}}{m-Z_{\dot{w}}} \\ \frac{1}{J_y} \left(M_u + \frac{M_w Z_u}{m-Z_{\dot{w}}} \right) & \frac{1}{J_y} \left(M_w + \frac{M_{\dot{w}} Z_w}{m-Z_{\dot{w}}} \right) & \frac{1}{J_y} \left(M_q + \frac{M_{\dot{w}} (Z_q + mu_{op})}{m-Z_{\dot{w}}} \right) & \frac{1}{J_y} \left(\frac{M_{\dot{w}} (-mg \sin \theta_{op})}{m-Z_{\dot{w}}} \right) \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta \theta \end{bmatrix} + \begin{bmatrix} \frac{1}{m}X_{\delta_e} & \frac{1}{m}X_{\delta_t} \\ \frac{Z_{\delta_e}}{m-Z_{\dot{w}}} & \frac{Z_{\delta_t}}{m-Z_{\dot{w}}} \\ \frac{1}{J_y} \left(M_{\delta_e} + \frac{M_{\dot{w}} Z_{\delta_e}}{m-Z_{\dot{w}}} \right) & \frac{1}{J_y} \left(M_{\delta_t} + \frac{M_{\dot{w}} Z_{\delta_t}}{m-Z_{\dot{w}}} \right) \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \delta_e \\ \Delta \delta_t \end{bmatrix} \quad (3.11)$$

La figure 3.1 présente un diagramme fonctionnel du système découplé longitudinal.

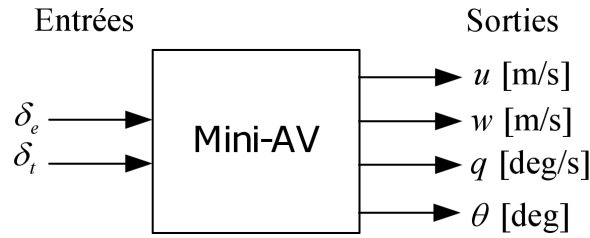


FIGURE 3.1 – Diagramme fonctionnel du système découplé longitudinal

Mouvements latéraux

Ce sont des mouvements dans les axes X_B et Z_B (gauche-droite-tourner). Les quantités utiles sont v , p , r , ϕ , δ_a et δ_r . Les vecteurs d'état et de contrôle latéraux sont :

$$x_{LD} = \begin{bmatrix} \Delta v \\ \Delta p \\ \Delta r \\ \Delta \phi \end{bmatrix} \quad \text{et} \quad u_{LD} = \begin{bmatrix} \Delta \delta_a \\ \Delta \delta_r \end{bmatrix}$$

Les équations linéarisées latérales prennent donc la forme suivante :

$$\dot{x}_{LD} = A_{LD}x_{LD} + B_{LD}u_{LD} \quad (3.12)$$

$$(3.13)$$

$$\begin{aligned}
\begin{bmatrix} \Delta \dot{v} \\ \Delta \dot{p} \\ \Delta \dot{r} \\ \Delta \dot{\phi} \end{bmatrix} &= \begin{bmatrix} \frac{1}{m} Y_v & \frac{1}{m} Y_p & \frac{1}{m} Y_r - u_{op} & g \cos \theta_{op} \\ \frac{J_z L_v + J_{xz} N_v}{J_x J_z - J_{xy}^2} & \frac{J_z L_p + J_{xz} N_p}{J_x J_z - J_{xy}^2} & \frac{J_z L_r + J_{xz} N_r}{J_x J_z - J_{xy}^2} & 0 \\ \frac{J_{xz} L_v + J_x N_v}{J_x J_z - J_{xy}^2} & \frac{J_{xz} L_p + J_x N_p}{J_x J_z - J_{xy}^2} & \frac{J_{xz} L_r + J_x N_r}{J_x J_z - J_{xy}^2} & 0 \\ 0 & 1 & \tan \theta_{op} & 0 \end{bmatrix} \begin{bmatrix} \Delta v \\ \Delta p \\ \Delta r \\ \Delta \phi \end{bmatrix} + \\
&+ \begin{bmatrix} \frac{1}{m} Y_{\delta_a} & \frac{1}{m} Y_{\delta_r} \\ \frac{J_z L_{\delta_a} + J_{xz} N_{\delta_a}}{J_x J_z - J_{xy}^2} & \frac{J_z L_{\delta_r} + J_{xz} N_{\delta_r}}{J_x J_z - J_{xy}^2} \\ \frac{J_{xz} L_{\delta_a} + J_x N_{\delta_a}}{J_x J_z - J_{xy}^2} & \frac{J_{xz} L_{\delta_r} + J_x N_{\delta_r}}{J_x J_z - J_{xy}^2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \delta_a \\ \Delta \delta_r \end{bmatrix} \quad (3.14)
\end{aligned}$$

La figure 3.2 présente un diagramme fonctionnel du système découplé longitudinal.

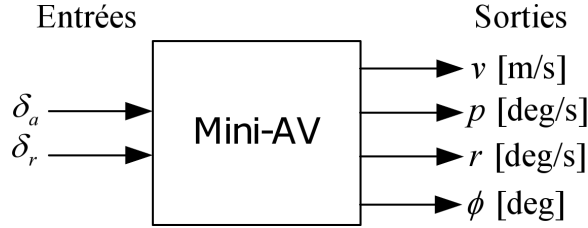


FIGURE 3.2 – Diagramme fonctionnel du système découplé latéral

3.1.2 Fonctions de transfert de l'avion en mode non-stationnaire

En partant du principe que :

$$Y(s) = G(s)U(s) \quad (3.15)$$

$$= [C(sI - A)^{-1}B + D]U(s) \quad (3.16)$$

avec $C = I$ et $D = 0$, on a :

$$G(s) = (sI - A)^{-1}B \quad (3.17)$$

Réponses longitudinales

En utilisant les matrices A_{LONG} et B_{LONG} ,

$$G(s) = (sI - A_{LONG})^{-1}B_{LONG} \quad (3.18)$$

$$(3.19)$$

$$G(s) = \frac{1}{Den_{LONG}(s)} \begin{bmatrix} Num_{\delta_e}^u(s) & Num_{\delta_t}^u(s) \\ Num_{\delta_e}^w(s) & Num_{\delta_t}^w(s) \\ Num_{\delta_e}^q(s) & Num_{\delta_t}^q(s) \\ Num_{\delta_e}^\theta(s) & Num_{\delta_t}^\theta(s) \end{bmatrix} \quad (3.20)$$

où les $Num(s)$ sont les numérateurs des différentes fonctions de transfert et $Den_{LONG}(s)$ est le dénominateur commun pour toutes les réponses longitudinales. $Den_{LONG}(s)$ est de la forme suivante :

$$Den_{LONG}(s) = (s^2 + 2\zeta_{ph}\omega_{n_{ph}}s + \omega_{n_{ph}}^2)(s^2 + 2\zeta_{sp}\omega_{n_{sp}}s + \omega_{n_{sp}}^2) \quad (3.21)$$

$$(3.22)$$

Dans [34], on présente, sous forme de fonctions de transfert, les structures des modèles du comportement longitudinal. En effet, si δ_t est maintenu constant, on obtient les structures de fonction de transfert, en fonction de la gouverne de profondeur δ_e , suivantes :

$$\frac{u(s)}{\delta_e(s)} \equiv \frac{Num_{\delta_e}^u(s)}{Den_{LONG}(s)} = \frac{k_u(s + 1/T_u)(s^2 + 2\zeta_u\omega_u s + \omega_u^2)}{(s^2 + 2\zeta_{ph}\omega_{ph}s + \omega_{ph}^2)(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (3.23)$$

$$\frac{w(s)}{\delta_e(s)} \equiv \frac{Num_{\delta_e}^w(s)}{Den_{LONG}(s)} = \frac{k_w(s + 1/T_\alpha)(s^2 + 2\zeta_\alpha\omega_\alpha s + \omega_\alpha^2)}{(s^2 + 2\zeta_{ph}\omega_{ph}s + \omega_{ph}^2)(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (3.24)$$

$$\frac{q(s)}{\delta_e(s)} \equiv \frac{Num_{\delta_e}^q(s)}{Den_{LONG}(s)} = \frac{k_q s(s + 1/T_{\theta_1})(s + 1/T_{\theta_2})}{(s^2 + 2\zeta_{ph}\omega_{ph}s + \omega_{ph}^2)(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (3.25)$$

$$\frac{\theta(s)}{\delta_e(s)} \equiv \frac{Num_{\delta_e}^\theta(s)}{Den_{LONG}(s)} = \frac{k_\theta(s + 1/T_{\theta_1})(s + 1/T_{\theta_2})}{(s^2 + 2\zeta_{ph}\omega_{ph}s + \omega_{ph}^2)(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (3.26)$$

On remarque que le dénominateur $Den_{LONG}(s)$ possède deux paires de pôles complexes conjugués. En effet :

$$\lambda_{1,2} = -\zeta_{ph}\omega_{n_{ph}} \pm j\omega_{n_{ph}}\sqrt{1 - \zeta_{ph}^2} \quad (3.27)$$

$$\lambda_{3,4} = -\zeta_{sp}\omega_{n_{sp}} \pm j\omega_{n_{sp}}\sqrt{1 - \zeta_{sp}^2} \quad (3.28)$$

On retrouve donc deux modes d'oscillation. La stabilité combinée de ces 2 modes détermine la stabilité dynamique longitudinale totale du système. Toutes les parties réelles des valeurs propres doivent être négatives pour assurer la stabilité longitudinale. Dans le cadre de ce projet, le MiniAV utilisé est stable longitudinalement en vol non stationnaire.

Ces modes permettent aussi d'effectuer des simplifications qui sont très utiles. Le premier mode provient de la réponse du système en réponse libre, avec $\Delta\delta_e = 0$ et

$\Delta\delta_t = 0$. Ce mode, appelé phugoïde (*phugoid*), correspond à une longue oscillation (en basse fréquence) en vitesse avant et en hauteur. Le deuxième mode, appelé mode rapide (*short period*), correspond à un changement rapide de θ lorsque la vitesse est constante (lorsqu'une variation de δ_e survient). On peut approximer ce mode en négligeant l'équation de force sur l'axe X_B est en supposant une variation de vitesse $\dot{u} = 0$. On obtient le système suivant dans le référentiel de l'avion :

$$\begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} z_W & z_Q & 0 \\ m_W & m_Q & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} z_{\delta_e} \\ m_{\delta_e} \\ 0 \end{bmatrix} \delta_e \quad (3.29)$$

Étant donné que l'on peut approximer $q \approx s\theta$ et, après quelques manipulations, on trouve :

$$\frac{q(s)}{\delta_e(s)} \approx \frac{k_q(s + 1/T_{\theta_1})}{(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (3.30)$$

$$\frac{\theta(s)}{\delta_e(s)} \approx \frac{q(s)}{s\delta_e(s)} \quad (3.31)$$

Ces approximations sont valides pour une vitesse u_0 donnée avec δ_t constant et elles servent à expliquer le comportement aérodynamique longitudinal pour un point d'opération. Les structures de fonction de transfert, en fonction de la manette des gaz δ_t , ne sont pas utilisées dans le cadre du projet.

Réponses latérales

De la même façon avec l'équation 3.17, en utilisant les matrices A_{LAT} et B_{LAT} ,

$$G(s) = (sI - A_{LAT})^{-1}B_{LAT} \quad (3.32)$$

on obtient :

$$G(s) = \frac{1}{Den_{LAT}(s)} \begin{bmatrix} Num_{\delta_a}^v(s) & Num_{\delta_r}^v(s) \\ Num_{\delta_a}^p(s) & Num_{\delta_r}^p(s) \\ Num_{\delta_a}^r(s) & Num_{\delta_r}^r(s) \\ Num_{\delta_a}^\phi(s) & Num_{\delta_r}^\phi(s) \end{bmatrix} \quad (3.33)$$

où les $Num(s)$ sont les numérateurs des différentes fonctions de transfert et $Den_{LAT}(s)$ est le dénominateur commun en s pour toutes les réponses latérales-directionnelles. $Den_{LAT}(s)$ est de la forme suivante :

$$Den_{LAT}(s) = (s + \lambda_s)(s + \lambda_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2) \quad (3.34)$$

Dans [34], on présente les fonctions de transfert du comportement latéral. On obtient typiquement les structures de fonction de transfert, en fonction des ailerons δ_a , suivantes :

$$\frac{v(s)}{\delta_a(s)} \equiv \frac{Num_{\delta_a}^v(s)}{Den_{LAT}(s)} = \frac{k_v(s + 1/T_{\beta_1})(s + 1/T_{\beta_2})}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.35)$$

$$\frac{p(s)}{\delta_a(s)} \equiv \frac{Num_{\delta_a}^p(s)}{Den_{LAT}(s)} = \frac{k_p s(s^2 + 2\zeta_\phi \omega_{n\phi} s + \omega_{n\phi}^2)}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.36)$$

$$\frac{r(s)}{\delta_a(s)} \equiv \frac{Num_{\delta_a}^r(s)}{Den_{LAT}(s)} = \frac{k_r(s + 1/T_\psi)(s^2 + 2\zeta_\psi \omega_{n\psi} s + \omega_{n\psi}^2)}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.37)$$

$$\frac{\phi(s)}{\delta_a(s)} \equiv \frac{Num_{\delta_a}^\phi(s)}{Den_{LAT}(s)} = \frac{k_\phi(s^2 + 2\zeta_\phi \omega_{n\phi} s + \omega_{n\phi}^2)}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.38)$$

Aussi, on obtient les structures de fonction de transfert, en fonction de la gouverne de direction δ_r , suivantes :

$$\frac{v(s)}{\delta_r(s)} \equiv \frac{Num_{\delta_r}^v(s)}{Den_{LAT}(s)} = \frac{k_v(s + 1/T_{\beta_1})(s + 1/T_{\beta_2})(s + 1/T_{\beta_3})}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.39)$$

$$\frac{p(s)}{\delta_r(s)} \equiv \frac{Num_{\delta_r}^p(s)}{Den_{LAT}(s)} = \frac{k_p s(s + 1/T_{\phi_1})(s + 1/T_{\phi_2})}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.40)$$

$$\frac{r(s)}{\delta_r(s)} \equiv \frac{Num_{\delta_r}^r(s)}{Den_{LAT}(s)} = \frac{k_r(s + 1/T_\psi)(s^2 + 2\zeta_\psi \omega_{n\psi} s + \omega_{n\psi}^2)}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.41)$$

$$\frac{\phi(s)}{\delta_r(s)} \equiv \frac{Num_{\delta_r}^\phi(s)}{Den_{LAT}(s)} = \frac{k_\phi(s + 1/T_{\phi_1})(s + 1/T_{\phi_2})}{(s + 1/T_s)(s + 1/T_r)(s^2 + 2\zeta_d \omega_{nd} s + \omega_{nd}^2)} \quad (3.42)$$

Le dénominateur est un polynôme du quatrième ordre. On remarque que le système est une combinaison de trois types de réponse lorsque le système est en réponse libre : deux modes non-oscillants (le mode du roulis amorti (*roll*) et le mode spirale (*spiral*)) et un mode oscillant (le mode roulis hollandais (*dutch roll*)). La stabilité globale latérale est donc en fonction de ces 3 modes (toutes les parties réelles des valeurs propres doivent être négatives pour assurer la stabilité latérale-directionnelle). Dans le cadre de ce projet, le MiniAV utilisé est stable latéralement en vol non stationnaire.

Le premier mode, appelé le mode du roulis amorti (*roll*), possède une plus petite constante de temps (T_r) que le mode spirale (*spiral*) et il représente la réponse à une perturbation sur l'axe X_B . Une perturbation de ce genre est plus rapidement amortie. On obtient son approximation en ignorant le couplage entre p et les autres variables d'état. Typiquement, on a :

$$\frac{p(s)}{\delta_a(s)} \approx \frac{k_p}{(s + 1/T_r)} \quad (3.43)$$

Le deuxième mode, appelé le mode spirale (*spiral*), possède la constante de temps la plus grande (T_s). Ce mode correspond à lorsque l'avion effectue un virage. On peut aussi utiliser un système du deuxième ordre pour approximer le mode du roulis amorti et le mode spirale. L'approximation est réalisée à partir d'une hypothèse de vol comme quoi la force latérale qui est due à la gravité produit la même vitesse angulaire qu'il existe avec un angle $\beta = 0$. On obtient le système d'état suivant :

$$\begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & -u0 & g \\ l_v & l_p & l_r & 0 \\ n_v & n_p & n_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} y_{\delta_a} & y_{\delta_r} \\ l_{\delta_a} & l_{\delta_r} \\ n_{\delta_a} & n_{\delta_r} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \quad (3.44)$$

On obtient alors les approximations utiles suivantes :

$$\frac{v(s)}{\delta_a(s)} \approx \frac{k_v}{(s + 1/T_s)(s + 1/T_{r2})} \quad (3.45)$$

$$\frac{p(s)}{\delta_a(s)} \approx \frac{k_p s}{(s + 1/T_s)(s + 1/T_{r2})} \quad (3.46)$$

$$\frac{r(s)}{\delta_a(s)} \approx \frac{k_v(s + 1/T_\Psi)}{(s + 1/T_s)(s + 1/T_{r2})} \quad (3.47)$$

$$\frac{\phi(s)}{\delta_a(s)} \approx \frac{p(s)}{s \delta_a(s)} \quad (3.48)$$

$$\frac{v(s)}{\delta_r(s)} \approx \frac{k_v}{(s + 1/T_s)(s + 1/T_{r2})} \quad (3.49)$$

$$\frac{p(s)}{\delta_r(s)} \approx \frac{k_p s}{(s + 1/T_s)(s + 1/T_{r2})} \quad (3.50)$$

$$\frac{r(s)}{\delta_r(s)} \approx \frac{k_v(s + 1/T_\Psi)}{(s + 1/T_s)(s + 1/T_{r2})} \quad (3.51)$$

$$\frac{\phi(s)}{\delta_r(s)} \approx \frac{p(s)}{s \delta_r(s)} \quad (3.52)$$

Le mode du roulis hollandais (*dutch roll*), est quant à lui, un mouvement complexe d'oscillation de lacet et de roulis.

Approximation du cap ψ

La deuxième loi de Newton ($F = ma$) peut être utilisée pour approximer la fonction de transfert du cap ψ . Pour effectuer un changement de direction de ψ_1 à ψ_2 , l'avion effectue un virage coordonné à une vitesse u_0 et à un taux angulaire $\dot{\psi}$. En effet, la figure 3.3 montre les différentes forces qui agissent sur l'avion durant un virage d'un rayon R . L'avion tourne avec un angle ϕ . En sommant dans l'axe Y_B les forces, on obtient :

$$mu_0\dot{\psi}\cos\phi = mg\sin\phi \quad (3.53)$$

$$\tan\phi = \frac{u_0\dot{\psi}}{g} \quad (3.54)$$

Typiquement, $\phi \ll 1$. On peut donc approximer :

$$\phi \approx \frac{u_0\dot{\psi}}{g} \quad (3.55)$$

Dans le domaine de Laplace, on a :

$$\Psi(s) \approx \frac{k_{psi}\phi(s)}{s} \quad (3.56)$$

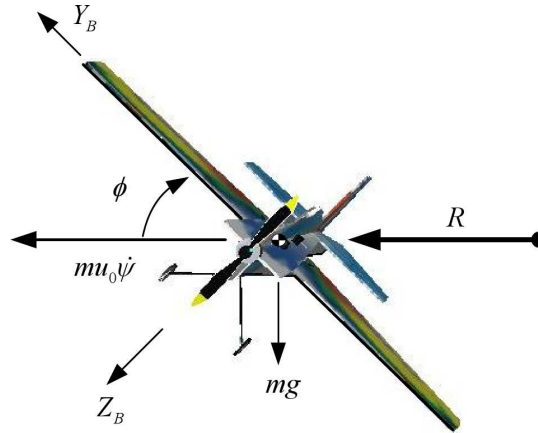


FIGURE 3.3 – Avion effectuant un virage

3.1.3 Identification

La figure 3.4, tirée de [7], présente les principales étapes pour réussir un processus d'identification. La connaissance a priori d'un système est à la base de toutes les étapes. Ainsi, dans la section 3.1.2, les fonctions de transfert qui décrivent les principaux comportements aérodynamiques ont été approximées. Ces informations sur les principales

fonctions de transfert permettent, entre autres, de planifier les essais et le type d'excitation à réaliser. On peut donc identifier les fonctions de transfert du mode longitudinal avec une excitation de δ_e (axe Y_B) et les fonctions de transfert du mode latéral avec des excitations de δ_a et δ_r (axes X_B et Z_B). On cherche à identifier les structures suivantes :

$$\frac{q(s)}{\delta_e(s)}, \frac{\theta(s)}{\delta_e(s)}, \frac{p(s)}{\delta_a(s)}, \frac{r(s)}{\delta_a(s)}, \frac{\phi(s)}{\delta_a(s)}, \frac{p(s)}{\delta_r(s)}, \frac{r(s)}{\delta_r(s)}, \frac{\phi(s)}{\delta_r(s)} \text{ et } \psi(s)$$

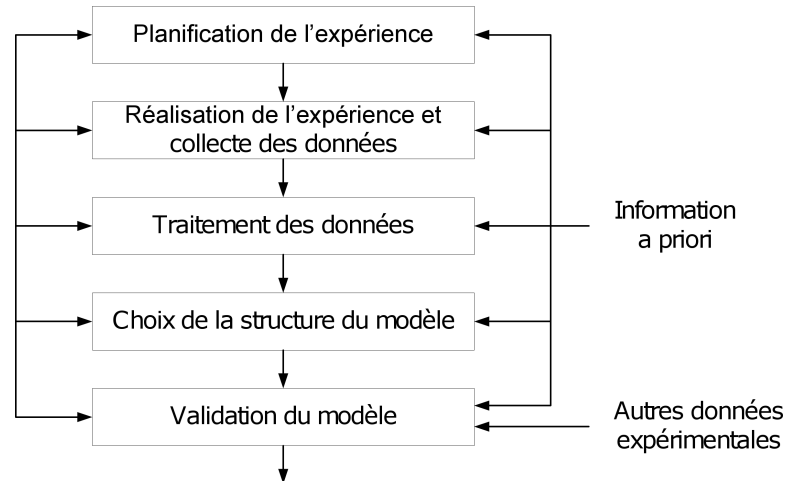
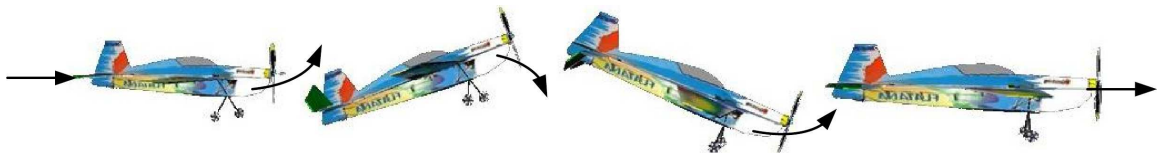
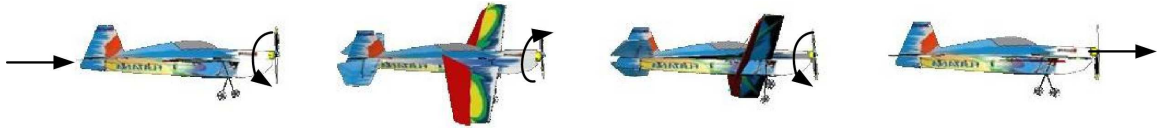
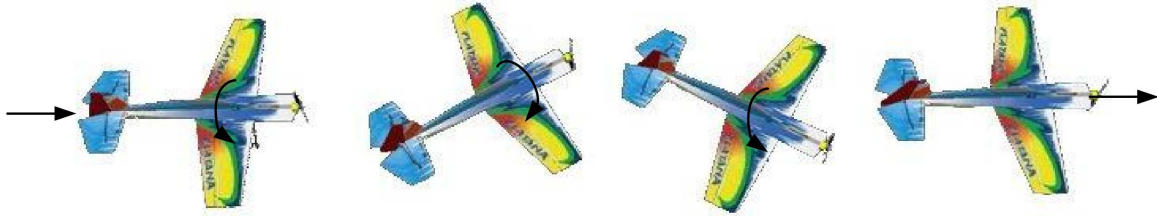


FIGURE 3.4 – Processus d'identification

Des fonctions de transfert linéaires sont donc identifiées pour décrire le comportement de l'avion, même si le comportement réel est non-linéaire. On fait l'hypothèse que ces fonctions de transfert identifiées représentent assez bien le comportement autour du point d'opération. En utilisant la méthodologie présentée par Klein et Moreilli [29], les manoeuvres d'excitation qui sont exécutées sont des doublets d'amplitude et de durée différentes. Les manoeuvres ont été exécutées convenablement de sorte que les mouvements engendrés n'amènent pas l'avion en décrochage ou en situation d'instabilité. Un pilote expérimenté a donc réalisé les manoeuvres d'excitation à partir de conditions de vol nominal avec δ_l le plus constant possible. Les figures 3.5 à 3.7 présentent les types de manoeuvres qui ont été exécutées.

FIGURE 3.5 – Manoeuvres d'excitation du δ_e

FIGURE 3.6 – Manoeuvres d’excitation des δ_a FIGURE 3.7 – Manoeuvres d’excitation du δ_r

Planification et réalisation de l’expérience

Les essais ont été réalisés en gymnase sur la base militaire de Valcartier. Les dimensions du gymnase sont illustrées à la figure 3.8. Comme on peut le constater, les dimensions physiques du gymnase ont contraint la durée et la faisabilité des essais. Les essais d’identification se sont déroulés entre les points A et B. Le pilote devait, lors du passage de l’avion au point A, s’assurer que celui-ci était en condition de vol nominal. De plus, un interrupteur sur la manette du pilote devait être déclenché manuellement entre les points A et B pour effectuer l’acquisition des données (pour diminuer le nombre de données et pour séparer les essais). La période d’échantillonnage est $t = 0.01$. Voici le plan de vol qui a été suivi lors de la séance d’identification en gymnase :

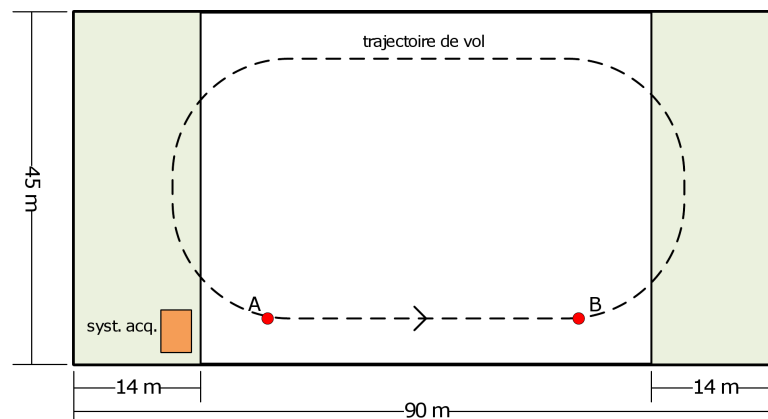


FIGURE 3.8 – Gymnase utilisé pour réaliser les essais

1. Installation des équipements et réchauffement du pilote avec l’avion de rechange
2. Vérification du centre de masse (répartition du poids) et de l’équipement sur

l'avion

3. Ajustement de la manette pour les variables manipulées (gains, signe, saturation min/max)
4. Routine du IMU (algorithme interne qui enlèvent les biais sur les vitesses angulaires p, q, r)
5. Vérification des états $\phi, \theta, \psi, p, q, r$ sur l'écran de l'ordinateur de base (signe et amplitude lors du mouvement).
6. Vols de réchauffement (ajustements si nécessaire)
7. Vols d'identification (7-8 par surfaces de contrôle)

Traitement des données et choix des structures

La première étape consiste à éliminer les séries de données qui ne sont pas représentatives (données incohérentes, données manquantes, système sous-excité, etc.). Pour ce faire, on effectue une analyse qualitative des séries temporelles des données de mesure. On supprime les mauvaises données au début et à la fin de chaque essai et on s'assure que lorsqu'une variable manipulée est excitée, les autres variables manipulées restent constantes. Par la suite, on soustrait les points d'opération des variables mesurées et manipulées. Les points d'opération soustraits sont les premières données de chaque essai, car on suppose l'avion en vol nominal (principe des conditions de vol stable, voir section 3.1.1 sur la linéarisation). La plage de fonctionnement des micros-servos est de 1 [ms] à 2 [ms]. Pour exprimer les variables manipulées en degrés ou en pourcentage, on effectue les multiplications suivantes :

$$\delta_a [deg] = \delta_a [ms] * 111 [deg/ms] \quad (3.57)$$

$$\delta_e [deg] = \delta_e [ms] * 112.4 [deg/ms] \quad (3.58)$$

$$\delta_t [\%] = \delta_t [ms] * 100 [\%/ms] \quad (3.59)$$

$$\delta_r [deg] = \delta_r [ms] * 168.5 [deg/ms] \quad (3.60)$$

Ces valeurs sont trouvées en mesurant la déflexion, en degrés, des variables manipulées à l'aide d'un appareil qui mesure l'angle de déflexion (*Control Surface Deflection Meter*).

Des séries de données sont choisies pour effectuer l'identification et les autres sont choisies pour faire la validation. Les séries de données sont fusionnées (la fonction *merge* dans Matlab) pour effectuer l'identification et la validation dans Matlab. La fonction *merge* met bout-à-bout les séries de données permettant ainsi d'ajouter les critères pour faire l'identification paramétrique. Les figures 3.9, 3.10 et 3.11 présentent des exemples de résultats de séries temporelles qui ont été obtenus lors des essais.

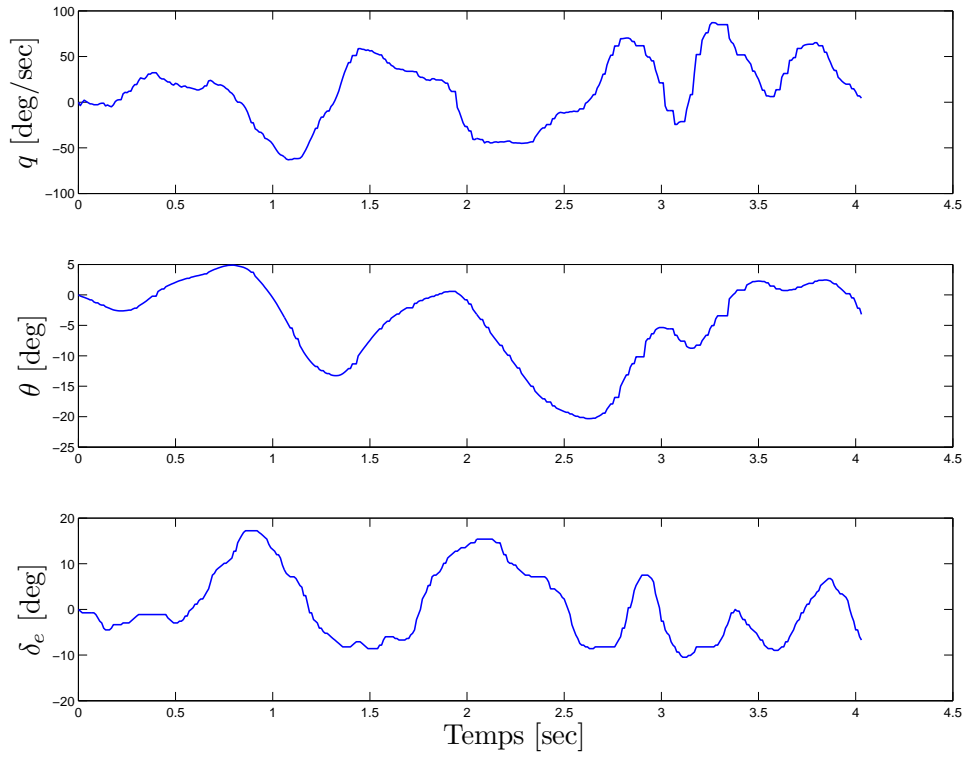


FIGURE 3.9 – Manœuvres δ_e

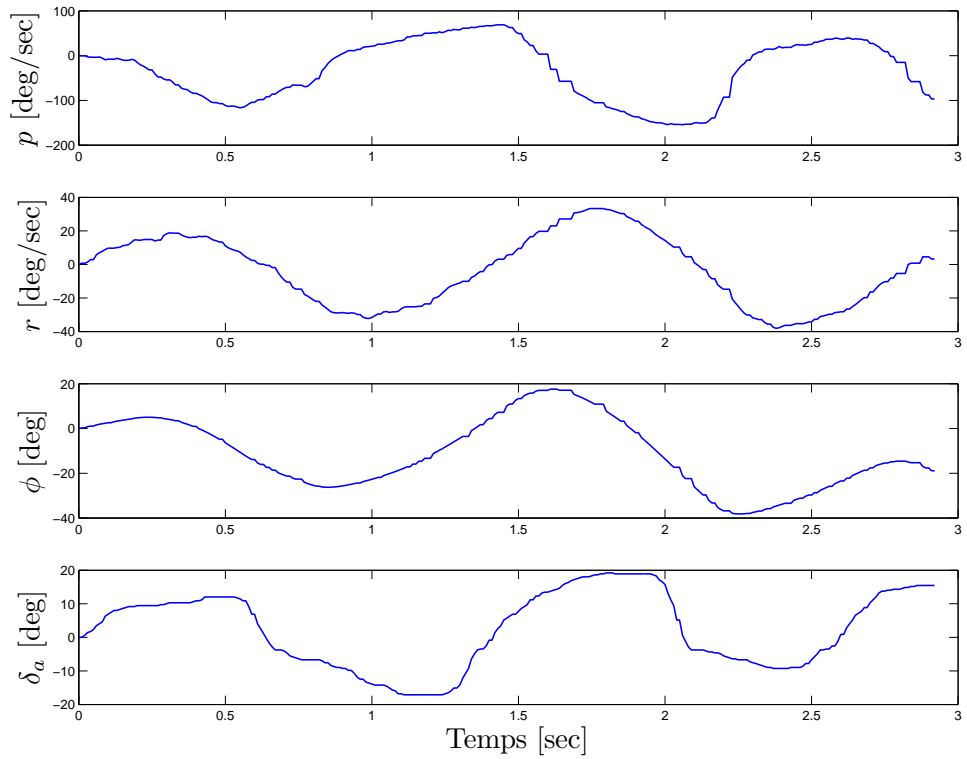


FIGURE 3.10 – Manœuvres δ_a

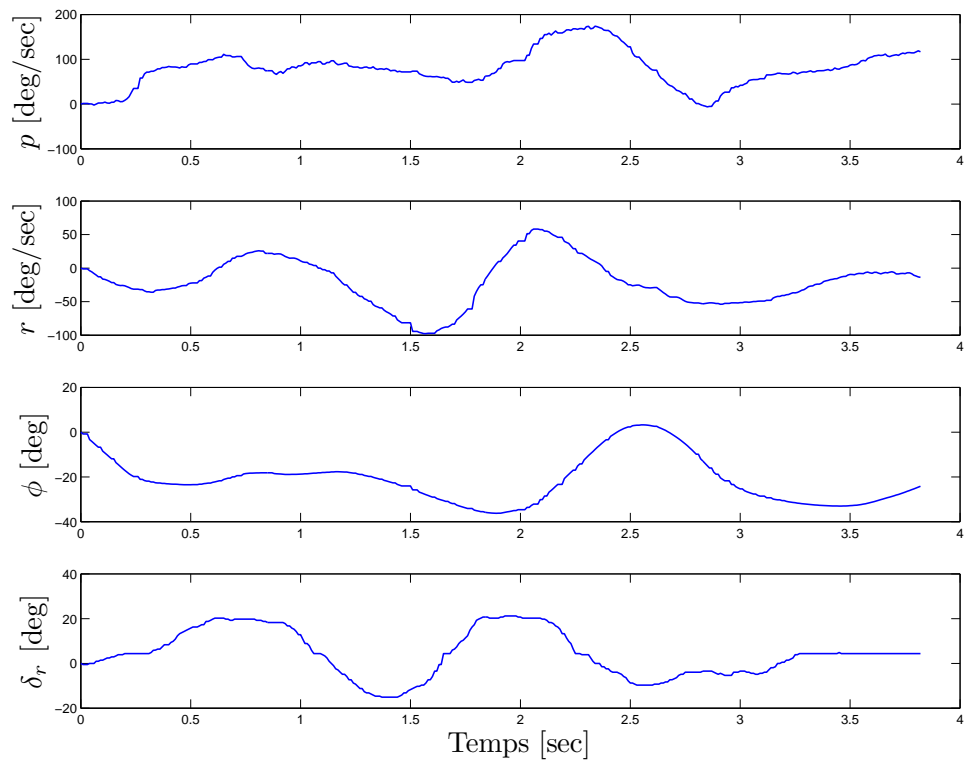


FIGURE 3.11 – Manoeuvres δ_r

Au début de chacun des essais d'identification, l'avion est déposé sur le sol et un algorithme interne, qui enlève les biais sur les vitesses angulaires p, q, r , est lancé. Ces biais sur p, q, r , que l'on considère constant durant l'essai (voir section 4.4.4), ont une répercussion sur les valeurs ϕ, θ, ψ générées par l'algorithme interne du IMU (*embedded software algorithms*). Ainsi, avec cette procédure, les biais sur p, q, r, ϕ, θ et ψ deviennent négligeables. L'estimation des biais peut se faire, entre autres, en utilisant la méthode basée sur l'erreur de sortie (*Output Error*). Le toolbox SIDPAC V2.0 [29] de Eugene A. Morelli peut être utilisé comme outil.

Voici la forme générale des structures que l'on utilise en identification paramétrique :

$$A(z^{-1})y(k) = \frac{B(z^{-1})}{F(z^{-1})}z^{-d}u(k) + \frac{C(z^{-1})}{D(z^{-1})}e(k) \quad (3.61)$$

où

$$A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_{na}z^{-na} \quad (3.62)$$

$$B(z^{-1}) = b_1z^{-1} + \dots + b_{nb}z^{-nb} \quad (3.63)$$

$$C(z^{-1}) = 1 + c_1z^{-1} + \dots + c_{nc}z^{-nc} \quad (3.64)$$

$$D(z^{-1}) = 1 + d_1z^{-1} + \dots + d_{nd}z^{-nd} \quad (3.65)$$

$$F(z^{-1}) = 1 + f_1z^{-1} + \dots + f_{nf}z^{-nf} \quad (3.66)$$

et $y(k)$, $u(k)$ et $e(k)$ sont respectivement la sortie, l'entrée et un bruit blanc. Il faut donc bien choisir la structure pour obtenir une bonne estimation des paramètres. On peut donc généraliser l'équation 3.61 par :

$$y(k) = G(z^{-1})z^{-d}u(k) + H(z^{-1})e(k) \quad (3.67)$$

où $G(z^{-1})$ est la partie déterministe et $H(z^{-1})$ est la partie stochastique du modèle. Le tableau 3.1 présente quelques structures de modèles qui sont utilisés.

ARX	ARMAX	OE	BJ
$nc = nd = nf = 0$	$nd = nf = 0$	$na = nc = nd = 0$	$na = 0$
$G(z^{-1}) = \frac{B(z^{-1})z^{-d}}{A(z^{-1})}$	$G(z^{-1}) = \frac{B(z^{-1})z^{-d}}{A(z^{-1})}$	$G(z^{-1}) = \frac{B(z^{-1})z^{-d}}{F(z^{-1})}$	$G(z^{-1}) = \frac{B(z^{-1})z^{-d}}{F(z^{-1})}$
$H(z^{-1}) = \frac{1}{A(z^{-1})}$	$H(z^{-1}) = \frac{C(z^{-1})}{A(z^{-1})}$	$H(z^{-1}) = 1$	$H(z^{-1}) = \frac{C(z^{-1})}{D(z^{-1})}$

TABLE 3.1 – Structures de modèles pour l'identification paramétrique

L'outil *System Identification Toolbox Version 7.1* de Matlab permet d'effectuer l'analyse de telles structures. Pour calculer les coefficients des polynômes, l'algorithme

minimise la somme des carrés des erreurs de prédiction (résidus) avec la méthode du maximum de vraisemblance.

De plus, cet outil possède une fonction, *Process Model*, qui permet d'identifier un modèle continu. Cet outil permet donc d'imposer, avec de simples options, des structures ayant une signification physique pouvant posséder 3 pôles (dont 2 oscillants), un intégrateur, un délai et un zéro. Ainsi, dans le cas présent, on peut choisir la bonne structure selon le comportement aérodynamique qui est recherché. En fait, cet outil transforme tout simplement le modèle continu en une structure de modèles discrets pour estimer les paramètres. Ceux-ci sont calculés de façon itérative pour trouver un modèle qui minimise la variance de l'erreur de prédiction. Le modèle *Process Model* a la forme suivante :

$$y(s) = G(s)u(s) + H(s)e(s) \quad (3.68)$$

où $G(s)$ est la structure à identifier et $H(s)$ est un modèle de perturbation de type *ARMA* $\left(\frac{C(s)}{D(s)}\right)$. Le modèle de perturbation peut être du premier ou du deuxième degré. Il est aussi possible d'imposer une valeur ou un intervalle sur les différents paramètres de $G(s)$ (exemple : imposer la valeur ou un intervalle sur un pôle).

Les structures de modèle peuvent aussi être *SISO* ou *MISO*. Un modèle *SISO* signifie une entrée unique avec une sortie unique (*single input, single output*). Un modèle *MISO* signifie des entrées multiples avec une sortie unique (*single input, single output*).

Pour effectuer la validation, on simule le modèle temporellement (partie déterministe sans la partie stochastique) et on compare la réponse obtenue avec des séries de données de validation. Finalement, l'analyse des résidus avec les données de validation permet de savoir si les coefficients ont bien été identifiés. En effet, si le modèle est parfait, les résidus tendent asymptotiquement vers un bruit blanc. De plus, on effectue une corrélation entre les résidus et les entrées du système. Si les résidus peuvent être expliqués seulement en partie par l'entrée du système, cela signifie que le modèle est incomplet. La période d'échantillonnage k est de 0.01 seconde. Le délai, introduit par le système d'acquisition à 100 Hz, a été approximé et imposé à 0.17 sec (17 échantillons).

Plusieurs types de structures de modèles ont donc été étudiés (*ARX SISO-MISO, ARMAX SISO-MISO, OE SISO-MISO, BJ SISO-MISO, Process Model SISO-MISO*). Les modèles *MISO* ont été considérés lors des essais d'excitation du mode latéral. Cependant, ceux-ci n'ont pas été choisis, car, lors des essais, une seule variable manipulée était excitée à la fois, les autres variables manipulées restaient assez constantes (le contenu fréquentiel était alors trop faible pour réaliser l'identification).

Tout d'abord, le modèle *Process Model SISO* a été choisi pour l'analyse, car cet outil permet d'imposer une structure ayant une signification physique et un modèle de perturbation de type *ARMA* $\left(\frac{C(s)}{D(s)}\right)$. Le modèle discret *ARMAX SISO* aussi a été choisi pour l'analyse, car le polynôme $C(z^{-1})$ ajoute de la flexibilité à la partie stochastique du modèle pour permettre de mieux blanchir les résidus. La section suivante montre un exemple d'identification avec $\frac{q(s)}{\delta_e(s)}$.

Exemple d'identification

Il a été démontré que $\frac{q(s)}{\delta_e(s)}$ pouvait être approximée par le mode rapide (*short period*) :

$$\frac{q(s)}{\delta_e(s)} \approx \frac{k_q(s + 1/T_{\theta_1})e^{-T_d s}}{(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (3.69)$$

$$(3.70)$$

La figure 3.12(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model SISO* et d'un modèle *ARMAXSISO* avec une série de données de mesure de validation. Les deux modèles semblent bien suivre les données de mesure. La figure 3.12(b) présente l'analyse des résidus du modèle choisi avec les données de validation et la figure 3.12(c) montre la corrélation entre $\delta_e(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures 3.12(b) et 3.12(c).

Le modèle choisi est un modèle continu de type *Process model* :

$$\frac{q(s)}{\delta_e(s)} = K * \frac{(1 + T_z s) e^{-T_d s}}{(T_w s)^2 + 2\zeta T_w s + 1} \quad (3.71)$$

avec

$$K = -2.106 \pm 0.18636 \quad (3.72)$$

$$T_w = 0.0803 \pm 0.0025554 \quad (3.73)$$

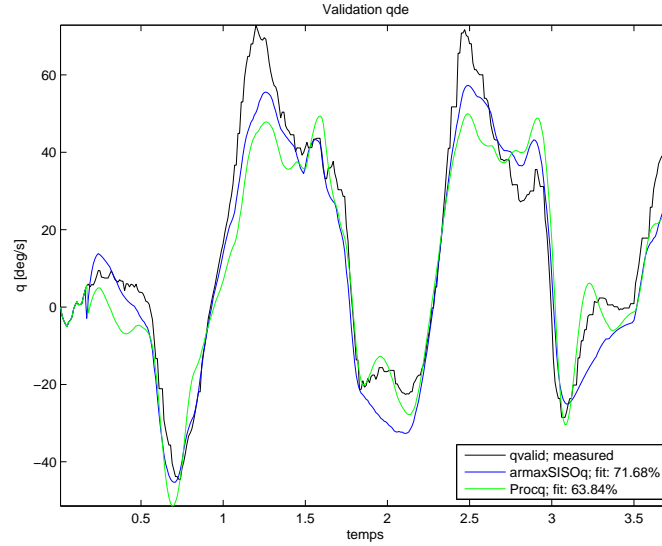
$$\zeta = 0.4 \pm 0.035621 \quad (3.74)$$

$$T_d = 0.17 \pm 0 \quad (3.75)$$

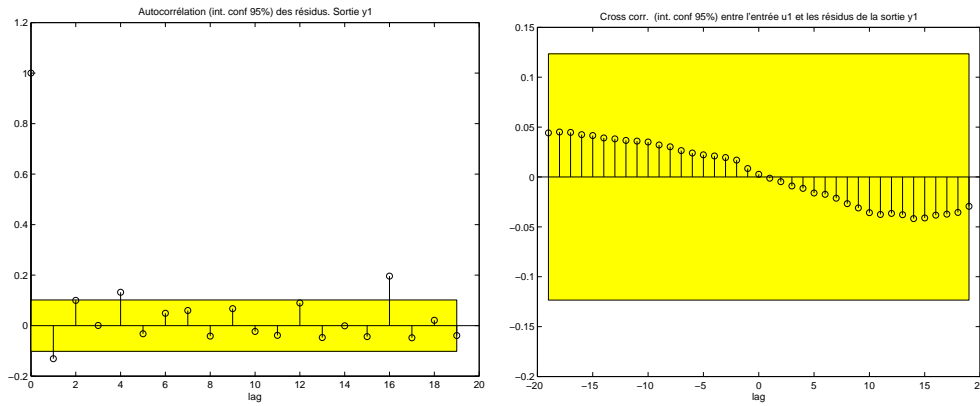
$$T_z = 0.205 \pm 0.031282 \quad (3.76)$$

Le modèle de perturbation, de type *ARMA* $\left(\frac{C(s)}{D(s)}\right)$, est :

$$y(s) = G(s)u(s) + H(s)e(s) \quad (3.77)$$



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus

 (c) Corrélation entre $\delta_e(k)$ et $\varepsilon(k+\tau)$

 FIGURE 3.12 – Identification et validation de $\frac{q(s)}{\delta_e(s)}$

avec

$$C(s) = s^2 + 154.9(\pm 30.41)s + 9988(\pm 4184) \quad (3.78)$$

$$D(s) = s^2 + 68.2(\pm 34.68)s + 202.8(\pm 130.8) \quad (3.79)$$

Le modèle stochastique est cependant moins précis. On trouve alors les valeurs propres $\lambda_{1,2} = -5 \pm 11.5i$ avec $\zeta = 0.4$ et $\omega_n = 12.3$ rad/s. Les autres structures identifiées sont dans l'Annexe A.1.

Modèles identifiés du mode longitudinal

Voici les modèles qui ont été identifiés :

$$\frac{q(s)}{\delta_e(s)} \approx \frac{-2.106(0.205s + 1)e^{-0.17s}}{(0.0064s^2 + 0.0642s + 1)} \quad (3.80)$$

$$\frac{\theta(s)}{\delta_e(s)} \approx \frac{-2.1147(0.22261s + 1)e^{-0.17s}}{s(0.0082s^2 + 0.0543s + 1)} \quad (3.81)$$

On retrouve donc le mode rapide (*short period*) de $\frac{q(s)}{\delta_e(s)}$ comme étant $\lambda_{sp} = -5 \pm 11.5i$. Ce mode est stable. Dans le cadre du projet, le mode phugoïde (*phugoid*) est approximé par l'intégrateur avec $\lambda_{ph} = 0$.

Modèles identifiés du mode latéral

Voici les modèles qui ont été identifiés :

$$\frac{p(s)}{\delta_a(s)} \approx \frac{-4.608e^{-0.17s}}{(0.047s + 1)} \quad (3.82)$$

$$\frac{r(s)}{\delta_a(s)} \approx \frac{-4.05(-0.1791s + 1)e^{-0.17s}}{(0.0773s + 1)(0.81257s + 1)} \quad (3.83)$$

$$\frac{\phi(s)}{\delta_a(s)} \approx \frac{-6.347e^{-0.17s}}{s(0.0401s + 1)} \quad (3.84)$$

$$\frac{p(s)}{\delta_r(s)} \approx \frac{0.18se^{-0.17s}}{(0.06s + 1)(1.71s + 1)} \quad (3.85)$$

$$\frac{r(s)}{\delta_r(s)} \approx \frac{0.37704(10s + 1)e^{-0.17s}}{(0.052s + 1)(1.501s + 1)} \quad (3.86)$$

$$\frac{\phi(s)}{\delta_r(s)} \approx \frac{-0.045se^{-0.17s}}{(0.07s + 1)(1.555s + 1)} \quad (3.87)$$

$$\psi(s) \approx \frac{1.41 \phi(s)}{s} \quad (3.88)$$

La constante de temps du mode roulis amorti (*roll*) est donc estimée entre 0.04 et 0.07 sec ($-25 < \lambda_r < -14$) et le mode spirale (*spiral*) entre 0.81 et 1.55 sec ($-1.17 < \lambda_s < -0.66$). Ces deux modes sont stables.

3.1.4 Conception des autopilotes

Un autopilote est un système qui permet de maintenir un véhicule sur une trajectoire, une position, une vitesse pré-établie, etc. Dans le cas d'un avion, l'autopilote a pour but de stabiliser l'avion et d'en faciliter la commande. Il été vu dans la section 3.1.2 qu'un avion possède plusieurs comportements aérodynamiques. La stabilité naturelle en vol fait en sorte que, après une perturbation en condition de vol stable, les forces aérodynamiques de l'avion redonnent à l'avion une condition de vol stable [11]. Cependant, ces caractéristiques de stabilité naturelles ne sont pas souhaitable car elle peuvent entraîner l'avion dans des conditions de vol non-souhaitables ou inconfortables pour les passagers d'un avion. Les autopilotes sont donc un système d'augmentation de stabilité.

Les structures d'autopilote pour des avions à ailes fixes se retrouvent facilement dans la littérature [4, 9, 35]. La manoeuvrabilité (*handling qualities*) des avions conventionnels est aussi bien comprise et documentée [11, 32]. Cependant, il existe très peu de normes sur la manoeuvrabilité des UAVs [6]. Dans le cadre de ce projet, les autopilotes doivent permettre de contrôler l'avion assez rapidement autour du point d'opération qui a été identifié, et ce, tout en respectant des marges de stabilité.

Il est possible de quantifier la stabilité d'un système en boucle fermée $H(s) = \frac{G(s)}{1+G(s)}$ en analysant la réponse en fréquences de $G(s) = G_c(s)G_p(s)$, où $G_c(s)$ est le régulateur et $G_p(s)$ est le procédé. La marge de gain G_m est l'augmentation du rapport d'amplitude de $G(s)$ à la fréquence ultime qui amènerait $H(s)$ à la limite de la stabilité. La marge de phase ϕ_m est la diminution de la phase de $G(s)$ à la fréquence ω_0 , donc la fréquence où le rapport d'amplitude de $G(s) = 0dB$, amènerait $H(s)$ à la limite de la stabilité. Des marges positives sont nécessaires pour assurer la stabilité d'un système en asservissement. Plus les marges de gain et de phase seront grandes, plus le système de commande sera robuste aux erreurs d'identification du procédé. Une marge de gain minimale de 6 *dB* et une marge de phase minimale de 60 degrés sont choisies pour concevoir les régulateurs.

Mode longitudinal en boucle fermée

Pour le contrôle en boucle fermée du mode longitudinal, l'angle θ est la variable d'intérêt. Pour obtenir des caractéristiques de manoeuvrabilité intéressante, on doit obtenir une réponse assez rapide, une erreur statique nulle ainsi qu'un bon amortissement des modes d'oscillation. La figure 3.13 présente le diagramme fonctionnel d'un contrôleur d'attitude θ . Dans ce cas-ci, δ_t est supposée constante à son point d'opération $\delta_{t \text{ level}}$.

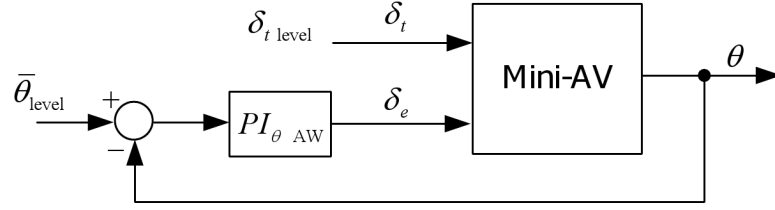


FIGURE 3.13 – Diagramme fonctionnel d'un contrôleur longitudinal

Un PI anti-windup ($PI_{\theta AW}$) peut être utilisé dans un contrôleur longitudinal [9] pour contrôler l'attitude en tangage. Un contrôleur anti-windup (AW) est un contrôleur qui tient compte de la valeur de la saturation des actionneurs. La figure 3.14 illustre ce principe [16].

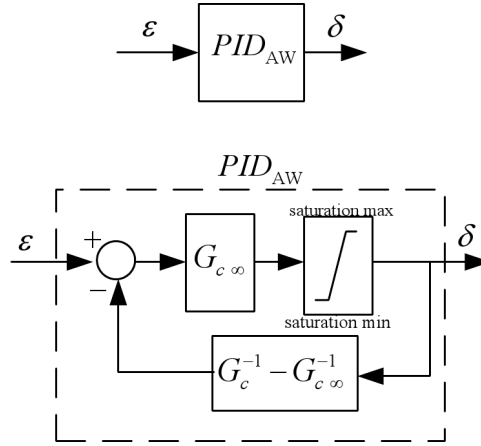


FIGURE 3.14 – Diagramme fonctionnel d'un contrôleur anti-windup

Le PID_{AW} suivant est utilisé :

$$G_c(s) = \frac{K_c (T_i s + 1) (T_d s + 1)}{T_i s (T_f s + 1)} \quad (3.89)$$

avec

$$G_{c \infty} = \lim_{s \rightarrow \infty} G_c(s) = \frac{K_c T_d}{T_f} \quad (3.90)$$

$$G_c^{-1}(s) - G_{c \infty}^{-1} = \frac{-1 \left((T_i T_f + T_f T_d - T_i T_d) s^2 + T_f \right)}{(K_c T_i T_d^2) \left(s^2 + \frac{T_i + T_d}{T_i T_d} s + \frac{1}{T_i T_d} \right)} \quad (3.91)$$

où ε est l'erreur, δ la valeur de la variable manipulée et $G_{c \infty}$ est le gain du régulateur lorsque la fréquence tend vers l'infini. Dans le système de commande, la fonction de transfert $G_c^{-1}(s) - G_{c \infty}^{-1}$ est transposée en représentation de l'espace d'état. Pour un PI_{AW} , $T_d = T_f = 1$ dans les équations 3.90 et 3.91.

Pour calculer les paramètres du PI, on peut trouver une approximation de $\frac{\theta(s)}{\delta_e(s)}$. On trouve alors $\frac{\theta(s)}{\delta_e(s)} \approx \frac{-1.35e^{-0.17s}}{s}$. La figure 3.15 montre la réponse à l'échelon de la fonction identifiée ainsi que la fonction approximée. Donc, la fonction approximée est un procédé stable non asymptotiquement. Il existe de nombreuses méthodes pour choisir les paramètres d'un PI d'un tel procédé, dont la méthode des contours [33]. Cependant, cette méthode ne permet pas de choisir la dynamique de la boucle fermée. On utilise

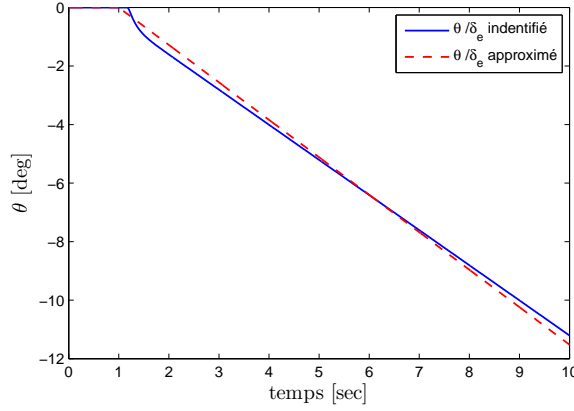


FIGURE 3.15 – Approximation de $\frac{\theta(s)}{\delta_e(s)}$

alors la méthode du réglage par placement de pôles. Cette méthode consiste à sélectionner, pour un procédé $G_p(s)$, un régulateur $G_c(s)$ de manière à obtenir une certaine dynamique $H(s)$ en boucle fermée :

$$H(s) = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)} \quad (3.92)$$

Ainsi, dans le cas présent, pour $G_p(s) \approx \frac{K_p}{s}$ (le retard est négligé dans la conception) et $G_c(s) = \frac{K_c(T_i s + 1)}{T_i s}$ on obtient en boucle fermée :

$$H(s) = \frac{T_i s + 1}{\frac{T_i}{K_p K_c} s^2 + T_i s + 1} \quad (3.93)$$

On remarque que ce système a un zéro à $-1/T_i$, causant un dépassement. Un filtre sur la consigne ($\frac{1}{T_i s + 1}$) pourrait éliminer ce dépassement. Pour placer les pôles, on peut comparer ce système à un système du deuxième ordre : $\frac{1}{\omega_n^2} s^2 + \frac{2\zeta}{\omega_n} s + 1$. En faisant égaliser et réarranger les termes en s , on obtient :

$$T_i = \frac{2\zeta}{\omega_n} \quad (3.94)$$

$$K_c = \frac{2\zeta\omega_n}{K_p} \quad (3.95)$$

En choisissant $\zeta = 0.7$ pour un système du deuxième ordre, on obtient le temps de réponse le plus court pour une valeur de ω_n [8]. Ainsi, si $\zeta = 0.7$:

$$\omega_n = \frac{3}{T_{rep}} \quad (3.96)$$

où T_{rep} est le temps de réponse à $\pm 5\%$. Ici, on choisit $T_{rep} = 3$. On obtient le PI suivant :

$$PI_{\theta}(s) = \frac{-1(1.4s + 1)}{1.4s} \quad (3.97)$$

La figure 3.17 présente la réponse à un changement de consigne unitaire (non filtrée) de θ du système (modèle de l'équation 3.81) en boucle fermée. Malgré le dépassement, on observe une erreur statique nulle. Pour l'analyse fréquentielle, on définit $G_{\theta} = PI_{\theta}(s) \frac{\theta(s)}{\delta_e(s)}$. La figure 3.16 présente l'analyse fréquentielle de G_{θ} . On observe une marge de gain de 7.66 dB et une marge de phase de 93.6 degrés.

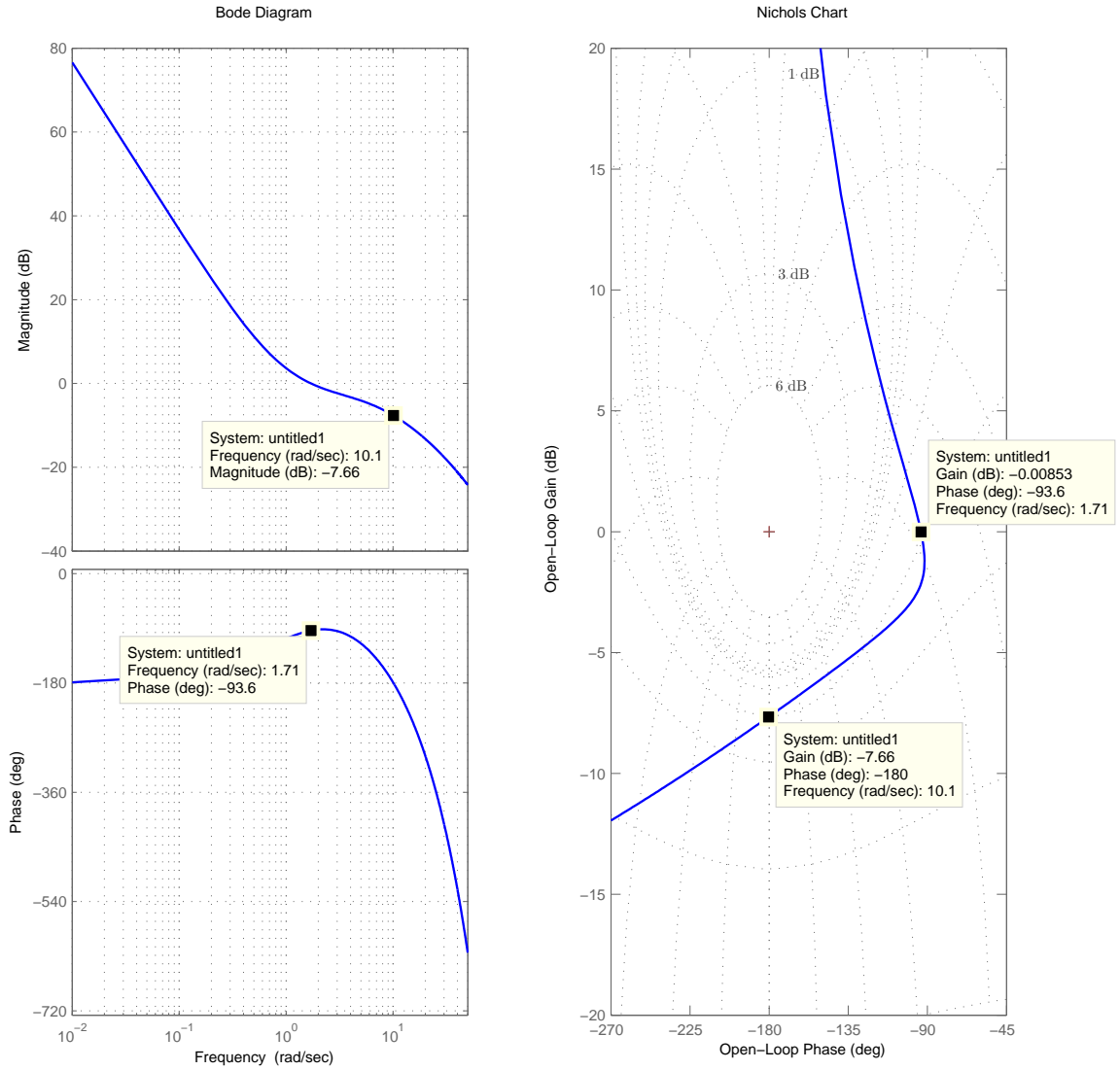


FIGURE 3.16 – Analyse fréquentielle - Réponse en fréquences de G_θ

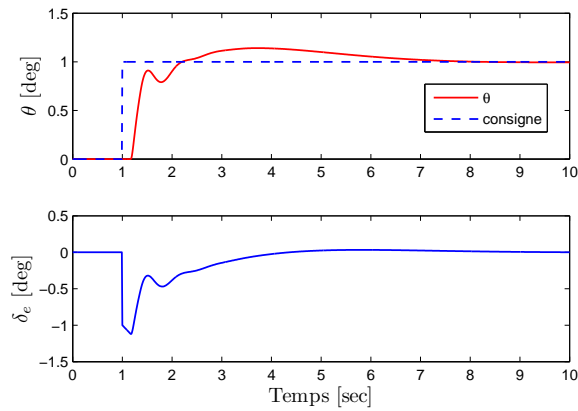


FIGURE 3.17 – Sortie du système θ en asservissement

Mode latéral en boucle fermée

Pour le contrôle latéral en boucle fermée, des boucles cascades sont utilisées [9] (figure 3.18). La première boucle interne contrôle p . La seconde boucle interne contrôle l'angle ϕ . L'erreur de l'angle ϕ est convertie en une consigne de p par un régulateur proportionnel P_p . Finalement, la boucle externe contrôle le cap (l'angle ψ). Ainsi, l'erreur de cap est convertie en une consigne de ϕ par un régulateur PI_ψ . En résumé, pour effectuer un changement de cap, l'avion doit effectuer un virage contrôlé.

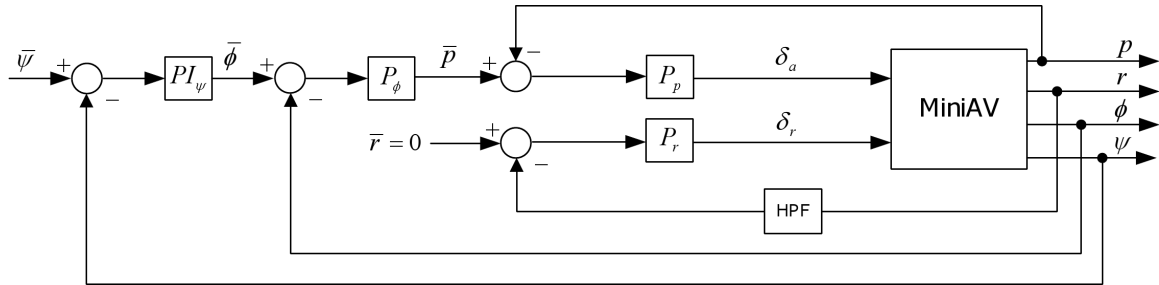


FIGURE 3.18 – Diagramme fonctionnel d'un contrôleur latéral

Un amortisseur de lacet (*yaw damper*) est aussi introduit dans la première boucle interne du contrôleur latéral. Les amortisseurs de direction sont utiles, car durant une manoeuvre de virage continu, la valeur de r n'est pas égale à zéro. Un filtre passe-haut (*HPF*) est utilisé pour que la variable manipulée δ_r soit calculée pour effectuer correctement la manoeuvre. La fréquence de coupure du filtre a été choisie comme étant 20 rad/sec, c'est-à-dire environ $1/T_r$ où T_r est le mode du roulis amorti (*roll*). Pour la boucle interne, on a les fonctions de transfert suivantes :

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \quad (3.98)$$

$$= \begin{bmatrix} \frac{p(s)}{\delta_a(s)} & \frac{p(s)}{\delta_r(s)} \\ \frac{r(s)}{\delta_a(s)} & \frac{r(s)}{\delta_r(s)} \end{bmatrix} \quad (3.99)$$

Pour les besoins de fonctionnalité du système, on veut contrôler la sortie p avec δ_a et r avec δ_r . Généralement, ce couplage est choisi lors de la conception d'un autopilote latéral [9, 35]. On peut confirmer le choix de ce couplage en faisant l'analyse de la matrice de gains relatifs statiques Λ . Les gains relatifs permettent de mesurer le degré de couplage d'un procédé multivariable. En fait, ils mesurent l'influence d'une variable sélectionnée sur une variable contrôlée particulière relativement aux autres variables

manipulées agissant sur le procédé. Ainsi :

$$\Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \begin{bmatrix} 1.7099 & -0.7099 \\ -0.7099 & 1.7099 \end{bmatrix} \quad (3.100)$$

λ_{11} étant positif, on peut confirmer le choix du couplage de p avec δ_a et r avec δ_r .

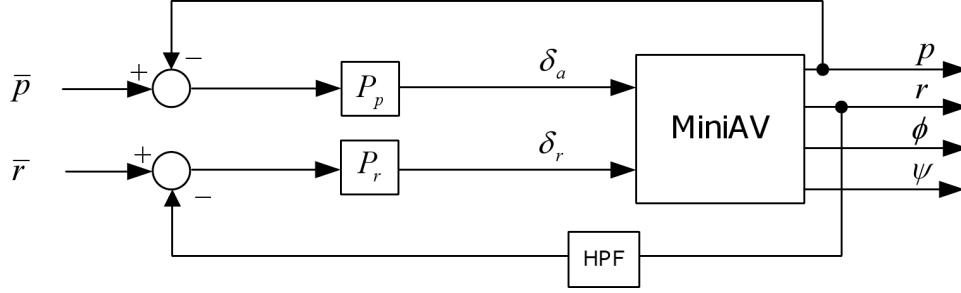


FIGURE 3.19 – Diagramme fonctionnel de la première boucle interne

Typiquement, des gains proportionnels P_p et P_r sont choisis dans la première boucle interne ([4, 9, 35]). On choisit aussi une commande distribuée plutôt qu'une commande multivariable avec découpleurs [4, 9, 35] pour simplifier cette boucle de commande. La figure 3.19 présente un diagramme de la boucle interne utilisée pour effectuer le réglage de P_p et P_r . Ici, un changement de consigne sur p est vu comme une perturbation sur r et vice-versa. Pour régler les gains proportionnels P_p et P_r , la méthode consiste à ouvrir la boucle qui est à régler et à trouver la fonction de transfert qui est vue par le régulateur. Ainsi, la fonction de transfert vue par P_p est G_1 et celle vue par P_r est G_2 :

$$G_1(s) = G_{11}(s) - \frac{HPF(s)G_{12}(s)G_{21}(s)P_r}{1 + HPF(s)G_{22}(s)P_r} \quad (3.101)$$

$$G_2(s) = G_{22}(s) - \frac{G_{12}(s)G_{21}(s)P_p}{1 + G_{11}(s)P_p} \quad (3.102)$$

L'entrée de $G_1(s)$ est δ_a et l'entrée de $G_2(s)$ est δ_r . On trouve de façon itérative les valeurs de P_p et P_r . En respectant les critères de conception des marges de stabilité, on trouve $P_p = -0.08$ et $P_r = 0.25$. On peut aussi trouver des fonctions de transfert pour approximer $G_1(s)$ et $G_2(s)$. Ainsi, avec $P_p = -0.08$ et $P_r = 0.25$, on trouve :

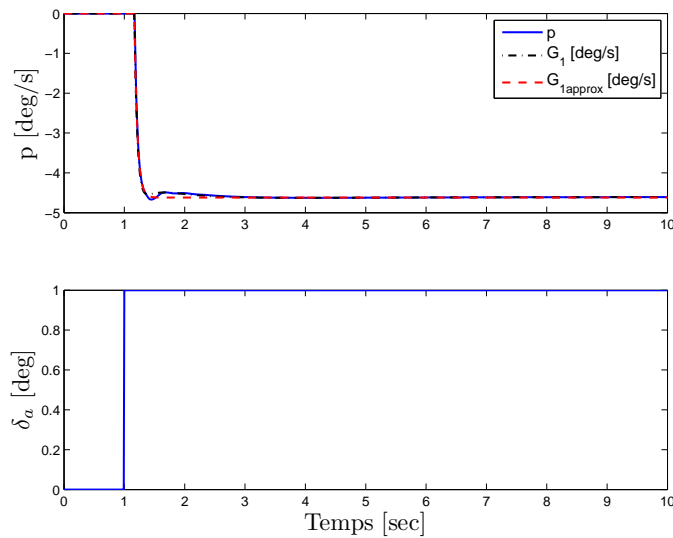
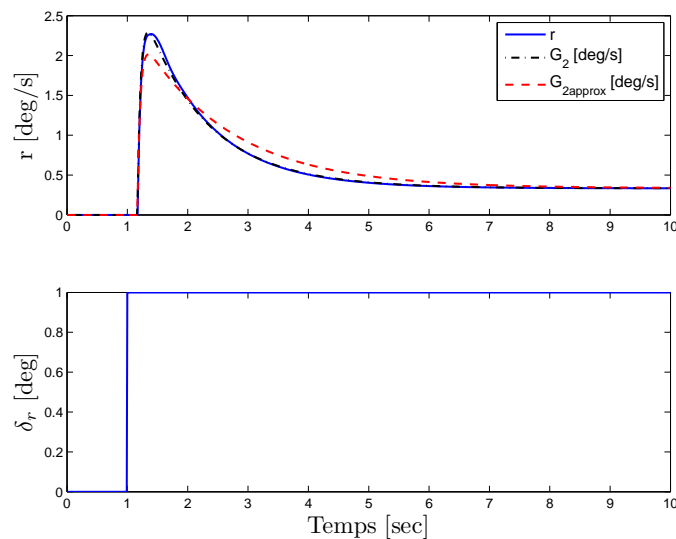
$$G_{1approx}(s) = \frac{-4.62 e^{-0.17s}}{0.05s + 1} \quad (3.103)$$

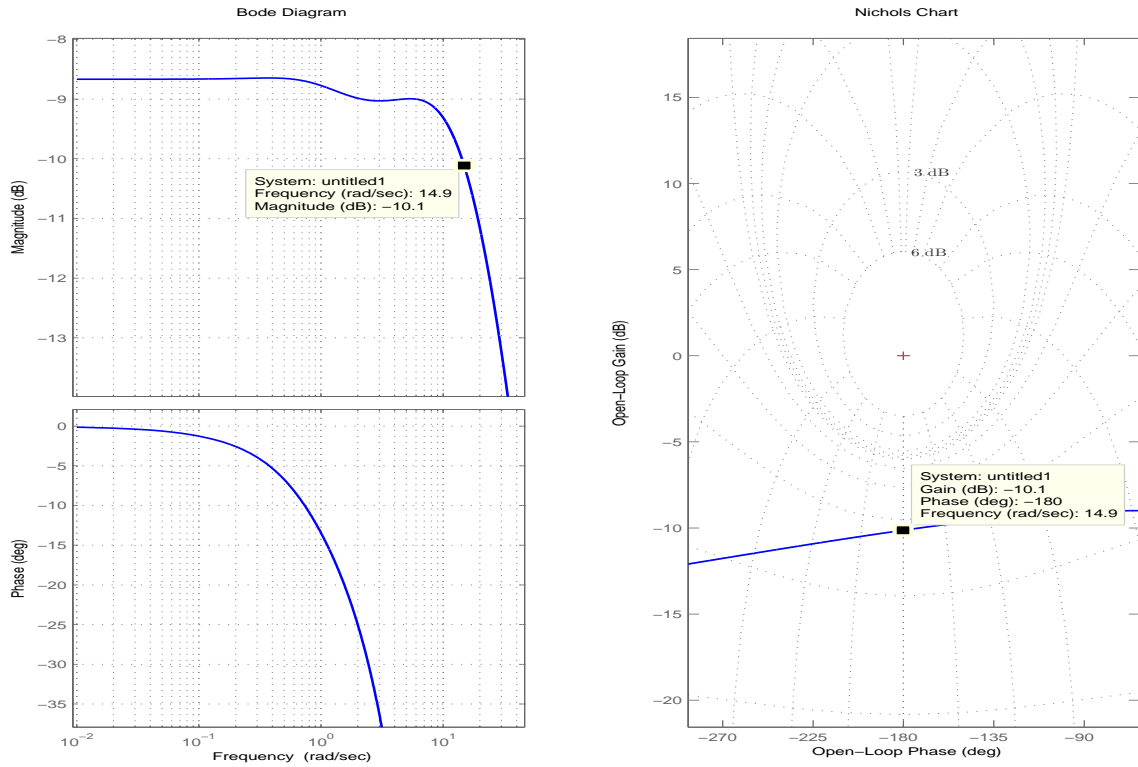
$$G_{2approx}(s) = \frac{0.335(8s + 1) e^{-0.17s}}{(0.85s + 1)(0.14s + 1)} \quad (3.104)$$

La figure 3.20(a) montre la réponse de la sortie du système p suite à un échelon δ_a (la boucle de r - δ_r est fermée). Les réponses à l'échelon de $G_1(s)$ (eq. 3.101) et $G_{1approx}(s)$ (eq. 3.103) sont aussi superposées. La figure 3.20(b) montre la réponse de la sortie du

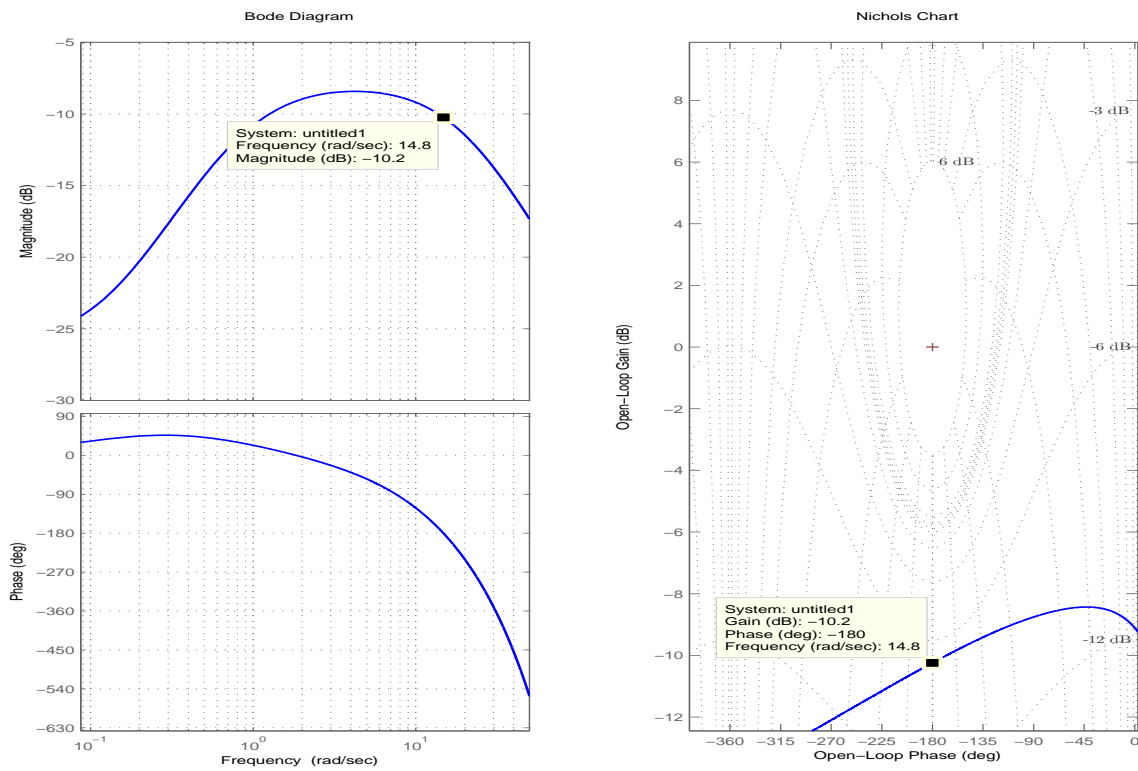
système r suite à un échelon δ_r (la boucle de p - δ_a est fermée). Les réponses à l'échelon des fonctions de transfert $G_2(s)$ (eq. 3.102) et $G_{2approx}(s)$ (eq. 3.104) sont aussi superposées. Les sorties p et r sont calculées avec les modèles des équations 3.82, 3.83, 3.85 et 3.86.

Pour l'analyse fréquentielle, on définit $G_p(s) = P_p G_1(s)$ et $G_r(s) = P_r G_2(s)$. On obtient respectivement des marges de gain de 10.1 dB (figure 3.21(a)) et 10.2 dB (figure 3.21(b)) avec des marges de phase à l'infini. La figure 3.22 montre les réponses de p et r obtenues suite à un changement de consigne p . On remarque l'erreur statique en p . On remarque aussi l'effet du *HPF*. On voit qu'en virage (avant le *HPF*), $r \neq 0$.

(a) Fonction de transfert vue par P_p (b) Fonction de transfert vue par P_r FIGURE 3.20 – Fonctions de transfert vues par P_p et P_r



(a) Analyse fréquentielle - Réponse en fréquences de G_p



(b) Analyse fréquentielle - Réponse en fréquences de G_r

FIGURE 3.21 – Analyse fréquentielle - Réponse en fréquences de $G_p(s)$ et $G_r(s)$

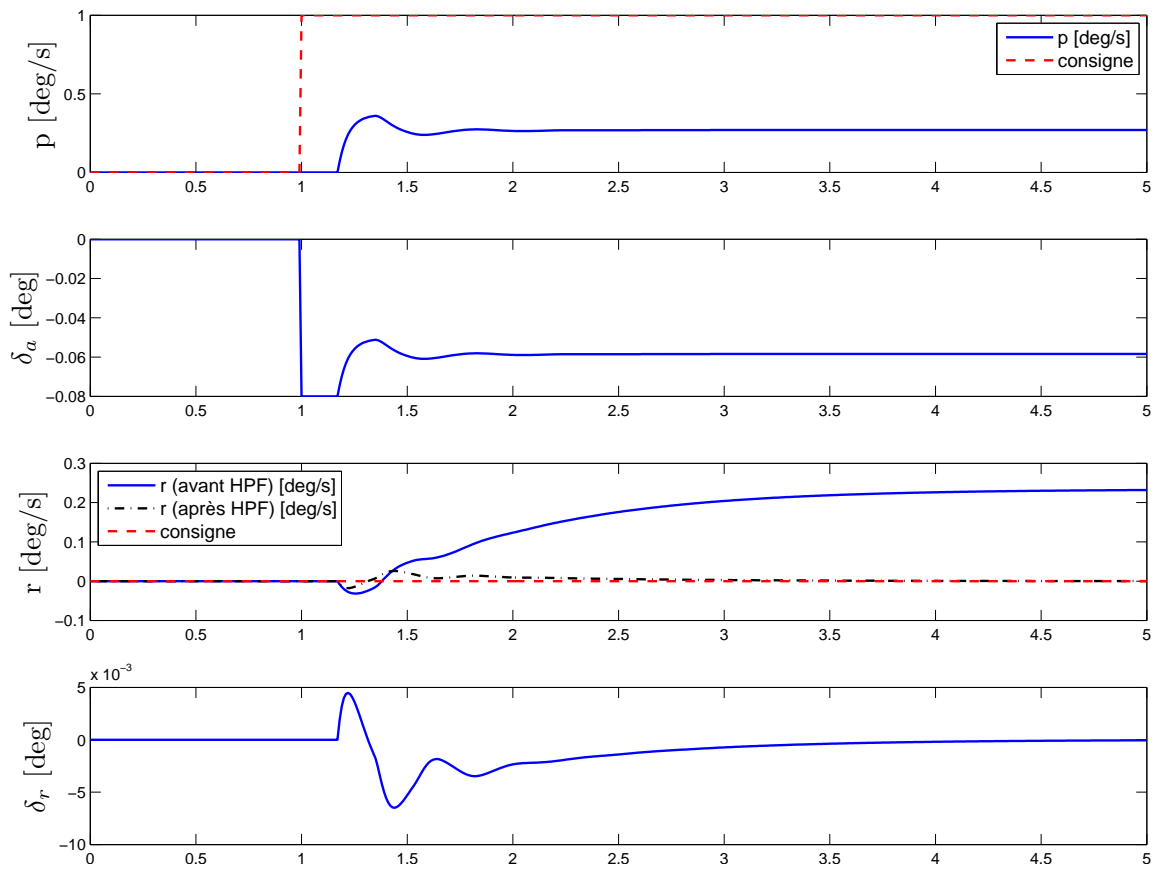


FIGURE 3.22 – Sorties du système p et r en asservissement

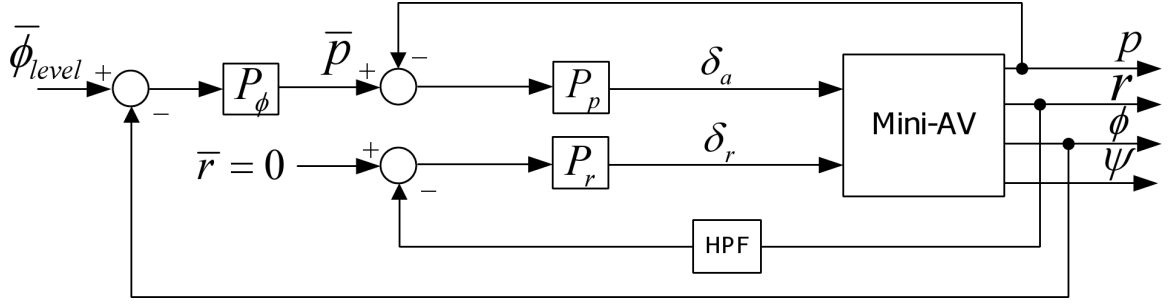


FIGURE 3.23 – Diagramme fonctionnel de la deuxième boucle interne

Par la suite, il faut calculer P_ϕ (figure 3.23). Pour ce faire, on trouve une approximation de la réponse de ϕ suite à un changement de consigne p . La figure 3.24 illustre cette réponse. Cette structure de contrôle permet d'effectuer des virages. La fonction de transfert vue par P_ϕ est :

$$\phi_{approx}(s) \approx \frac{0.38e^{-0.17s}}{s} \quad (3.105)$$

On utilise alors la méthode du réglage par placement de pôles (le retard est négligé dans la conception). Cette méthode consiste à sélectionner, pour un procédé $G_p(s)$ un régulateur $G_c(s)$ de manière à obtenir une certaine dynamique $H(s)$ en boucle fermée :

$$H(s) = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)} \quad (3.106)$$

$$= \frac{P_\phi \frac{0.38}{s}}{1 + P_\phi \frac{0.38}{s}} \quad (3.107)$$

Avec $P_\phi = 8$, on obtient une dynamique de $H(s) = \frac{1}{0.33s+1}$ (constante de temps $T_{phi} = 0.33$ sec). Pour l'analyse fréquentielle (le retard n'est pas négligé), on définit $G_\phi = P_\phi \phi_{approx}(s)$. On utilise le modèle ϕ_{approx} pour l'analyse fréquentielle pour des fins de simplification. On obtient une marge de gain de 9.63 dB et une marge de phase de 119 deg (figure 3.25). La figure 3.26 présente la réponse à un changement de consigne unitaire de ϕ du système en boucle fermée. Les sorties p , r et ϕ sont calculées avec les modèles des équations 3.82, 3.83, 3.84, 3.85, 3.86 et 3.87 (les retards ne sont pas négligés dans la simulation). On remarque que le temps de réponse de la sortie ϕ est de moins 1 seconde, ce qui respecte la dynamique $T_{phi} = 0.33$ seconde.

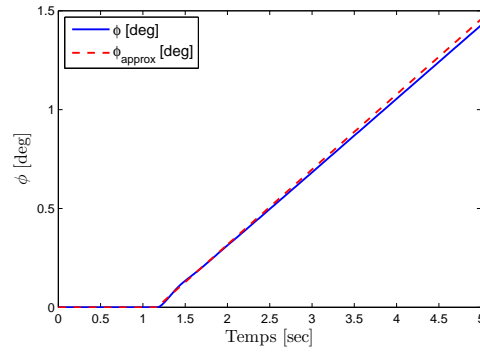


FIGURE 3.24 – Approximation de ϕ suite à un changement de consigne de p

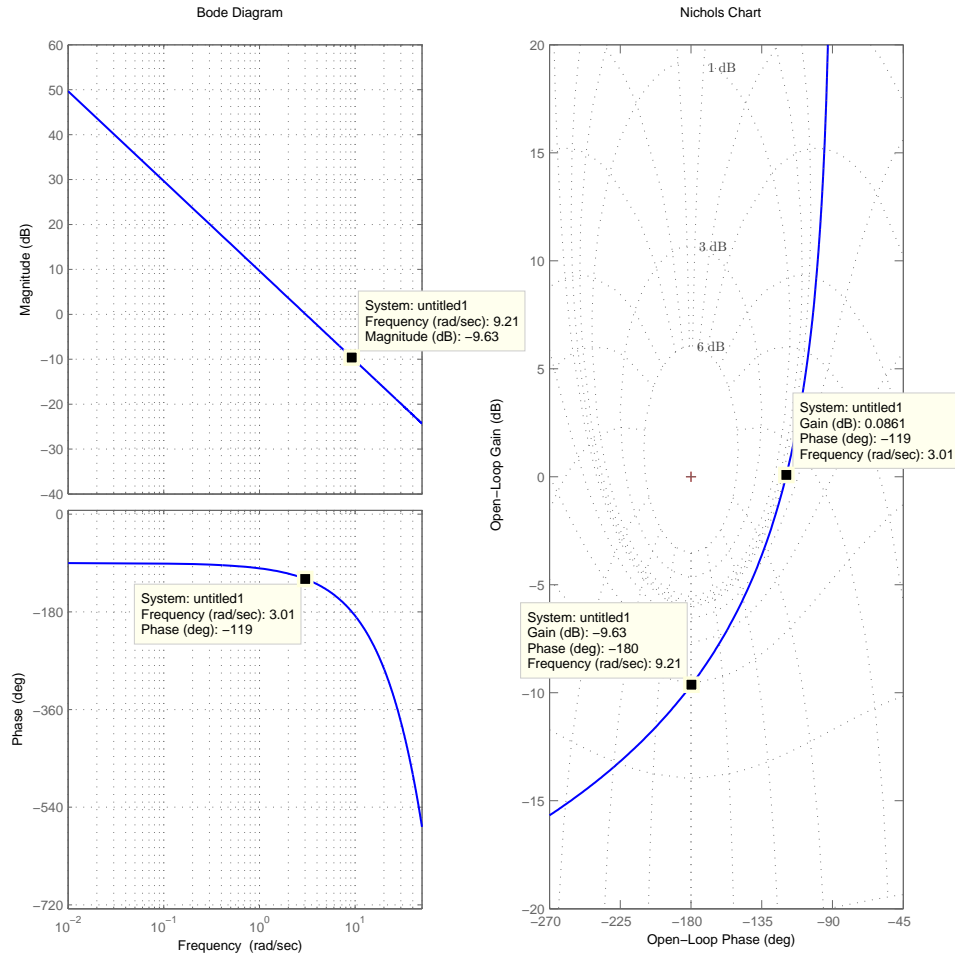


FIGURE 3.25 – Analyse fréquentielle - Réponse en fréquences de G_ϕ

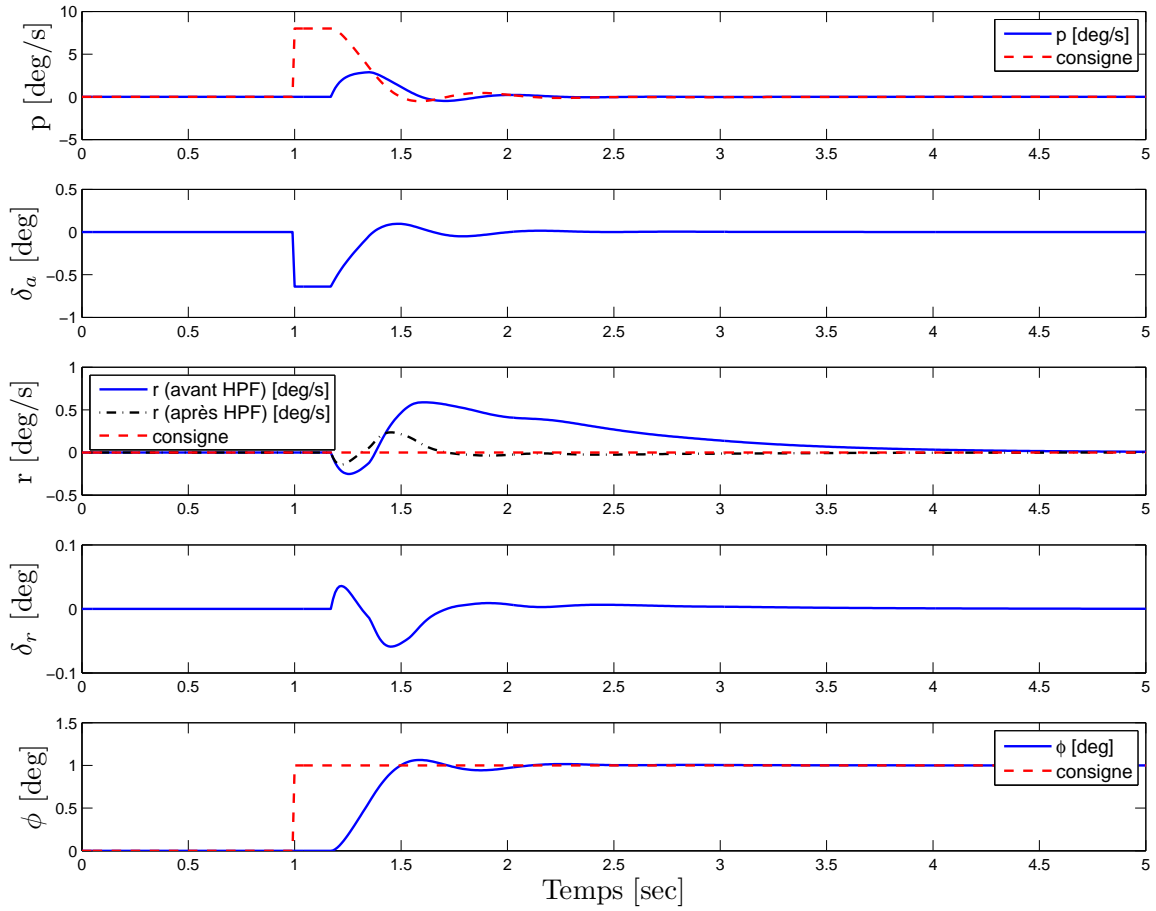


FIGURE 3.26 – Simulateur linéaire - changement de consigne ϕ de 1 degré - variables latérales

Enfin, il faut calculer PI_{Ψ} (figure 3.18). Pour ce faire, on trouve une approximation de la réponse de Ψ suite à un changement de consigne ϕ . La figure 3.27 illustre cette réponse. La fonction de transfert vue par P_{ϕ} est :

$$\Psi_{approx}(s) \approx \frac{1.4e^{-0.17s}}{s} \quad (3.108)$$

On utilise alors la même méthode de réglage de placement des pôles qu'avec $\frac{\theta(s)}{\delta_e(s)}$ (le retard est négligé dans la conception). Avec les équations 3.94 et 3.95, $\zeta = 0.7$, $T_{rep} = 3$ et en négligeant le retard, on obtient le PI suivant :

$$PI_{\Psi}(s) = \frac{1(1.4s + 1)}{1.4s} \quad (3.109)$$

La figure 3.29 présente la réponse à un changement de consigne unitaire non filtré de Ψ du système en boucle fermée (les retards ne sont pas négligés dans la simulation). Les sorties p , r , ϕ et Ψ sont calculées avec les modèles des équations 3.82, 3.83, 3.84, 3.85, 3.86, 3.87 et 3.88. Malgré le dépassement, on observe une erreur statique nulle. La figure 3.30 présente la réponse à un changement de consigne unitaire filtré de Ψ du système en boucle fermée. On remarque que l'ajout d'un filtre permet de rejoindre la consigne tout en enlevant le dépassement. Pour l'analyse fréquentielle (le retard n'est pas négligé), on définit $G_{\Psi} = PI_{\Psi}(s)\Psi_{approx}(s)$. On utilise le modèle Ψ_{approx} pour l'analyse fréquentielle pour des fins de simplification. La figure 3.28 présente l'analyse fréquentielle de G_{Ψ} . On observe une marge de gain de 15.9 dB et une marge de phase de 130 degrés.

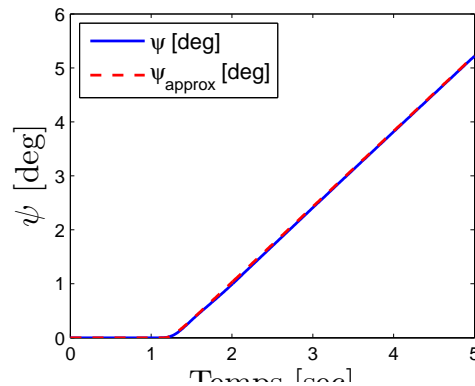


FIGURE 3.27 – Approximation de Ψ suite à un changement de consigne de ϕ

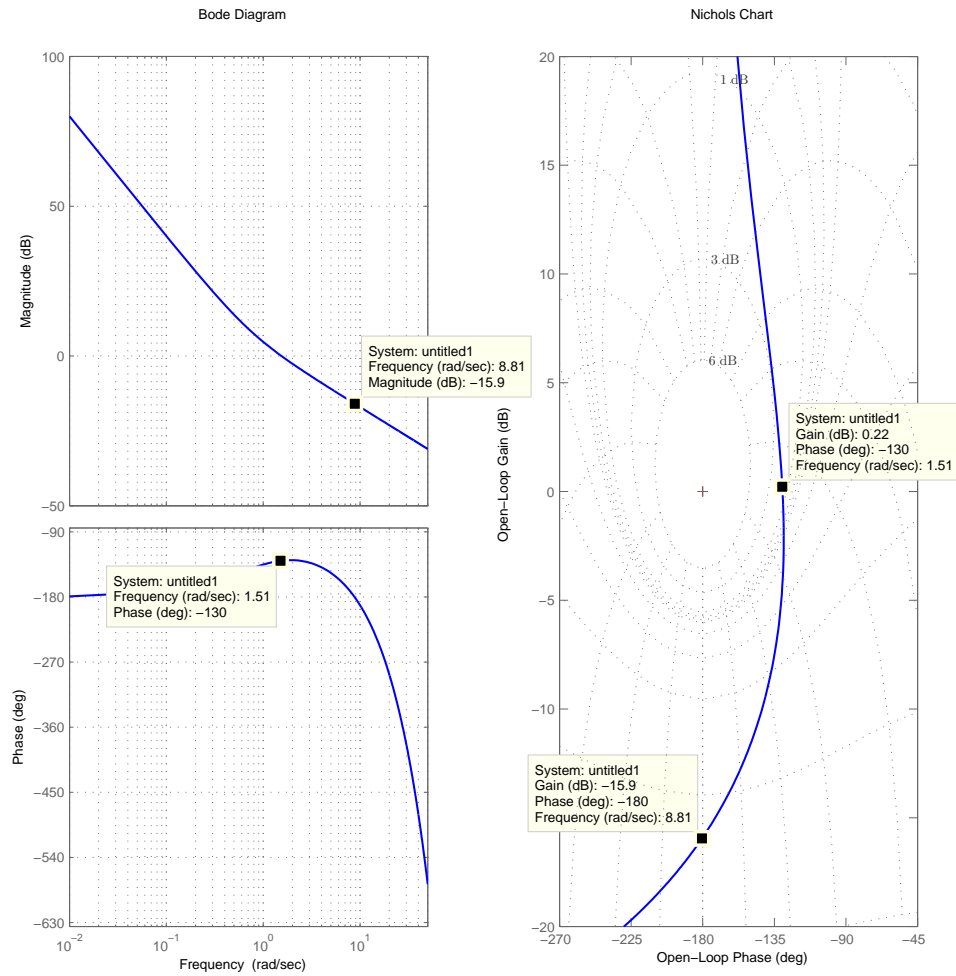


FIGURE 3.28 – Analyse fréquentielle - Réponse en fréquences de G_0

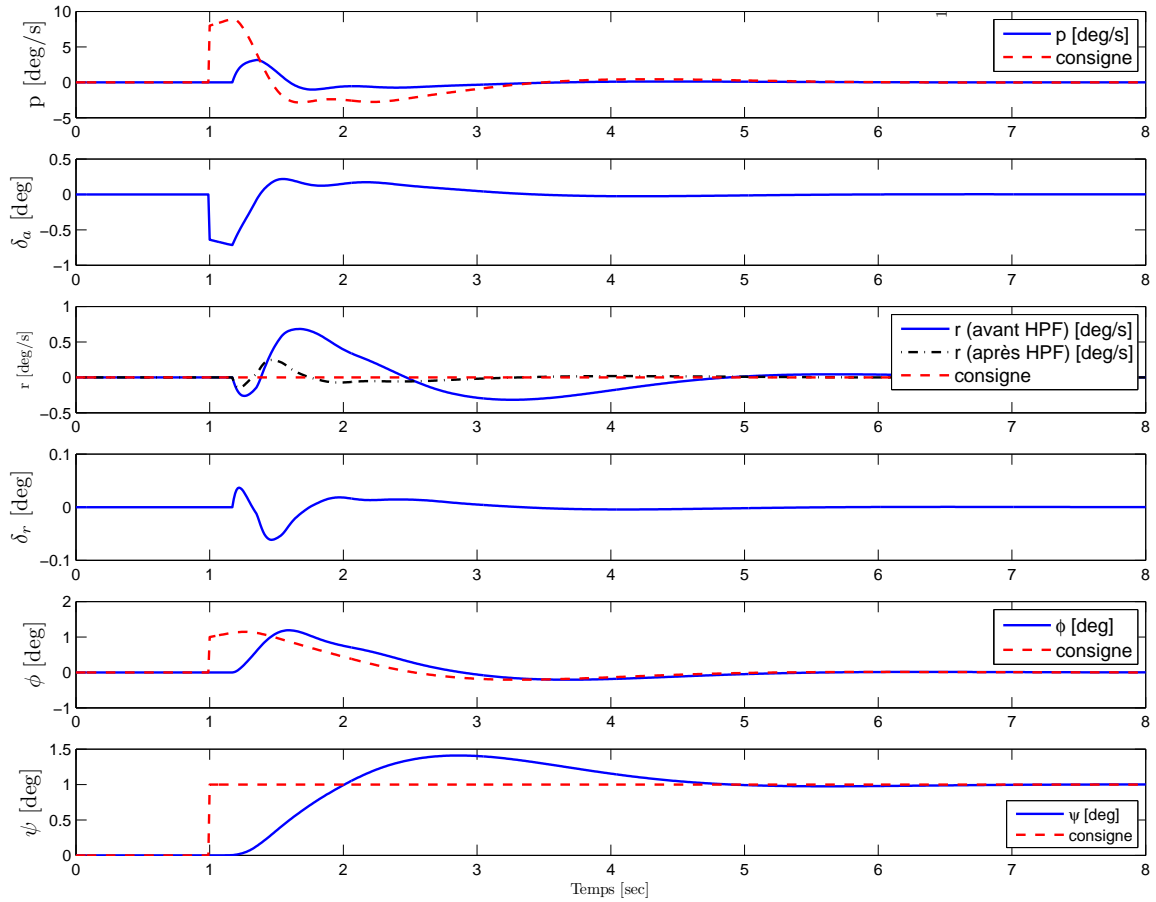


FIGURE 3.29 – Simulateur linéaire - changement de consigne (sans filtre) ψ de 1 degré - variables latérales

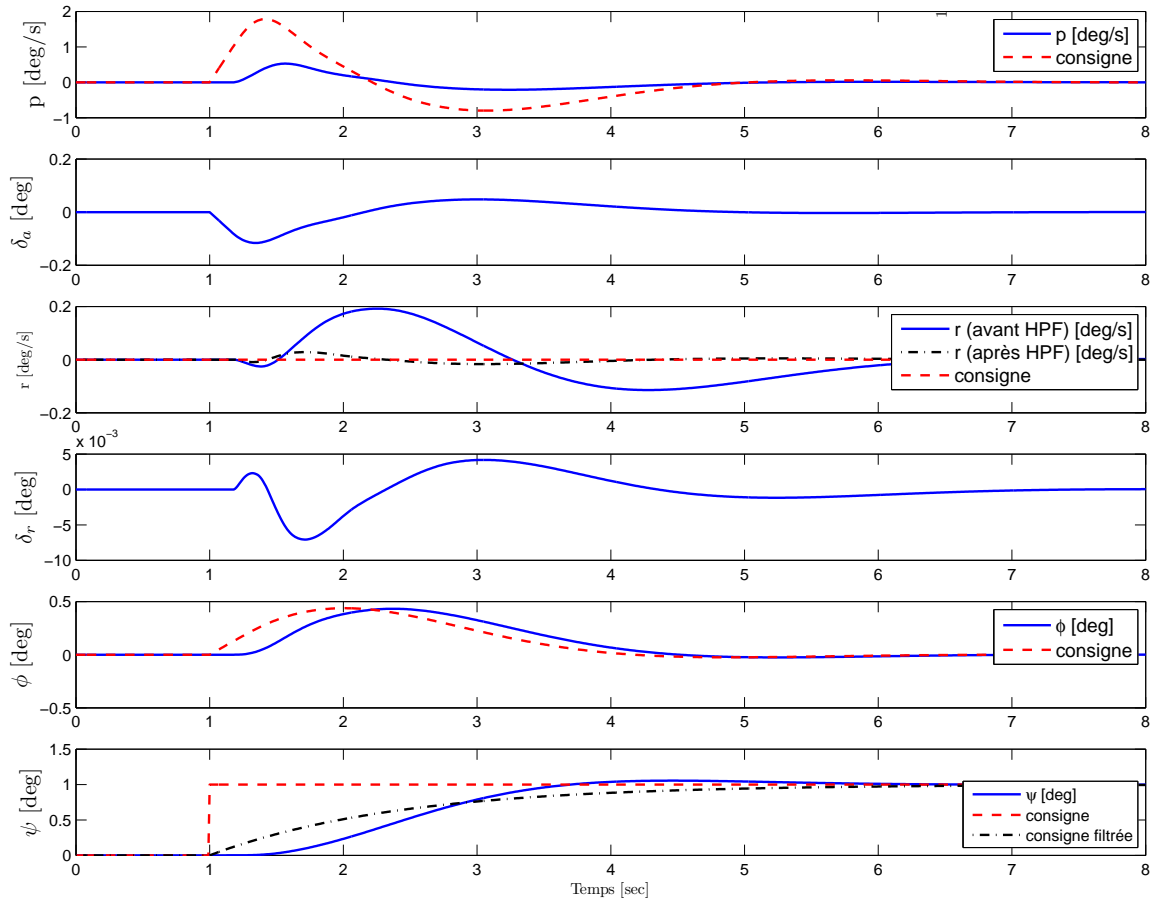


FIGURE 3.30 – Simulateur linéaire - changement de consigne (avec filtre) ψ de 1 degré - variables latérales

3.1.5 Simulateur non-linéaire du NRC-CNRC

Lors du processus itératif du choix de la plateforme, le RDDC Valcartier et l'Institut de recherche aérospatiale du Conseil national de recherches Canada (NRC-CNRC) ont collaboré [21] afin de développer des modèles aérodynamiques des modes non stationnaires et stationnaires de quelques plateformes de type MiniAV, dont le Flatana. Avec des données de vols qui ont été collectées, le NRC-CNRC a utilisé une technique de reconstruction des données de vols et une méthode d'estimation du maximum de vraisemblance (*MLE*) pour estimer les modèles aérodynamiques [21]. En plus des données du IMU, les données de vitesse V_t et d'altitude provenant du tube de Pitot ainsi des données de position provenant d'un système de caméra ont été utilisées pour développer, en plus des modèles $p, q, r, \phi, \theta, \psi$, un modèle de vitesse u, v, w et de position x, y, z .

Un modèle à 6 degrés de liberté couplé, pour tenir compte de l'effet de δ_e sur la dynamique du lacet, a été identifié. Ce comportement couplé est probablement dû à une asymétrie de la plateforme qui a été utilisée lors des tests d'identification (moteur désenligné, distribution asymétrique du poids, la gouverne de profondeur δ_e désenlignée). Le modèle mathématique (coefficients des matrices de stabilité et de contrôle, sections 3.1.1 et 3.1.1) du Flatana est incorporé dans Matlab/Simulink avec l'outil *Aerosim* en utilisant le bloc *6-DOF Aircraft Model (Body-frame EOM)*.

Les matrices des coefficients de stabilité et de contrôle dimensionnel identifiées par le NRC sont données dans l'annexe A.2. En utilisant les matrices de l'équation 3.11 et 3.14 avec $U_{op} = 28f/s$, $\theta_{op} = 18 \text{ deg}$ et $g \approx 32.2f/s^2$ et en supposant que les termes $Z_{\dot{w}} \approx 0$ et $M_{\dot{w}} \approx 0$ (ils sont supposés très petits), on obtient :

- Mode longitudinal : on retrouve le mode *phugoid* ($\lambda_{ph} = -0.146 \pm 0.584i$) et le mode *short period* ($\lambda_{sp} = -2.89 \pm 7.27i$). Tous les pôles sont stables.
- Mode latéral : on retrouve le mode *roll* avec $\lambda_r = -8.1$, le mode *spiral* $\lambda_s = 0.281$ et le *dutch roll mode* avec $\lambda_d = -2.28 \pm 8.14$. On remarque que le mode *roll* et le mode *dutch roll* sont stables, tandis que le mode *spiral* est instable (probablement dû à l'asymétrie de la plateforme).

Les tableaux 3.2 et 3.3 présentent une comparaison entre les deux méthodes d'identification. Même si le Flatana identifié est quelque peu différent mécaniquement de celui qui est utilisé dans le cadre de ce projet, on remarque que le modèle en mode non stationnaire du NRC reste assez valide pour effectuer des simulations avec les autopilotes qui ont été développés dans la section 3.1.4.

	<i>Phugoid</i> (λ_{ph})	<i>Short Period</i> (λ_{sp})
Section 3.1.3	0	$-5 \pm 11.5i$
NRC-CNRC [21]	$-0.146 \pm 0.584i$	$-2.89 \pm 7.27i$

TABLE 3.2 – Tableau comparatif d’identification - Modes longitudinaux

	<i>Roll</i> λ_r	<i>Spiral</i> (λ_s)	<i>Dutch roll</i> (λ_d)
Section 3.1.3	$-25 < \lambda_r < -14$	$(-1.17 < \lambda_s < -0.66)$	
NRC-CNRC [21]	-8.1	0.281	-2.28 ± 8.14

TABLE 3.3 – Tableau comparatif d’identification - Modes latéraux

Les autopilotes conçus dans la section 3.1.4 sont intégrés dans le simulateur du NRC-CNRC (structures de commande des figures 3.13 et 3.18). Dans le contrôleur latéral de la figure 3.18, un interrupteur est intégré avant la consigne $\bar{\phi}$ pour pouvoir faire un changement de consigne pour réaliser un essai où l’avion effectue un virage coordonné avec un angle ϕ (structure de commande de la figure 3.23). Des régulateurs proportionnels P_z et P_u (figure 3.31), qui ont été trouvés itérativement, sont aussi inclus pour permettre à l’avion de garder une altitude et une vitesse quasi constante durant l’essai. L’erreur en altitude z est convertie en une consigne de θ par le régulateur P_z et l’erreur en vitesse u est convertie en une commande δ_t par le régulateur P_u .

Quand la simulation débute, l’avion est en vol stable et on peut choisir les conditions initiales voulues (vitesse u_0, v_0, w_0 , vitesse angulaire p_0, q_0, r_0 , attitude ϕ_0, θ_0, ψ_0 et position x_0, y_0, z_0). L’avion débute son vol avec une vitesse de $u_0 = 8,4m/s$, une altitude de $z_0 = 3m$ et une attitude de $\theta_0 = 18$ degrés. Les autres conditions initiales sont nulles. Pendant les 5 premières secondes, les autopilotes stabilisent l’avion autour de son point d’opération. Aussi, durant les essais, l’avion doit garder une vitesse et une altitude constantes. À $t = 5$ secondes, on effectue les changements de consigne pour tester les autopilotes en poursuite.

Les figures 3.32 et 3.33 présentent les réponses à un changement de consigne ϕ de 10 degrés (structure de commande de la figure 3.23). Ainsi, l’avion se met à tourner en rond. On remarque, lors du changement de consigne ϕ , que le mouvement en lacet engendré génère une perturbation sur l’altitude z (il y a une perte d’altitude). Le régulateur $PI_{\theta AW}$ parvient à stabiliser l’angle θ pour permettre de maintenir une altitude quasi constante. On remarque aussi l’erreur statique de l’angle ϕ , due au régulateur proportionnel P_ϕ .

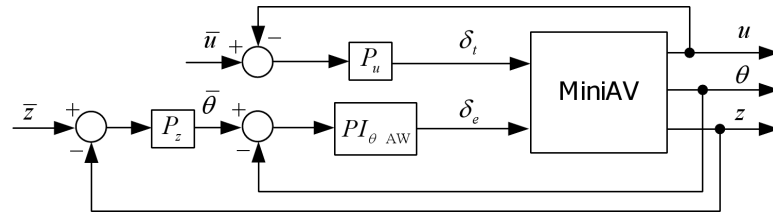


FIGURE 3.31 – Diagramme fonctionnel d'un contrôleur longitudinal pour maintenir une vitesse u et une altitude z constantes

Les figures 3.34 et 3.35 présentent respectivement les réponses à un changement de consigne ψ de 10 degrés non-filtrée et filtrée. L'avion effectue donc un changement de cap de 10 degrés. Dans la figure 3.34, on remarque un dépassement assez important de l'angle ψ lors du changement de consigne. Le filtre aide donc à limiter le dépassement.

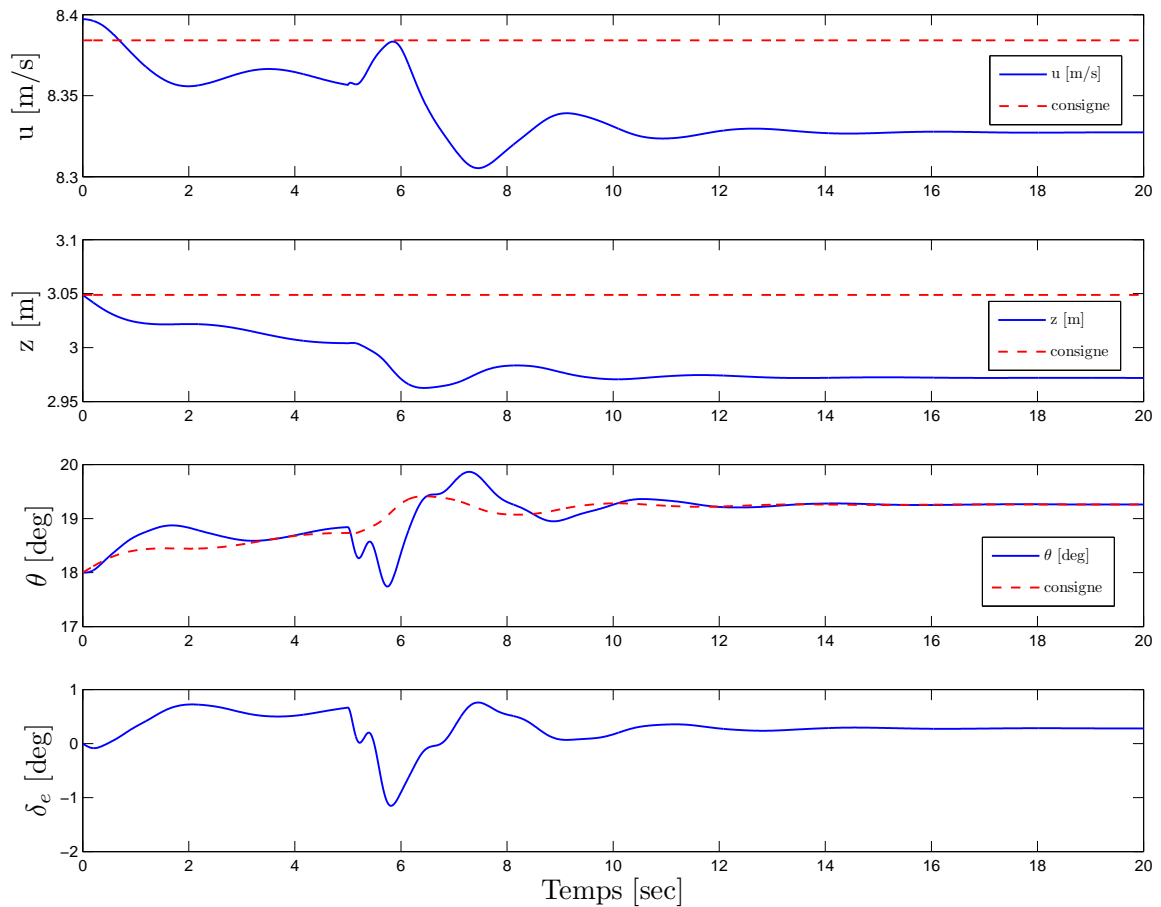


FIGURE 3.32 – Simulateur non-linéaire NRC - changement de consigne ϕ de 10 degrés - variables longitudinales

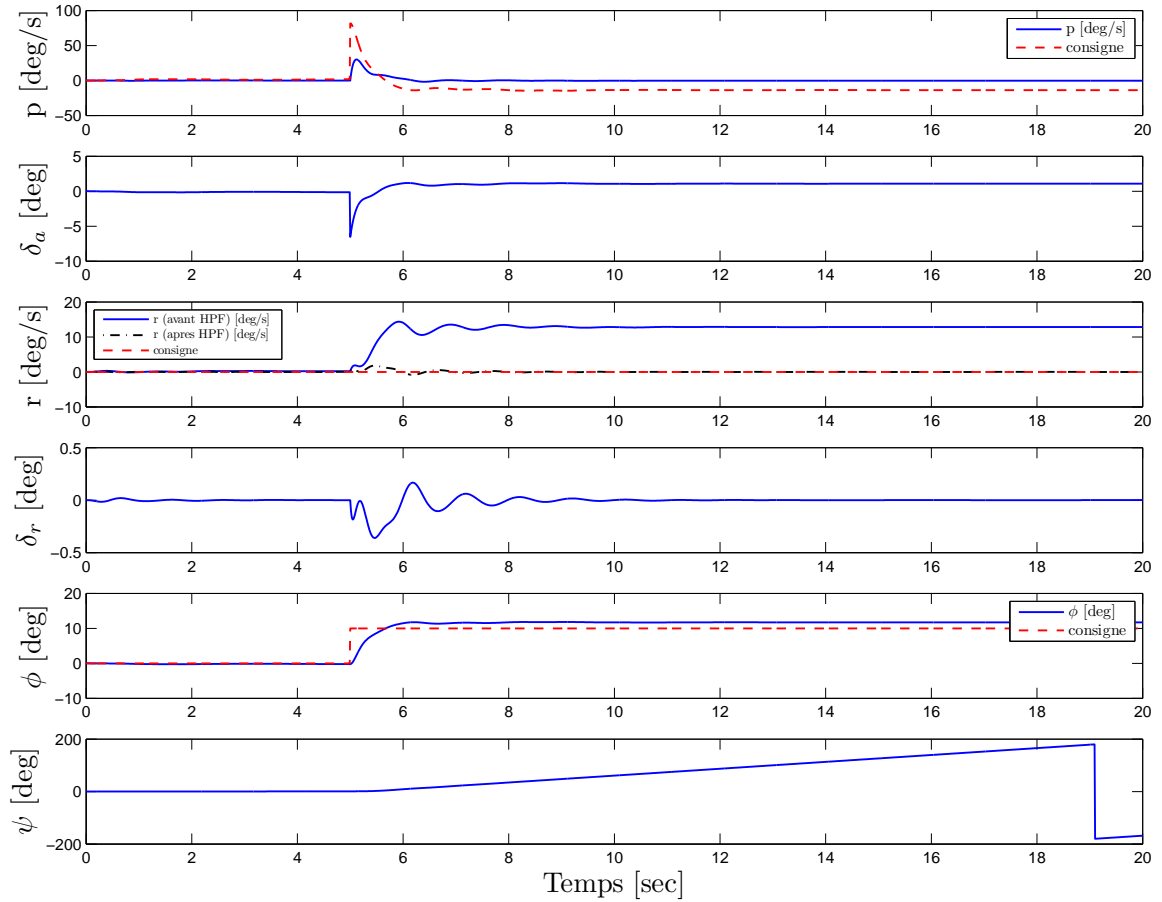


FIGURE 3.33 – Simulateur non-linéaire NRC - changement de consigne ϕ de 10 degrés - variables latérales

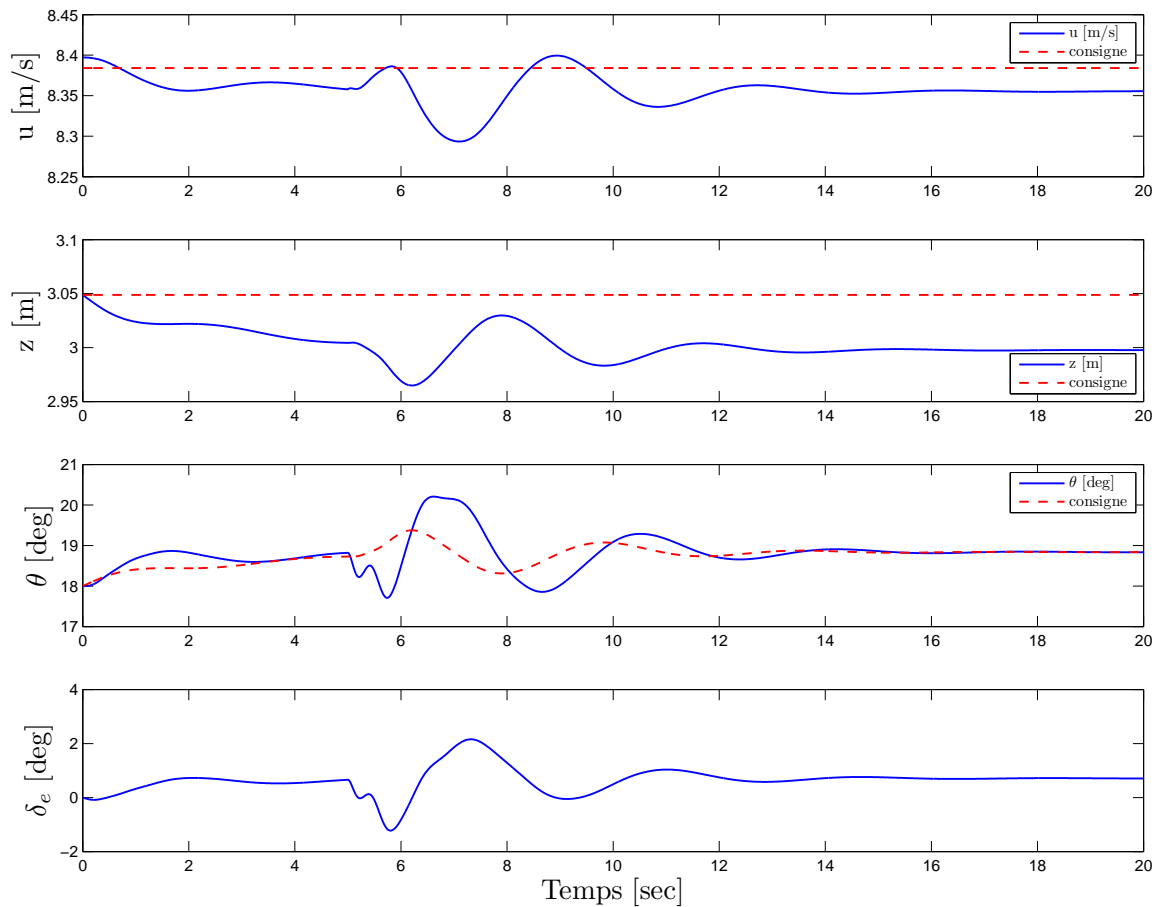


FIGURE 3.34 – Simulateur non-linéaire NRC - changement de consigne (sans filtre) ψ de 10 degrés - variables latérales

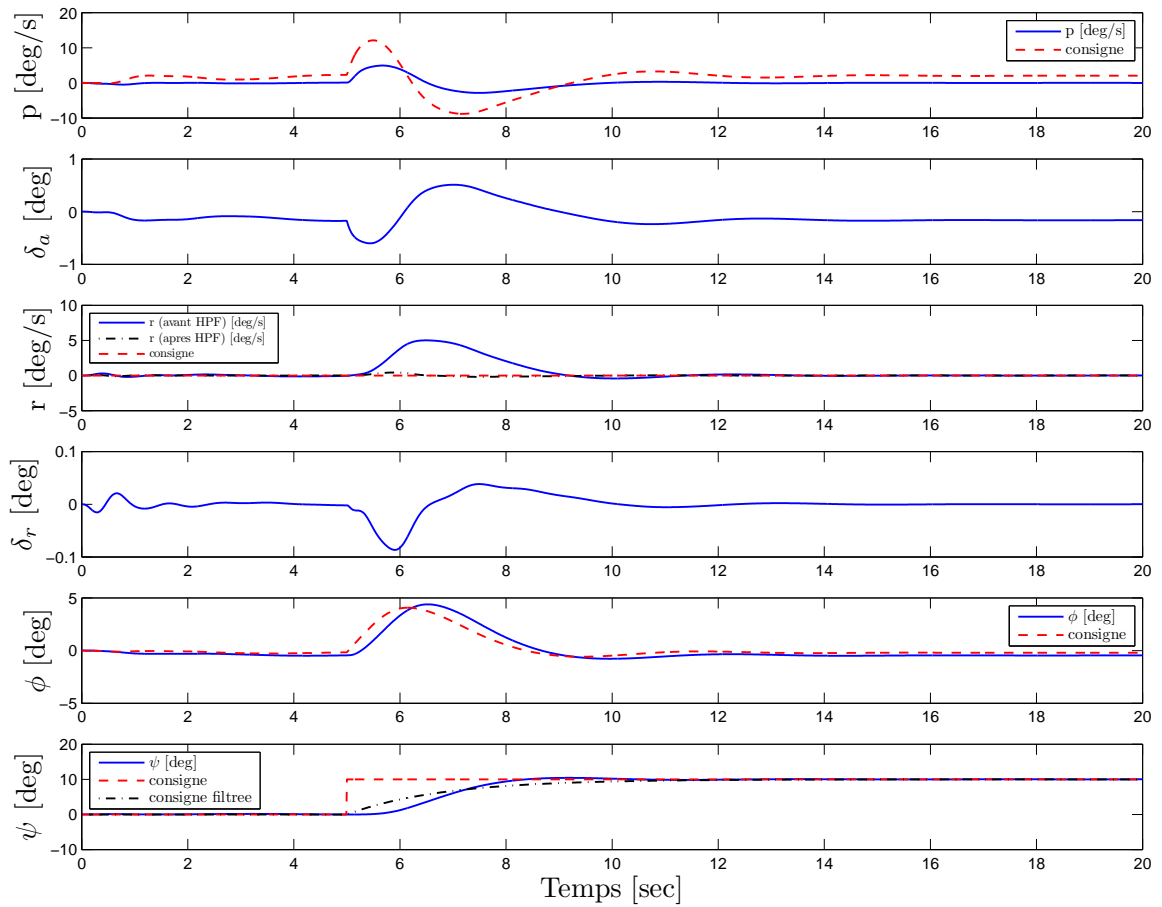


FIGURE 3.35 – Simulateur non-linéaire NRC - changement de consigne (avec filtre) ψ de 10 degrés - variables latérales

3.2 Vol stationnaire

L'identification des structures de modèle et des contrôleurs en mode stationnaire pour cette plateforme a été réalisée par Bilodeau et al. [2, 3]. Des hypothèses de simplification du comportement du mode stationnaire ont été formulées et une linéarisation des équations non-linéaires à 6 degrés de liberté d'un corps rigide a été réalisée pour obtenir une structure qui est découplée sur chaque axe. Les hypothèses suivantes sont émises :

- le repère avion (section 2.1.1) est utilisé ;
- le système peut être découplé en trois sous-systèmes, un pour chaque axe ;
- on suppose une vitesse de translation verticale faible, de sorte que les forces générées par la gouverne des gaz δ_t sont égales à la force gravitationnelle ;
- les vitesses angulaires de chaque sous-système sont petites ;
- les termes en accélération, qui sont le produit d'une vitesse angulaire et d'une vitesse de déplacement (accélération de Coriolis, ex : $r\dot{v}$), sont négligés.

Ainsi, chaque axe possède une interaction entrée-sortie significative (X_B avec δ_a , Y_B avec δ_e et Z_B avec δ_r). Le modèle a été identifié avec des données d'expérimentation, suivant un peu le même principe d'identification de la section 3.1.3. Les structures découplées identifiées sont les suivantes :

$$\frac{p(s)}{\delta_a(s)} = \frac{-5.44 e^{-0.17s}}{(0.67s + 1)} \quad (3.110)$$

$$\frac{q(s)}{\delta_e(s)} = \frac{-7.96 s e^{-0.17s}}{(5s - 1)(0.08s + 1)} \quad (3.111)$$

$$\frac{r(s)}{\delta_r(s)} = \frac{2.46 s e^{-0.17s}}{(1.43s - 1)(0.09s + 1)} \quad (3.112)$$

On fait l'hypothèse que l'attitude est obtenue en intégrant les vitesses angulaires :

$$\frac{\phi(s)}{\delta_a(s)} = \frac{p(s)}{s \delta_a(s)} \quad (3.113)$$

$$\frac{\theta(s)}{\delta_e(s)} = \frac{q(s)}{s \delta_e(s)} \quad (3.114)$$

$$\frac{\psi(s)}{\delta_r(s)} = \frac{r(s)}{s \delta_r(s)} \quad (3.115)$$

Comme les angles d'Euler possèdent des singularités à $\theta = \pm 90^\circ$ (section 2.2), les quaternions sont utilisés pour décrire l'attitude du MiniAV en mode stationnaire. Le quaternion de consigne est généré avec l'équation 2.44. Des quaternions d'erreur sont générés pour trouver les erreurs angulaires. L'algorithme de Green est exécuté à chaque

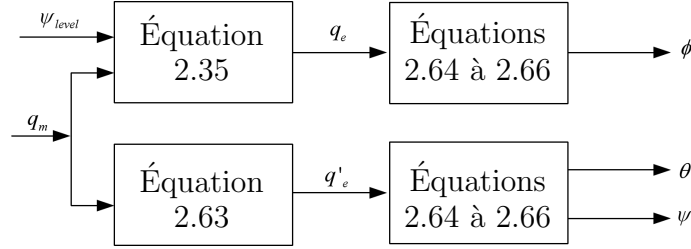


FIGURE 3.36 – Diagramme fonctionnel du calcul de l’attitude du MiniAV en mode stationnaire

k (équation 2.63) pour générer les erreurs en tangage θ et en lacet ψ de sorte que l’erreur ϕ soit toujours près de 0 (les erreurs en roulis sont découplés de la direction du véhicule). Ainsi peu importe la direction du ventre de l’avion lors des essais, cet algorithme fait en sorte que l’axe X_B de l’avion soit colinéaire avec l’axe Z_E . Le quaternion d’erreur en ϕ est calculé avec l’équation 2.35 à chaque itération k (période d’échantillonnage $T_s = 0.01$) avec le cap ψ_{level} qui est transmis par le superviseur. Les erreurs angulaires sont calculées grâce aux équations 2.64 à 2.66. La figure 3.36 présente un diagramme fonctionnel du calcul de l’attitude du MiniAV en mode stationnaire.

Même s’il existe des perturbations provenant de la déflexion de l’air par δ_a sur δ_e et δ_r , le couplage est négligé dans la conception. Des boucles cascades sont utilisées pour contrôler chaque axe du mode stationnaire (figure 3.38 à 3.40). La figure 3.37 présente un contrôleur générique monovariable du mode stationnaire où $\bar{\Theta}$ représente la consigne en attitude $[\bar{\phi}, \bar{\theta}, \bar{\psi}]$, Θ est l’attitude mesurée $[\phi, \theta, \psi]$, Ω est la vitesse angulaire $[p, q, r]$ et δ représente les variables manipulées $[\delta_a, \delta_e, \delta_r]$. La boucle interne a un gain proportionnel P pour permettre d’accélérer le système. La boucle externe est utilisée pour assurer d’avoir une erreur statique nulle. Des PI_{AW} sont alors choisis pour contrôler chacun des axes. L’ajustement des paramètres est réalisé en utilisant une méthode fréquentielle basée sur le diagramme de Nichols. Dans le contrôleur de l’axe X_B axis, $\bar{\phi}$ est choisi de sorte que le ventre de l’avion soit face à une certaine direction ψ_{level} . Dans le contrôleur de l’axe Y_B , $\bar{\theta}$ est choisi à 90 deg. Dans le contrôleur de l’axe Z_B , $\bar{\psi}$ est choisi 0 deg. Le tableau 3.4 présente les fonctions de transfert des contrôleurs du mode stationnaire.

TABLE 3.4 – Fonctions de transfert des contrôleurs du mode stationnaire

	$\phi[deg]$	$\theta[deg]$	$\psi[deg]$	$p[deg/s]$	$q[deg/s]$	$r[deg/s]$
P_Ω				-0.3	-0.18	0.16
PI_Θ	$\frac{-2.5(3s+1)}{3s}$	$\frac{-7(2.5s+1)}{2.5s}$	$\frac{-5(2s+1)}{2s}$			

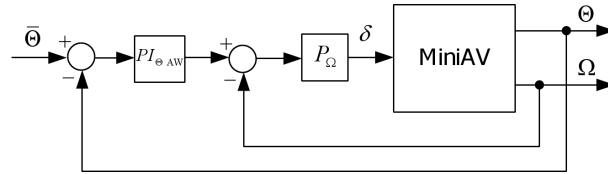
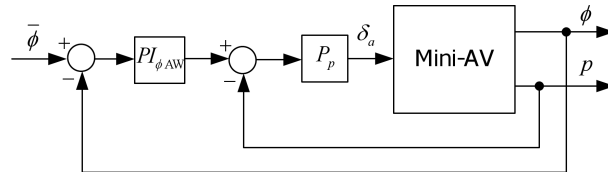
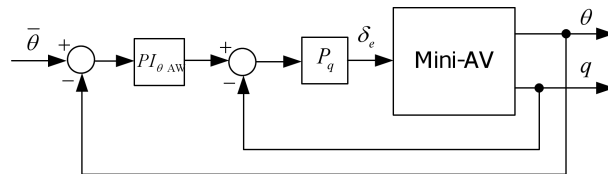
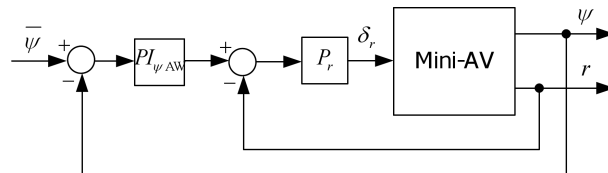


FIGURE 3.37 – Diagramme fonctionnel générique d'un contrôleur monovariante en mode stationnaire

FIGURE 3.38 – Diagramme fonctionnel du contrôleur de l'axe X_B en mode stationnaireFIGURE 3.39 – Diagramme fonctionnel du contrôleur de l'axe Y_B en mode stationnaireFIGURE 3.40 – Diagramme fonctionnel du contrôleur de l'axe Z_B en mode stationnaire

En mode stationnaire, δ_t est utilisé pour maintenir une altitude z . En premier lieu, il est possible de contrôler manuellement l'altitude en boucle ouverte en contrôlant manuellement δ_t . L'utilisateur doit cependant avoir une certaine expérience pour réaliser un tel contrôle. Ainsi, il est préférable de réaliser le contrôle de l'altitude en boucle fermée. Pour ce faire, l'altitude doit être connue. Un capteur ultrasonique, comme le propose Green [18] et Kita [26], peut être utilisé. Malheureusement, un capteur ultrasonique a une portée maximale de 6 mètres par rapport au sol et la performance diminue sur des surfaces inégales comme du gazon. Un capteur de pression atmosphérique peut aussi être utilisé pour des altitudes plus élevées, mais la précision est d'environ 2-3 mètres, ce qui peut être problématique pour ce type de véhicule. Finalement, un *GPS* peut générer l'altitude (avec quelques mètres de précision), mais la couverture satellitaire peut être intermittente ou perdue dans certaines zones ou dans un immeuble, ce qui en réduit la précision.

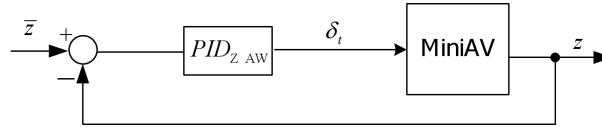


FIGURE 3.41 – Diagramme de blocs du contrôleur de l'altitude en mode stationnaire

Comme outil de développement pour le contrôle de l'altitude z , un système de vision sert de capteur. Ainsi, un $PID_z AW$ (voir section 3.1.4) est utilisé pour effectuer le contrôle de l'altitude en boucle fermée (Fig. 3.41). Enfin, le tableau 3.5 montre la fonction de transfert identifiée $\frac{z(s)}{\delta_t(s)}$ de l'altitude z en fonction de δ_t et le contrôleur $PID_z AW$ qui lui est associé.

TABLE 3.5 – Fonctions de transfert des contrôleurs du mode stationnaire

$\frac{z(s)}{\delta_t(s)}$	$\frac{-32.58 e^{-0.22s}}{s(1.21s+1)}$
$PID_z AW$	$\frac{-0.05(2s+1)(1.25s+1)}{2s(0.1s+1)}$

3.3 Conclusion

La méthodologie qui a été utilisée pour développer les lois de commande pour les vols non stationnaires et stationnaires a été présentée. Une linéarisation des équations non-linéaires à 6 degrés de liberté a été utilisée pour approximer le comportement de l'avion. Les principaux comportements aérodynamiques et leurs caractéristiques, sous forme de fonctions de transfert, ont été présentés et détaillés. Cette connaissance du comportement du système a permis ensuite de réaliser l'identification du système selon des étapes précises (planification, réalisation, traitement des données et choix des structures). À partir de ces modèles identifiés, des autopilotes ont été conçus avec des structures bien connues dans la littérature et réglés avec une méthode fréquentielle. Enfin, la stratégie pour la commande du vol stationnaire développée par Bilodeau et al. [2, 3] est résumée. En utilisant les loi de commande pour les deux principaux modes de vol qui ont été développés dans ce chapitre, le chapitre 4 définit les stratégies pour réaliser la transition entre les deux modes.

Chapitre 4

Stratégies de transition

Ce chapitre présente les différentes stratégies de transition qui ont été conçues pour réaliser les manoeuvres de transition. En premier lieu, une analyse des équations de forces aérodynamiques des variables longitudinales lors d'une transition et une analyse comparative des fonctions de transfert des deux principaux modes de vol sont effectuées. Par la suite, l'approche systématique pour réaliser les transitions est détaillée. Tout d'abord, le superviseur basé sur la logique, qui s'occupe du moment de la commutation entre les contrôleurs, est décrit et le principe de la commutation sans-à-coup est expliqué. Enfin, les différentes étapes nécessaires aux transitions L2H et H2L sont décrites.

4.1 Équations de forces aérodynamiques

A priori, il est facile de constater que le comportement entre les deux modes de vols principaux est non-linéaire. On peut comprendre ce phénomène analytiquement avec les équations de forces aérodynamiques. Les 4 forces en vol qui agissent sur l'avion sont les suivantes : la force ascendante (*lift*, L), la force du poids (*weight*, G), la force de poussée (*trust*, T) et la force de traînage (*drag*, D). Ici, on considère les variables longitudinales pour illustrer le comportement durant les transitions (figure 4.1). La force T est colinéaire avec l'axe X_B , la force G est colinéaire avec l'axe Z_E , la portance L est la composante normale à la vitesse V et la force de traînage est perpendiculaire à la portance L .

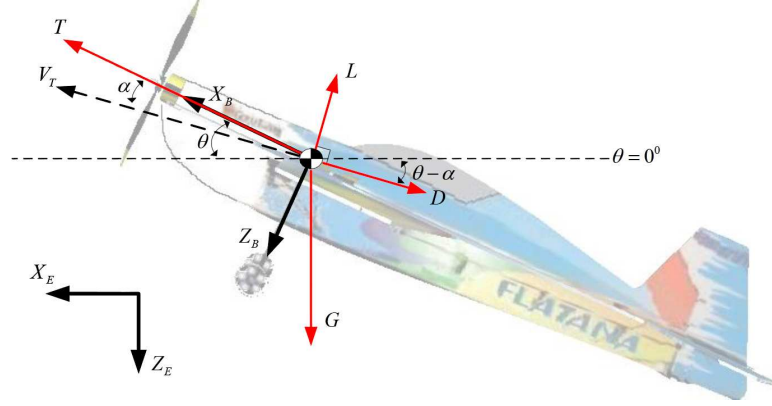


FIGURE 4.1 – Équations de forces - plan longitudinal

En faisant la sommation des vecteurs de force dans le référentiel de l'avion, on a :

$$m \begin{bmatrix} \ddot{X}_B \\ \ddot{Z}_B \end{bmatrix} = G + L + D + T \quad (4.1)$$

$$\begin{bmatrix} \ddot{X}_B \\ \ddot{Z}_B \end{bmatrix} = g \begin{bmatrix} \sin \theta \\ \cos \theta \end{bmatrix} + \frac{L}{m} \begin{bmatrix} \sin \alpha \\ -\cos \alpha \end{bmatrix} + \frac{D}{m} \begin{bmatrix} -\cos \alpha \\ \sin \alpha \end{bmatrix} + \frac{T}{m} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (4.2)$$

Si on transforme dans le référentiel inertiel NED , on obtient :

$$\begin{bmatrix} \ddot{X}_E \\ \ddot{Z}_E \end{bmatrix} = -g \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{L}{m} \begin{bmatrix} -\sin(\theta - \alpha) \\ -\cos(\theta - \alpha) \end{bmatrix} + \frac{D}{m} \begin{bmatrix} -\cos(\theta - \alpha) \\ \sin(\theta - \alpha) \end{bmatrix} + \frac{T}{m} \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \quad (4.3)$$

avec :

$$L = \frac{1}{2} \rho V^2 S C_L(\alpha) \quad (4.4)$$

$$D = \frac{1}{2} \rho V^2 S C_D(\alpha) \quad (4.5)$$

où $V = \sqrt{\dot{X}_E^2 + \dot{Z}_E^2}$ est la vitesse tangentielle à la trajectoire de vol, ρ est la densité de l'air et S est la surface des ailes. $C_L(\alpha)$ et $C_D(\alpha)$ sont respectivement les coefficients sans dimension de portance et de traînage qui varient en fonction de α . Ces fonctions ne sont pas linéaires. Aussi, tout objet qui se déplace dans l'air ou qui est soumis à un vent relatif subit une résistance à l'avancement (traînage). Finalement, on obtient :

$$\ddot{X}_E = \frac{-\rho V^2 S}{2m} (C_L(\alpha) \sin(\theta - \alpha) + C_D(\alpha) \cos(\theta - \alpha)) + \frac{T}{m} \cos \theta \quad (4.6)$$

$$\ddot{Z}_E = \frac{-\rho V^2 S}{2m} (C_L(\alpha) \cos(\theta - \alpha) - C_D(\alpha) \sin(\theta - \alpha)) + \frac{T}{m} \sin \theta \quad (4.7)$$

On remarque que lorsque l'avion est près de la position stationnaire (i.e. avec $\theta \approx 90$ et $\alpha \approx 0$), la force de poussée T générée par l'hélice doit balancer pour le poids de l'avion. Aussi, il existe un angle d'attaque α pour lequel les ailerons ne peuvent plus contribuer à la force ascendante. On a donc un système d'équations non-linéaires qui dépendent à la fois des angles θ et α , de la force de poussée T et de coefficients qui varient en fonction de α . Créer un seul système unifié capable de modéliser parfaitement l'avion dans les modes principaux et dans les transitions requiert des modèles beaucoup plus complexes comme les systèmes planifiés, les systèmes non-linéaires, les systèmes adaptatifs ou les réseaux de neurones. Ainsi, des stratégies de commande non-linéaires pourraient être élaborées pour avoir un système de contrôle unique pour l'ensemble de l'enveloppe de vol. Par exemple, Johnson et al.[23, 24] ont développé un réseau de neurones adaptatif en temps réel et un contrôleur adaptatif pour contrôler leur plateforme dans l'ensemble de l'enveloppe de vol. Aussi, Anderson et Stone [1] ont développé une stratégie de commande prédictive et un contrôleur unique est utilisé pour le système en entier.

4.2 Analyse comparative des deux principaux modes de vol

Il y a 4 variables manipulées : δ_e , δ_t , δ_a et δ_r . Dans le mode non stationnaire (figure 4.2(a)), δ_e est principalement utilisé pour contrôler l'altitude, via l'angle θ ; δ_t est utilisé pour contrôler la vitesse de translation u ; δ_a et δ_r sont couplées et sont utilisées en combinaison pour le contrôle latéral, c'est-à-dire pour effectuer un virage coordonné avec un angle ϕ_{level} ou maintenir un cap ψ_{level} . En mode stationnaire (figure 4.2(b)), les variables manipulées jouent des rôles un peu différents. δ_e est utilisé pour stabiliser l'axe Y_B de la plateforme autour de $\theta = 90^0$; δ_t est utilisé pour contrôler l'altitude; δ_a est utilisé pour contrôler l'axe X_B de sorte que le ventre du véhicule puisse faire face à un cap ψ_{level} ; δ_r est utilisé pour stabiliser l'axe Z_B autour de 0^0 .

4.2.1 Analyse comparative des fonctions de transfert

Une analyse comparative des fonctions de transfert est utile pour comprendre la différence entre les deux principaux modes de vol. Pour ce faire, on compare les fonctions $\frac{p(s)}{\delta_a(s)}$, $\frac{q(s)}{\delta_r(s)}$ et $\frac{r(s)}{\delta_r(s)}$. Les tableaux 4.1 et 4.2 présentent ces fonctions de transfert identifiées.

En mode non stationnaire, l'attitude ϕ et θ sont obtenues en intégrant les vitesses angulaires p, q et l'attitude ψ est obtenue en intégrant ϕ . En mode stationnaire, l'atti-

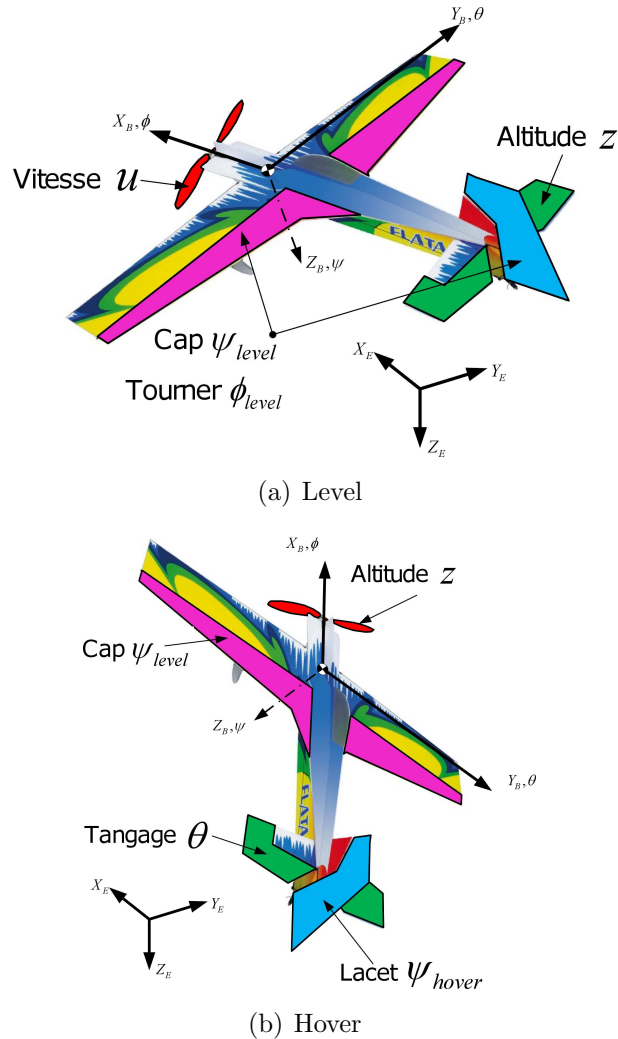


FIGURE 4.2 – Analyse comparative des variables manipulées

tude ϕ, θ, ψ est obtenue en intégrant les vitesses angulaires p, q, r . En mode de palier, le contrôle de l'angle ψ ou le cap, sert principalement à naviguer dans le plan $X_E - Y_E$. En mode stationnaire, la stabilisation de ψ est utilisée pour garder la plateforme droite dans le plan $X_E - Z_E$. La fonction de l'angle ψ change donc du tout au tout entre les deux principaux modes. C'est pourquoi on remarque que, en mode non stationnaire, p et r sont couplés, tandis qu'en en mode stationnaire, les variables p et r sont découplés.

Tout d'abord, la réponse à l'échelon de $\frac{p(s)}{\delta_a(s)}$ (figure 4.3(a)) nous montrant le comportement en roulis, par rapport à δ_a , est sensiblement le même dans les deux principaux modes. La différence peut s'expliquer par le fait qu'en mode non stationnaire, le régime du moteur (poussée T) est plus grand sur le véhicule (voir les explications sur les différents points d'opération de δ_t dans la section 4.4.1). On remarque que $\frac{q(s)}{\delta_e(s)}$ possède un pôle instable en mode stationnaire, tandis que $\frac{q(s)}{\delta_e(s)}$ possède une paire de pôles com-

TABLE 4.1 – Fonctions de transfert du mode non stationnaire

	$\delta_a[deg]$	$\delta_e[deg]$	$\delta_r[deg]$
$p[deg/s]$	$\frac{-4.61e^{-0.17s}}{(0.05s+1)}$		$\frac{0.18se^{-0.17s}}{(0.06s+1)(1.71s+1)}$
$q[deg/s]$		$\frac{-2.106(0.205s+1)}{(0.0064s^2+0.0642s+1)}$	
$r[deg/s]$	$\frac{-4.05(-0.18s+1)e^{-0.17s}}{(0.08s+1)(0.81s+1)}$		$\frac{0.38(10s+1)e^{-0.17s}}{(0.05s+1)(1.5s+1)}$

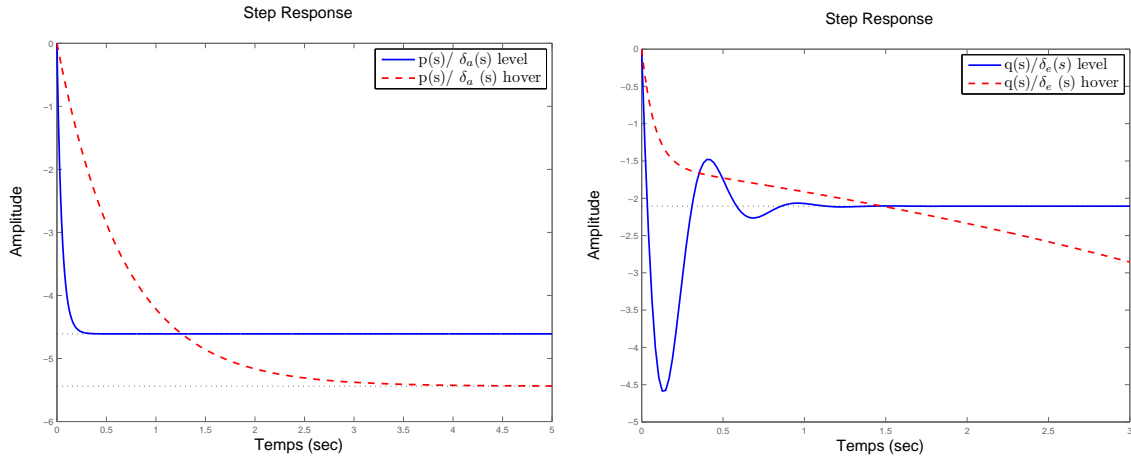
TABLE 4.2 – Fonctions de transfert du mode non stationnaire

	$\delta_a[deg]$	$\delta_e[deg]$	$\delta_r[deg]$
$p[deg/s]$	$\frac{-5.44e^{-0.17s}}{(0.67s+1)}$		
$q[deg/s]$		$\frac{-7.96se^{-0.17s}}{(5s-1)(0.08s+1)}$	
$r[deg/s]$			$\frac{2.46se^{-0.17s}}{(1.43s-1)(0.09s+1)}$

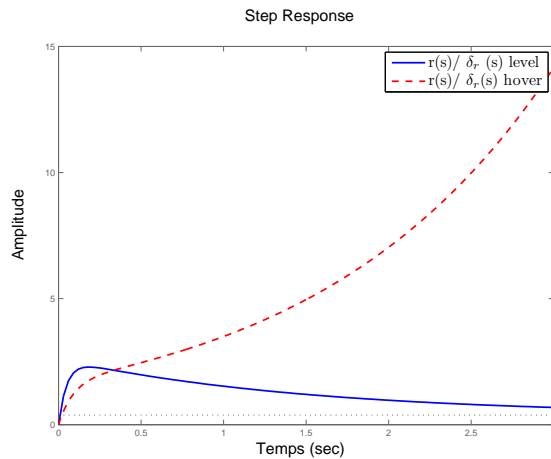
plexe stable en mode non stationnaire. Le comportement est similaire lors des premiers instants (image 4.3(b)). Finalement, on remarque que $\frac{r(s)}{\delta_r(s)}$ possède un pôle instable en mode stationnaire, tandis que $\frac{r(s)}{\delta_r(s)}$ possède deux pôles stables en mode non stationnaire. Le comportement est aussi similaire lors des premiers instants (image 4.3(c)). Malgré ces différences, on remarque que le mode roulis (*roll*) se retrouve dans les deux principaux modes de vol.

4.2.2 Utilisation des deux modes principaux pour une approche systématique

Comme il a été vu dans le chapitre 3, les systèmes linéaires peuvent être utilisés pour modéliser et contrôler un avion dans des modes stationnaire et non stationnaire idéaux. Étant donné que le comportement de la plateforme durant les transitions est extrêmement non-linéaire et que l'identification d'un modèle réaliste et complet peut être difficile, une approche systématique, basée sur la connaissance a priori de l'avion dans les deux modes principaux, a été choisie pour développer les stratégies de transition. Cette méthodologie peut aussi être utilisée pour réaliser et intégrer des stratégies de transitions sur d'autres types de plateforme.



(a) Comparaison : Réponse à l'échelon de $\frac{p(s)}{\delta_a(s)}$ (b) Comparaison : Réponse à l'échelon de $\frac{q(s)}{\delta_e(s)}$



(c) Comparaison : Réponse à l'échelon de $\frac{r(s)}{\delta_r(s)}$

4.3 Superviseur basé sur la logique des états

Le superviseur basé sur la logique s'occupe du moment de la commutation entre les contrôleurs et les différents modes. La figure 4.3 présente la machine à états du superviseur. Il y a deux principales structures de contrôleurs : le contrôleur du mode non stationnaire et le contrôleur du mode stationnaire. Par conséquent, le système de contrôle est divisé en 4 modes de vol : le mode non stationnaire (*Level*), le mode non stationnaire vers le mode stationnaire (*L2H*), le mode stationnaire (*Hover*) et le mode stationnaire vers le mode non stationnaire (*H2L*). Les modes *Level* et *L2H* sont gérés par le contrôleur du mode non stationnaire, tandis que les modes *Hover* et *H2L* sont gérés par le contrôleur du mode stationnaire. On a donc la structure suivante :

- **Initialisation** : L'initialisation est réalisée une seule fois, lorsque le système est

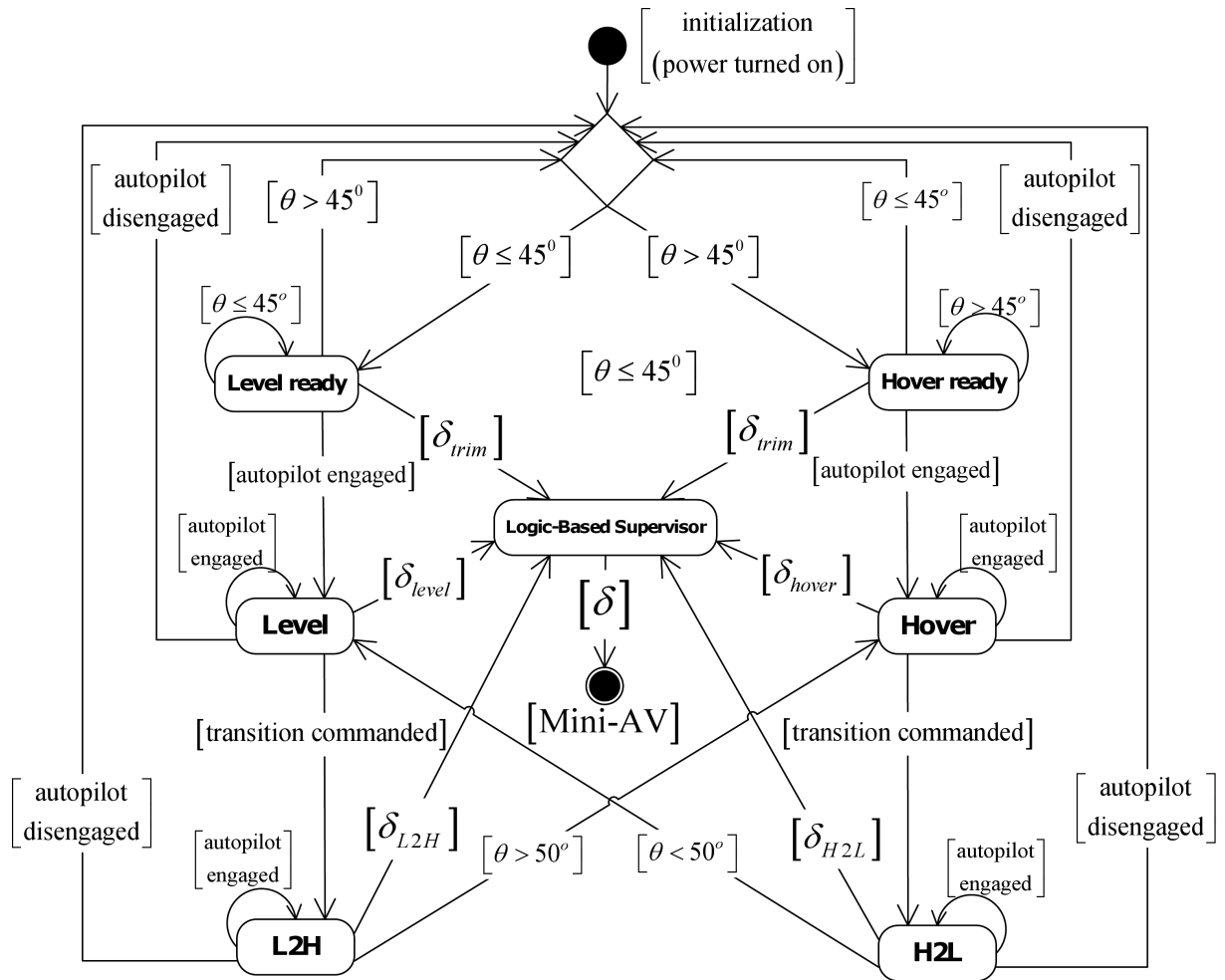


FIGURE 4.3 – Diagramme de la machine d'état du superviseur

mis sous alimentation. La valeur de tous les états est nulle (*false*) et la sortie des variables manipulées est égale aux valeurs d'équilibre. Le système bascule ensuite en attente que l'autopilote soit engagé.

- **Level ready / Hover ready** : Le système est en attente que l'autopilote soit engagé par l'utilisateur. Si $\theta \leq 45$, la plateforme attend en mode non stationnaire. Sinon, le véhicule attend en mode stationnaire. Il est aussi possible pour un pilote d'expérience de contrôler manuellement les variables manipulées pour faire des vols sans autopilote (voir section 4.3.3).
- **Level** : C'est le contrôleur du mode non stationnaire qui est actif. Il est possible de contrôler manuellement les consignes θ et ψ pour effectuer de la navigation (voir section 4.3.2).
- **L2H** : C'est la transition du mode non stationnaire vers le mode stationnaire. Ce mode est activé par l'utilisateur. Le superviseur change alors les consignes des contrôleurs du mode non stationnaire. Ce mode est utilisé jusqu'à $\theta < 50$ deg. Cette transition est entièrement autonome.
- **Hover** : C'est le contrôleur du mode stationnaire. Il est possible de contrôler manuellement les consignes des axes X_B , Y_B et Z_B pour effectuer de la navigation (voir section 4.3.2). Il est aussi possible de contrôler manuellement δ_f .
- **H2L** : C'est la transition du mode stationnaire vers le mode non stationnaire. Ce mode est activé par l'utilisateur. Le superviseur change alors les consignes des contrôleurs du mode non stationnaire. Ce mode est utilisé jusqu'à $\theta > 50$ deg. Cette transition est entièrement autonome.

Chaque contrôleur transmet au superviseur son état ainsi que la valeur des variables manipulées qu'il calcule. L'état lors de l'attente (*Level ready* ou *Hover ready*) indique au superviseur quel contrôleur celui-ci doit activer lorsque l'autopilote est engagé. Lorsque l'autopilote est désengagé, la véhicule retourne en attente. Le choix de la valeur de l'angle de commutation ($\theta = 50$ deg) est expliqué dans la section 4.4.2.

Pour illustrer le propos, la figure 4.4 présente un exemple (flèches rouges, numéros ① à ⑫). Le véhicule est positionné au sol (à l'horizontal), le système est initialisé (alimenté électriquement) et l'autopilote est désengagé ①. L'angle θ mesuré par le *IMU* étant inférieur à 45 degrés ②, le superviseur attend en mode non stationnaire et la sortie des variables manipulées est égale aux valeurs d'équilibre $\delta_{trim\ level}$ ③. L'utilisateur engage l'autopilote ④. Le superviseur active le contrôleur du mode non stationnaire

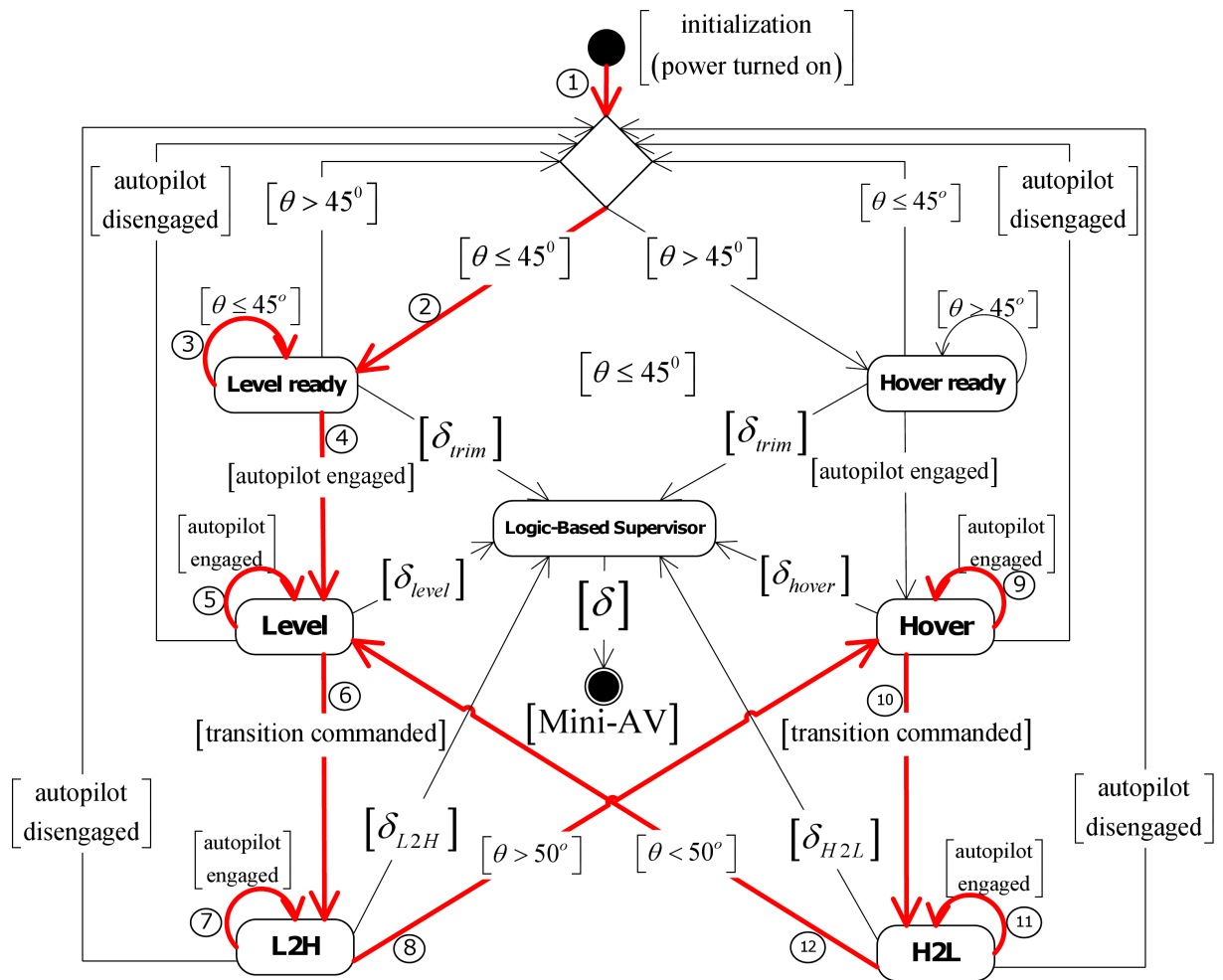


FIGURE 4.4 – Diagramme de la machine d'état du superviseur : exemple

⑤, et ce, jusqu'à ce que une autre action soit commandée. Si l'utilisateur demande une transition ⑥, le superviseur active le mode *L2H* ⑦. Le superviseur change alors pour le contrôleur du mode stationnaire lorsque l'attitude de θ est plus grande que 50 degrés ⑧. Le contrôleur en mode stationnaire est actif jusqu'à ce qu'une autre action soit commandée ⑨. Quand une transition est requise ⑩, le mode *H2L* est activé ⑪ et le superviseur commute pour le contrôleur du mode non stationnaire lorsque l'angle θ est en dessous de 50 degrés ⑫. L'algorithme 4.1 présente le pseudo-code qui a été intégré dans Matlab.

Algorithme 4.1 Machine à état du superviseur basée sur la logique

```

for  $k = 0$  to  $\infty$  do
  if  $k == 0$  then
    {Initialisation du système}
    LevelState = false
    L2HState = false
    HoverState = false
    H2LState = false
     $\delta = \delta_{trim\ level}$ 
  else
    if TrigAutopilot = false then
      {Attente (Level ou Hover) que l'autopilote soit engagé}
      if  $\theta \leq 45^0$  then
        LevelState = true
         $\delta = \delta_{trim\ level}$ 
      else
        HoverState = true
         $\delta = \delta_{trim\ hover}$ 
      end if
    else
      {Autopilote engagé}
      if LevelState == true and RequestLevel2Hover == false then
        TrigLevel=true
         $\delta = \delta_{level}$ 
      else if RequestLevel2Hover == true and  $\theta \leq 50^0$  and L2HState == true then
        TrigLevel2Hover=true
         $\delta = \delta_{L2H}$ 
      else if RequestHover2Level == false and HoverState == true then
        TrigHover=true
         $\delta = \delta_{hover}$ 
      else if ConsigneHover2Level == true and  $\theta > 50^0$  and H2LState == true then
        TrigHover2Level= true
         $\delta = \delta_{H2L}$ 
      end if
    end if
  end if
end for

```

4.3.1 Principe de la commutation sans à-coup

La figure 4.5 montre un diagramme fonctionnel du processus de la commutation d'un contrôleur. Cette figure est générique et elle sert à illustrer le principe de la commutation d'un contrôleur d'une variable de sortie y ainsi que la variable manipulée δ qui lui est associée. \bar{r} représente la consigne, ε l'erreur, G_c le contrôleur, δ_{G_c} la commande calculée par G_c , δ_{trim} la valeur d'équilibre de la variable manipulée, δ la valeur de la variable manipulée transmise au MiniAV et y est la sortie du système. Les diagrammes fonctionnels de la commutation pour les 4 modes de vols sont présentés à la section 4.3.4.

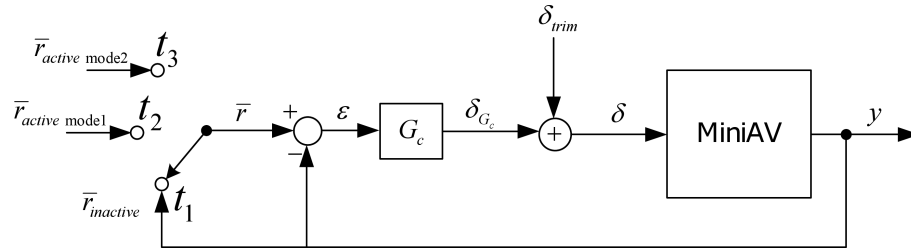


FIGURE 4.5 – Diagramme fonctionnel du processus de commutation sans à-coup

À la position t_1 , le contrôleur est inactif. La consigne est donc forcée à être égale à sa sortie ($\bar{r} = \bar{r}_{inactive} = y$). Ainsi, une erreur $\varepsilon = 0$ est transmise au régulateur G_c et une commande $\delta_{G_c} = 0$ est calculée. Il est à noter que les intégrateurs des différents contrôleurs sont désactivés lorsqu'un mode est inactif (section 3.1.4 sur les contrôleurs anti-windup). De plus, le contrôleur de la figure 4.5 reçoit en paramètre la valeur d'équilibre de la variable manipulée δ_{trim} . Les valeurs d'équilibre représentent les angles de déflexion nuls des micros-servos de δ_e, δ_a et δ_r et une valeur de repos pour δ_t . Ces valeurs sont additionnées aux valeurs calculées δ_{G_c} , car G_c effectue les calculs en ne tenant pas compte de la valeur des angles de déflexion nuls des micros-servos.

En mode non stationnaire et en mode *L2H*, les valeurs d'équilibre $\delta_{trim level}$ sont $\delta_e = 0^0$, $\delta_t = 0\%$, $\delta_a = 0^0$ et $\delta_r = 0^0$. En mode stationnaire et en mode *H2L*, les valeurs d'équilibre $\delta_{trim hover}$ sont $\delta_e = 0^0$, $\delta_t = 0\%$, $\delta_a = -40^0$ (valeur moyenne des ailerons en mode stationnaire) et $\delta_r = 0^0$. Ainsi, lorsqu'un contrôleur de mode est inactif, la sortie des variables manipulées est égale aux valeurs d'équilibre.

Quand un mode est activé, $t_1 \rightarrow t_2$ ou $t_2 \rightarrow t_3$, la consigne \bar{r} est changée à une valeur désirée (soit pour un mode de vol principal $\bar{r}_{active mode1}$ ou soit pour une transition $\bar{r}_{active mode2}$) et la nouvelle valeur calculée par G_c est additionnée à la valeur d'équilibre de la variable manipulée. Ainsi, la commutation est sans à-coup (*bumpless*).

4.3.2 Modes semi-autonomes

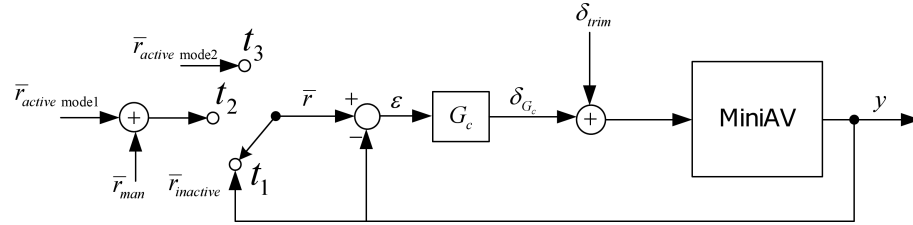


FIGURE 4.6 – Diagramme fonctionnel du contrôle manuel des consignes

Lorsque l'autopilote est engagé, il est possible pour l'utilisateur de naviguer selon les besoins de la mission. Le contrôle semi-autonome permet donc à l'utilisateur de ne pas se soucier de la stabilisation de la plateforme. En utilisant les structures de contrôleurs des deux principaux modes (figure 4.6), un utilisateur peut modifier les consignes pour naviguer. Les gains de la manette peuvent être programmés de cette façon. Par exemple, sur la figure 4.6 générique, lorsque $\bar{r}_{active\ mode1}$ est actif à t_2 , un signal externe \bar{r}_{man} provenant de la manette peut être ajouté.

En mode non stationnaire, il est possible de contrôler l'altitude avec $\bar{\theta}$ (dans l'axe Z_E) et la direction avec $\bar{\psi}$ (dans le plan $X_E - Y_E$). En mode stationnaire, on peut modifier les consignes de chaque axe pour naviguer. En modifiant les consignes des contrôleurs des axes Y_B et Z_B , il est possible de naviguer dans le plan $X_E - Y_E$. En modifiant la consigne de l'axe X_B , l'avion peut tourner sur lui-même. Les modes *L2H* et *H2L* sont entièrement autonomes.

4.3.3 Contrôle manuel des variables manipulées

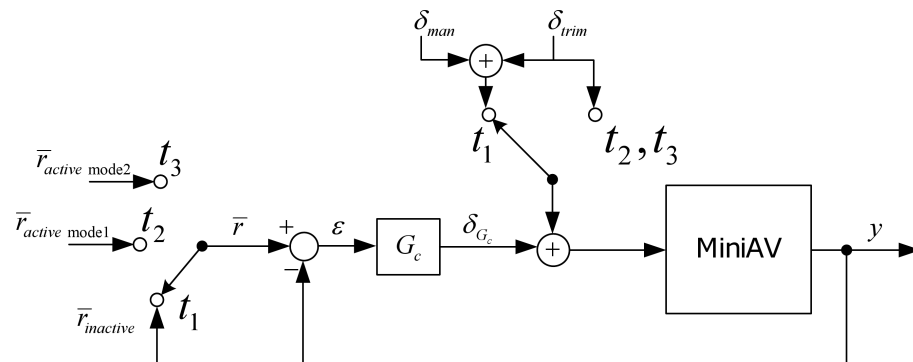


FIGURE 4.7 – Diagramme fonctionnel du contrôle manuel des variables manipulées

Lorsque la plateforme est en état d'attente *Level ready* (à t_1), un utilisateur expérimenté peut contrôler manuellement les variables manipulées pour effectuer des vols.

Les manoeuvres servant à l'identification ont été réalisées grâce à cette structure de contrôle. Dans le contrôleur *Level*, les valeurs des 4 canaux (*channel*) de la manette ainsi que celles des interrupteurs sont transmises en paramètre. Les gains et les signes des canaux sont choisis pour permettre à un pilote d'effectuer facilement des vols. La figure 4.7 montre un diagramme fonctionnel générique de cette structure.

4.3.4 Commutation des contrôleurs des différents modes de l'enveloppe de vol

Suivant les principes de la commutation entre les contrôleurs et les modes, les figures 4.8 à 4.11 illustrent cette commutation avec les contrôleurs des modes non stationnaire et stationnaire. La figure 4.12 présente le principe de la commutation intégré au diagramme de bloc du superviseur, selon la position t . Les sections 4.4 et 4.5 expliquent en détail la stratégie derrière les transitions.

La figure 4.8 illustre le contrôleur longitudinal et la figure 4.9 illustre le contrôleur latéral. Voici le fonctionnement de ces contrôleurs, selon la position t :

- t_1 : Les contrôleurs sont inactifs.
- $t_1 \rightarrow t_2$: Les contrôleurs font la commutation vers le mode actif *Level*.
- $t_2 \rightarrow t_1$: Les contrôleurs deviennent inactifs.
- t_2 : Les contrôleurs sont en mode actif *Level*.
- $t_2 \rightarrow t_3$: Les contrôleurs font la commutation vers le mode actif *L2H*.
- t_3 : Les contrôleurs sont en mode actif *L2H*.
- $t_3 \rightarrow t_1$: Les contrôleurs deviennent inactifs.

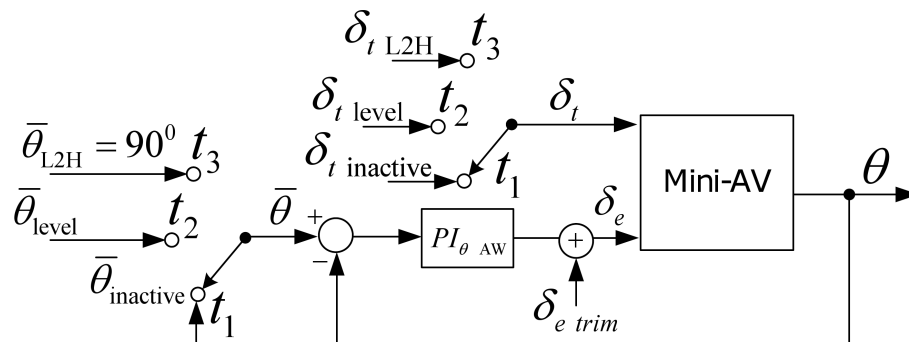


FIGURE 4.8 – Diagramme fonctionnel des différents modes de la commutation du contrôleur longitudinal

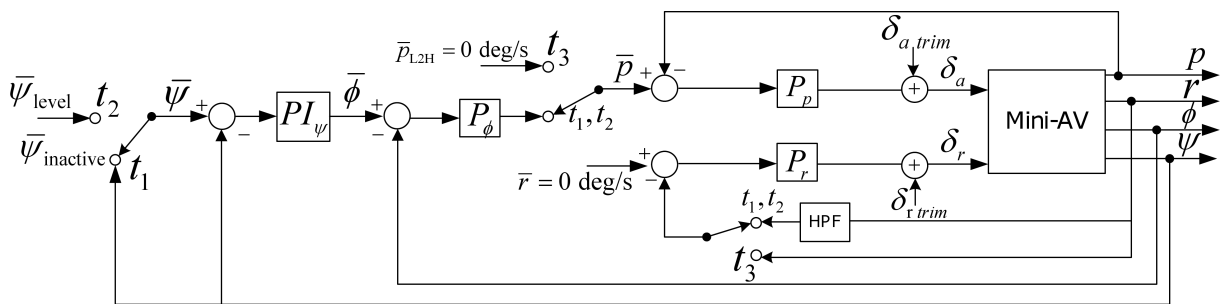


FIGURE 4.9 – Diagramme fonctionnel des différents modes de la commutation du contrôleur latéral

La figure 4.10 illustre le contrôleur du mode stationnaire (axes X_B , Y_B et Z_B) et la figure 4.11 illustre le contrôleur de l'altitude. Voici le fonctionnement de ces contrôleurs, selon la valeur de t :

- t_4 : Les contrôleurs sont inactifs.
- $t_4 \rightarrow t_5$: Les contrôleurs font la commutation vers le mode actif *Hover*.
- $t_5 \rightarrow t_4$: Les contrôleurs deviennent inactifs.
- t_5 : Les contrôleurs sont en mode actif *Hover*.
- $t_5 \rightarrow t_6$: Les contrôleurs font la commutation vers le mode actif *L2H*.
- t_6 : Les contrôleurs sont en mode actif *H2L*.
- $t_6 \rightarrow t_4$: Les contrôleurs deviennent inactifs.

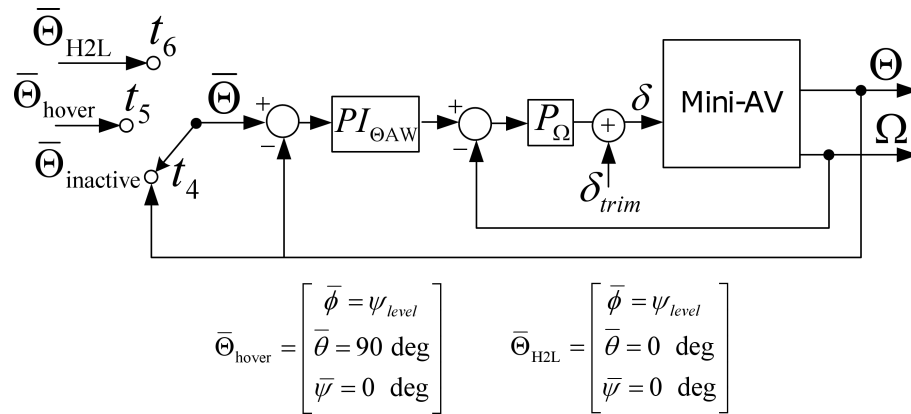


FIGURE 4.10 – Diagramme fonctionnel des différents modes et de la commutation du contrôleur du mode stationnaire

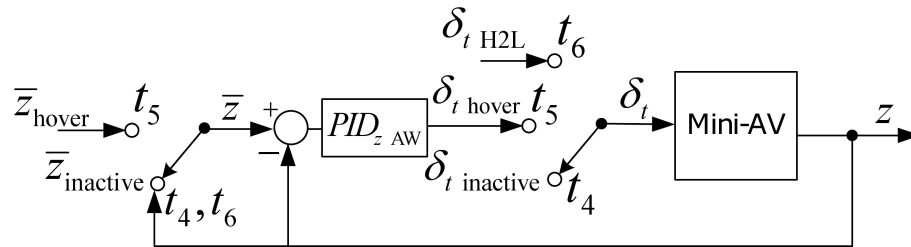


FIGURE 4.11 – Diagramme fonctionnel des différents modes de la commutation du contrôleur de l'altitude

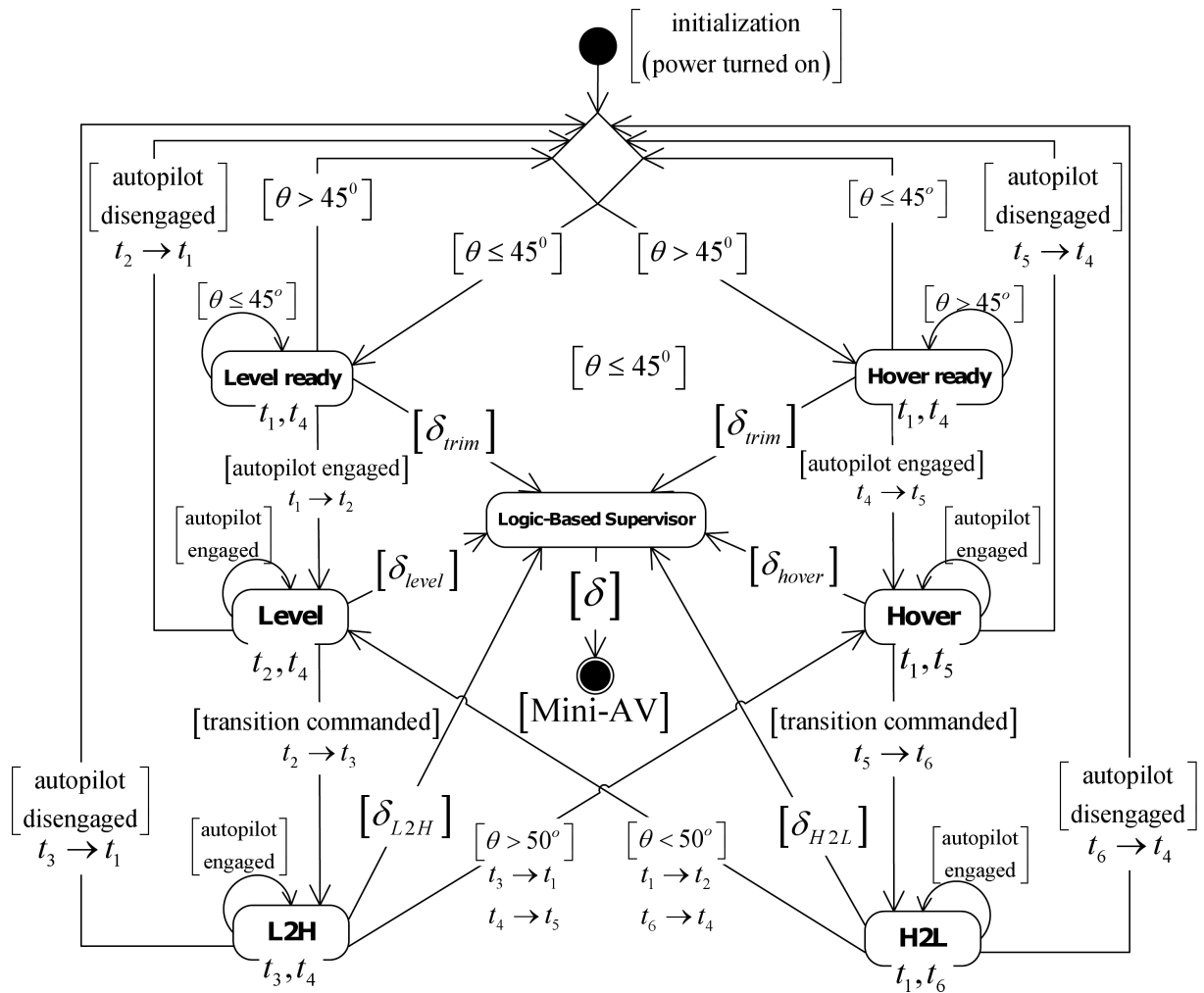


FIGURE 4.12 – Diagramme de la machine d'état du superviseur avec le principe de commutation, selon la valeur de t

4.4 Transition du vol non stationnaire vers le vol stationnaire (L2H)

L'objectif de cette transition est d'amener le plus rapidement possible la plateforme en position stationnaire. L'avion doit passer d'un angle θ_{level} à un angle $\theta_{hover} = 90^\circ$ et la vitesse en translation $u_{level} \neq 0$ doit rapidement devenir $u_{hover} = 0$. Ainsi, l'avion possède une énergie cinétique $E_{k\ level} = \frac{1}{2}mu_{level}^2$ et toute cette énergie cinétique doit être transformée en énergie potentielle ($u_{hover} = 0$). Aussi, tous les axes du mode stationnaire doivent être stabilisés le plus rapidement autour de leurs consignes. L'avion ne doit pas non plus gagner ou perdre trop d'altitude z et la transition doit être réalisée sur la plus petite distance x possible.

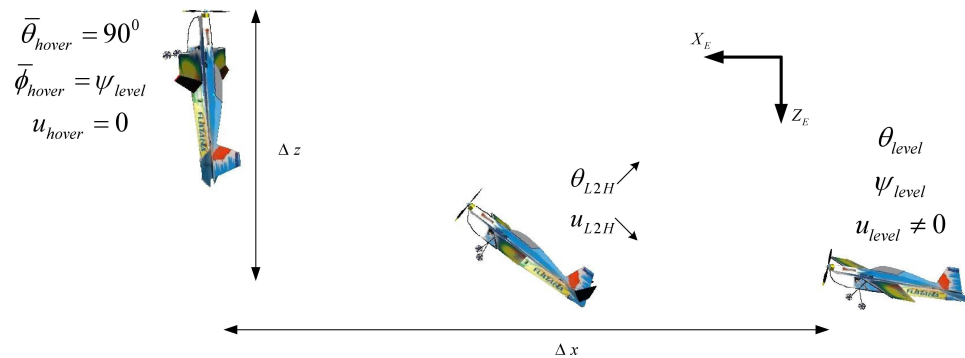


FIGURE 4.13 – Défis de la transition L2H

La connaissance a priori du comportement aérodynamique des deux principaux modes de vol du système nous permet d'établir certaines hypothèses pour effectuer la transition *L2H*. Tout d'abord, les fonctionnalités des variables manipulées du mode non stationnaire devraient rester les mêmes jusqu'au décrochage de la plateforme. Le décrochage est la perte, plus ou moins brusque, de la force de portance : l'avion s'enfonce et perd de l'altitude. Plusieurs facteurs peuvent amener l'avion en situation de décrochage dont l'angle d'attaque, la vitesse, la masse, etc. Par la suite, les contrôleurs du mode stationnaire devraient être capables de stabiliser la plateforme si les conditions initiales d'entrées en mode stationnaire respectent certaines limites d'opération. Donc, il faut amener subitement l'avion en mode stationnaire avec des conditions de telle sorte que les contrôleurs du mode stationnaire puissent rapidement stabiliser le véhicule.

En premier lieu, on utilise la même architecture et structure de contrôle que celle du mode non stationnaire. L'hypothèse est avancée que les fonctions de transfert du mode longitudinal et du mode latéral peuvent être utilisées pour approximer le comportement de l'avion jusqu'à son décrochage. Ici, on considère θ_{stall} l'angle pour lequel l'avion décroche (évaluation de l'angle de décrochage décrite à la section 4.4.1).

Considérons un MiniAV en vol nominal non stationnaire avec θ_{level} et $\delta_{t\ level}$ (figure 4.8 et figure 4.12 : t_2), en direction de ψ_{level} (figure 4.9 et figure 4.12 : t_2). Une transition est commandée par l'utilisateur. Les étapes pour réaliser la transition sont décrites dans ce qui suit (sections 4.4.1 à 4.4.4).

4.4.1 Étape 1 - Consignes

Au niveau du contrôleur longitudinal (figure 4.8 et figure 4.12 : $t_2 \rightarrow t_3$), une consigne de $\bar{\theta}_{12h}$ de 90 degrés est commandée. δ_e sature et le véhicule est lancé vers le haut en position stationnaire. Dans le contrôleur latéral (figure 4.9 et figure 4.12 : $t_2 \rightarrow t_3$), seule la première boucle cascade du contrôleur non stationnaire, avec des consignes de $\bar{p} = 0$ et $\bar{r} = 0$, est utilisée. Le filtre *HPF* n'est pas utilisé, car la consigne $\bar{r} = 0$ n'est pas seulement requise qu'aux hautes fréquences comme le vol en palier (section 3.1.4). En effet, pour réaliser une transition *L2H*, un mouvement pur en tangage est requis. On considère donc les consignes en vitesse angulaire suivantes durant la transition :

$$\bar{p} = 0 \quad (4.8)$$

$$\bar{q} > 0 \quad (4.9)$$

$$\bar{r} = 0 \quad (4.10)$$

Ainsi, on s'assure d'une réponse assez rapide et que le ventre de l'avion se dirige dans la direction ψ_{level} durant la manoeuvre. Ce choix de structure de contrôleur latéral est suffisant pour permettre à la plateforme de garder son cap pendant la courte période que la plateforme est dans ce mode.

δ_t est spécifiée en boucle ouverte selon une planification de l'angle θ . Ainsi, on permet à la plateforme d'augmenter rapidement son angle de tangage sans gagner trop d'altitude. On augmente δ_t de 10% par rapport à son point d'opération en mode de palier jusqu'à ce que l'avion atteigne un angle $\theta = 25^0$ et on le diminue graduellement par la suite pour atteindre un régime d'opération inférieur (-10 % du régime en mode non stationnaire). De ce fait, on accélère rapidement l'angle θ sur une courte période et, vers la fin de la séquence planifiée, δ_t est diminué plus bas que le point d'opération δ_t du mode stationnaire : on permet à la force de gravité de contribuer à ralentir l'ascension de la plateforme. Il faut cependant s'assurer qu'il y ait un flux d'air assez important pour que δ_a et δ_r puissent être effective sur p et r . Cette séquence δ_t planifiée est trouvée expérimentalement. L'algorithme 4.2 présente le pseudo-code.

Voici une comparaison de δ_t , selon le mode d'opération choisi :

$$\delta_{t\ level} > \delta_{t\ hover} \quad (4.11)$$

Algorithme 4.2 Séquence planifiée de δ_t L2H

```

for  $k=0$  to  $\infty$  do
  {À chaque itération  $k$  (période d'échantillonnage  $T_s = 0.01$ ), lorsque le mode L2H
  est activé}
  if  $\theta(k) < 25^0$  then
     $\delta_t(k) \leftarrow \delta_{t \text{ level}} + 10\%$ 
  else if  $\theta(k) \geq 25^0$  and  $\theta(k) < 40^0$  then
     $\delta_t(k) \leftarrow \delta_{t \text{ level}} + 10\% - \left(\frac{\theta(k)-25^0}{15^0}\right) 20\%$  {Diminuer  $\delta_t$ , de façon planifiée, jusqu'à
    ce l'avion atteigne un angle  $\theta(k) = 40^0$ }
  else
     $\delta_t(k) \leftarrow \delta_{t \text{ level}} - 10\%$  {Ce point d'opération est inférieur à  $\delta_{t \text{ hover}}$ }
  end if
end for

```

4.4.2 Étape 2 - Commutation du contrôleur *hover*

Le contrôleur en mode stationnaire est commuté 15 degrés avant l'angle de décrochage, c'est-à-dire à un angle de $\theta = 50^0$ (figure 4.10 et figure 4.12 : $t_4 \rightarrow t_5$; figure 4.8, figure 4.9 et figure 4.12 : $t_3 \rightarrow t_1$). Ainsi, on s'assure d'avoir une marge de sécurité suffisante pour les cas où l'angle de décrochage serait inférieur (plateforme abîmée ou pile faible). Lors de la transition, le cap ψ_{level} vers laquelle la plateforme se dirigeait est transmise par le superviseur au contrôleur du mode stationnaire. Les consignes des axes sont $\bar{\theta} = 90$ deg, $\bar{\phi} = \psi_{level}$ et $\bar{\psi} = 0$ deg. Ainsi, tel qu'illustré à la section 2.3.2, le quaternion de consigne en mode stationnaire $q_{c_{hover}}$ devient :

$$q_{0c_{hover}} = (\cos(-\psi_{level}/2) \cos(90/2) + \sin(-\psi_{level}/2) \sin(90/2)) \quad (4.12)$$

$$q_{1c_{hover}} = (\sin(-\psi_{level}/2) \cos(90/2) - \cos(-\psi_{level}/2) \sin(90/2)) i \quad (4.13)$$

$$q_{2c_{hover}} = (\cos(-\psi_{level}/2) \sin(90/2) + \sin(-\psi_{level}/2) \cos(90/2)) j \quad (4.14)$$

$$q_{3c_{hover}} = (\cos(-\psi_{level}/2) \cos(90/2) - \sin(-\psi_{level}/2) \sin(90/2)) k \quad (4.15)$$

La figure 3.36 présente un diagramme fonctionnel du calcul des erreurs angulaires en mode stationnaire.

4.4.3 Étape 3 - Contrer l'élan vers l'avant

Lors de l'expérimentation, il a été observé que l'élan engendré par la saturation de δ_e et la séquence de δ_t et crée un élan vers l'avant (x positif). Il faut donc stopper cet

élan pour réduire la distance x sur laquelle est effectuée la transition. En effet, on peut expliquer ce comportement avec l'équation 4.6 :

$$\ddot{X}_E = \frac{-\rho V^2 S}{2m} (C_L(\alpha) \sin(\theta - \alpha) + C_D(\alpha) \cos(\theta - \alpha)) + \frac{T}{m} \cos \theta$$

Si on néglige le terme de gauche (les termes $V, C_L(\alpha)$ et $C_D(\alpha)$ sont petits à cet instant), on a :

$$\ddot{X}_E \approx \frac{T}{m} \cos \theta \quad (4.16)$$

On remarque que lors de l'entrée en mode stationnaire (à 50°), on a $\frac{T}{m} \cos(\theta = 50^\circ) = \frac{T}{m} 0.643$. Ainsi, δ_r propulse encore l'avion vers l'avant (x positif). C'est pourquoi il faut amener le plus rapidement possible l'angle θ autour de sa consigne de 90 deg. Ainsi, durant les premiers moments en mode stationnaire, et ce, jusqu'à 75 deg, une consigne de 30 deg, vers la direction x négatif, dans le contrôleur de l'axe Y_B est commandée pour augmenter rapidement θ et ainsi diminuer la distance parcourue vers l'avant.

Dès que l'erreur en tangage est de moins de 15 degrés par rapport à sa consigne en tangage 90 deg et que la plateforme a dépassé sa consigne ou atteint une altitude maximum, la boucle de contrôle de la variable manipulée δ_r (altitude en boucle fermée ou en boucle ouverte) est activée (figure 4.11 et figure 4.12 : $t_4 \rightarrow t_5$). Dans le cas du contrôle de l'altitude en boucle fermée, la consigne en altitude est choisie 50 cm au dessus de la position de la plateforme lors de l'activation du PID. L'algorithme 4.3 présente cette séquence.

4.4.4 Étape 4 - Contrôle de l'altitude

Une des difficultés de cette transition est de contrôler l'altitude autour d'une consigne. Les transitions doivent être effectuées dans un espace restreint : le véhicule ne peut pas gagner ou perdre trop d'altitude par rapport à son altitude en vol de palier. Il faut donc éviter une grande variation de flux d'air de δ_r . Plus la variation de δ_r est grande, plus le contrôle de l'altitude sera difficile. Idéalement, il faut que le contrôle de l'altitude se réalise en boucle fermée. La connaissance de l'altitude, avec certaine précision, est donc requise.

Trois méthodes pour contrôler l'altitude ont été testées et étudiées. La première méthode est un contrôle en boucle ouverte de l'altitude : quand le mode stationnaire est activé, un utilisateur expérimenté peut contrôler manuellement δ_r . La seconde méthode est un contrôle en boucle fermée qui utilise un capteur pour connaître l'altitude, comme

Algorithme 4.3 Premiers instants en mode stationnaire

```

for  $k=0$  to  $\infty$  do
  {À chaque itération  $k$  (période d'échantillonnage  $T_s = 0.01$ )}
  if  $\theta(k) < 75^0$  and Condition1 == false then
    {Pour réduire la distance  $x$ , jusqu'à ce que  $\theta > 75^0$ }
     $\bar{\theta}_{hover}(k) \leftarrow$  Consigne  $\theta \leftarrow 30^0$  vers  $x$  négatif
     $\delta_t(k) \leftarrow \delta_t_{level} - 10\%$ 
  else
     $\bar{\theta}_{hover}(k) \leftarrow$  Consigne  $\theta \leftarrow 90^0$ 
    if Altitude boucle ouverte == true then
      {L'utilisateur contrôle manuellement  $\delta_t$ }
       $\delta_t(k) \leftarrow \delta_{tman}(k)$ 
      Condition1  $\leftarrow$  true
    else
      if Condition1 == false then
         $\bar{z} \leftarrow z(k) - 50$  cm
        Condition1  $\leftarrow$  true
        {Lorsque  $\theta(k) > 75$ , on choisit la consigne  $\bar{z}$  comme étant  $z(k)$ }
      else if Condition1 == true or Condition2 == true or  $z(k) < z(\bar{k})$  or  $z(k) >$ 
 $z(k-1)$  then
         $\delta_t(k) \leftarrow PID_z_{active}$ 
        Condition2  $\leftarrow$  true
      else
         $\delta_t(k) \leftarrow \delta_t_{level} - 10\%$ 
      end if
    end if
  end if
end for

```

un système de vision numérique. Malheureusement, un système de vision ne peut pas être utilisé en situation réelle et d'autres capteurs, comme les capteur ultrasoniques, les capteurs d'altitude barométrique ou le *GPS*, ne sont pas assez précis pour réaliser ce genre de manoeuvre. La troisième méthode est aussi une stabilisation en boucle fermée, mais l'altitude est estimée en utilisant une technique de navigation inertielle de type plateforme *strap-down inertial navigation system (INS)*. La section qui suit détaille le principe derrière cette méthode.

Navigation inertielle

L'altitude est estimée en intégrant numériquement les mesures d'accéléromètres (a_x, a_y, a_z) , les vitesses angulaires (p, q, r) et les positions angulaires (ϕ, θ, ψ) provenant de l'IMU. On utilise les équations de translation pour trouver la vitesse de la plateforme dans son repère (u, v, w) et ensuite les équations de navigation pour trouver la position dans le référentiel NED (X_E, Y_E, Z_E) . On assume que le *IMU* est placé au *CG* et qu'il n'est pas mécaniquement déplacé du *CG* de la plateforme durant les vols.

Dans les équations de forces développées au chapitre 2, les forces aérodynamiques, qui sont appliquées en unité de g , peuvent être remplacées par les mesures d'accélération qui sont mesurées par le capteur IMU (que l'on considère placé au *CG*). En effet, les accélérations mesurées sont équivalentes à :

$$a_x = \frac{f_x}{mg} \quad (4.17)$$

$$a_y = \frac{f_y}{mg} \quad (4.18)$$

$$a_z = \frac{f_z}{mg} \quad (4.19)$$

Ainsi, on peut réécrire les équations 2.101 à 2.103 :

$$\dot{u} = rv - qw - g \sin \theta + ga_x \quad (4.20)$$

$$\dot{v} = -ru + pw - g \cos \theta \sin \phi + ga_y \quad (4.21)$$

$$\dot{w} = qu - pv - g \cos \theta \cos \phi + ga_z \quad (4.22)$$

En utilisant cette forme, les équations utilisent uniquement des quantités de mouvement et on les résout pour générer des valeurs de sorties reconstruites u, v, w . Par la suite, les équations de navigation 2.118 à 2.120 sont utilisées pour trouver les positions x, y, z .

Pour solutionner les équations différentielles, on utilise la méthode d'intégration numérique classique de Runge-Kutta d'ordre quatre (rk4). Cette méthode effectue l'approximation des solutions d'équations différentielles suivante :

$$\dot{x} = f(x(k), u(k)) \quad (4.23)$$

où $x(k)$ qui est le vecteur d'état du système et $u(k)$ qui est le vecteur d'entrée. Pour utiliser *en ligne* cette technique d'intégration numérique, on utilise $x(k-1), u(k-1), u(k)$ pour trouver la valeur de $x(k)$. Ainsi, pour l'itération k avec $\Delta_t = 0.01$ comme période

Algorithme 4.4 Intégration numérique : navigation inertielle

```

for  $k = 0$  to  $\infty$  do
  {À chaque itération  $k$  (période d'échantillonnage  $T_s = 0.01$ )}
  if  $k == 0$  then
     $[u(k) \ v(k) \ w(k)]^T \leftarrow [u_0 \ v_0 \ w_0]^T$  {Vitesses initiales}
     $[X_E(k) \ Y_E(k) \ Z_E(k)]^T \leftarrow [X_{E0} \ Y_{E0} \ Z_{E0}]^T$  {Positions initiales}
  else
     $[u(k) \ v(k) \ w(k)]^T \leftarrow rk4(x_{vit}(k-1), u_{vit}(k-1), u_{vit}(k))$ 
     $[X_E(k) \ Y_E(k) \ Z_E(k)]^T \leftarrow rk4(x_{pos}(k-1), u_{pos}(k-1), u_{pos}(k))$ 
  end if
end for

```

d'échantillonnage, on trouve $x(k)$ en effectuant les opérations suivantes :

$$x_{d1} = f(x(k-1), u(k-1)) \quad (4.24)$$

$$x_{int1} = x(k-1) + \frac{x_{d1}}{2} \Delta_t \quad (4.25)$$

$$u_{int1} = \frac{u(k-1) + u(k)}{2} \quad (4.26)$$

$$x_{d2} = f(x_{int1}, u_{int1}) \quad (4.27)$$

$$x_{int2} = x(k-1) + \frac{x_{d2}}{2} \Delta_t \quad (4.28)$$

$$x_{d3} = f(x_{int2}, u_{int1}) \quad (4.29)$$

$$x_{int3} = x(k-1) + \frac{x_{d3}}{2} \Delta_t \quad (4.30)$$

$$u_{int3} = u(k) \quad (4.31)$$

$$x_{d4} = f(x_{int3}, u_{int3}) \quad (4.32)$$

$$x(k) = x(k-1) + \frac{x_{d1} + 2x_{d2} + 2x_{d3} + x_{d4}}{6} \Delta_t \quad (4.33)$$

Pour les équations de force (vitesse), on a le vecteur d'état $x_{vit}(k) = [u(k) \ v(k) \ w(k)]^T$ et le vecteur d'entrée $u_{vit}(k) = [p(k) \ q(k) \ r(k) \ \phi(k) \ \theta(k) \ \psi(k) \ a_x(k) \ a_y(k) \ a_z(k)]^T$. Pour les équations de navigation, on a le vecteur d'état $x_{pos}(k) = [X_E(k) \ Y_E(k) \ Z_E(k)]^T$ et le vecteur d'entrée $u_{pos}(k) = [u(k) \ v(k) \ w(k) \ \phi(k) \ \theta(k) \ \psi(k)]^T$. L'algorithme 4.4 présente le calcul qui est effectué à chaque itération k (période d'échantillonnage $T_s = 0.01$) pour calculer les vitesses et les positions.

Malheureusement, comme tout capteur, le *IMU* n'est pas à l'abri des erreurs d'instrumentation. Tout d'abord, les biais et les dérives sont les erreurs qui causent les plus grands effets. Les données du *IMU* sont aussi sensibles aux changements de température et aux vibrations. Un modèle d'équation de mesure d'un capteur d'avion générant une

erreur systématique d'instrumentation est [22] :

$$z(k) = (1 + \lambda(k))y(k) + b(k) + v(k) \quad (4.34)$$

ou $z(k)$ est la k^{eme} données de sortie mesurée, $y(k)$ est la vraie sortie, $v(k)$ est un bruit de mesure aléatoire, $\lambda(k)$ est une erreur de dérive et $b(k)$ est une erreur de biais. Si les paramètres d'erreurs d'instrumentation $b(k)$ et $\lambda(k)$ sont nuls, les données de mesures seront égales aux vraies sorties, plus un bruit de mesure aléatoire ajouté. Ces hypothèses d'erreurs d'instrumentation sont suffisantes pour expliquer la plupart des erreurs. En pratique, ce ne sont pas tous les capteurs qui possède une erreur de dérive et/ou de biais [29]. Typiquement, les accélérations de translation et les vitesses angulaires possèdent des erreurs de biais et de dérive. Ceux-ci changent à chaque fois que le *IMU* est alimenté électriquement et ils varient dans le temps. Par exemple, un biais sur les données d'accéléromètres du *IMU* produit une erreur quadratique sur la position calculée. Les erreurs de la navigation inertielle *INS* ont typiquement des dynamiques lentes et des modèles complexes ont été développés [10].

Habituellement, avant de les utiliser dans le processus d'intégration numérique, les données de mesure sont filtrées avec un filtre passe-bas dans le but de diminuer le bruit de haute-fréquence. Quand du bruit aléatoire est filtré, cela produit une marche aléatoire. L'intégration de cette marche aléatoire résulte en des vitesses et des positions calculées qui peuvent varier à différents taux durant un essai, et ce, même si le *IMU* bouge dans la même orientation et subit les mêmes accélérations à chaque essai.

La précision se détériore avec le temps si le système est laissé à lui-même et qu'aucune mesure externe n'est accessible. Ainsi, en utilisant des capteurs externes pouvant fournir des mises-à-jour sur les vitesses ou les positions, comme un GPS ou un capteur barométrique, les erreurs systématiques peuvent être éliminées. Des filtres de Kalman peuvent être intégrés si une solution d'estimation de l'altitude à long terme est requise [20, 25, 39, 40]. La méthode *INS* donne cependant une grande précision pour les premières secondes. À court terme, on considère alors que les biais et les dévires sont négligeables pour réaliser ce type de manoeuvre.

4.5 Transition du vol stationnaire vers le vol non stationnaire (*H2L*)

L'objectif de cette transition est d'amener le plus rapidement possible la plateforme en position horizontale. L'avion doit passer d'un angle $\theta_{hover} = 90^0$ à un angle $\theta_{level} = 18^0$

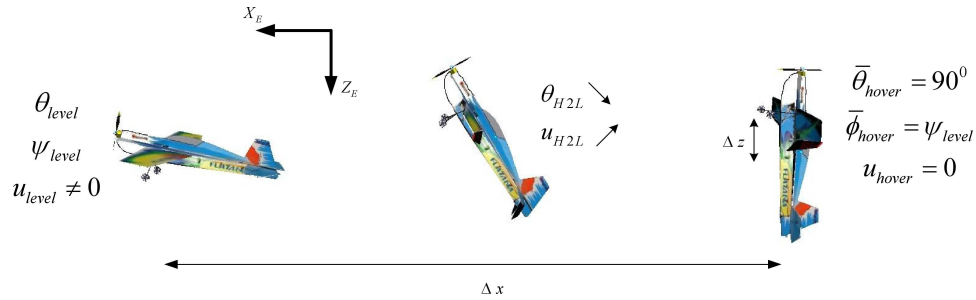


FIGURE 4.14 – Défis de la transition H2L

et la vitesse en translation $u_{hover} = 0$ doit rapidement devenir $u_{level} \neq 0$. Aussi, le cap de l'avion doit être contrôlé le plus rapidement autour de sa consigne. L'avion ne doit pas non plus gagner ou perdre trop d'altitude z et la transition doit être réalisée sur la plus petite distance x possible.

Encore une fois, la connaissance a priori du comportement aérodynamique des deux principaux modes de vol du système nous permet d'établir certaines hypothèses pour effectuer la transition *H2L*. Tout d'abord, les fonctionnalités des variables manipulées du mode stationnaire devraient rester les mêmes jusqu'à l'angle de décrochage θ_{stall} . Comme l'axe Y_B est instable en mode stationnaire, la manoeuvre vers l'avant déstabilise le système. Par la suite, les contrôleurs du mode non stationnaire devraient être capables de stabiliser la plateforme si les conditions initiales d'entrées respectent une certaine limite. Donc, il faut amener subitement l'avion en mode non stationnaire avec des conditions de telle sorte que les contrôleurs du mode non stationnaire puissent rapidement contrôler la plateforme.

En premier lieu, on utilise la même architecture et structure de contrôle que celle du mode stationnaire. L'hypothèse est avancée que les fonctions de transfert du mode longitudinal et du mode lateral peuvent être utilisées pour approximer le comportement de l'avion jusqu'à θ_{stall} .

Considérons un MiniAV en vol nominal stationnaire avec δ_t_{hover} (figure 4.10 et figure 4.12 : t_5), le ventre de l'avion en direction de ψ_{level} (figure 4.11 et figure 4.12 : t_5). Une transition est commandée par l'utilisateur. Les étapes pour réaliser la transition sont décrites dans ce qui suit (sections 4.5.1 à 4.5.2).

4.5.1 Étape 1 - Consignes

Dans le contrôleur de l'axe Y_B , $\bar{\theta}$ est mis à 0 deg. δ_e sature pour lancer vers l'avant l'avion en vol de palier. Les contrôleurs des axes X_B et Z_B gardent les mêmes consignes, soit $\bar{\phi} = \psi_{level}$ et $\bar{\theta} = 0$ deg. δ_t est augmenté de 10% par rapport à son point d'opération en mode stationnaire pour éviter la perte d'altitude durant la manoeuvre.

4.5.2 Étape 2 - Commutation du contrôleur *level*

Le contrôleur du mode non stationnaire est commuté au même angle i.e. vers un angle de $\theta = 50^0$ (figure 4.10 et figure 4.12 : $t_4 \rightarrow t_5$; figure 4.8, figure 4.9 et figure 4.12 : $t_3 \rightarrow t_1$). Lors de la transition, le cap ψ_{level} vers laquelle le ventre de la plateforme pointait est transmise par le superviseur au contrôleur du mode non stationnaire.

δ_t est spécifiée en boucle ouverte selon une planification de l'angle θ . Jusqu'à ce que l'angle de tangage de la plateforme rejoigne sa consigne θ_{level} , δ_t est graduellement diminué jusqu'à son point d'opération $\delta_{t\ level}$. L'algorithme 4.5 présente cette séquence.

Algorithme 4.5 Séquence planifiée de δ_t du mode H2L vers le mode LEVEL

```

for  $k=0$  to  $\infty$  do
  {À chaque itération  $k$  (période d'échantillonnage  $T_s = 0.01$ )}
  if  $\theta(k) > 50^0$  and Condition == false then
     $\delta_t(k) \leftarrow \delta_{t\ hover} + 10\%$ 
  else if  $\theta(k) > 18^0$  and Condition == false then
     $\delta_t(k) \leftarrow \delta_{t\ hover} + 10\% - \left( \frac{\theta - 18^0}{50^0 - 18^0} \right) (\delta_{t\ hover} + 10\% - \delta_{t\ level_{op}})$ 
    {Diminuer graduellement de  $\delta_{t\ hover} + 10\%$  vers  $\delta_{t\ level}$  jusqu'à  $\theta(k) = 18^0$ }
  else
     $\delta_t(k) \leftarrow \delta_{t\ level}$ 
    Condition  $\leftarrow$  true
    {Dès  $\theta(k)$  devient  $< 18^0$ ,  $\delta_t(k)$  doit rester à  $\delta_{t\ level}$ }
  end if
end for

```

Lors de l'expérimentation, il a été observé que l'élan engendré par la saturation de δ_e et la séquence de δ_t crée un moment vers le bas (vers les z positif). Ainsi, pour ralentir ce moment, on maintient $\bar{\theta} = 50$ degrés jusqu'à ce que l'angle de tangage rejoigne un angle $\theta = 18$ degrés. Par la suite, la consigne en mode stationnaire devient $\bar{\theta} = 18$ degrés.

4.6 Conclusion

Dans ce chapitre, les différentes stratégies de transition qui ont été conçues pour réaliser les manoeuvres de transition ont été décrites. En premier lieu, une analyse des équations de forces aérodynamiques des variables longitudinales et une analyse comparative des fonctions de transfert des deux principaux modes de vol ont été effectuées. Une approche systématique, basée sur la connaissance a priori des comportements de la plateforme dans les deux principaux modes de vol, est choisie pour développer le superviseur basé sur la logique. Une structure de commutation simple entre les contrôleurs est proposée. Enfin, les différentes étapes nécessaires aux manoeuvres de transitions L2H et H2L sont décrites. Cette approche systématique permet donc d'éviter d'avoir à identifier un modèle non-linéaire complet et de concevoir une loi de commande non-linéaire pour gérer l'ensemble de l'enveloppe de vol. Les principaux types d'essais qui ont été réalisés pour tester les algorithmes de contrôle, la logique du superviseur et la commutation entre les contrôleurs sont présentés dans le chapitre suivant.

Chapitre 5

Résultats expérimentaux en vol

Ce chapitre présente plusieurs essais de vol qui ont été réalisés. Tout d'abord, le banc d'essais (équipement, dimension...) , situé dans un local du RDDC Valcartier, est décrit. La plateforme utilisée pour réaliser les essais, le Flatana modifié, est présentée et la structure des algorithmes sous Matlab-Simulink est résumée. Par la suite, 9 types d'essais sont présentés dans les sections 5.3, 5.4 et 5.5. Les résultats sont comparés avec les performances de plateforme similaires rencontrées dans la littérature. Certains essais peuvent aussi être visionnés sur le CD-ROM inclus avec ce mémoire.

5.1 Description du banc d'essais

Les essais expérimentaux de vol ont lieu dans un local du RDDC Valcartier. Le véhicule peut décoller en mode non stationnaire d'une rampe de décollage (figures 5.1(a) et 5.1(b)). Cette rampe, muni de roues, a été construite avec la collaboration du personnel de RDDC Valcartier. La rampe a une longueur de 1.5 m, une largeur de 60 cm et elle est positionnée à une hauteur de 1.5 m. Le véhicule peut franchir une dizaine de mètres en vol de palier avant d'être arrêté par un filet de sécurité (figure 5.2(b)). Il y a environ 3 mètres entre le filet de sécurité et le plafond (figure 5.2(a)).

Les 15 caméras du système de vision sont placées sur le plafond et un volume de capture d'environ 3-m x 2-m x 2-m est généré au dessus du filet La figure B.1(a) de l'annexe B présente la disposition des caméras avec le corps de l'avion reconnu par le système de vision. La figure B.1(b) de l'annexe B présente, quant à elle, le volume qui est généré par le système de caméra.



(a) Vue de côté



(b) Vue de face avec caméra de vision numérique au plafond

FIGURE 5.1 – Rampe pour le décollage



(a) Vue de côté



(b) Vue d'en dessous de la rampe de décollage

FIGURE 5.2 – Banc d'essais

5.1.1 Flatana

Les figures 5.3(a) et 5.3(b) montrent la plateforme qui est utilisée pour réaliser les essais. Quelques modifications au véhicule original ont cependant dû être apportées afin de pouvoir réaliser à répétition les essais en vol de transition.

1. Pour que le système de vision puisse reconnaître l'avion comme étant un corps rigide dans son volume de capture, de petites boules luminescentes ont été collées sur le corps de l'avion (figures 5.3(a) et 5.3(b)). Les boules ont été disposées de façon asymétrique sur le corps pour permettre au système de vision de reconnaître l'orientation du véhicule dans son repère.
2. L'avion étant en carton mousse rigide, des tiges de renforcement ont été rajoutées sur l'axe longitudinal de l'avion (axe X_B). Il a été observé qu'avec les nombreuses chutes dans le filet, l'ajout des tiges permet d'amortir les chocs et d'augmenter considérablement la vie utile de la plateforme. En effet, le principal moyen de terminer un essai est de laisser tomber le Flatana dans le filet de sécurité.
3. Pour permettre de réaliser les essais en mode de palier, il faut que le MiniAV puisse décoller en ligne droite en tout temps de la rampe de lancement. Ainsi, un train de décollage, muni d'une tige au centre, a été fabriqué. Une tige a aussi été fixée à l'arrière du MiniAV (figures 5.4(a) et 5.4(b)). Une rainure au centre de la rampe permet donc au MiniAV de décoller en ligne droite.
4. L'orientation du moteur électrique (manette des gaz, δ_t) sur le nez de l'avion est critique. En effet, celui-ci doit être parallèle le plus possible avec le plan $X_B - Z_B$. Un mauvais alignement du moteur peut rendre les manoeuvres de transition L2H et la navigation en mode stationnaire plus difficiles à réaliser. Ainsi, le moteur a dû être posé sur une plaquette de bois supplémentaire et des tiges de carbone ont été rajoutées pour amortir les chocs.
5. Il est très important que le poids de l'avion, avec tout l'équipement ajouté, soit bien réparti au centre de gravité naturel de l'avion. Ainsi, les batteries ont été disposées de façon à équilibrer le poids.

Quelques avions ont été fabriqués et utilisés pour réaliser les tests. Toutes ces modifications ont été réalisées grâce à un processus itératif. Malgré toutes ces modifications, les modèles développés de l'avion sont demeurés fiables pour réaliser les tests.

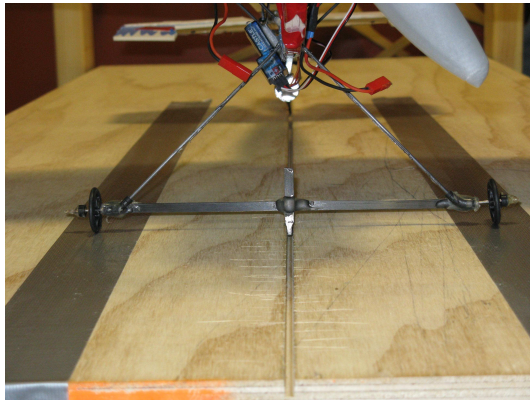


(a) Vue de côté

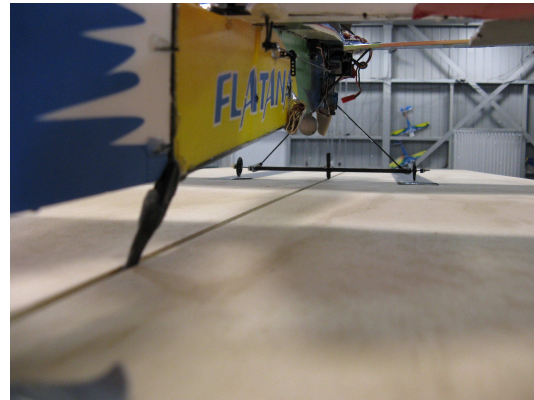


(b) Vue de haut

FIGURE 5.3 – Flatana



(a) Train pour le décollage - Vue de devant



(b) Train pour le décollage - Vue de derrière



(c) Renforcement du nez

FIGURE 5.4 – Exemples de modifications apportées au Flatana

5.2 Description de la structure des algorithmes sous xPC-Target

Les détails du montage du système de contrôle et de communication entre les différents éléments sont présentés dans [3]. En résumé, un ordinateur (un Pentium II 266 MHz avec 256 Mo de mémoire vive) fait fonctionner un kernel Xpc Target 3.2 de Mathworks en temps réel. Il est donc possible de compiler un modèle Simulink (qui contient les algorithmes de contrôle) sur cet ordinateur. Tous les calculs des variables manipulées et la gestion des contrôleurs sont alors faits sous l'environnement Xpc Target-Simulink. L'ordinateur communique numériquement avec une télécommande R/C à l'aide d'un signal PPM afin d'envoyer les commandes à l'avion et une carte d'acquisition est utilisée pour l'acquisition des données de vol. Les données de mesure provenant de l'IMU sont décodées par une station de base sans-fils qui communique avec l'ordinateur avec le protocole RS232. La manette USB et le système de vision communiquent avec l'ordinateur via un protocole UDP.

L'utilisation de l'interface Xpc Target permet d'effectuer tous les calculs des algorithmes sur un ordinateur qui n'est pas à bord de l'avion. L'avantage de cette interface est donc de pouvoir développer et de tester rapidement des algorithmes de contrôle. Son interface est de plus très intuitive à utiliser. La structure de base de ces algorithmes sous Matlab/Simulink est décrite dans [3]. Dans le cadre de ce projet sur les vols de transition, la structure de base a été augmentée afin d'inclure l'enveloppe de vol complète, c'est-à-dire les 4 modes de vol. La première augmentation a été d'ajouter, à la structure de contrôle du mode stationnaire déjà développé par Bilodeau et al. [2, 3], une structure de contrôle pour le mode non stationnaire. Par la suite, un superviseur a été développé pour réaliser la commutation, selon le plan de vol, entre les contrôleurs.

Chaque structure de contrôle reçoit les données de mesures (IMU, système vision, positions et vitesses reconstruites), les valeurs des différents canaux de la manette USB, les valeurs d'équilibre des variables manipulées, les signaux booléens d'activation de mode et les consignes en cap et tangage. Les sorties des structures de contrôle sont les valeurs des variables manipulées calculées, les états du système et les valeurs des différentes consignes. La figure 5.6 présente le modèle Simulink utilisé avec les deux structures de contrôle ainsi que le superviseur.

Le superviseur reçoit les valeurs des variables manipulées calculées des deux structures de contrôle, les états du système des deux contrôleurs et des signaux booléens provenant des interrupteurs de la manette USB qui permettent à un utilisateur d'engager l'autopilote ou de commander des transitions. En sortie, le superviseur transmet les

valeurs des variables manipulées qui sont transmises à l'avion, l'état présent du système et un vecteur de signaux booléens pour indiquer quel contrôleur de mode est actif.

La figure 5.6 présente les blocs Simulink qui forment le superviseur. Il est composé d'un bloc qui détermine quel mode doit être actif, un bloc qui gère les demandes de transitions (les consignes en transition peuvent être programmées de façon automatique et/ou manuelle) et un autre bloc qui gère les consignes.

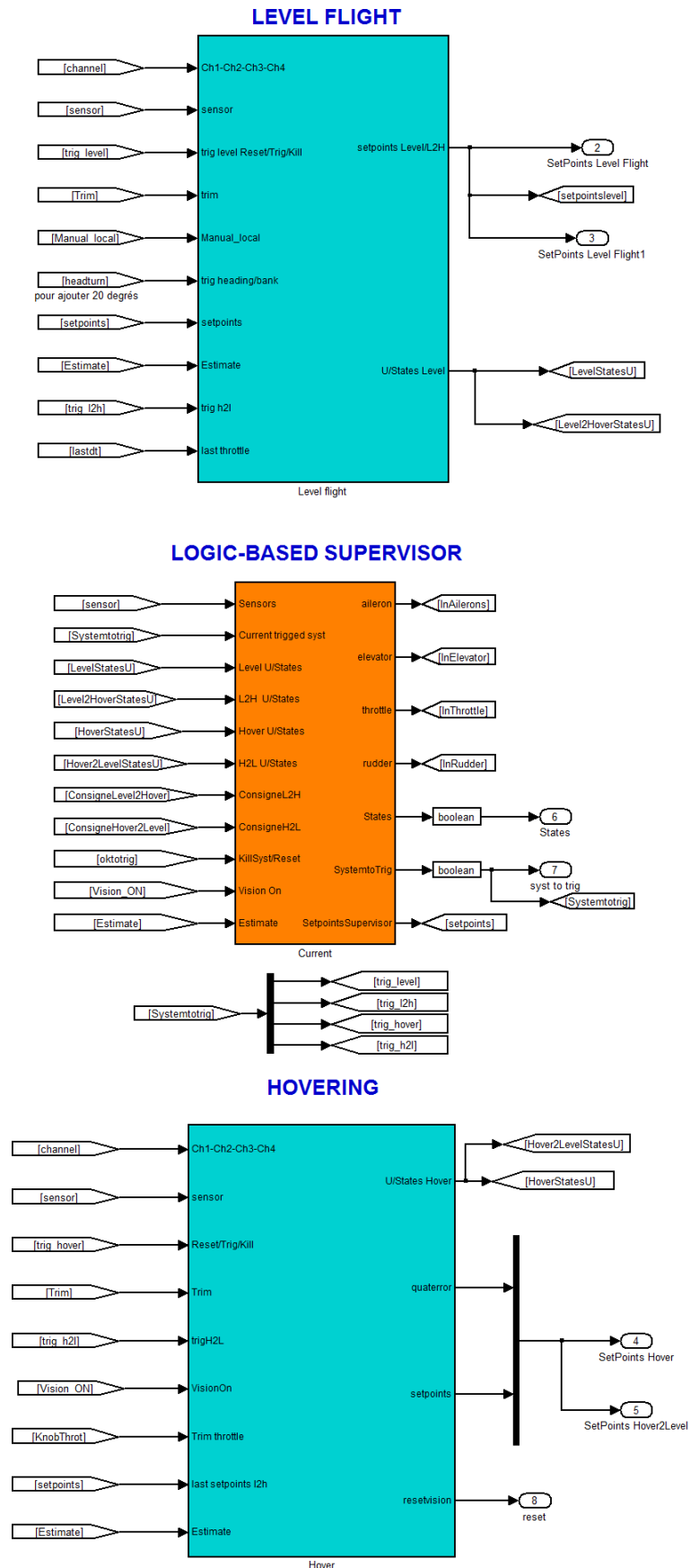
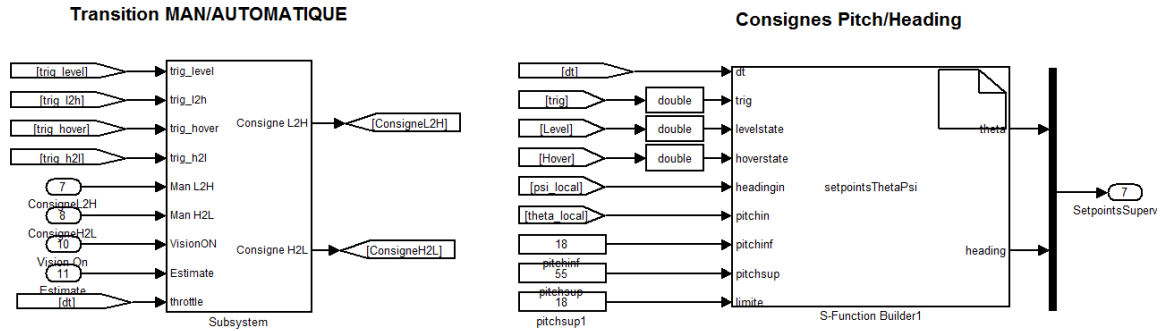
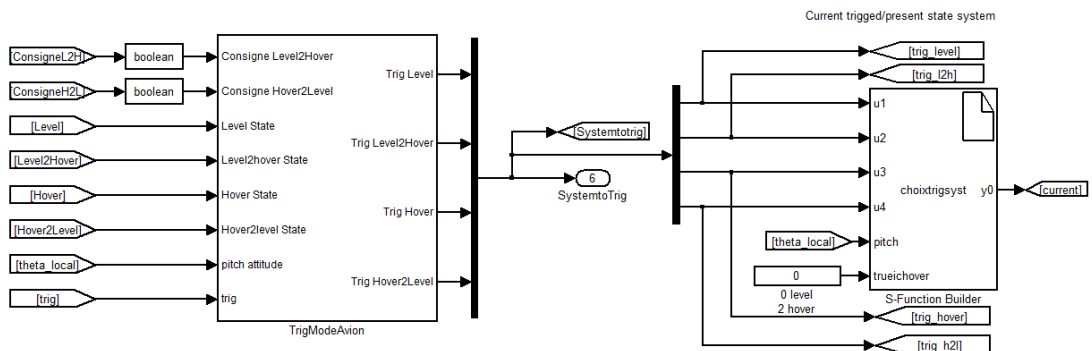


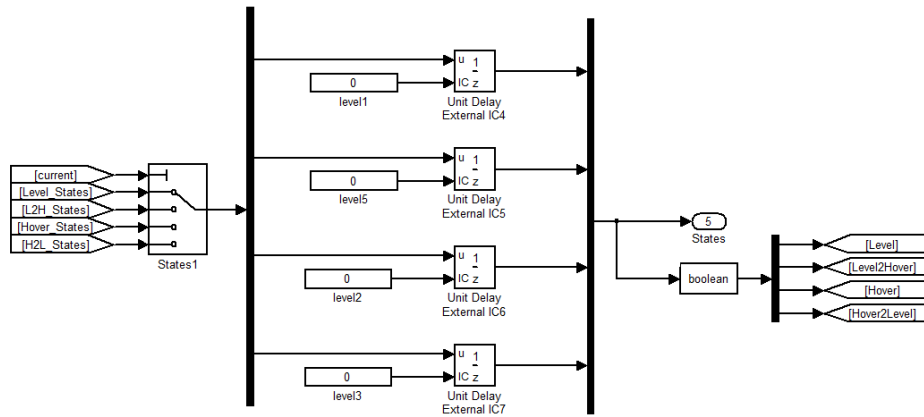
FIGURE 5.5 – Blocs Simulink des deux structures de contrôleur ainsi que du superviseur



Gestion contrôleurs et modes



États



Variables manipulés

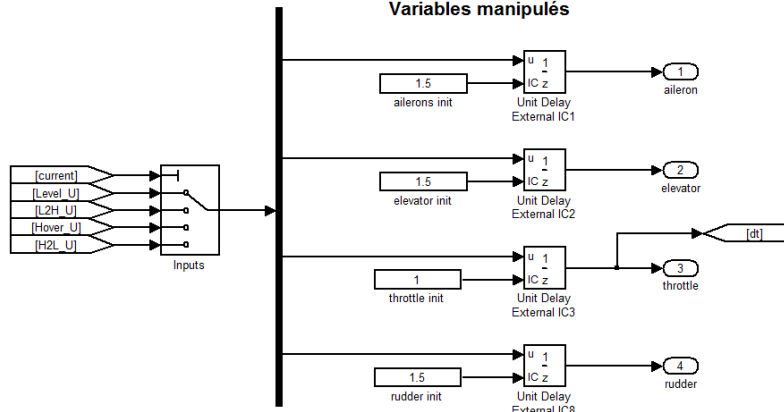


FIGURE 5.6 – Superviseur basé sur la logique

5.2.1 Présentation des essais

Plusieurs types d'essais ont été réalisés pour tester les algorithmes de contrôle, la logique du superviseur et la commutation entre les contrôleurs. En tout, 9 types d'essais sont présentés dans les sections 5.3, 5.4 et 5.5. Certains essais peuvent être visionnés sur le CD-ROM inclus avec ce mémoire.

Pour chacun des essais, on présente tout d'abord l'objectif. Par la suite, on décrit le plan de vol qui est nécessaire pour la réalisation de l'essai. Pour l'analyse des résultats, les réponses temporelles des différentes variables et consignes pertinentes sont présentées et analysées. Les variables longitudinales (θ , δ_e , z et δ_r) et les variables latérales (ϕ , δ_a , ψ et δ_r) sont présentées sur deux graphiques différents. Les angles d'Euler normaux sont utilisés pour représenter l'attitude du véhicule des modes non stationnaire et L2H (section 2.2). Pour les modes stationnaire et H2L, l'attitude est représentée par les erreurs angulaires calculées par le quaternion de déviation (section 2.3.1). Il est à noter qu'il existe d'importantes discontinuités qui sont dues au changement de représentation des attitudes ϕ et ψ entre le mode non stationnaire et le mode stationnaire. Évidemment, il n'existe pas de discontinuités pour les angles ϕ et ψ lors de la transition. La section 2.3.4 explique ce phénomène. Pour l'attitude θ en mode stationnaire et H2L, celle-ci est ramenée (*unwrapped*) autour de 90 degrés pour bien illustrer le phénomène du changement d'attitude en tangage θ . Enfin, dans ces deux graphiques, les différents modes sont séparés par des lignes et les colonnes sont identifiées.

Les variables manipulées sont physiquement limitées à des angles de déviation maximale par les micros-servos. Ainsi, δ_e est limitée à ± 55 degrés, δ_a à ± 55 degrés et δ_r à ± 80 degrés. Il est à noter que dans les modes stationnaire et H2L, les ailerons δ_a sont limités entre 0 degré (ailerons plats) et -55 degrés. Cette stratégie de contrôle a été choisie par Bilodeau et al. [2, 3]. En effet, il a été supposé que les ailerons δ_a ne doivent pas dévier de façon positive. En effet, la force mécanique δ_r crée un moment dans le sens horaire et les ailerons δ_a servent à créer un moment dans le sens anti-horaire, évitant ainsi que l'avion ne tourne sur lui-même. Si les ailerons se retrouvent à être déviés de façon positive, cela crée un moment dans le sens horaire, ce qui n'est pas souhaitable. Dans certains essais, il est possible d'observer ce phénomène de saturation à 0 degré. Cependant, une réserve est émise face à ce choix de conception. En effet, il est souhaitable que les δ_a soient déviés positivement transitoirement pour permettre au MiniAV d'effectuer plus rapidement les manoeuvres en roulis lorsque les conditions de vol l'exigent.

Il est à noter qu'en mode non stationnaire, la consigne θ est choisie à 18 degrés. Lors de l'étape de l'identification, il a été noté que l'angle tangage moyen en mode de

palier était de 18 degrés. Ainsi, lors du décollage à partir de la plateforme, l'angle θ est d'environ 10 degrés. Aussi, ϕ est rarement de 0 degré. Lorsque l'essai débute en mode de palier, la plateforme n'a pas le temps de rejoindre la consigne de θ avant d'effectuer la transition. À cause du manque de temps, des essais plus longs en mode non stationnaire n'ont pas pu être réalisés. Enfin, lors de certains essais en mode stationnaire, il est possible d'observer une oscillation des variables mesurées et manipulées. Ce phénomène est dû à un mauvais réglage des contrôleurs du mode stationnaire. En effet, la plateforme peut légèrement se modifier d'un essai à l'autre à cause de réparations ou de l'usure des micros-servos.

La figure 5.7 présente un schéma explicatif des différents tests de vol. Ainsi la séquence suivante est possible : LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER (① \rightarrow ② \rightarrow ③ \rightarrow ④ \rightarrow ⑤ \rightarrow ⑥ \rightarrow ⑦).

Enfin, pour certains essais, on présente les données de mesure du système de vision pour les distances parcourues $x - y$, c'est-à-dire la position du véhicule, et la vitesse V_t . Les données de mesure du système de vision sont présentées lorsque le Flatana se trouve dans ③,④,⑤,⑥ et ⑦ de la figure 5.7. Dans ces graphiques, la vitesse V_t est calculée avec l'équation 2.3.

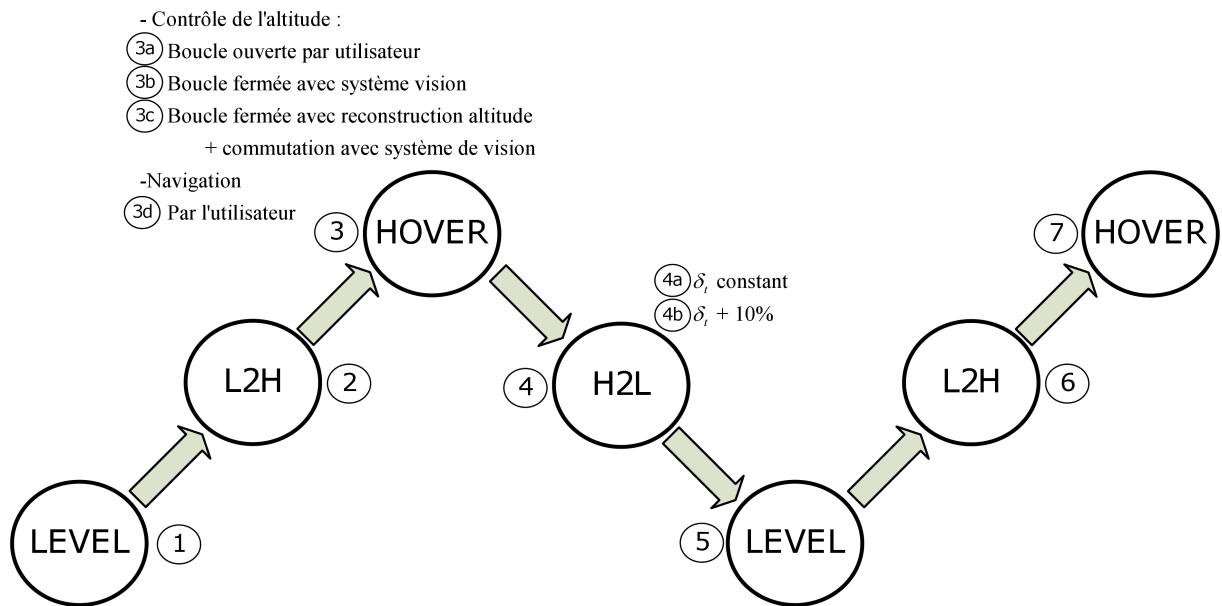


FIGURE 5.7 – Schéma des essais

5.3 Essais de transition du vol non stationnaire vers le vol stationnaire (L2H)

Les essais de cette section portent sur la réalisation de transitions du vol non stationnaire vers le vol stationnaire. Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H \rightarrow HOVER (① \rightarrow ② \rightarrow ③).

5.3.1 Angle de décrochage θ_{stall}

Afin de choisir le moment de la commutation, il faut caractériser l'instant où l'angle θ atteint une valeur maximum. Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H (① \rightarrow ②).

Plan de vol pour la réalisation de l'essai

L'avion est déposé sur la rampe de lancement. La rampe est placée de façon à ce que l'avion se retrouve à tout moment dans la zone de capture du système de vision. À $t = 0$ sec, l'utilisateur engage l'autopilote et l'avion décolle en mode non stationnaire avec $\bar{\theta} = 18^0$ et $\bar{\psi} = \psi_{t=0}$. À $t = 1$ sec, une transition est commandée automatiquement. Le superviseur active le mode L2H et le superviseur reste dans ce mode jusqu'à ce que l'utilisateur désengage l'autopilote.

Analyse des résultats

Tout d'abord, on remarque sur la figure 5.8 la séquence planifiée δ_t qui est décrite dans la section 4.4.1. On remarque aussi, lors de cet essai, que l'angle θ atteint un maximum à θ_{stall} à 62.7 degrés. L'avion continue son ascension encore pendant quelques instants, mais θ se met à diminuer rapidement. On remarque que δ_e est saturée au début de la manoeuvre (à cause du PI anti-reset). Par la suite, la déviation diminue (avec l'erreur qui diminue) jusqu'à θ_{stall} . Avec cette séquence planifiée δ_t , on peut aussi remarquer que l'on atteint une altitude maximale (-1.126 m), limitant ainsi la plateforme à gagner trop d'altitude. Après avoir atteint une altitude maximale, le Flatana s'écrase rapidement dans le filet, et ce, même si δ_e redevient saturé. À cause du phénomène du décrochage, δ_e perd de son efficacité et le MiniAV est incapable de reprendre de l'alti-

tude ou d'augmenter son angle de tangage. En moyenne, lors des expérimentations, il a été observé que $\theta_{stall} \approx 65$ degrés.

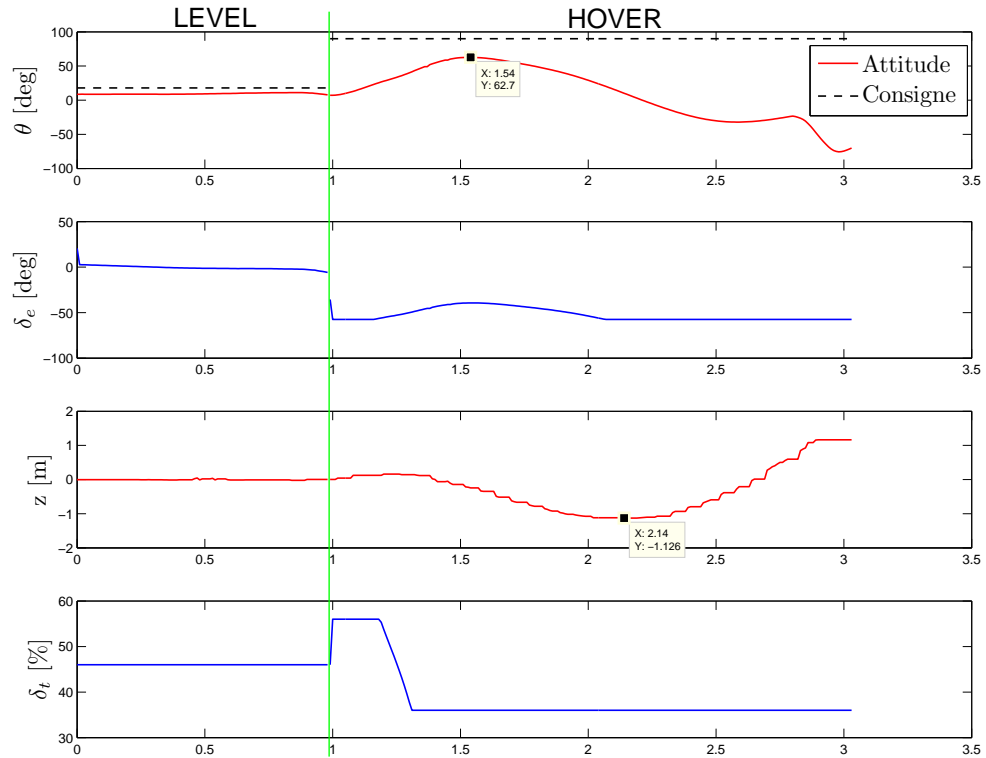


FIGURE 5.8 – Essai LEVEL \rightarrow L2H : variables longitudinales

5.3.2 Variables latérales

Cet essai vise à étudier le comportement des variables latérales durant le mode L2H ainsi que durant les premiers instants en mode stationnaire. Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) (① \rightarrow ② \rightarrow ③b).

Plan de vol pour la réalisation de l'essai

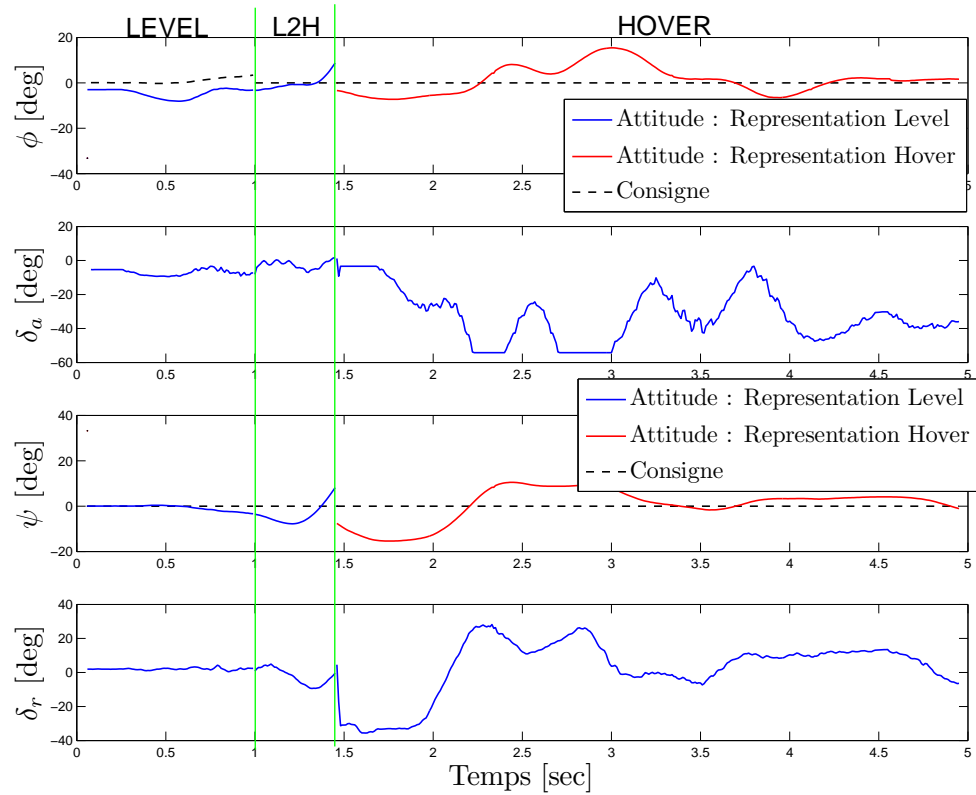
L'avion est déposé sur la rampe de lancement. La rampe est placée le plus loin possible et elle est enlignée avec la zone de capture du système de vision. À $t = 0$ sec, l'utilisateur engage l'autopilote et l'avion décolle en mode non stationnaire avec $\bar{\theta} = 18^0$ et $\bar{\psi} = \psi_{t=0}$. À $t = 1$ sec, une transition est commandée automatiquement. Le superviseur active le mode L2H. À 50^0 , le superviseur active le mode stationnaire. Pour les besoins d'analyse des variables latérales, le système de vision est utilisé comme données de mesure pour la rétroaction de l'altitude. Lorsque tous les axes et l'altitude sont stabilisés, l'utilisateur peut désengager l'autopilote.

Analyse des résultats

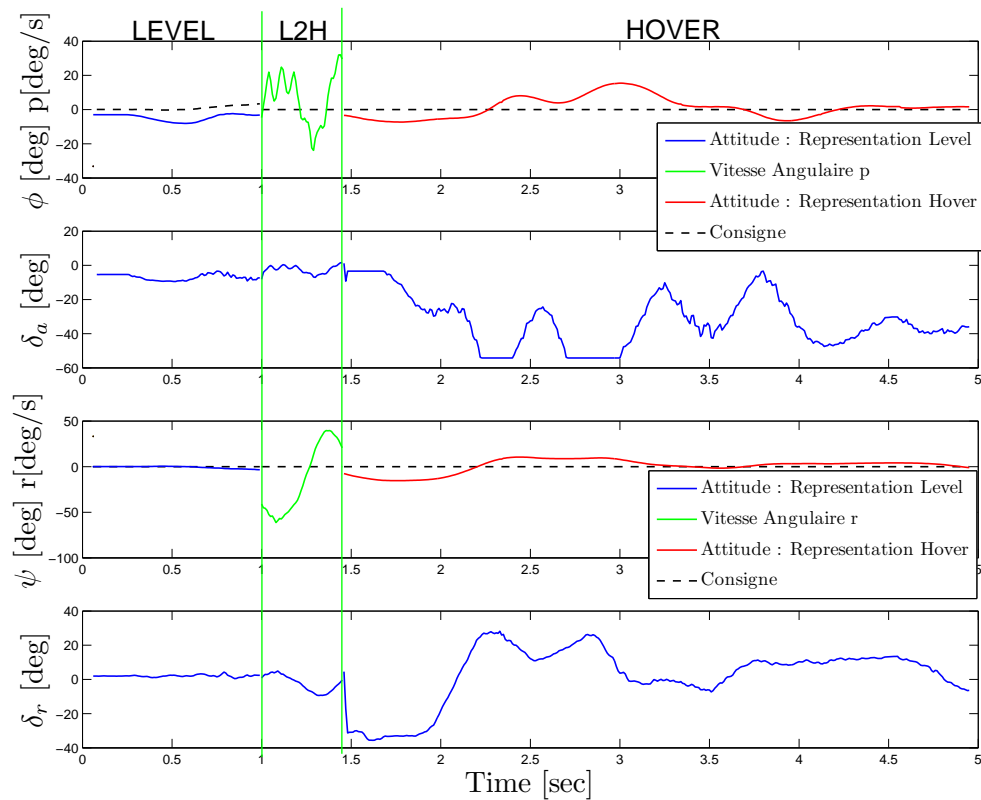
Les figures 5.9(a) et 5.9(b) montrent les réponses temporelles des variables latérales de cet essai. Le Flatana est dans le mode L2H moins de 0.5 seconde. La différence entre ces deux figures est la représentation du mode L2H durant la manoeuvre : la figure 5.9(a) présente les attitudes ϕ et ψ et la figure 5.9(b) présente les vitesses angulaires p et r . On remarque que, malgré le court temps dans le mode L2H, les régulateurs P_p et P_r de la figure 3.18 arrive à maintenir, en moyenne, des vitesses angulaires autour de 0, ce qui permet un mouvement en tangage. On remarque le même comportement pour les attitudes. Cette manoeuvre est très rapide et critique. En effet, l'avion doit monter vers le mode stationnaire le plus droit possible. Il a été remarqué, au début de l'expérimentation, que l'utilisation de la structure du contrôleur latéral du mode non stationnaire, avec ψ comme consigne de la boucle externe, n'était pas assez rapide pour réaliser convenablement la manoeuvre L2H.

Comme il a été expliqué dans la section 2.3.4, il existe des discontinuités dues au changement de la représentation de l'attitude entre les modes non stationnaire et stationnaire. Ainsi, lors de l'entrée dans le mode stationnaire, on remarque une importante déviation (-35 degrés) de δ_r . Cette importante déviation est due à l'erreur en ψ (ex-

plication à la section 2.3.4) et à l'agressivité du contrôleur de l'axe Z_B pour compenser cette erreur. L'attitude en ψ se stabilise tout de même assez rapidement autour de sa consigne de 0 degré. Concernant δ_a , on remarque le changement du point d'opération. En effet, lorsque l'attitude ϕ est contrôlée autour de sa consigne, le point d'opération de $\delta_a = -40$ degrés, comparé à 0 degré en mode non stationnaire.



(a) Attitude : Représentation Level



(b) Vitesse angulaire

FIGURE 5.9 – Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) : variables latérales

5.3.3 Contrôle de l'altitude en boucle ouverte

Cet essai vise à réaliser une transition semi-autonome du vol non stationnaire vers le vol stationnaire : l'utilisateur doit contrôler l'altitude via δ_r . Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H \rightarrow HOVER (z boucle ouverte) ($\textcircled{1} \rightarrow \textcircled{2} \rightarrow \textcircled{3a}$).

Plan de vol pour la réalisation de l'essai

L'avion est déposé sur la rampe de lancement. La rampe est placée le plus loin possible et elle est enlignée avec la zone de capture du système de vision. À $t = 0$ sec, l'utilisateur engage l'autopilote et l'avion décolle en mode non stationnaire avec $\bar{\theta} = 18^\circ$ et $\bar{\psi} = \psi_{t=0}$. À $t = 1$ sec, une transition est commandée automatiquement. Le superviseur active le mode L2H. À 50° , le superviseur active le mode stationnaire. À cet instant, l'utilisateur doit contrôler, à l'aide de la manette USB, l'altitude en manipulant δ_r . Les autres variables manipulées (δ_a, δ_e et δ_r) sont autonomes. Lorsque tous les axes sont stabilisés et que l'altitude est contrôlée à une hauteur désirée par l'utilisateur, celui-ci peut désengager l'autopilote.

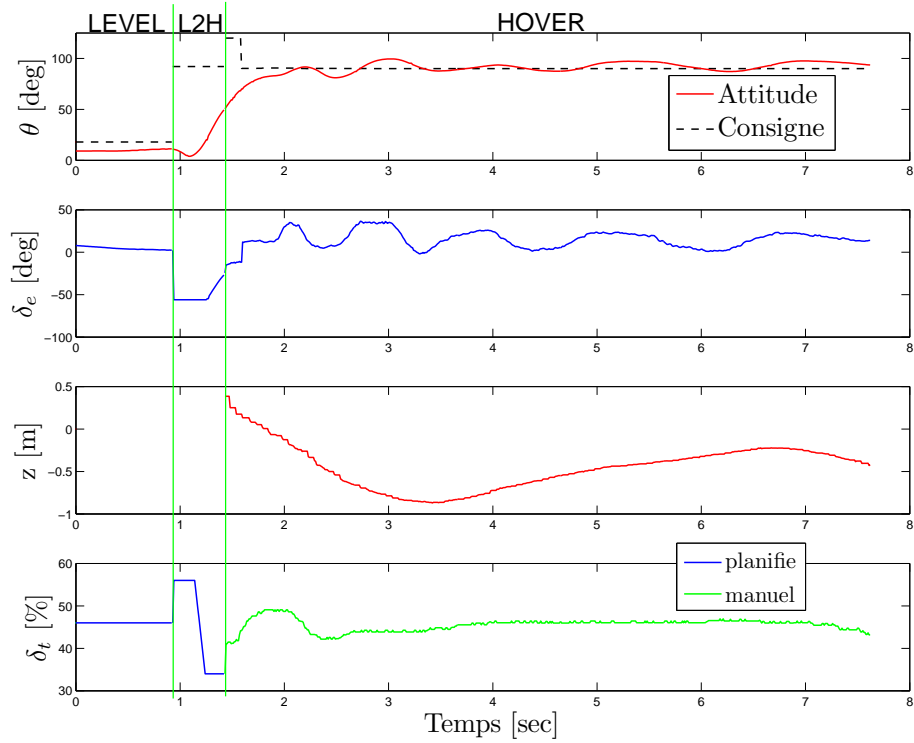
Analyse des résultats

La figure 5.11(a) montre les réponses temporelles des variables longitudinales et la figure 5.11(b) montre les réponses temporelles des variables latérales. Dans 5.10(a), on remarque qu'il faut plus de 6 secondes avant que le véhicule se positionne à une altitude désirée par l'utilisateur. Les variables latérales prennent environ 3 secondes avant de se stabiliser autour de leur consigne.

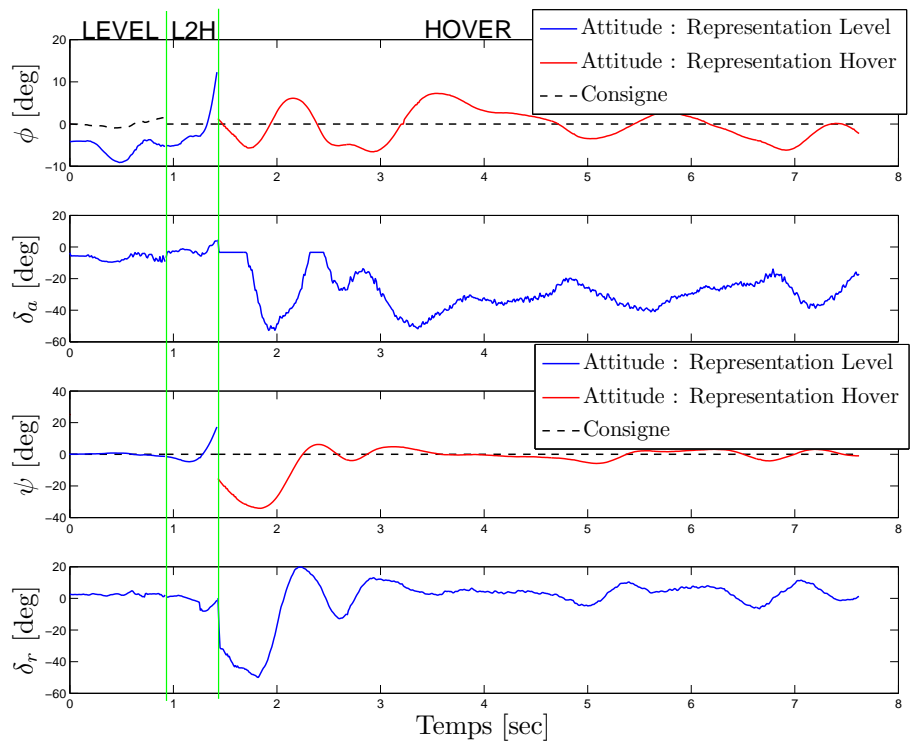
Il est à noter que cet essai a été réalisé par un utilisateur expérimenté et que celui-ci était à seulement quelques mètres de l'avion et qu'il pouvait voir le véhicule. Ainsi, on remarque que la courbe ' δ_r manuel' ne subit pas d'importante variation. On peut donc conclure que, durant une mission, si l'utilisateur a une connaissance de l'environnement visuel de la plateforme, cette technique de contrôle de l'altitude peut être envisageable. Dans le cadre d'une mission, une caméra sur le ventre de l'avion, transmettant des images en temps réel, serait alors nécessaire. Évidemment, un entraînement plus spécifique serait nécessaire, mais cette technique pourrait être réalisable.

Il est possible de comparer avec Green [18, 17] ce type d'essai. On retrouve à [19] une vidéo de la plateforme de Green [18, 17] qui montre deux essais de transition L2H où

δ_t est contrôlée manuellement par l'utilisateur. Le premier essai est réalisé à l'extérieur et le second à l'intérieur dans un gymnase. Tout d'abord, on peut voir que lorsque la transition est réalisée, il n'y a pas de boucle fermée sur ϕ qui a été intégrée dans le système de contrôle. Ainsi, à cause du moment de force mécanique (*torque*) créé par δ_t , l'avion tourne lentement sur lui-même. Dans [17], des réponses temporelles d'essais de la variable θ sont présentées. Il faut environ 2 secondes pour atteindre et contrôler la consigne en tangage autour de 90 degrés. Aucune donnée de mesure sur l'attitude ou sur les variables manipulées durant une manoeuvre de transition n'a été publiée jusqu'à ce jour par Green [17, 18]. Un lien Internet est disponible sur le CD-ROM pour visionner la vidéo (`Green_Autonomous_Cruise-to-Hover_Transition`).



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.10 – Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle ouverte) : variables longitudinales et latérales

5.3.4 Contrôle de l'altitude en boucle fermée avec le système de vision

Cet essai vise à réaliser une transition autonome du vol non stationnaire vers le vol stationnaire. Toutes les variables sont en boucle fermée. Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) (① \rightarrow ② \rightarrow ③b).

Plan de vol pour la réalisation de l'essai

L'avion est déposé sur la rampe de lancement. La rampe est placée le plus loin possible et enlignée avec la zone de capture du système de vision. L'utilisateur engage l'autopilote et l'avion décolle en mode non stationnaire à $t = 0$ avec des consigne $\bar{\theta} = 18^0$ et $\bar{\psi} = \psi_{t=0}$. À $t = 1$ sec, une transition est commandée automatiquement. Le superviseur active le mode L2H. À 50^0 , le superviseur active le mode stationnaire. Le système de vision est utilisé comme données de mesure pour la rétroaction de l'altitude. Lorsque tous les axes et l'altitude sont stabilisés, l'utilisateur peut désengager l'autopilote.

Analyse des résultats

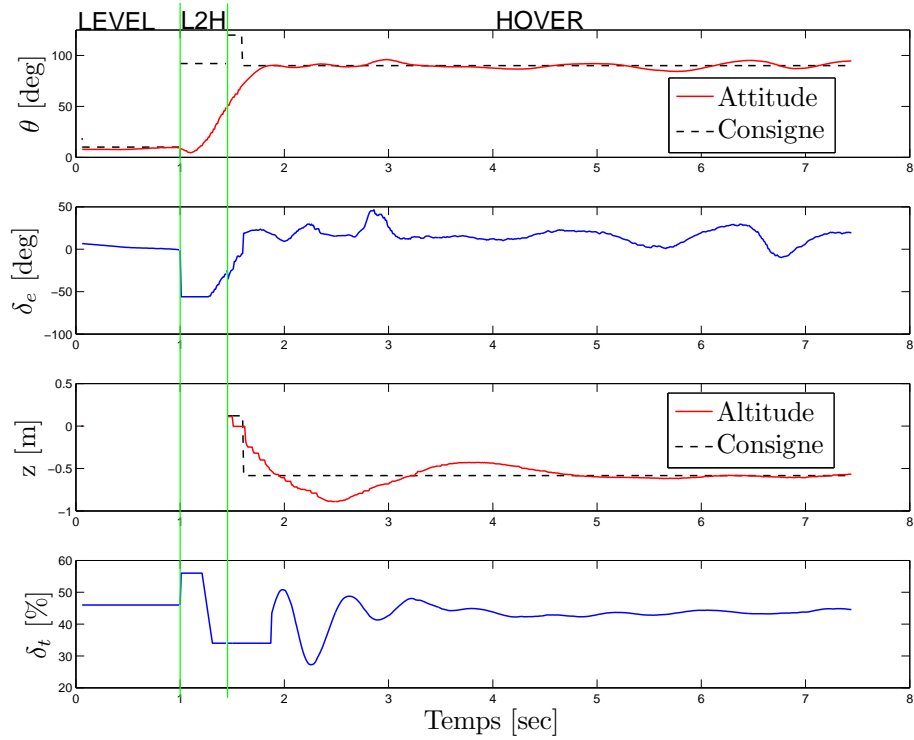
La figure 5.11(a) montre les réponses temporelles des variables longitudinales et la figure 5.11(b) montre les variables latérales. La figure 5.12 présente la position ($x - y$) et la vitesse V_t du véhicule. Cet essai peut être visionné sur le CD-ROM sous le nom de `Transition_L2H.avi`.

La figure 5.11(a) montre que le MiniAV est dans le mode L2H moins de 0.5 secondes. Il faut moins de 3 secondes après qu'une transition soit commandée pour que les attitudes ϕ , θ et ψ soient stabilisées autour de leur consigne (oscillations à ± 5 deg). Enfin, un dépassement en altitude de 30 cm est observé et le temps de réponse est de moins de 3.5 secondes. Dans la figure 5.12, on remarque que le véhicule atteint une vitesse maximale V_t d'environ 5.8 m/s. Il faut moins de 3 secondes pour que la vitesse V_t diminue autour de 0 m/s. Les figures 5.13 et 5.14 présentent des captures d'images d'une vidéo d'un essai L2H.

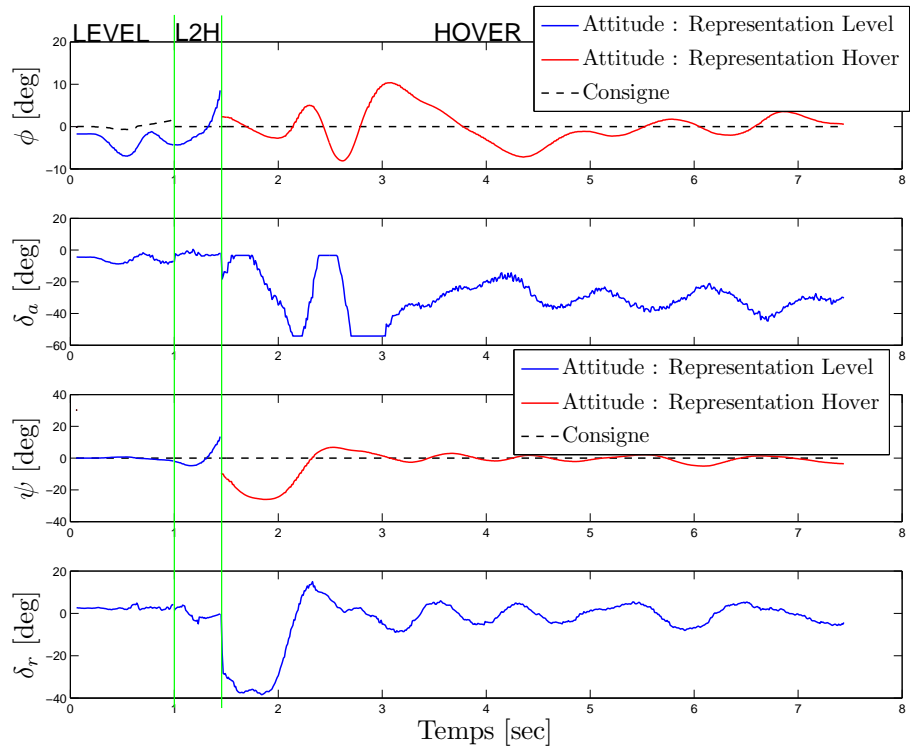
Il est possible de comparer avec Frank et al. [12] la performance de ce type d'essai. Pour réaliser la manoeuvre, δ_e et δ_r sont utilisées en boucle ouverte pour créer un moment de tangage, les ailerons δ_a sont utilisés en boucle fermée pour garder le cap et

δ_t est diminué légèrement en dessous du point d'opération en mode stationnaire. Ainsi, lors de la transition du mode en palier vers le mode stationnaire, leur plateforme prend plus de 6 secondes avant de stabiliser θ autour de sa consigne en tangage à 90 degrés. Aussi, il y a une oscillation d'environ 1.4 mètre en altitude avant que celle-ci ne soit contrôlée à une consigne désirée. Aucune réponse temporelle n'est présentée pour les variables ϕ et ψ et les variables manipulées. Enfin, avec une vitesse initiale (avant la transition) d'environ 6 m/s cela prend environ 6 secondes avant que la vitesse V_t soit autour de 0 m/s.

On peut retrouver à [13] une vidéo de la plateforme de Frank et al. [12] effectuant un essai de ce genre. L'avion décolle du sol et réalise une transition vers le vol stationnaire. On remarque sur la vidéo que l'avion prend plusieurs secondes avant d'être contrôlé en altitude et pour que les variables latérales se stabilisent autour de leur consigne. Un lien Internet est disponible sur le CD-ROM pour visionner la vidéo (MIT_Autonomous_Airplane_Ground_Take-off_to_Hover).



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.11 – Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) : variables longitudinales et latérales

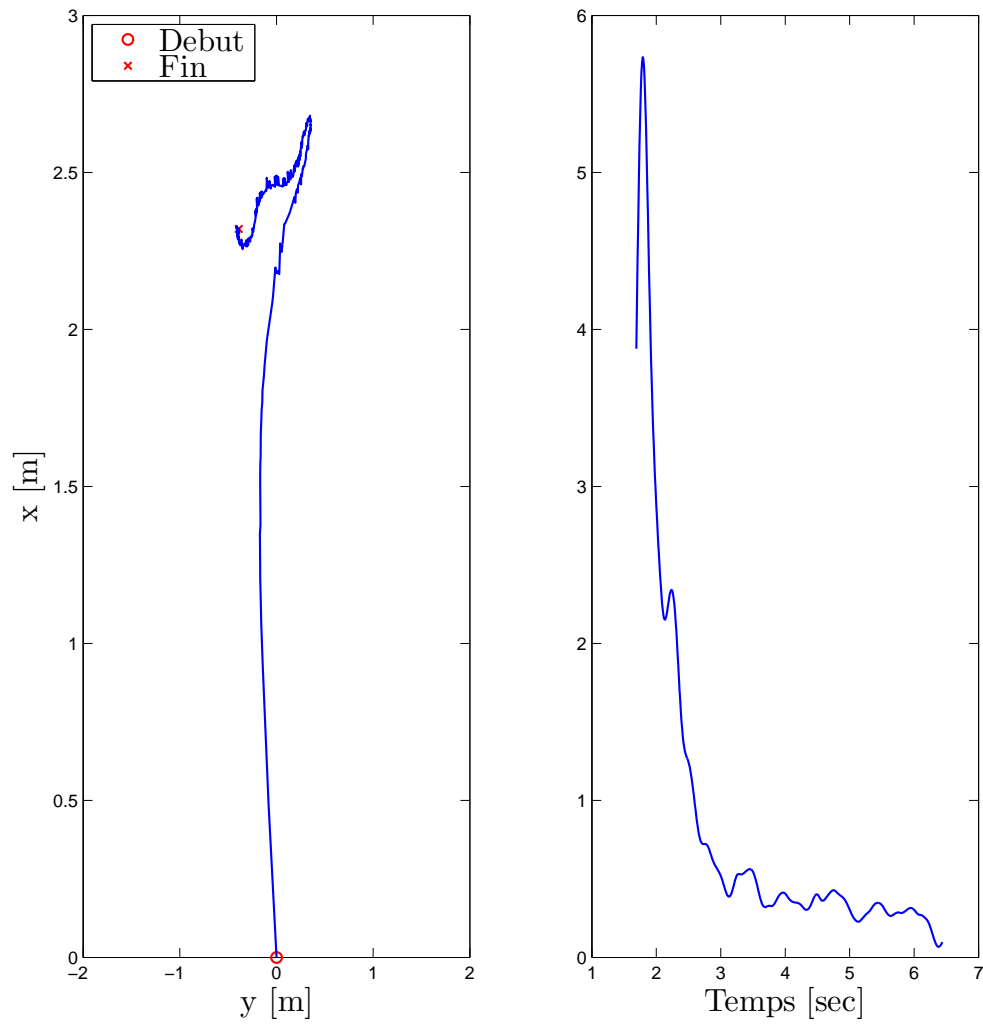


FIGURE 5.12 – Essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) : position ($x-y$) et vitesse V_t



(a) Mode non stationnaire



(b) Mode L2H



(c) Mode stationnaire

FIGURE 5.13 – Images capturées d’une vidéo : transition L2H, vue d’en dessous de la rampe de lancement



(a) Mode non stationnaire



(b) Mode L2H



(c) Mode stationnaire

FIGURE 5.14 – Images capturées d’une vidéo : transition L2H, vue de côté

5.3.5 Contrôle de l'altitude en boucle fermée avec estimation de l'altitude

Évidemment, une technique de contrôle de l'altitude en mode stationnaire suppose l'existence d'un capteur robuste capable de mesurer l'altitude avec une précision au centimètre.

Cette section vise à étudier et démontrer la faisabilité du contrôle en boucle fermée avec l'estimation de l'altitude en boucle fermée à l'aide de la technique INS (section 4.4.4). Avant de réaliser des essais, on se doit de comparer les positions reconstruites utilisant les équations de la section 4.4.4 avec les données de mesure du système de vision numérique. Pour ce faire, on effectue un essai LEVEL \rightarrow L2H \rightarrow HOVER (① \rightarrow ② \rightarrow ③), figure 5.7) et on compare les séries de données recueillies. Les conditions initiales de la technique INS sont les suivantes : $u_0 = 0, v_0 = 0, w_0 = 0, x_0 = 0, y_0 = 0$ et $z_0 = 0$. Dans le cadre de cette étude, les données de mesure du IMU ne sont pas filtrées. Les données en position du système de vision sont filtrées à l'aide d'un filtre numérique passe-bas de Butterworth (4e ordre, $\omega_n = 0.08$).

La figure 5.15 compare les vitesses u, v, w . Pour obtenir les vitesses u, v, w reconstruites, on utilise les équations 4.20 à 4.22. Pour obtenir les vitesses u, v, w du système de système de vision, on effectue une dérivation des données de mesure de position et on utilise l'équation 2.9 pour obtenir les vitesses u, v, w dans le repère avion. Il est à noter que l'avion pénètre dans le système de vision uniquement lorsque l'avion se retrouve en mode stationnaire. On remarque donc que la vitesse u atteint une valeur maximale d'environ 6 m/s et que durant la manoeuvre de transition, les valeurs en v et w ne sont pas nulles. Les vitesses mesurées par le système de vision sont plus faibles, mais on remarque qu'en mode stationnaire, la même dynamique est observée.

La figure 5.16 compare les positions x, y, z reconstruites avec les données de mesure en position du système de vision. On peut remarquer clairement que, après environ 5 secondes, les distances parcourues x et y commencent à diverger. L'altitude z prend un peu plus de temps à diverger dans cet essai. Si le IMU est bien calibré, les erreurs d'instrumentation peuvent être négligées sur une courte période (moins de 5 secondes) si on utilise une technique d'intégration numérique.

Enfin, il peut être intéressant de comparer l'attitude du système de vision avec celle générée par le IMU. Comme il a été mentionné dans la section 2.7, le système de vision retourne comme données de mesure l'attitude en quaternion. La figure 5.17 présente une comparaison entre l'attitude générée par le IMU et celle générée par le système de

système de vision. Les données sont transformées en erreurs angulaires (équations 2.64 à 2.66). On remarque que les deux capteurs génèrent des attitudes très similaires.

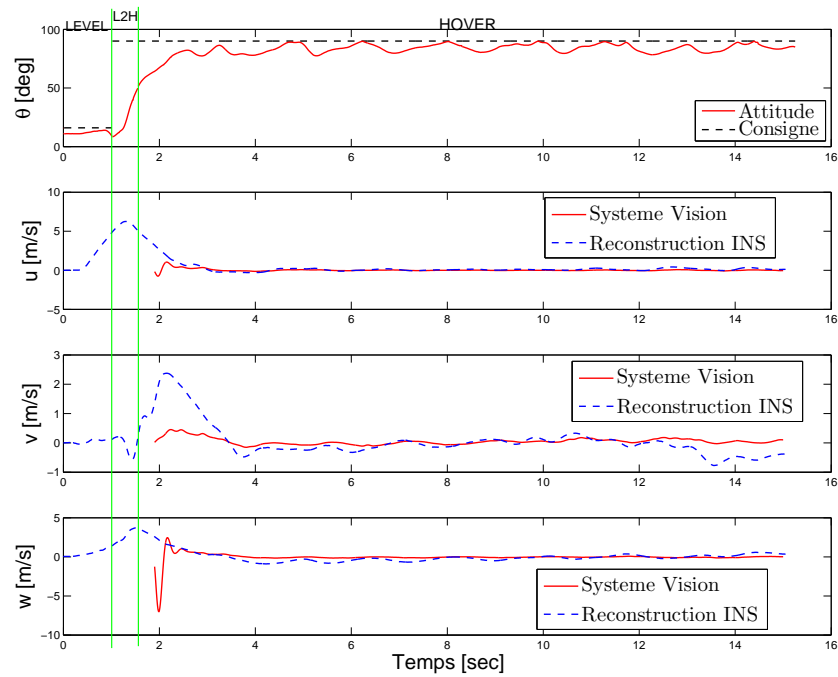


FIGURE 5.15 – Reconstruction de la vitesse versus les données du système de vision : θ , u , v et w

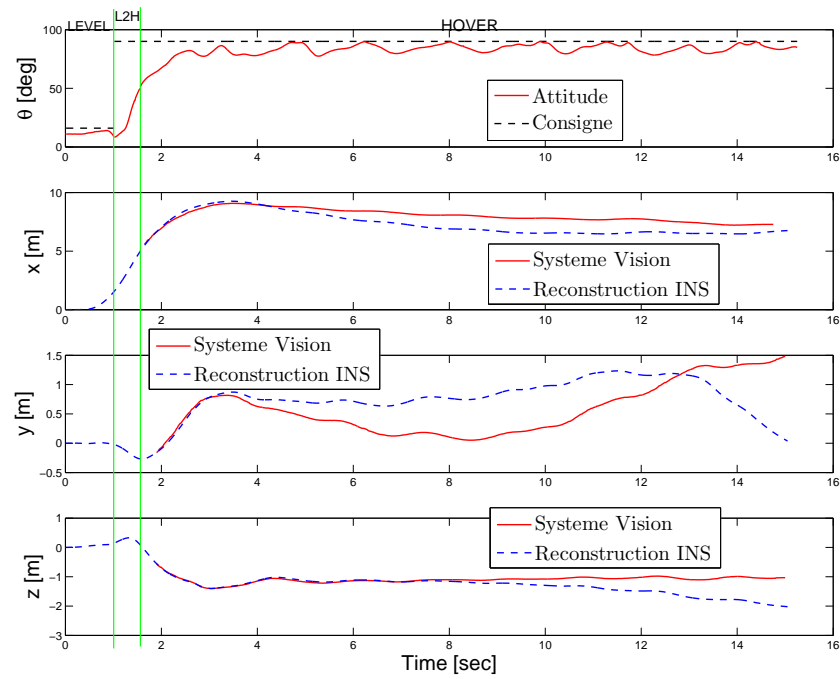


FIGURE 5.16 – Reconstruction de la position versus les données du système de vision : θ , x , y et z

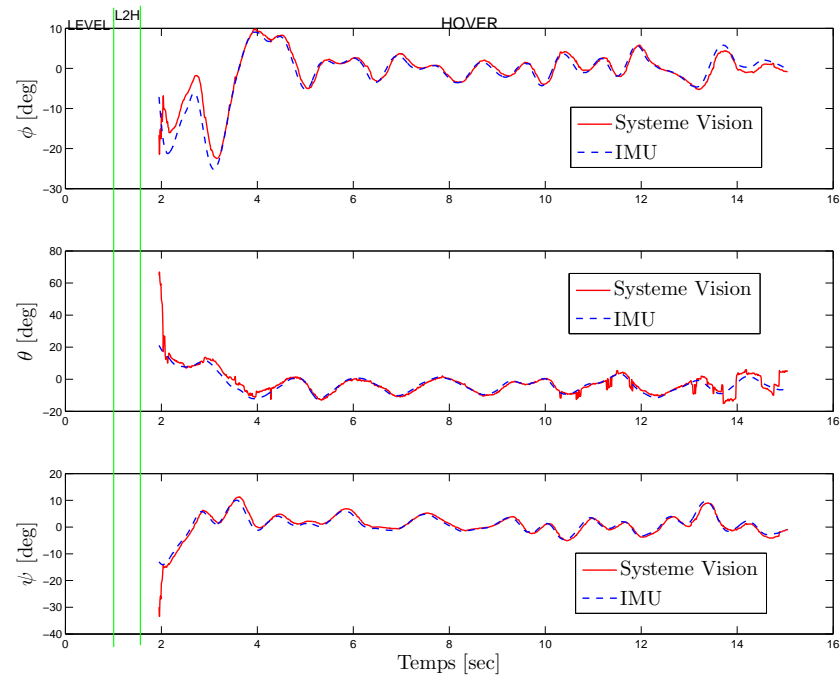


FIGURE 5.17 – Comparaison de l'attitude en utilisant les données du IMU versus celles du système de vision

On suppose donc qu'avec la technique INS, il est possible de réaliser un contrôle de l'altitude en boucle fermée sur une courte période. Ainsi, en estimant l'altitude durant la manoeuvre, il est possible de stabiliser la plateforme en mode stationnaire pendant quelques secondes. Par la suite, un autre capteur (sonar, système de traitement d'images, GPS, capteur barométrique) peut être utilisé pour permettre un contrôle en boucle fermée de l'altitude à long terme. Les sections 4.4.4 et 3.2 exposent ce problème. Ici, on choisit d'utiliser la technique INS pendant 5 secondes pour stabiliser la plateforme en mode stationnaire et on réalise la commutation avec un autre capteur plus robuste (le système de vision).

Cet essai vise à réaliser une transition complète. Toutes les variables sont en boucle fermée. Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H \rightarrow HOVER (z boucle fermée, reconstruction z et commutation avec système vision) (① \rightarrow ② \rightarrow ③c).

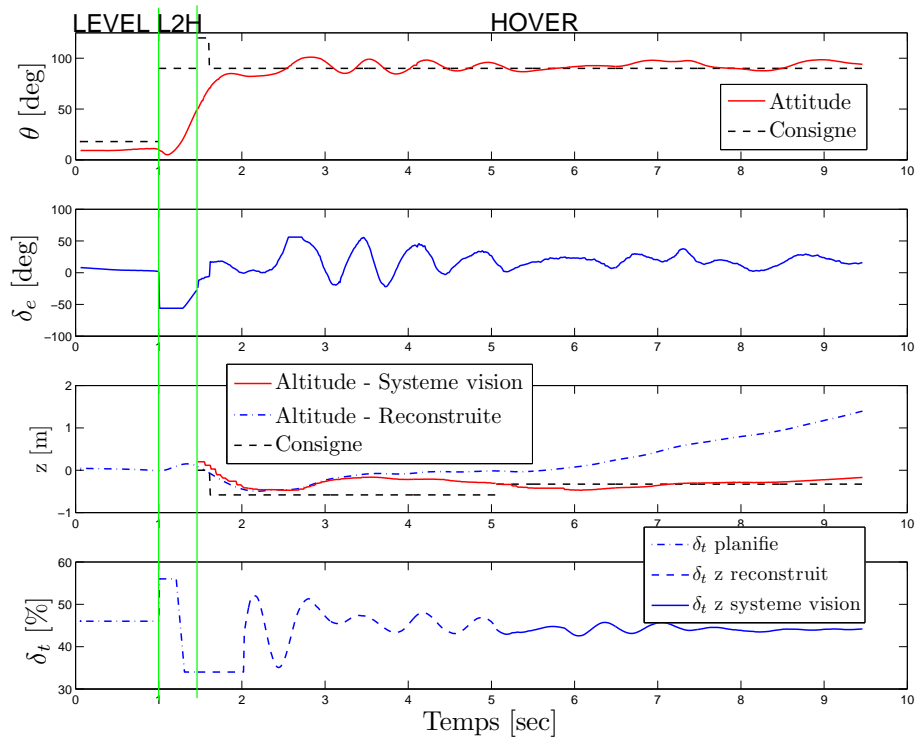
Plan de vol pour la réalisation de l'essai

L'avion est déposé sur la rampe de lancement. La rampe est placée le plus loin possible et enlignée avec la zone de capture du système de vision. L'utilisateur engage l'autopilote et l'avion décolle en mode non stationnaire à $t = 0$ avec des consignes $\bar{\theta} = 18^0$ et $\bar{\psi} = \psi_{t=0}$. Les conditions initiales de la technique INS sont les suivantes : $u_0 = 0, v_0 = 0, w_0 = 0, x_0 = 0, y_0 = 0$ et $z_0 = 0$. À $t = 1$ sec, une transition est commandée automatiquement. Le superviseur active le mode L2H. À 50^0 , le superviseur active le mode stationnaire. L'altitude reconstruite est utilisée comme signal pour la rétroaction. À $t = 5$ sec, le système de vision remplace le signal reconstruit comme signal utilisé pour la rétroaction et $\bar{z} = z_{vision\ t=5}$. Lorsque tous les axes et l'altitude sont stabilisés, l'utilisateur peut désengager l'autopilote.

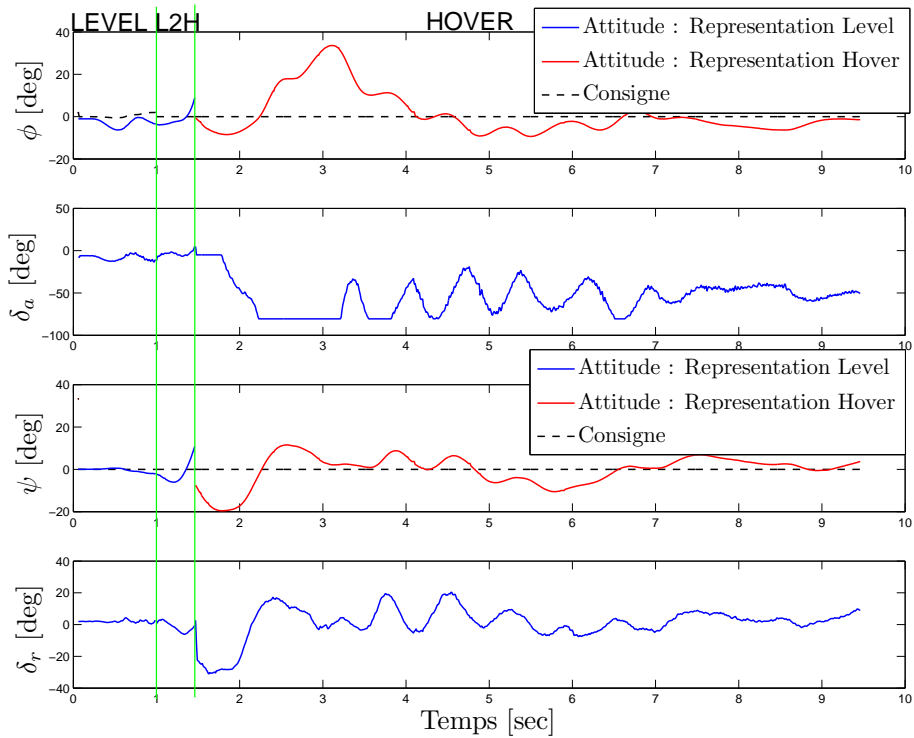
Analyse des résultats

La figure 5.18(a) montre les réponses temporelles des variables longitudinales et la figure 5.18(b) montre les variables latérales. Dans la figure 5.18(a), on remarque que l'altitude reconstruite commence à diverger à environ 4 secondes. À 5 secondes, les données de mesures du système de vision sont utilisées. Des performances semblables à l'essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision) sont atteintes.

Fait à noter, on peut observer le phénomène de l'intégration d'une marche aléatoire décrit dans la section 4.4.4. En effet, on remarque que l'altitude ne diverge pas dans la même direction dans les figures 5.18(a) et 5.16.



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.18 – Essai LEVEL \rightarrow L2H \rightarrow HOVER (z boucle fermée, reconstruction de z et commutation du système de vision) : variables longitudinales et latérales

5.4 Essais de transition du vol stationnaire vers le vol non stationnaire (H2L)

Les essais de cette section porte sur la réalisation des transitions du vol stationnaire vers le vol non stationnaire. Pour la réalisation de ces essais, on ne considère pas le problème du contrôle de l'altitude.

5.4.1 δ_t constante

Cet essai vise à réaliser une transition autonome du vol stationnaire vers le vol non stationnaire. Durant le mode H2L, δ_t est égale à la dernière mesure δ_t du mode stationnaire. Sur la figure 5.7, on réalise la séquence suivante : HOVER \rightarrow H2L (δ_t constante) \rightarrow LEVEL (③b) \rightarrow (④a) \rightarrow (⑤).

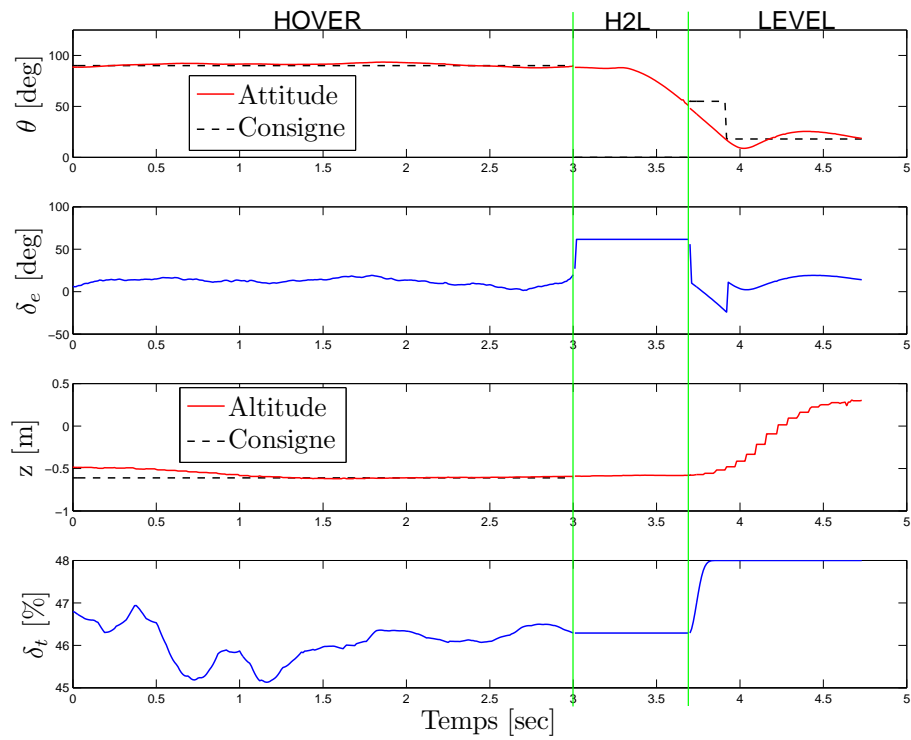
Plan de vol pour la réalisation de l'essai

L'avion est en mode stationnaire à la limite de la zone de capture du système de vision. À $t = 0$ sec, tous les axes sont stabilisés autour de leur consigne ($\bar{\phi} = \psi_{level}$, $\bar{\theta} = 90$ et $\bar{\psi} = 0$). À $t = 3$ sec, une transition est commandée manuellement par l'utilisateur à l'aide d'un interrupteur de la manette USB. Le superviseur active le mode H2L et $\delta_t = \delta_t(4.99) + 10\%$. À 50^0 , le superviseur active le mode non stationnaire. Lorsque l'avion est rendu près du filet de sécurité, l'utilisateur désengage l'autopilote.

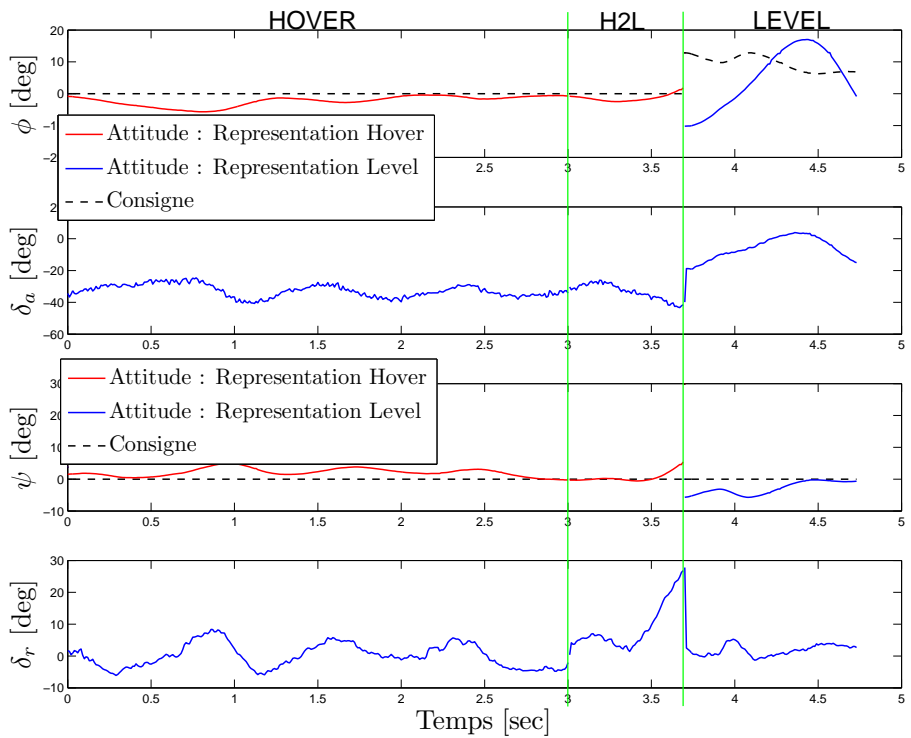
Analyse des résultats

La figure 5.19(a) montre les réponses temporelles des variables longitudinales et la figure 5.19(b) montre les variables latérales. Dans la figure 5.19(a), on peut voir que lorsque la transition H2L est commandée, δ_e sature à 55 degrés. La figure 5.11(a) montre que le MiniAV est dans le mode H2L moins de 0.6 seconde. Aussi, la valeur de δ_t est égale à sa dernière valeur en mode stationnaire pendant tout le mode H2L et l'altitude reste constante dans ce mode. On voit aussi δ_t rejoindre, grâce à l'algorithme 4.5, le point d'opération du mode non stationnaire qui est supérieur. Cependant, lorsque le mode non stationnaire est activé, il y a une perte en altitude d'environ 80 cm par rapport à l'altitude de référence en mode stationnaire. La figure 5.21 (section suivante) présente

la position $(x - y)$ et la vitesse V_t du véhicule. On remarque que le véhicule atteint une vitesse maximale V_t d'environ 5.8 m/s. Les données de cet essai sont comparées avec les données de l'essai de la section 5.4.2.



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.19 – Essai HOVER \rightarrow H2L (δ_t constante) \rightarrow LEVEL : variables longitudinales et latérales

5.4.2 Augmentation de δ_t

Cet essai vise à réaliser une transition autonome du vol stationnaire vers le vol non stationnaire. Durant le mode H2L, δ_t est égale à la dernière mesure du mode stationnaire + 10%. Sur la figure 5.7, on réalise la séquence suivante : HOVER \rightarrow H2L ($\delta_t + 10\%$) \rightarrow LEVEL (③b \rightarrow ④b \rightarrow ⑤).

Plan de vol pour la réalisation de l'essai

L'avion est en mode stationnaire à la limite de la zone de capture du système de vision. À $t = 0$ sec, tous les axes sont stabilisés autour de leur consigne ($\bar{\phi} = \psi_{level}$, $\bar{\theta} = 90$ et $\bar{\psi} = 0$). À $t = 3$ sec, une transition est commandée manuellement par l'utilisateur à l'aide d'un interrupteur de la manette USB. Le superviseur active le mode H2L et $\delta_t = \delta_t(4.99) + 10\%$. À 50^0 , le superviseur active le mode non stationnaire. Lorsque l'avion est rendu très près du filet de sécurité, l'utilisateur désengage l'autopilote.

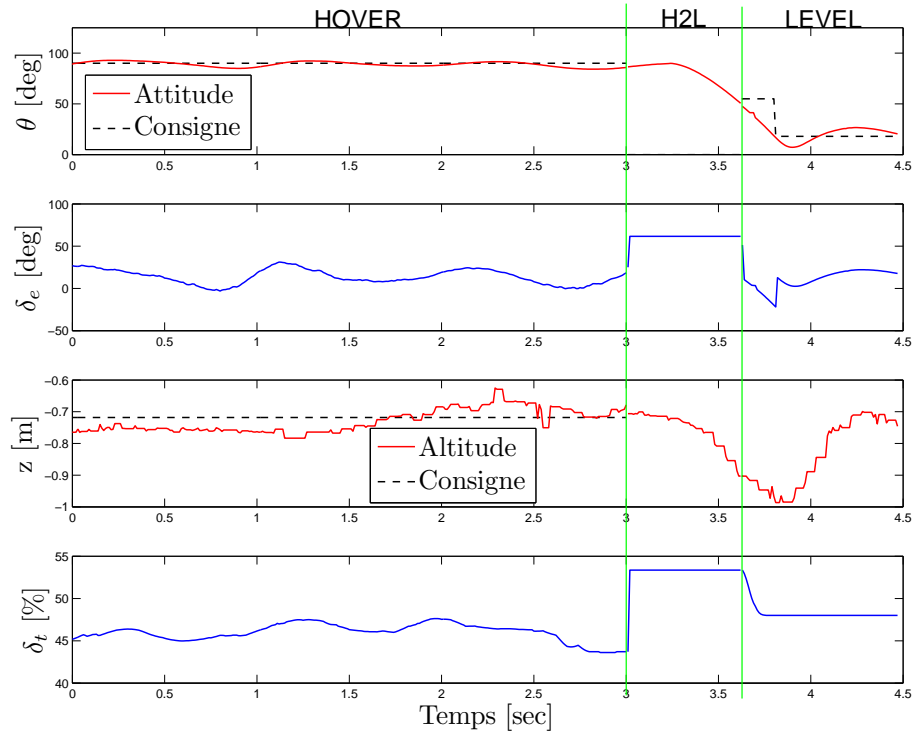
Analyse des résultats

La figure 5.20(a) montre les réponses temporelles des variables longitudinales et la figure 5.11(b) montre les variables latérales. Cet essai peut être visionné sur le CD-ROM sous le nom de `Transition_H2L.avi`.

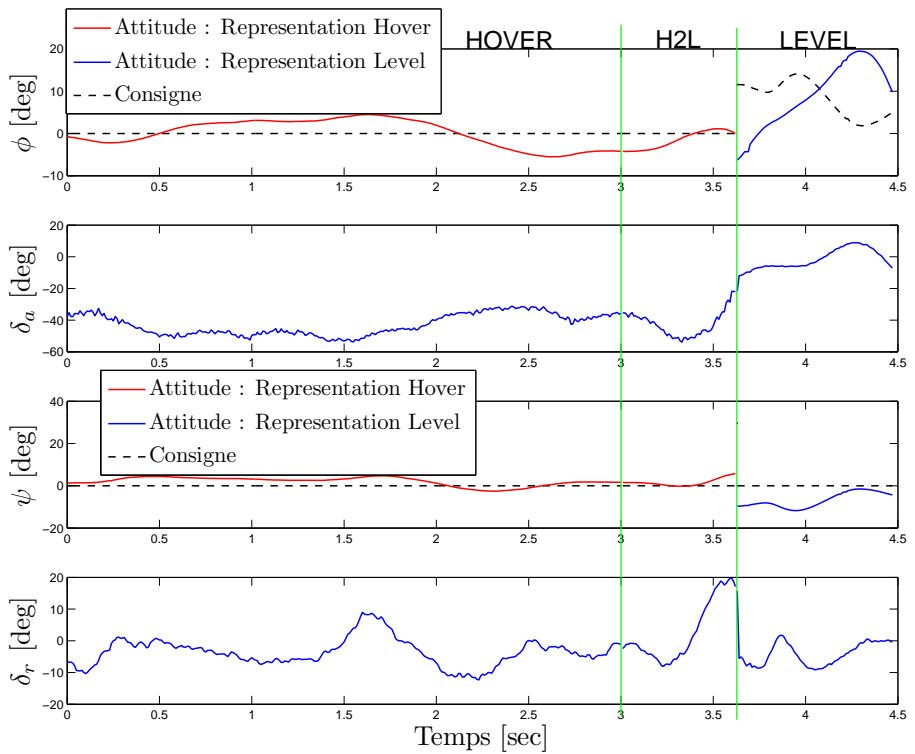
Sur la figure 5.11(a), on peut remarquer que le MiniAV est dans le mode H2L moins de 0.6 seconde. Aussi, pendant tout le mode H2L, la valeur de δ_t est égale à sa dernière valeur en mode stationnaire +10% et elle rejoint, grâce à l'algorithme 4.5, le point d'opération du mode non stationnaire (celui-ci est inférieur). Cela prend moins de 2 secondes après qu'une transition soit commandée pour que les attitudes θ et ψ soient contrôlées autour de leur consigne. On remarque aussi, durant la manoeuvre, un dépassement de 30 cm par rapport à l'altitude de référence en mode stationnaire. Ce dépassement est nécessaire pour que l'altitude en mode stationnaire revienne à l'altitude de référence du mode stationnaire, comme le montre la figure 5.11(a).

La figure 5.21 présente la position $(x - y)$ et la vitesse V_t du véhicule. Comme l'essai précédent, on remarque que le véhicule atteint une vitesse maximale V_t d'environ 5.8 m/s. La différence est que la vitesse augmente légèrement plus vite avec cet essai. Enfin, les figures 5.22 et 5.23 présentent des captures d'images d'une vidéo d'un essai H2L de ce type.

Il est possible de comparer avec Frank et al. [12] la performance de ce type d'essai H2L. Pour réaliser la manoeuvre, une réduction de 5 % de δ_r en mode stationnaire est utilisée pour aider l'avion à basculer vers l'avant. Ainsi, lors de la transition du vol stationnaire vers le vol non stationnaire, leur plateforme prend environ 2 secondes avant de contrôler θ autour de sa consigne en tangage et pour contrôler l'altitude à une consigne désirée. Il y a aussi une perte en altitude d'environ 60 cm par rapport à la valeur de référence en vol stationnaire. Aucune réponse temporelle n'est présentée pour les variables ϕ et ψ . On peut retrouver à [14] une vidéo de la plateforme de Frank et al. [12] d'un essai de ce genre. Un lien Internet est disponible sur le CD-ROM pour visionner la vidéo (`MIT_Autonomous_Airplane_Level_Flight_Testing`).



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.20 – Essai HOVER \rightarrow H2L ($\delta_t + 10\%$) \rightarrow LEVEL : variables longitudinales et latérales

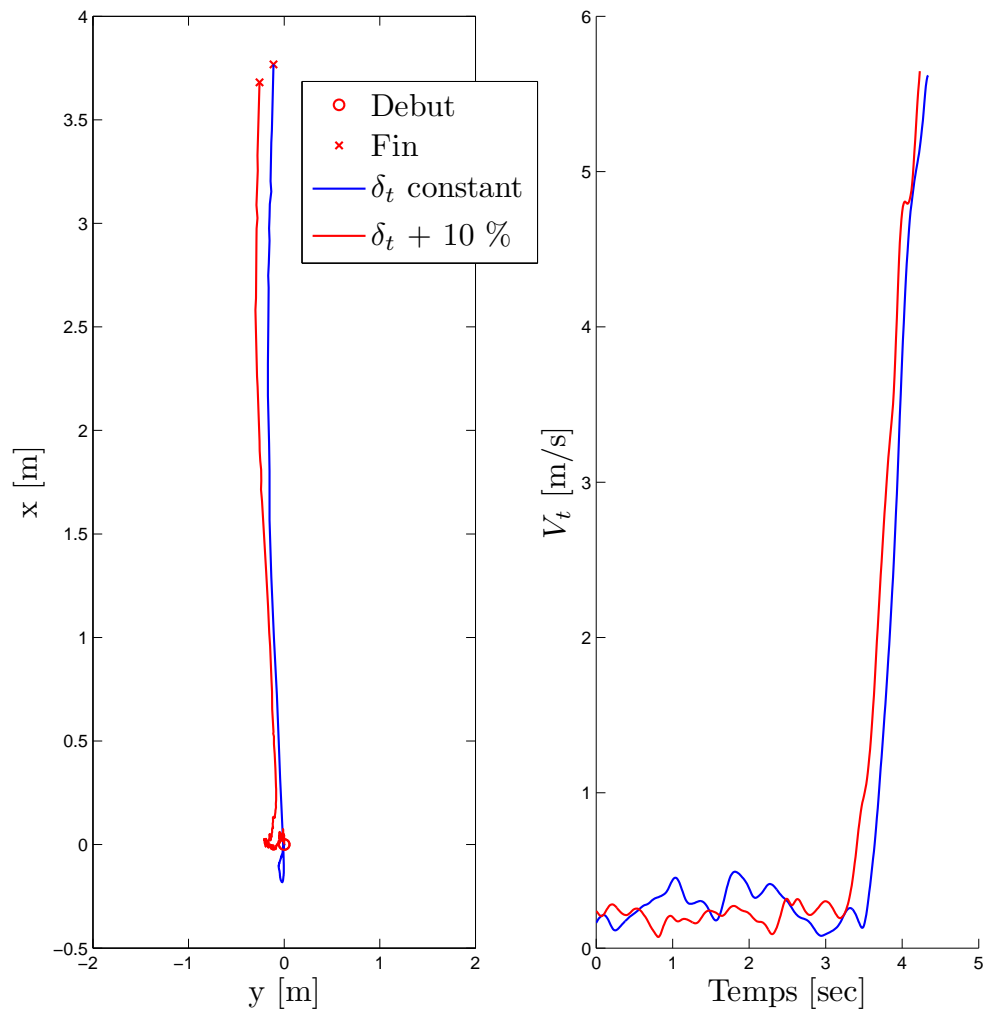


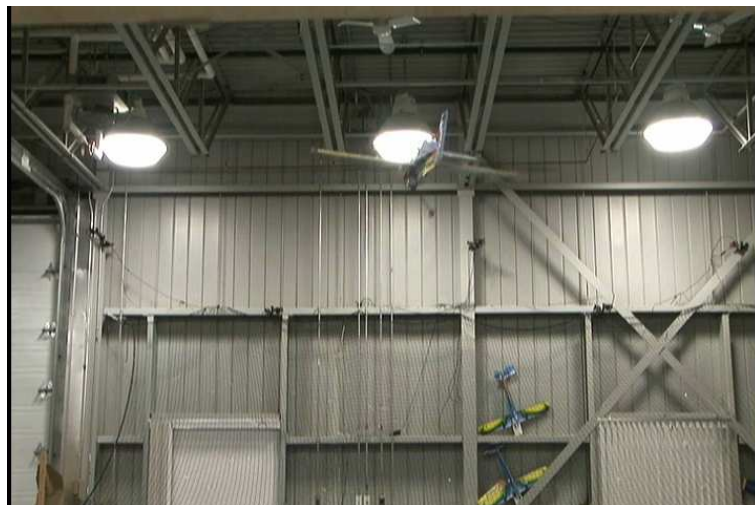
FIGURE 5.21 – Essai HOVER \rightarrow H2L ($\delta_t + 10\%$) \rightarrow LEVEL : position ($x-y$) et vitesse V_t



(a) Mode stationnaire



(b) Mode H2L



(c) Mode non stationnaire

FIGURE 5.22 – Images capturées d’une vidéo : transition H2L, vue d’en dessous de la rampe de lancement



(a) Mode stationnaire



(b) Mode H2L



(c) Mode non stationnaire

FIGURE 5.23 – Images capturées d’une vidéo : transition L2H, vue de côté

5.5 Manœuvres de transitions multiples

Les essais de cette section portent sur la réalisation d'essais de séquences de transition entre le vol non stationnaire et stationnaire. L'objectif est de tester la logique du superviseur et de réaliser des commutations rapides entre les différents modes. On peut aussi comparer les points d'opération des différentes variables manipulées pendant les différents modes. En mode stationnaire, le système de vision est utilisé pour la rétroaction de l'altitude et, en mode H2L, $\delta_t = \delta_t(4.99) + 10\%$.

5.5.1 Manœuvre : vol non stationnaire \rightarrow vol stationnaire \rightarrow vol non stationnaire

Cet essai vise à réaliser une transition complète du vol non stationnaire vers le vol stationnaire, de naviguer de façon semi-autonome et de réaliser une transition du vol stationnaire vers le vol non stationnaire. Sur la figure 5.7, on réalise la séquence suivante : LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow Navigation \rightarrow H2L \rightarrow LEVEL (① \rightarrow ② \rightarrow ③ \rightarrow ③d \rightarrow ④ \rightarrow ⑤).

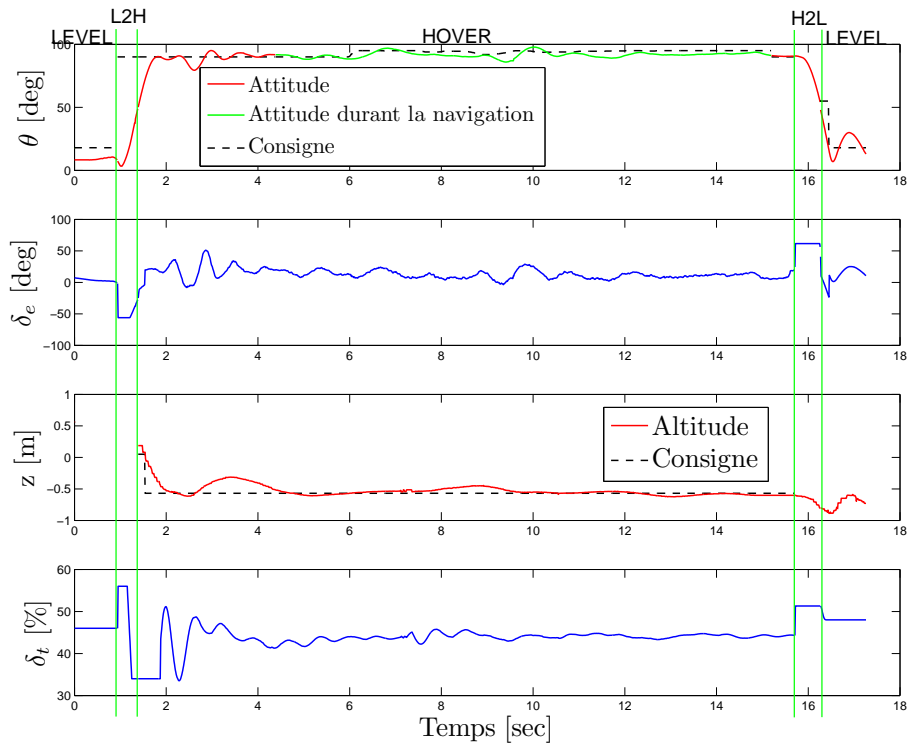
Plan de vol pour la réalisation de l'essai

L'avion est déposé sur la rampe de lancement. La rampe est placée le plus loin possible et enlignée avec la zone de capture du système de vision. L'utilisateur engage l'autopilote et l'avion décolle en mode non stationnaire à $t = 0$ avec des consigne $\bar{\theta} = 18^0$ et $\bar{\psi} = \psi_{t=0}$. À $t = 1$ sec, une transition est commandée automatiquement. Le superviseur active le mode L2H. À 50^0 , le superviseur active le mode stationnaire. Le système de vision est utilisé comme données de mesure pour la rétroaction de l'altitude. Lorsque tous les axes et l'altitude sont stabilisés, l'utilisateur utilise la manette USB pour faire naviguer le véhicule dans le but de le faire reculer jusqu'à la limite de la zone de capture du système de vision. Lorsque tous les axes et l'altitude sont stabilisés, une transition est commandée manuellement par l'utilisateur à l'aide d'un interrupteur de la manette USB. Le superviseur active le mode H2L. À 50^0 , le superviseur active le mode non stationnaire. Lorsque l'avion est rendu très près du filet de sécurité, l'utilisateur désengage l'autopilote.

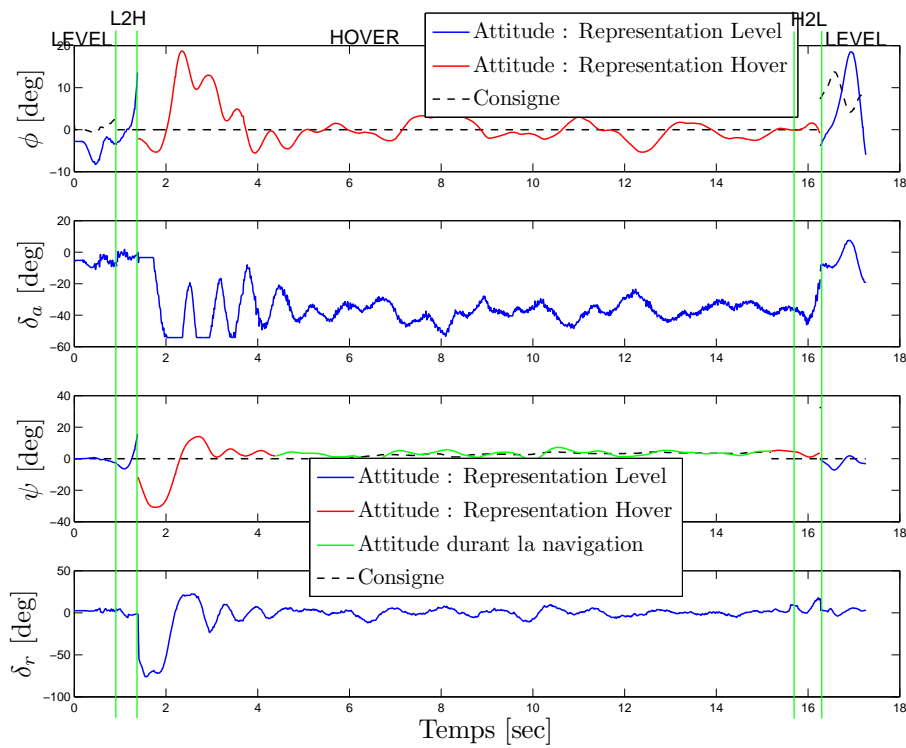
Analyse des résultats

Ce type d'essai permet d'observer une vue d'ensemble de la plage d'utilisation des variables manipulées. Tout d'abord, on peut voir que δ_e est utilisée sur toute sa plage d'utilisation. En mode non stationnaire et stationnaire, δ_e est utilisée autour de 0 degré. Durant les modes de transition L2H et H2L, δ_e est saturée pour lancer la manoeuvre de tangage. Pour δ_r , on peut visualiser les différents points d'opération, selon le mode de vol, comme le montre l'équation 4.11. On remarque aussi la différence d'utilisation des ailerons δ_a , selon le mode. Enfin, la gouverne de direction δ_r est pratiquement tout le temps utilisée autour de 0 degré, sauf lors de l'entrée en mode stationnaire. Dans cet essai, le superviseur active chacun des modes au moins une fois. Aussi, la première transition est commandée automatiquement à 1 seconde, tandis que la deuxième transition est commandée manuellement par l'utilisateur à l'aide de la manette USB. Cet essai peut être visionné sur le CD-ROM sous le nom de `Manoeuvre_LEVEL_HOVER_LEVEL.avi`.

On peut retrouver à [14] une vidéo de la plateforme de Frank et al. [12] d'un essai avec des transitions L2H et H2L. Il est à noter que la navigation de l'avion se fait en boucle fermée. Un lien Internet est disponible sur le CD-ROM pour visionner la vidéo (`MIT_Autonomous_Airplane_Level_Flight_Testing`).



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.24 – Essai LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow Navigation \rightarrow H2L \rightarrow LEVEL : variables longitudinales et latérales

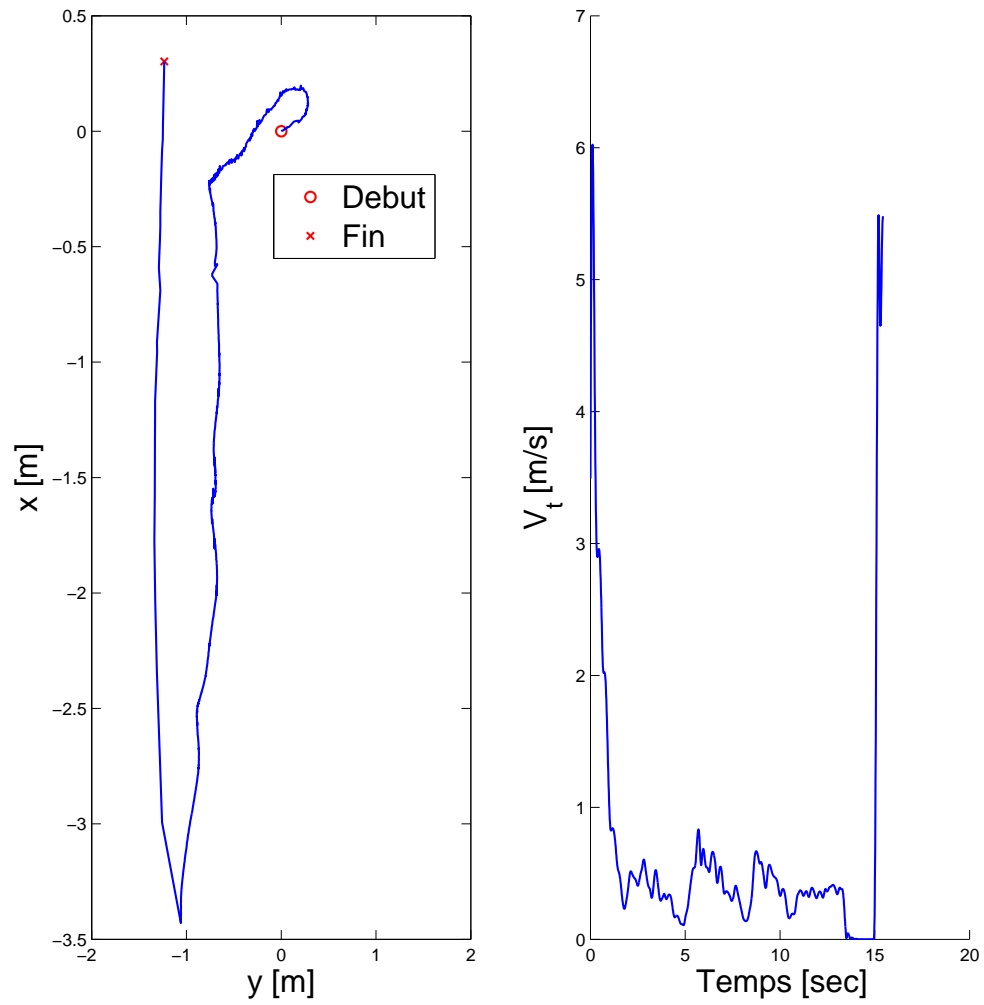


FIGURE 5.25 – Essai LEVEL \rightarrow L2H \rightarrow HOVER \rightarrow Navigation \rightarrow H2L \rightarrow LEVEL : position $(x-y)$ et vitesse V_t



(a) Mode non stationnaire



(b) Mode stationnaire 1



(c) Mode stationnaire 2



(d) Mode stationnaire 3



(e) Mode non stationnaire

FIGURE 5.26 – Capture d’images d’une manoeuvre :LEVEL → L2H → HOVER → Navigation → H2L → LEVEL

5.5.2 Manoeuvre : vol stationnaire \rightarrow vol non stationnaire \rightarrow vol stationnaire

Cet essai vise à réaliser une transition du vol stationnaire vers le vol non stationnaire, d'y rester un court laps de temps, et de réaliser une deuxième transition du vol non stationnaire vers le vol stationnaire. Sur la figure 5.7, on réalise la séquence suivante : HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER (③ \rightarrow ④ \rightarrow ⑤ \rightarrow ⑥ \rightarrow ⑦).

Plan de vol pour la réalisation de l'essai

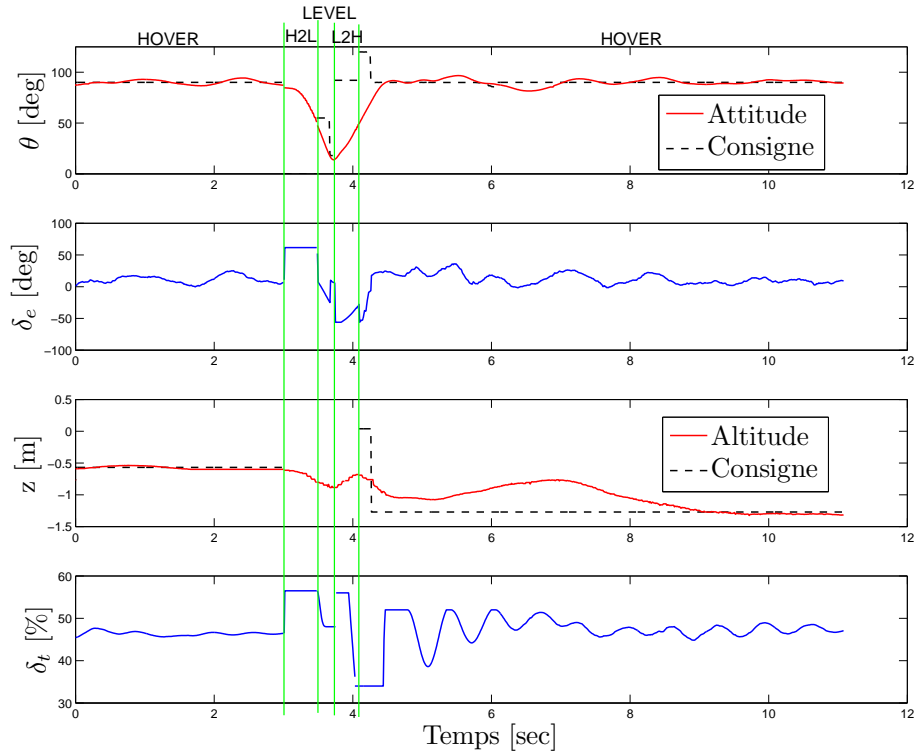
L'avion est en mode stationnaire à la limite de la zone de capture du système de vision. À $t = 0$ sec, tous les axes sont stabilisés autour de leur consigne ($\bar{\phi} = \psi_{level}$, $\bar{\theta} = 90$ et $\bar{\psi} = 0$). À $t = 3$ sec, une transition est commandée manuellement par l'utilisateur à l'aide d'un interrupteur de la manette USB. Le superviseur active le mode H2L. À 50° , le superviseur active le mode non stationnaire et $\delta_t = \delta_t(4.99) + 10\%$. Après 0.3 sec en mode non stationnaire, une transition L2H est commandée automatiquement. Le superviseur active le mode L2H. À 50° , le superviseur active le mode stationnaire. Le système de vision est utilisé comme données de mesure pour le contrôle de l'altitude. Lorsque tous les axes et l'altitude sont stabilisés, l'utilisateur peut désengager l'autopilote.

Analyse des résultats

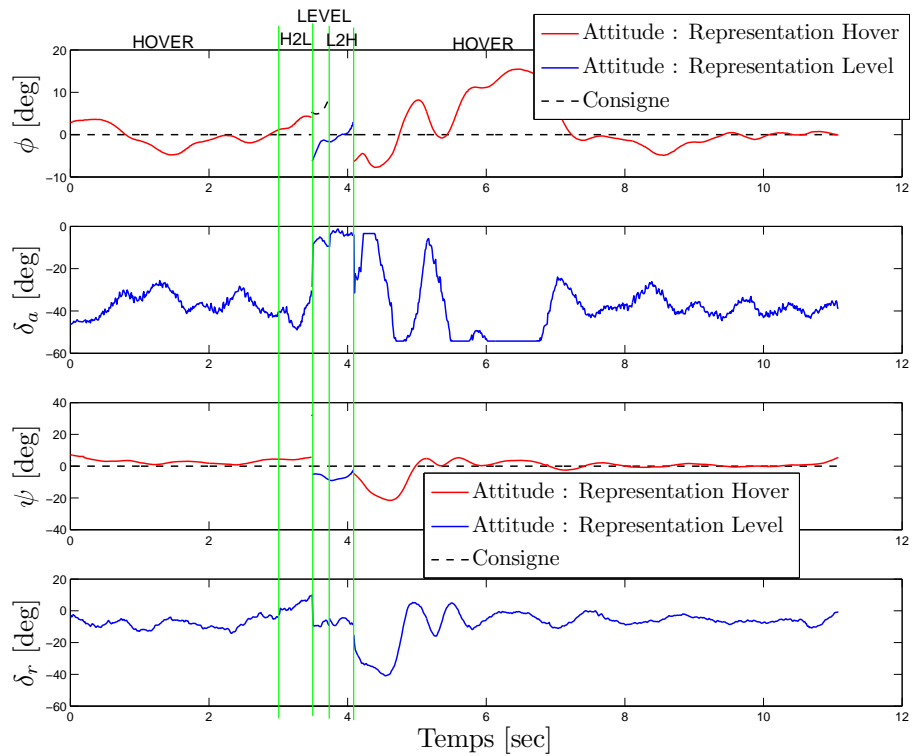
Ce type d'essai permet aussi d'observer une vue d'ensemble de la plage d'utilisation des variables manipulées. La différence observée, avec cet essai, est la commutation très rapide entre les 4 modes durant la manoeuvre. Durant les modes de transition H2L, non stationnaire et L2H, on observe que δ_e est utilisée dans toute sa plage d'utilisation, et ce, en moins de 1 seconde. Durant la manoeuvre, V_T atteint une valeur maximale d'environ 5.5 m/s. Aussi, on observe que l'avion ne dépasse pas la consigne en altitude. Par conséquent, dès que le superviseur détecte que l'altitude atteint un maximum, le PID est activé. Ainsi, on voit qu'il faut environ 4 secondes avant que l'altitude soit contrôlée à sa consigne. La première transition est commandée manuellement par l'utilisateur à l'aide de la manette USB, tandis que la deuxième transition est commandée automatiquement. Enfin, la figure 5.29 présente des captures d'images d'une vidéo d'un essai de ce type. Cet essai peut être visionné sur le CD-ROM sous le nom de `Manoeuvre_HOVER_LEVEL_HOVER.avi`.

Il est possible de comparer avec Frank et al. [12] la performance de ce type d'essai.

Pour réaliser cette manoeuvre, l'avion est tout d'abord positionné en mode stationnaire, une transition est commandée et l'avion réalise une transition H2L. Dès que l'angle de tangage atteint 30 degrés, c'est-à-dire dès que l'avion débute le vol en palier, une transition vers le mode stationnaire est commandée immédiatement. Durant la manoeuvre H2L, une perte en altitude d'environ 60 cm est observée et il y a une oscillation d'environ 1.4 mètre avant que l'altitude soit contrôlée à l'altitude de référence du départ. Aussi, une vitesse maximale V_t de 4.5 m/s est atteinte durant la manoeuvre et il faut plus de 5 secondes avant que V_t soit ramenée autour de 0 m/s. Le temps de réponse pour θ est semblable à ce qui a été observé pour l'essai LEVEL \rightarrow L2H \rightarrow HOVER (z en boucle fermée avec le système de vision). On peut retrouver à [15] une vidéo de la plateforme de Frank et al. [12] effectuant un essai de ce genre. On remarque sur la vidéo la façon dont la transition est réalisée : pour diminuer l'angle θ , l'avion tombe de côté. L'avion prend aussi plusieurs secondes avant d'être contrôlé en altitude et pour que les variables latérales se stabilisent autour de leurs consignes. Un lien Internet est disponible sur le CD-ROM pour visionner la vidéo (MIT_Autonomous_Airplane_Transition_(Hover-Level_Flight-Hover)_Tests).



(a) Variables longitudinales : θ , δ_e , z et δ_t



(b) Variables latérales : ϕ , δ_a , ψ et δ_r

FIGURE 5.27 – Essai HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER : variables longitudinales et latérales

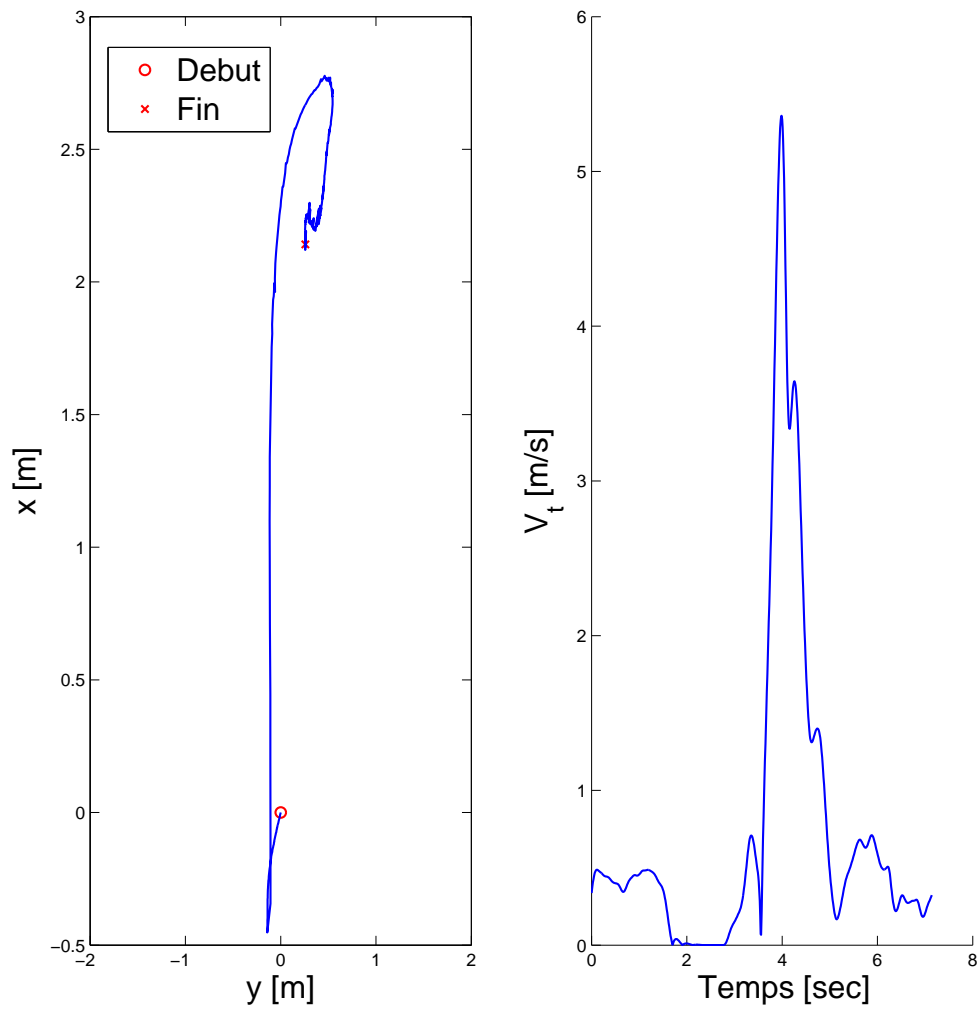


FIGURE 5.28 – Essai HOVER \rightarrow H2L \rightarrow LEVEL \rightarrow L2H \rightarrow HOVER : position $(x - y)$ et vitesse V_t

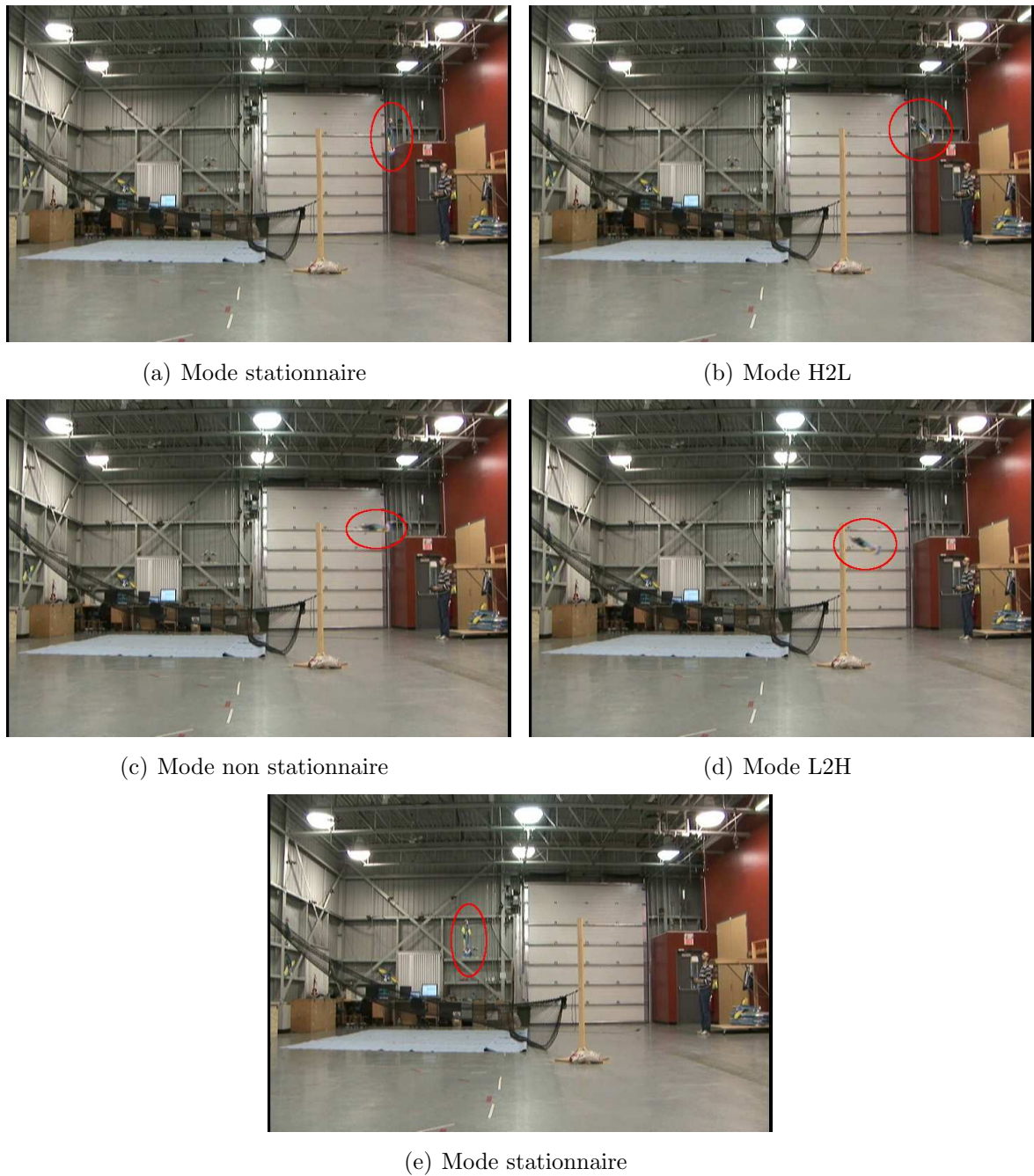


FIGURE 5.29 – Capture d’images d’une manoeuvre HOVER → H2L → LEVEL → L2H → HOVER

5.6 Conclusion

Ce chapitre a présenté plusieurs essais de vol. Pour chacun des essais, les réponses temporelles des différentes variables et consignes pertinentes ont été présentées et analysées. Les résultats des tests en vol sur le banc de test au RDDC Valcartier ont montré que les transitions L2H et H2L peuvent être réalisées en moins de 3 secondes en utilisant la stratégie développée dans les chapitres 3 et 4. Ces performances sont supérieures à celles obtenues par Green [17, 18] et Frank et al. [12], qui ont utilisé des plateformes semblables au type utilisé dans ces travaux.

Conclusion

Ce mémoire décrit un contrôleur du mode non stationnaire, un contrôleur du mode stationnaire ainsi que des stratégies de transition entre le vol non stationnaire et le vol stationnaire d'un avion miniature à ailes fixes. Ces transitions sont réalisées grâce à un superviseur qui est basé sur la logique du système qui permet de réaliser une commutation sans-à-coup entre les différents modes de vol.

La plateforme et la problématique de la représentation mathématique sont présentées et analysées. La représentation de l'attitude, à l'aide des angles d'Euler et des erreurs angulaires calculées avec le quaternion d'erreur, est présentée. Des exemples sont détaillés pour illustrer la différence entre les deux représentations.

Le modèle de l'avion est représenté sous forme de fonctions de transfert. Ces fonctions de transfert sont des modèles linéaires simples représentant les principaux comportements aérodynamiques d'un avion. Elles ont été déterminées par une linéarisation d'un corps rigide à 6 degrés de liberté. Les étapes du processus d'identification (planification, réalisation de l'expérience, traitement des données, choix des structures, estimation et validation) sont décrites. Le réglage des contrôleurs est réalisé avec les fonctions de transfert identifiées en utilisant une méthode précise basée sur l'analyse fréquentielle. Les contrôleurs sont de simples P,PI et PID.

Une approche systématique, basée sur la connaissance a priori des comportements de la plateforme dans les deux principaux modes de vol, est choisie pour développer le superviseur basé sur la logique. Une structure de commutation simple entre les contrôleurs est proposée. Cette approche permet d'éviter d'avoir à identifier un modèle non-linéaire complet et de concevoir une loi de commande non-linéaire, plus complexe, capable de gérer l'ensemble de l'enveloppe de vol.

Plusieurs essais ont été réalisés. Les résultats de tests en vol expérimentaux ont montré que les transitions L2H et H2L peuvent être réalisées en moins de 3 secondes en utilisant la stratégie développée dans ce mémoire. Ces performances sont supérieures à

celles obtenues par Green [17, 18] et Frank et al. [12], qui ont utilisé des plateformes semblables au type de plateformes utilisées dans ces travaux.

Il a donc été démontré que l'approche systématique développée dans ce mémoire pour réaliser des manoeuvres de transition pour de petites plateformes est avantageuse. Cependant, il n'est pas certain que cette méthode puisse être appliquée pour de plus grosses plateformes dont les manoeuvres de transitions sont réalisées sur plusieurs secondes et sur une grande distance. La stratégie développée dans ce mémoire repose sur la possibilité d'une commutation très rapide entre deux modes de vol.

Évidemment, les travaux réalisés dans ce mémoire et ceux réalisés par Bilodeau et al. [2, 3] ne sont que la première phase du projet. En effet, beaucoup de chemin reste à faire.

- Idéalement, un modèle non-linéaire ou linéaire par morceaux, doit être identifié afin d'avoir une meilleure connaissance du comportement aérodynamique. Des essais en soufflerie seraient probablement nécessaires. Par conséquent, avec un modèle non-linéaire complet, une loi de commande non-linéaire, capable de contrôler le système dans l'enveloppe de vol complète, pourrait être développée.
- Une représentation unifiée de l'attitude dans le système de commande est souhaitable. Comme les angles d'Euler possèdent des singularités à $\theta = \pm 90^\circ$ (section 2.2), les quaternions peuvent être utilisés pour représenter l'attitude pour toutes les conditions de vol. Les contrôleurs, avec des consignes en quaternion, devraient alors être développés.
- Pour la réalisation de la transition L2H, une connaissance précise de l'altitude est requise (voir section 4.4.4). Des filtres de Kalman peuvent être intégrés pour obtenir une solution d'estimation de l'altitude à long terme. Avec un IMU, plusieurs groupes de recherche [20, 25, 39, 40] ont développé des algorithmes de fusion à l'aide de modèles aérodynamiques, de GPS, de magnétomètres ou de capteurs barométriques. D'autres méthodes, basées sur l'analyse d'images prises par une caméra [28, 42], peuvent aussi être considérées pour estimer l'altitude et la position d'un véhicule aérien. Les senseurs ultrasoniques [18, 26] peuvent aussi être utilisés pour des missions en basse altitude en terrain non-accidenté.
- Des modifications aux algorithmes en mode stationnaire seraient souhaitables [2, 3]. En effet, comme il a été mentionné dans la section 5.2.1, la commande des ailerons δ_a est limitée entre 0 degré (ailerons plat) et -55 degrés, ce qui limite transitoirement l'action des ailerons sur le système.

- Évidemment, l'intégration d'un algorithme de navigation semi-autonome fonctionnel et la réalisation d'essais dans un environnement extérieur (avec perturbations atmosphériques) seraient souhaitables pour la suite du projet.

Bibliographie

- [1] Anderson P. and R.H. Stone, *Predictive Guidance and Control for a Tail-Sitting Unmanned Aerial Vehicle*, Information, Decision and Control, 2007. IDC '07 (2007), pp. 148-153.
- [2] Bilodeau, P.-R., É. Poulin, F. Wong, E. Gagnon and A. Desbiens, *Control of a Hovering Mini Fixed Wing Aerial Vehicle*, AIAA Conference on Guidance, Navigation and Control, Chicago, 2009, AIAA 2009-5794
- [3] Bilodeau, P.-R., *Commande d'un mini véhicule aérien sans pilote en vol stationnaire pour l'opération en milieu restreint*, Mémoire accepté à l'Université Laval en vue de l'obtention de la maîtrise ès sciences, Département de génie électrique et de génie informatique, 2009.
- [4] Blakelock J.H., *Automatic Control of Aircraft and Missiles*, Second Edition, Wiley, 1991.
- [5] Chu. D. and J. Sprinkle, R. Randall, and S. Shkarayev, *Automatic Control of VTOL Micro Air Vehicle During Automatic Control of VTOL Micro Air Vehicle During*, 2002 Florida Conference on Recent Advances in Robotics, AIAA Conference on Guidance, Navigation and Control, Chicago, 2009, AIAA 2009-5875
- [6] Colgren. R. and L. Holly, *Flight Dynamic Requirements for UAVs - Do They Really Exist ?*, 2002 Florida Conference on Recent Advances in Robotics, AIAA Conference on Guidance, Navigation and Control, Chicago, 2009, AIAA 2009-6323
- [7] Desbiens, A., *Identification des systèmes*, Notes de cours - Hiver 2008, Département de génie électrique et de génie informatique, Université Laval, 2008
- [8] Desbiens, A., *Systèmes et commande linéaires*, Notes de cours - Hiver 2008, Département de génie électrique et de génie informatique, Université Laval, 2008
- [9] Etkin, B. and L. D. Reid, *Dynamics of Flight, Stability and Control*, John Wiley & Sons, Inc., 3rd ed., January 1996
- [10] Fong, W.T., S. K. Ong and A. Y. C. Nee, *Methods for in-field user calibration of an inertial measurement unit without external equipment*, Meas. Sci. Technol. 19 (2008) 085202 (11pp)

- [11] Foster T.M., *Dynamic Stability and Handling Qualities of Small Unmanned-Aerial-Vehicle*, Thèse soumise à la Faculté de Brigham Young University en vue de l'obtention de la maîtrise ès sciences, Department of Mechanical Engineering, 2005
- [12] Frank, A., J. Mcgrew, M. Valenti, D. Levine and J. P. How, *Hover, Transition, and level Flight Control design for a Single-propeller Indoor Airplane*, AIAA GNC, Hilton Head, United states, 2007, AIAA 2007-6318.
- [13] Frank, A., J. Mcgrew, M. Valenti, D. Levine and J. P. How, *Autonomous Airplane Ground Take-off to Hover*, Autonomous UAV Aerobatics Project - Videos, Consulté le 12/12/2009,
<http://aerobatics.mit.edu/videos.html>
- [14] Frank, A., J. Mcgrew, M. Valenti, D. Levine and J. P. How, *Autonomous Airplane Level Flight Testing*, Autonomous UAV Aerobatics Project - Videos, Consulté le 12/12/2009,
<http://aerobatics.mit.edu/videos.html>
- [15] Frank, A., J. Mcgrew, M. Valenti, D. Levine and J. P. How, *Autonomous Airplane Transition (Hover-Level Flight-Hover) Tests*, Autonomous UAV Aerobatics Project - Videos, Consulté le 12/12/2009,
<http://aerobatics.mit.edu/videos.html>
- [16] Goodwin G.C., S.F. Graebe and M.E. Salgado, *Control System Design*, Prentice-Hall, Upper Saddle River, NJ, USA, 2001.
- [17] Green, W.E. and P.Y. Oh, *A MAV That Flies Like an airplane and Hovers Like a Helicopter*, International Conference on Advanced Intelligent Mechatronics, Monterey, United states, 2005, pp 694-698.
- [18] Green, W.E., *A Multimodal Micro Air Vehicle for Autonomous Flight in Near-Earth Environments*, Thèse soumise à la Faculté de Drexel University en vue de l'obtention du doctorat, Department of Mechanical Engineering, 2007.
- [19] Green, W.E, *Autonomous Cruise-to-Hover Transition*, Consulté le 12/12/2009,
http://prism2.mem.drexel.edu/~billgreen/Videos/3D_flying/autoCTH/autoCTH012006.mpg
- [20] Huang Z. J. and J. C. Fang , *Integration of MEMS Inertial Sensor-Based GNC of a UAV*, International Journal of Information Technology, Vol. 11 No. 10, 2005, pp 123-132
- [21] Hui K., L. Auriti, B. Hachem, and J. Werk, *Flight Characteristics and Aerodynamic Model Development of Micro Unmanned Aerial Vehicles : Flatana and U-CAN-DO*, 2008, NRC-CNRC Flight Research Laboratory, Report No. : LTR-FR-290
- [22] Jategaonkar R.V., *Flighth Vehicule System Identification : A Time Domain Methodology*, American Institute of Aeronautics and Astronautics, Progress in Astronautics and Aeronautics, 2006

- [23] Johnson Eric N, A. Wu, J. C. Neidhoefer, S. K. Kannan and M.A. Turbe , *Flight-Test Results of Autonomous Airplane Transitions Between Steady-Level and Hovering Flight*, Journal of Guidance, Control, and Dynamics, Vol. 31, No. 2, March-April 2008, pp 358-370
- [24] Johnson Eric N, A. Wu, J. C. Neidhoefer, S. K. Kannan and M.A. Turbe , *Flight Test Results of Autonomous Fixed-Wing UAV Transitions to and from Stationary Hover*, AIAA Guidance, Navigation, and Control Conference and Exhibit, 2006, Keystone, Colorado, AIAA 2006-6775
- [25] Kim J.-H., S. Wishart. and S. Sukkarieh, *Real-time Navigation, Guidance, and Control of a UAV using Low-cost Sensors*, Field and Service Robotics, STAR 24, 2006, pp. 299-309
- [26] Kita K., A. Konno, and M. Uchiyama, *Hovering Control of a Tail-Sitter VTOL Aerial Robot*, Journal of Robotics and Mechatronics, Vol.21 No.2, 2009, pp 277-283
- [27] Knoebel, N.B., *Adaptative Quaternion Control of a Miniature Tailsitter UAV*, Mémoire soumis à la Faculté de Brigham Young University en vue de l'obtention de la maîtrise ès sciences, Department of Mechanical Engineering, 2007
- [28] Millet P. T. , B. B. Ready and T. W. McLain, *Vision-Based Precision Landings of a Tailsitter UAV*, AIAA Guidance, Navigation, and Control Conference, 10 - 13 August 2009, Chicago, Illinois, AIAA 2009-5680
- [29] Morelli E. and V. Klein, *Aircraft System Identification : Theory and Practice*, American Institute of Aeronautics and Astronautics, AIAA Education Series, 2006.
- [30] Myrand-Lapierre, V., A. Desbiens, E. Gagnon, F. Wong and E. Poulin, *Transitions Between Level Flight and Hovering for a Fixed-Wing Mini Aerial Vehicle*, 2010 American Control Conference, Baltimore, Maryland, USA (soumis pour publication)
- [31] Osborne S.R. , *Transition Between Hover and Level Flight for a Tailsitter UAV*, Thèse soumise à la Faculté de Brigham Young University en vue de l'obtention de la maîtrise ès sciences, Department of Mechanical Engineering, 2007.
- [32] Pamadi B. N. , *Performance, Stability, Dynamics, and Control of Airplanes*, American Institute of Aeronautics and Astronautics Inc, Second Edition, 2004
- [33] Poulin É. and A. Pomerleau, *A Unified Method Based on a Maximum peak Resonance Specification*, IEE Proc-Control Theory Appl., Vol. 144, No. 6, November 1997, pp 566-574.
- [34] Pratt R.W, *Flight Control Systems*, Progress in Astronautics and Aeronautics, Volume 184, 2000.
- [35] Stevens B.L. and F.L. Lewis , *Aircraft Control and Simulation*, 2nd edition, John Wiley & sons Inc, New Jersey, 2003.
- [36] Stone R.H., *Control Architecture for a Tail-Sitter Unmanned Air Vehicle*, 5th Asian Control Conference, 2004, Vol.2, pp 736- 744

- [37] Stone R.H. and G. Clarke, *Optimization of Transition Manoeuvres for a Tail-Sitter Unmanned Air Vehicle (UAV)*, Australian International Aerospace Congress, Paper 105, Canberra, March 2001.
- [38] Stone R.H., P. Anderson and P. Gibbens, *Predictive Control for a Tail-Sitter UAV : Final Report (Part 2)*, AOARD, UNIT 45002, APO, AP, 96337-5002.
- [39] Walchko. K. J. and P. A.C. Mason, *Inertial Navigation*, 2002 Florida Conference on Recent Advances in Robotics
- [40] Wendel J., O. Meister, C. Schlaile, G. F. Trommer, *An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter*, 2006 Elsevier, Aerospace Science and Technology 10 (2006) 527-533
- [41] Wikipedia, *Yaw, pitch, and roll*, Consulté le 12/12/2009, http://en.wikipedia.org/wiki/Yaw,_pitch,_and_roll
- [42] Wu A. D., E. N. Johnson and A. A. Proctorz, *Vision-Aided Inertial Navigation for Flight Control*, AIAA Guidance, Navigation, and Control Conference and Exhibit, 15 - 18 August 2005, San Francisco, California, AIAA 2005-5998

Annexe A

Identification

A.1 Identification et validation

Identification et validation de $\frac{\theta(s)}{\delta_e(s)}$

Il a été montré que $\frac{\theta(s)}{\delta_e(s)}$ pouvait être approximé par :

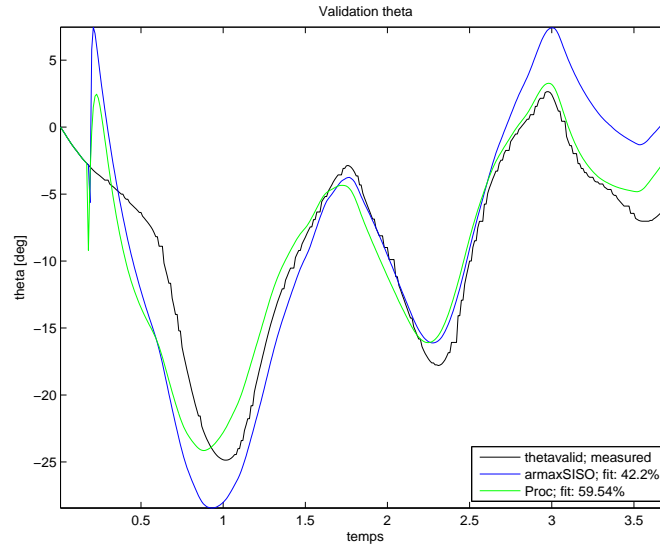
$$\frac{q(s)}{\delta_e(s)} \approx \frac{k_q(s + 1/T_{\theta_1}) e^{-T_d s}}{(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} \quad (\text{A.1})$$

$$\frac{\theta(s)}{\delta_e(s)} \approx \frac{q(s)}{s\delta_e(s)} \quad (\text{A.2})$$

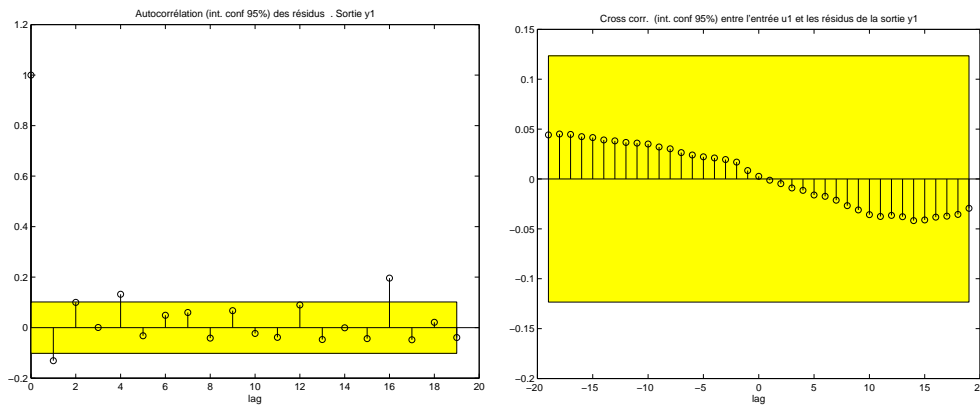
La figure A.1(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.1(b) et A.1(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_e(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.1(b) et A.1(c). Le modèle choisi est le suivant :

```
Process model with transfer function
          1+Tz*s
G(s) = K * ----- * exp(-Td*s)
          s(1+2*Zeta*Tw*s+(Tw*s)^2)
```

```
with   K = -2.1147+-0.10201
        Tw = 0.090545+-0.0021867
        Zeta = 0.3+-0.033195
```



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus

(c) Corrélation entre $\delta_e(k)$ et $\varepsilon(k+\tau)$ FIGURE A.1 – Identification et validation de $\frac{\theta(s)}{\delta_e(s)}$

$$T_d = 0.17+-0$$

$$T_z = 0.22261+-0.021522$$

An additive ARMA disturbance model has been estimated:

$$y = G u + (C/D)e$$

with

$$C(s) = s + 102.4 (+-2.441)$$

$$D(s) = s + 0.3262 (+-0.1289)$$

Avec :

Eigenvalue	Damping	Freq. (rad/s)
0.00e+000	-1.00e+000	0.00e+000
-3.31e+000 + 1.05e+001i	3.00e-001	1.10e+001
-3.31e+000 - 1.05e+001i	3.00e-001	1.10e+001

Identification et validation de $\frac{p(s)}{\delta_a(s)}$

Il a été montré que $\frac{p(s)}{\delta_a(s)}$ pouvait être approximé par :

$$\frac{p(s)}{\delta_a(s)} = \frac{k_p e^{-T_d s}}{(s + 1/T_r)} \quad (\text{A.3})$$

La figure A.2(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.2(b) et A.2(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_e(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.2(b) et A.2(c). Le modèle choisi est le suivant :

Process model with transfer function

$$G(s) = \frac{K}{1 + T_{p1}s} * \exp(-T_d s)$$

with $K = -4.6078 \pm 0.32016$
 $T_{p1} = 0.047098 \pm 0.0072297$
 $T_d = 0.17 \pm 0$

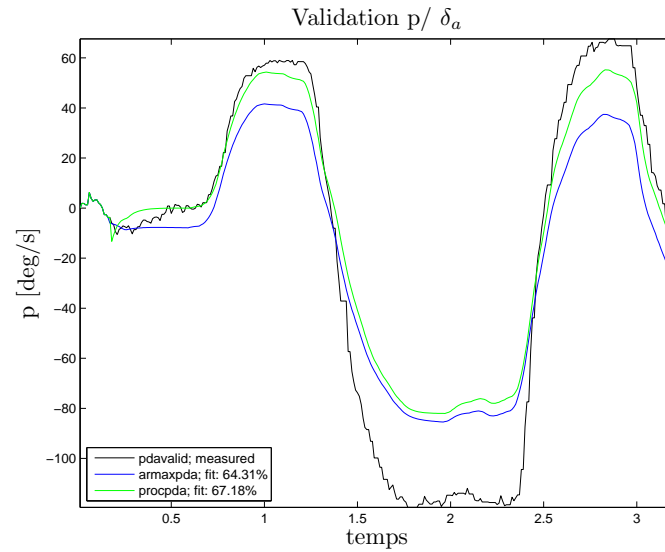
An additive ARMA disturbance model has been estimated:

$$y = G u + (C/D)e$$

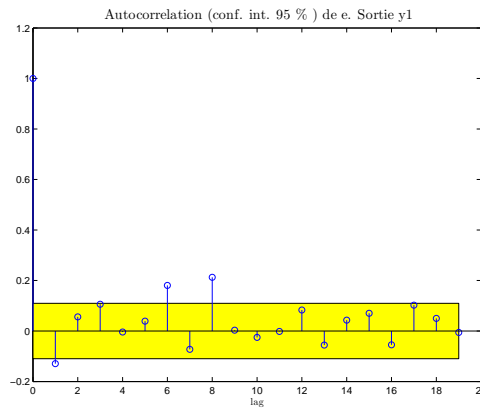
with
 $C(s) = s + 91.27 \quad (+ - 3.307)$
 $D(s) = s + 1.063e-007 \quad (+ - 0.2753)$

Avec :

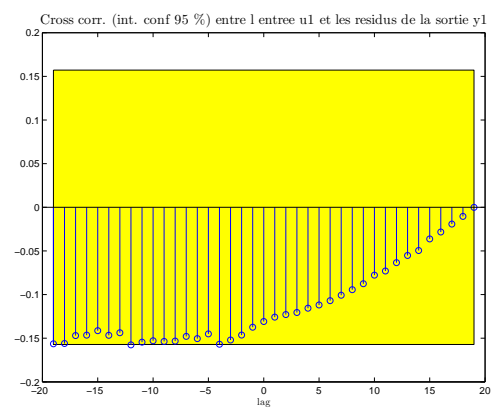
Eigenvalue	Damping	Freq. (rad/s)
-2.12e+001	1.00e+000	2.12e+001



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus



(c) Corrélation entre $\delta_e(k)$ et $\varepsilon(k + \tau)$

FIGURE A.2 – Identification et validation de $\frac{p(s)}{\delta_a(s)}$

Identification et validation de $\frac{p(s)}{\delta_r(s)}$

Il a été montré que $\frac{p(s)}{\delta_r(s)}$ pouvait être approximé par :

$$\frac{p(s)}{\delta_r(s)} \approx \frac{k_p s e^{-T_d s}}{(s + 1/T_s)(s + 1/T_{r2})} \quad (\text{A.4})$$

La figure A.3(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.3(b) et A.3(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_r(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.3(b) et A.3(c). Le modèle choisi est le suivant :

Process model with transfer function

$$G(s) = K * \frac{1 + T_z * s}{(1 + T_{p1} * s)(1 + T_{p2} * s)} * \exp(-T_d * s)$$

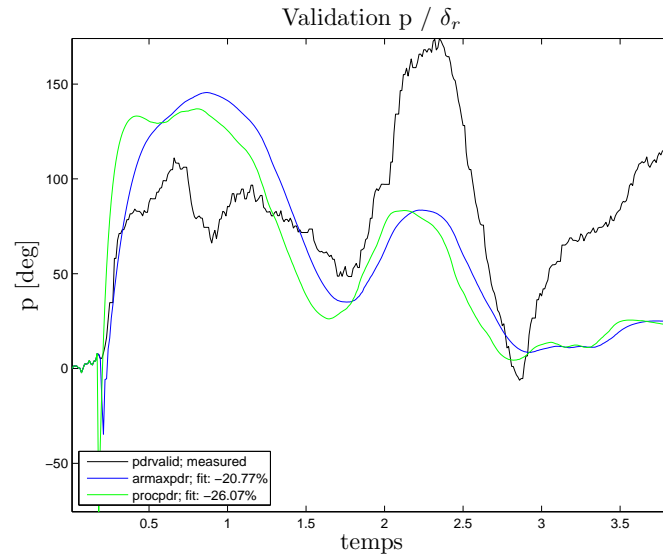
with K = 0.18032+-1.3954
 Tp1 = 0.055383+-0.014305
 Tp2 = 1.7184+-1.013
 Td = 0.17+-0
 Tz = 181+-15.42

An additive ARMA disturbance model has been estimated:

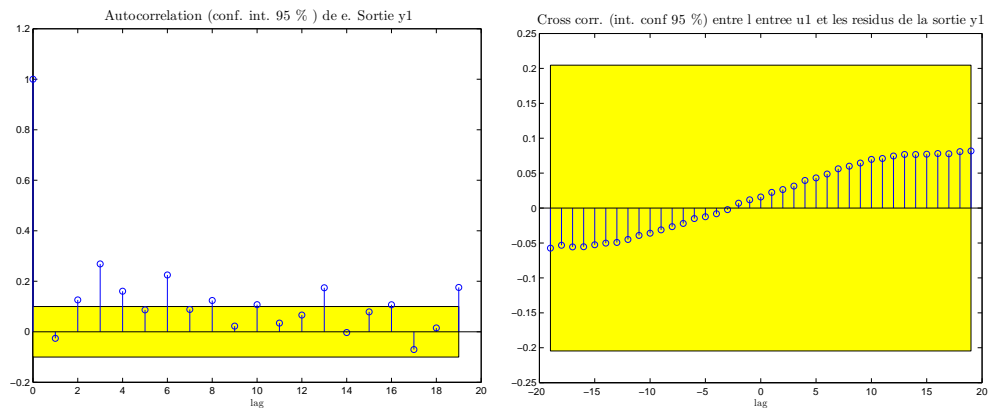
 y = G u + (C/D)e
 with
 C(s) = s + 120.7 (+-3.267)
 D(s) = s + 1.124 (+-0.6862)

Avec :

Eigenvalue	Damping	Freq. (rad/s)
-5.82e-001	1.00e+000	5.82e-001
-1.81e+001	1.00e+000	1.81e+001



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus

(c) Corrélation entre $\delta_a(k)$ et $\varepsilon(k + \tau)$

FIGURE A.3 – Identification et validation de $\frac{p(s)}{\delta_r(s)}$

Identification et validation de $\frac{r(s)}{\delta_a(s)}$

Il a été montré que $\frac{r(s)}{\delta_a(s)}$ pouvait être approximé par :

$$\frac{r(s)}{\delta_a(s)} \approx \frac{k_v(s + 1/T_\Psi) e^{-T_d s}}{(s + 1/T_s)(s + 1/T_{r2})} \quad (\text{A.5})$$

La figure A.4(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.4(b) et A.4(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_a(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.4(b) et A.4(c). Le modèle choisi est le suivant :

Process model with transfer function

$$G(s) = K * \frac{1 + T_z * s}{(1 + T_{p1} * s)(1 + T_{p2} * s)} * \exp(-T_d * s)$$

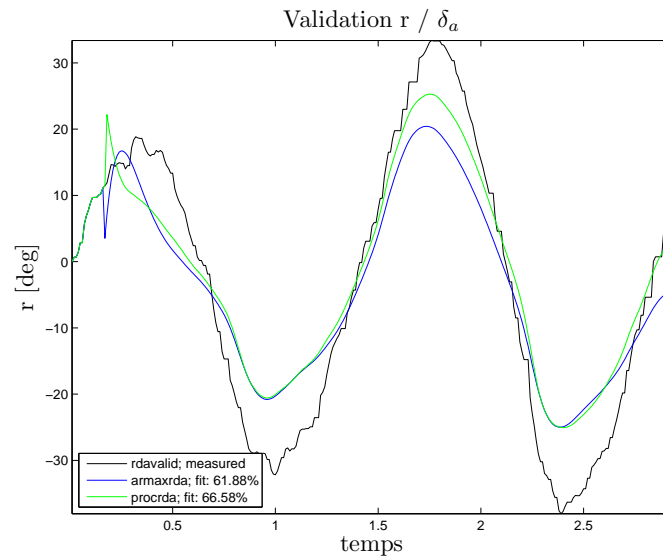
with K = -4.05+-1.3348
 Tp1 = 0.81257+-0.2756
 Tp2 = 0.077307+-0.022118
 Td = 0.17+-0
 Tz = -0.1791+-0.038396

An additive ARMA disturbance model has been estimated:

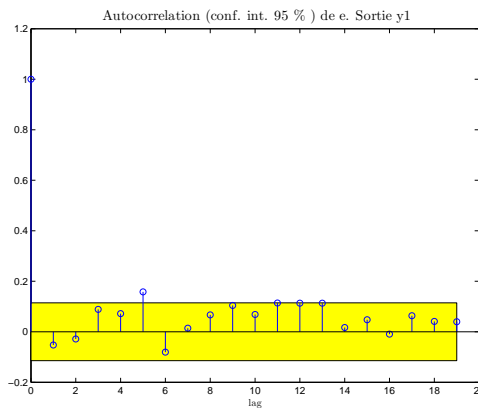
 y = G u + (C/D)e
 with
 C(s) = s + 100.1 (+-3.759)
 D(s) = s - 0.3745 (+-0.2079)

Avec :

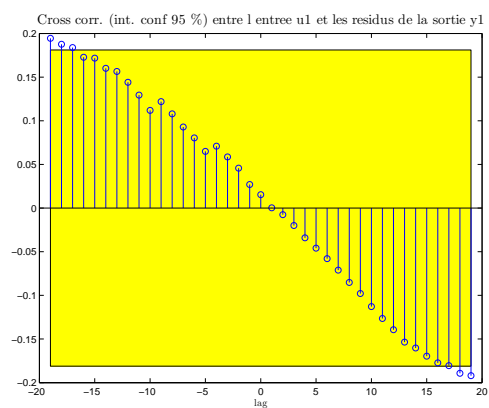
Eigenvalue	Damping	Freq. (rad/s)
-1.23e+000	1.00e+000	1.23e+000
-1.29e+001	1.00e+000	1.29e+001



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus



(c) Corrélation entre $\delta_a(k)$ et $\varepsilon(k + \tau)$

FIGURE A.4 – Identification et validation de $\frac{r(s)}{\delta_a(s)}$

Identification et validation de $\frac{r(s)}{\delta_r(s)}$

Il a été montré que $\frac{r(s)}{\delta_r(s)}$ pouvait être approximé par :

$$\frac{r(s)}{\delta_r(s)} \approx \frac{k_v(s + 1/T_\Psi) e^{-T_d s}}{(s + 1/T_s)(s + 1/T_{r2})} \quad (\text{A.6})$$

$$(\text{A.7})$$

La figure A.5(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.5(b) et A.5(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_r(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.5(b) et A.5(c). Le modèle choisi est le suivant :

Process model with transfer function

$$G(s) = K * \frac{1 + T_z * s}{(1 + T_{p1} * s)(1 + T_{p2} * s)} * \exp(-T_d * s)$$

with $K = 0.37704 + -1.2474$
 $T_{p1} = 0.052085 + -0.0097371$
 $T_{p2} = 1.501 + -0.86231$
 $T_d = 0.17 + -0$
 $T_z = 10 + -35.856$

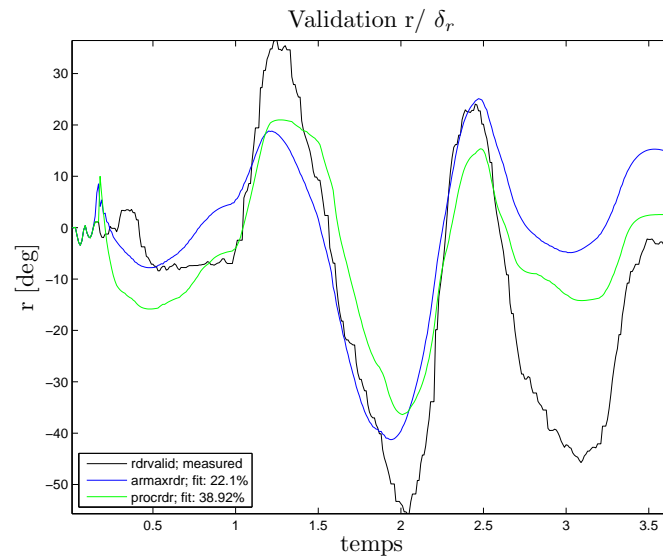
An additive ARMA disturbance model has been estimated:

$$y = G u + (C/D)e$$

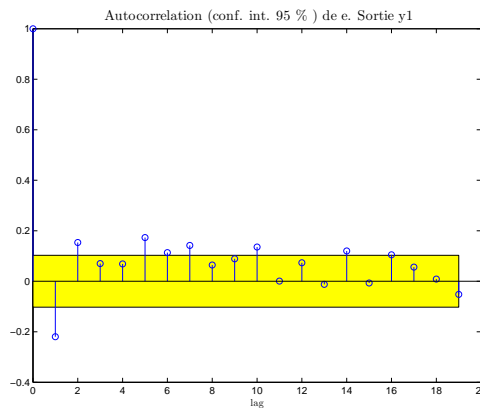
with
 $C(s) = s + 121.7 (+ -2.996)$
 $D(s) = s + 1.027 (+ -0.6496)$

Avec :

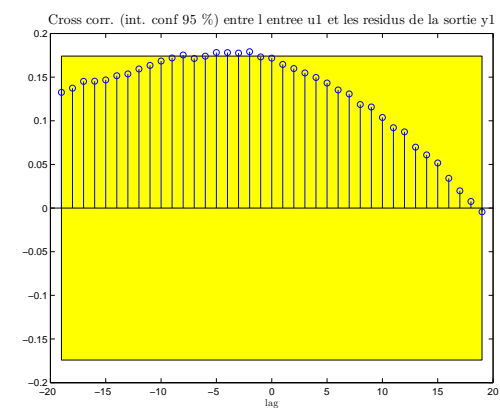
Eigenvalue	Damping	Freq. (rad/s)
-6.67e-001	1.00e+000	6.67e-001
-1.92e+001	1.00e+000	1.92e+001



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus



(c) Corrélation entre $\delta_a(k)$ et $\varepsilon(k + \tau)$

FIGURE A.5 – Identification et validation de $\frac{r(s)}{\delta_r(s)}$

Identification et validation de $\frac{\phi(s)}{\delta_a(s)}$

Il a été montré que $\frac{\phi(s)}{\delta_a(s)}$ pouvait être approximé par :

$$\frac{\phi(s)}{\delta_a(s)} \approx \frac{p(s)}{s \delta_a(s)} \quad (\text{A.8})$$

La figure A.6(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.6(b) et A.6(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_a(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.6(b) et A.6(c). Le modèle choisi est le suivant :

Process model with transfer function

$$G(s) = \frac{K}{s(1+Tp1*s)} * \exp(-Td*s)$$

with $K = -6.347+-0.24292$
 $Tp1 = 0.040011+-0.0070357$
 $Td = 0.17+-0$

An additive ARMA disturbance model has been estimated:

$$y = G u + (C/D)e$$

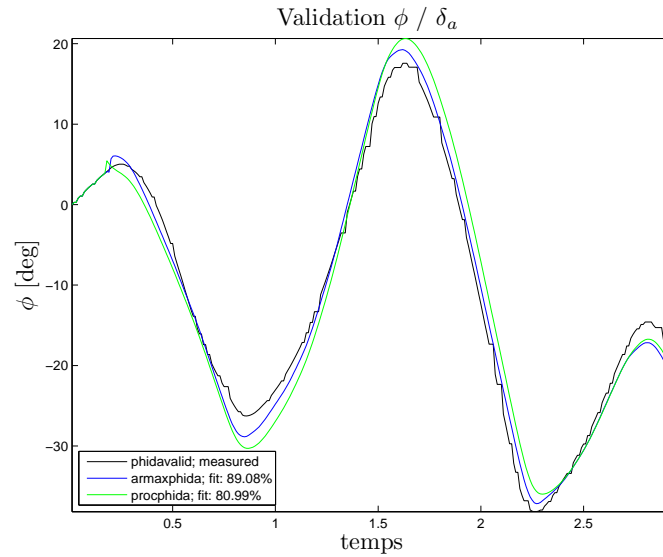
with

$$C(s) = s + 81.94 (+-4.3)$$

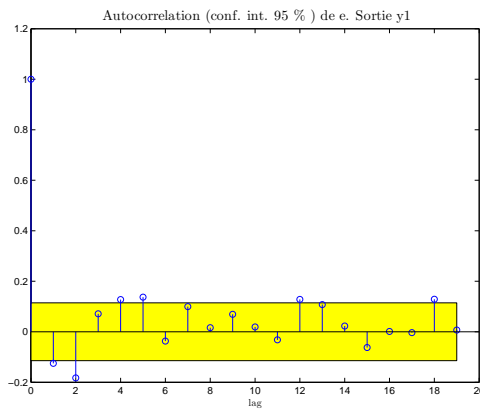
$$D(s) = s + 0.0002997 (+-0.1209)$$

Avec :

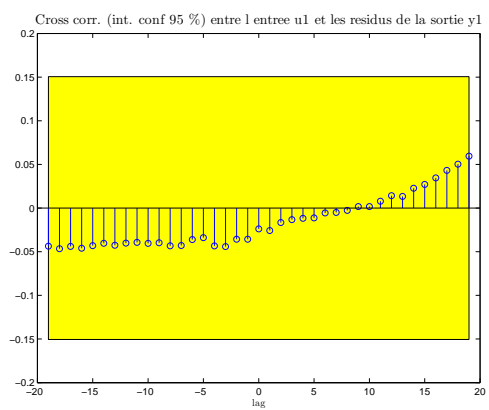
Eigenvalue	Damping	Freq. (rad/s)
0.00e+000	-1.00e+000	0.00e+000
-2.50e+001	1.00e+000	2.50e+001



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus



(c) Corrélation entre $\delta_a(k)$ et $\varepsilon(k + \tau)$

FIGURE A.6 – Identification et validation de $\frac{\phi(s)}{\delta_a(s)}$

Identification et validation de $\frac{\phi(s)}{\delta_r(s)}$

Il a été montré que $\frac{\phi(s)}{\delta_r(s)}$ pouvait être approximé par :

$$\frac{\phi(s)}{\delta_r(s)} \approx \frac{p(s)}{s \delta_r(s)} \quad (\text{A.9})$$

La figure A.7(a) présente une comparaison entre la simulation de la sortie d'un modèle *Process Model Siso* et ARMAX SISO et des données de mesure de validation. Les figures A.7(b) et A.7(c) présentent l'analyse de résidus effectuée du modèle choisi avec les données de validation ainsi que la corrélation entre $\delta_r(k)$ et $\varepsilon(k + \tau)$. Les intervalles de confiance à 95 % sont aussi présentés sur les figures A.7(b) et A.7(c). Le modèle choisi est le suivant :

```
Process model with transfer function
          1+Tz*s
G(s) = K * ----- * exp(-Td*s)
          s(1+Tp1*s)(1+Tp2*s)
```

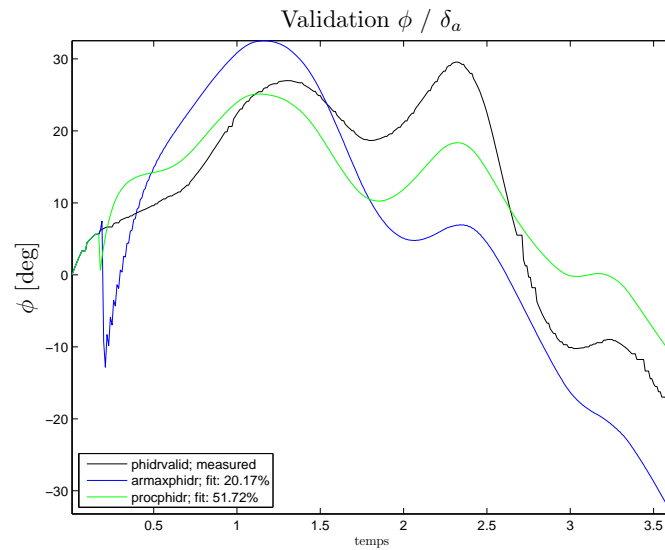
```
with   K = -0.045217+-0.019175
        Tp1 = 0.07+-0.049033
        Tp2 = 1.5554+-0.98801
        Td = 0.17+-0
        Tz = -100+-0
```

An additive ARMA disturbance model has been estimated:

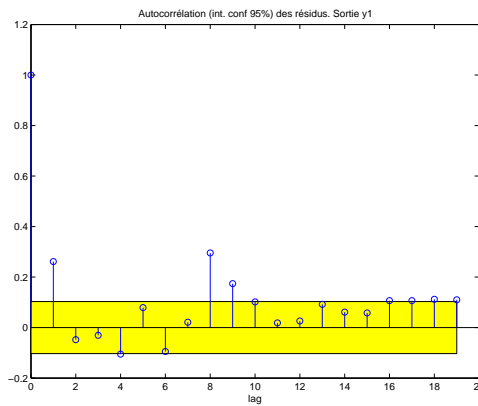
```
y = G u + (C/D)e
with
C(s) = s^2 + 48.87 (+-4.264) s + 1944 (+-261.7)
D(s) = s^2 + 4.238 (+-1.91) s + 26.78 (+-8.262)
```

Avec :

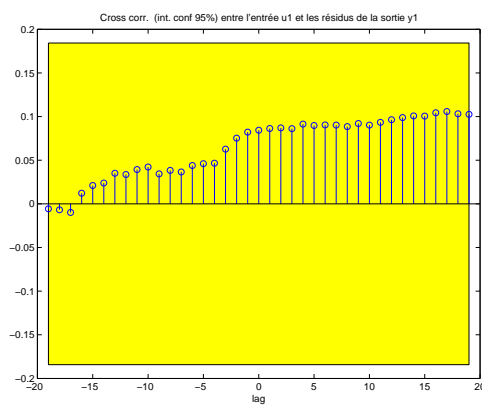
Eigenvalue	Damping	Freq. (rad/s)
0.00e+000	-1.00e+000	0.00e+000
-6.43e-001	1.00e+000	6.43e-001
-1.43e+001	1.00e+000	1.43e+001



(a) Comparaison - simulation de la sortie du modèle et données de mesure de validation



(b) Autocorrélation des résidus



(c) Corrélation entre $\delta_r(k)$ et $\varepsilon(k + \tau)$

FIGURE A.7 – Identification et validation de $\frac{\phi(s)}{\delta_r(s)}$

A.2 Identification du NRC-CNRC

Les tableaux A.1 à A.4 listent les matrices des coefficients de stabilité et de contrôle dimensionnel identifiées par le NRC-CNRC [21] :

	u	v	w	p	q	r
$x_{(.)}$	-0.195	0.2714	1.075	-0.1551	0.3501	-0.474
$y_{(.)}$	0.5500	-1.761	0.4651	-1.4878	-3.5853	0.6173
$z_{(.)}$	1.121	0.7819	-5.743	2.688	-7.185	-2.658
$l_{(.)}$	0.1882	-1.573	-1.929	-9.547	-5.024	15.33
$m_{(.)}$	0.7301	-0.3394	-3.126	-1.17118	-0.1359	-3.4605
$n_{(.)}$	-0.3249	0.9729	-0.3230	-0.8180	1.944	-0.8906

TABLE A.1 – Matrice de stabilité

	u	v	w	p	q	r
$x_{(.)}$	[1/sec]	[1/sec]	[1/sec]	[ft/sec · rad]	[ft/sec · rad]	[ft/sec · rad]
$y_{(.)}$	[1/sec]	[1/sec]	[1/sec]	[ft/sec · rad]	[ft/sec · rad]	[ft/sec · rad]
$z_{(.)}$	[1/sec]	[1/sec]	[1/sec]	[ft/sec · rad]	[ft/sec · rad]	[ft/sec · rad]
$l_{(.)}$	[rad/ft · sec]	[rad/ft · sec]	[rad/ft · sec]	[1/sec]	[1/sec]	[1/sec]
$m_{(.)}$	[rad/ft · sec]	[rad/ft · sec]	[rad/ft · sec]	[1/sec]	[1/sec]	[1/sec]
$n_{(.)}$	[rad/ft · sec]	[rad/ft · sec]	[rad/ft · sec]	[1/sec]	[1/sec]	[1/sec]

TABLE A.2 – Matrice de stabilité - Unités

	δ_e	δ_t	δ_a	δ_r
$x_{(.)}$	0.2350	0.3509	-0.0332	-0.0245
$y_{(.)}$	-0.08969	0.6412	-0.3401	0.1915
$z_{(.)}$	-0.1958	0.6826	0.4015	-0.1644
$l_{(.)}$	0.2710	-0.4943	-1.746	0.7194
$m_{(.)}$	-0.8904	1.229	0.3157	0.07179
$n_{(.)}$	-0.0424	-0.1266	-0.1212	-0.3847

TABLE A.3 – Matrice de contrôle

	δ_e	δ_t	δ_a	δ_r
$x_{(.)}$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$
$y_{(.)}$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$
$z_{(.)}$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$
$l_{(.)}$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$
$m_{(.)}$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$
$n_{(.)}$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$	$[ft/sec^2 \cdot deg]$

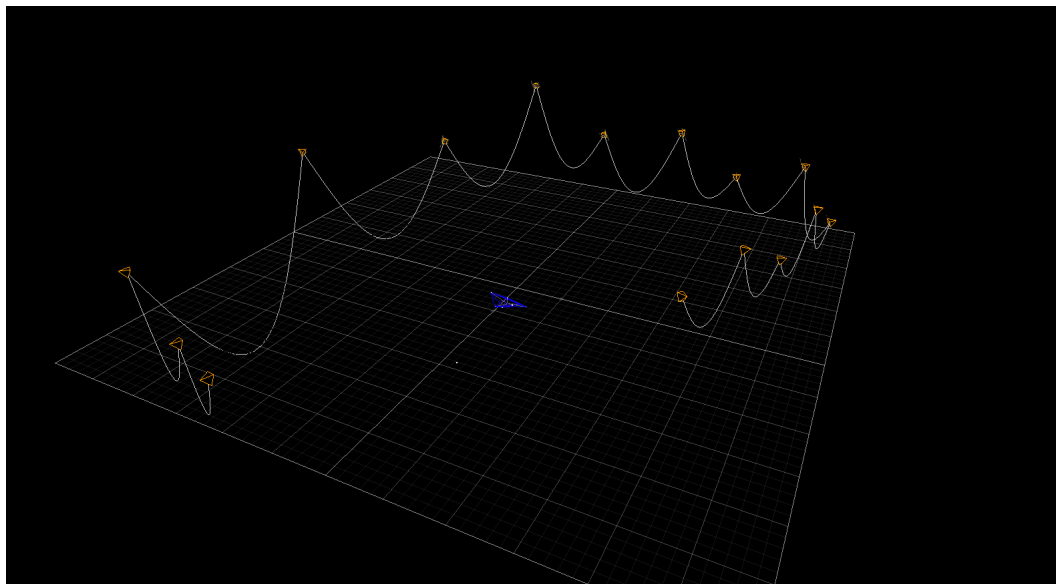
TABLE A.4 – Matrice de contrôle - unités

Il est à noter que les matrices de stabilité et de contrôle sont trouvées à partir de l'axe référentiel de l'avion. Tous les termes sont donc normalisés soit par la masse (les forces $x_{(.)}$, $y_{(.)}$, $z_{(.)}$), ou soit par leur inertie respective (les moments $l_{(.)}$, $m_{(.)}$, $n_{(.)}$). Aussi, la valeur de J_{xz} est supposée nulle par symétrie.

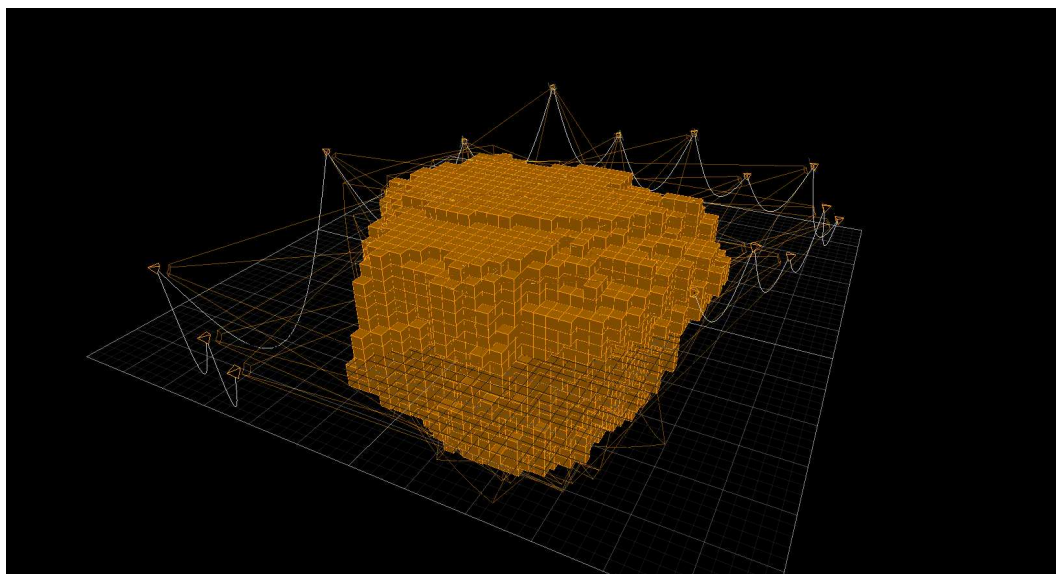
Annexe B

Système de vision

Les figures B.1(a) et B.1(b) montrent la disposition des caméras avec le corps de l'avion et le volume généré par le système de vision *Tracking tools*.



(a) Disposition des caméras avec le corps de l'avion reconnu par le système de vision *Tracking tools*



(b) Volume qui est généré par le système de vision *Tracking tools*

FIGURE B.1 – Tracking tools

Annexe C

Articles

C.1 Article soumis pour l'*American Control Conference 2010*

Transitions Between Level Flight and Hovering for a Fixed-Wing Mini Aerial Vehicle

Vincent Myrand-Lapierre, André Desbiens, Eric Gagnon, Frank Wong and Éric Poulin

Abstract— This article describes a control strategy to bridge the autonomous transition between level and hovering flight of a fixed-wing mini-aerial vehicle. These autonomous transitions, combined with the level-flight and hovering modes, would permit dangerous missions like reconnaissance in hostile or restricted areas. Then, there are four flight modes during a mission : the level-flight, the level-flight to hovering (L2H), the hovering, and the hovering to level-flight (H2L). The model structures for both level-flight and hovering modes are based on the linearization of a six-degree-of-freedom rigid body model of a fixed-wing mini-aerial vehicle. Controllers for both main flying modes are presented. The L2H mode is managed by the level-flight controller, whereas the H2L is managed by the hovering controller. A systematic approach based on a logic-based switching supervisor is developed to manage the transition between modes. Experimental results of a mini-aerial vehicle testbed which uses the switching supervisor are presented.

I. INTRODUCTION

Fixed-wing mini aerial vehicles (Mini-AV) are small aircrafts (wingspan less than 1000 mm) of low weight (less than 500 grams). Thanks to technological advances that have been made in recent years, Mini-AVs may be used for crowd surveillance and rescue in restricted or hostile areas. The high maneuverability of Mini-AVs may allow innovative operations, such as semi-autonomous flight inside a building or in urban area. To do so, the Mini-AV must be able to perform semi-autonomous level flight and hovering and autonomous transitions between these two modes so that the operator can focus on the higher level tasks of gathering information during a reconnaissance mission rather than focus on the lower level tasks of flying the vehicle. Level flight is the traditional flight mode that allows the aircraft to move horizontally at high speed, with relatively low energy consumption. Hovering mode refers to a vertical stationary mode, like that of a helicopter. However, this mode consumes more energy. Semi-autonomous control allows the operator to run the Mini-AV with simple commands, regardless of the problems of stabilization. Transitions must be fully autonomous, retaining as much as possible the current altitude of the vehicle. They must be made as fast as possible and within the shortest possible distance.

Myrand-Lapierre, Desbiens and Poulin are with the LOOP (Laboratoire d'observation et d'optimisation des procédés), Département of Electrical and Computer Engineering, Université Laval, Québec City, Québec, G1V 0A6, Canada. (e-mail : vincent.myrand-lapierre.1@ulaval.ca; andre.desbiens@gel.ulaval.ca; eric.poulin@gel.ulaval.ca)

Gagnon and Wong are with Defence Research and Development Canada Valcartier, Québec City, Québec, G3J 1X5, Canada (e-mail : eric.gagnon@drdc-rddc.gc.ca; franklin.wong@drdc-rddc.gc.ca)

In the literature, some authors proposed transition strategies between level flight and hovering for UAVs. Green and Oh [1] and Green [2] proposed PD controllers which were obtained iteratively to handle the transitions. Platform and sensors were similar to those presented in this paper. The level-flight mode was manually controlled, the hovering mode was semi-autonomous, and the transitions were performed only from level flight to hovering. The vehicle transitioned with a 2 seconds response time. However, the user had to manually control the throttle precisely, which cause a significant amount of overshoot in altitude. Finally, when the vehicle was entering the hovering mode, there was no closed-loop control of the roll angle. It should be noted that Green [2] was the first to succeed transition maneuvers for a Mini-AV. Frank et al. [3] also developed a similar experimental platform, but the attitude and position of the aircraft were measured by a motion capture system. All control surfaces were pre-programmed during the autonomous transitions, except the ailerons. The transition from level flight to hovering was done in 6 seconds with an oscillation of 1.4-m in altitude. The transition from hovering to level flight took 1.5 seconds with an altitude loss of 50-cm. The work of Frank et al. [3] was not focused on reducing the settling time and the altitude variation.

Johnson et al. [4] developed a real-time adaptive neural network model of their platform and they designed a single adaptive controller to control the entire flight envelope. Their platform was much larger (wingspan of 8.75 ft. and length of 7.8 ft.) than the platform of the present paper. Transitions to and from hovering were achieved with an altitude variation of 30-m and took between 5 and 9 seconds to complete. Stone et al. [5] identified an aerodynamic database of their tail-sitter to get a 6-DOF nonlinear rigid-body aircraft model. LQR and classical controllers were used to control both flying modes and the transitions. Relying on the work of Stone et al. [5], Anderson and Stone [6] developed a predictive control strategy that created, at each sampling period, a linear model of the nonlinear tail-sitter. Thus, a unique controller was used for the entire system. Flight test results of Stone et al. [7] showed that there was an altitude gain of 40-m for the transition from level to hovering flight, and an altitude loss of 5-m for the transition from hovering to level flight. Knoebel [8] and Osborne [9] also developed transition strategies for a tail-sitter. An approximation of the aerodynamic coefficients was used in the aircraft model. Osborne [9] tested three types of controllers: a simple linear controller, a feedback linearization adaptive controller, and an adaptive controller. Flight test results were similar in magnitude to those of

Stone et al. [7]. Obviously, these tail-sitters are quite different from the vehicle presented in this paper by their size and model structure; however, their transition strategies remain interesting for this study.

The objective of this paper is to present a new control strategy that bridges level and hovering flight for a Mini-AV. Based on simplified model structures and controllers for both flying modes, a systematic approach is proposed to perform autonomous transition maneuvers. This paper is organized as follows : Section II describes an overview of the system ; Section III describes the switching between the controllers and the model structures and controllers for both level flight and hovering mode ; then, test bed results of transitions are presented in Section IV.

II. SYSTEM OVERVIEW

The Mini-AV used for this study is a commercial-off-the-shelf fixed-wing airplane (Flatana, Great Planes). It has a 0.91-m wingspan and weighs 475 grams when all equipment is mounted. The propulsion system consists of a RIMFIRE 22M-1000 brushless motor with a 10x3.8 propeller coupled to a 25A electronic speed controller. The aerodynamic surfaces are controlled using a standard 72 MHz receiver and micro servos. The vehicle attitude is measured by a wireless IMU (Microstrain 3DM-GX2). It has 3 accelerometers, 3 rate-gyros, and 3 flux-magnetometers. The IMU calculates and transmits orientation via a cosine matrix at 100 Hz. A vision system is used to measure the altitude in hovering mode.

As shown in Fig. 1, there are four manipulated variables: ailerons (δ_a), elevator (δ_e), rudder (δ_r) and throttle (δ_t). X_B , Y_B , and Z_B are the body axis with the origin at the center of gravity (CG) of the aircraft. ϕ , θ , and ψ are the body frame angular attitude; p , q , and r are the body frame angular rates; a_x , a_y , and a_z are the body frame accelerations; u , v , and w are the velocity components of CG in the body reference frame. For navigation, the NED axis system (North-East-Down) is used as the frame of reference on the Earth. The Earth is considered as flat and stationary in inertial space. The NED origin is arbitrarily located. X_E points to the North, Y_E points to the East, and Z_E points vertically downward. The reference attitude correspond to $\phi = 0$, $\theta = 0$ and $\psi = 0$ when the body axis and the NED axis are aligned.

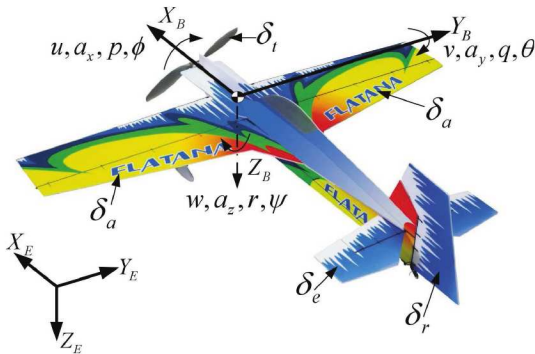


Fig. 1. Definitions of axis, control surfaces, and main variables

III. FLIGHT MODES

This section describes the different flight mode controllers. A logic-based supervisor (Fig. 2) manages the switching between two controllers, namely, the level-flight controller and the hovering controller. The control system is divided into four modes: the level-flight, the level-flight to hovering (L2H), the hovering, and the hovering to level-flight (H2L). The level-flight and L2H modes are managed in the level-flight controller, whereas the hovering and the H2L are managed in the hovering controller.

Each controller receives from the logic-based supervisor (shown in Fig. 2) the trim values of the manipulated variables (δ_{trim}). The trim values are the 0 deg deflection of the micro servos. At any given time the autopilot is engaged, one controller is inactive, whereas the other controller is active and calculates the manipulated variables. The set points of the inactive controller are forced to be equal to their outputs. When a mode is activated, the set points are switched to their desired values and the new calculated manipulated variables are added to the trim values. Thus, switching is bumpless.

As an example (Fig. 2, bold arrows, numbers ① to ⑫), the system is initialized on the ground : the power supply is turned on ① and the autopilot is disengaged. If the pitch angle is less than 45 deg ②, the supervisor waits in the level flight mode ③ with the manipulated variables being equal to the trim values δ_{trim} . When the user engages the autopilot ④, the supervisor triggers the level flight mode controller until another action is commanded ⑤. If the user commands a transition ⑥, then the supervisor triggers the L2H mode ⑦. The supervisor switches to the hover mode controller when the pitch attitude exceeds 50 deg ⑧. Then, the hover controller is triggered until another action is commanded ⑨. When transition is commanded ⑩, the H2L controller is triggered ⑪ and the supervisor switches to the level flight controller when the pitch attitude is below 50 deg ⑫.

A. Level-Flight Mode

The standard Euler angles are used to represent the vehicle attitude. Using the linearization of the 6-DOF nonlinear equations of a rigid body in level flight from Etkin and Reid [10], the longitudinal ($u, w, q, \theta, \delta_e, \delta_t$) and lateral ($v, p, r, \phi, \psi, \delta_a, \delta_r$) state variables are decoupled. Transfer functions (TFs) that describe the main aerodynamic behavior of the longitudinal and lateral modes are approximated. To identify these transfer functions, an experienced pilot executed excitation maneuvers from trim conditions.

Following the methodology for airplane identification presented by Morelli and Klein [11], doublets with sufficient amplitudes and durations were executed to obtain large angular flight maneuvers. Each doublet was applied on one manipulated variable at a time. In the level-flight mode, the elevator was used to control the pitch angle while the throttle was used for speed control. The ailerons and rudder were used for lateral control, that is, to control the roll and yaw angles, respectively. Tables I and II show the identified transfer functions of the system.

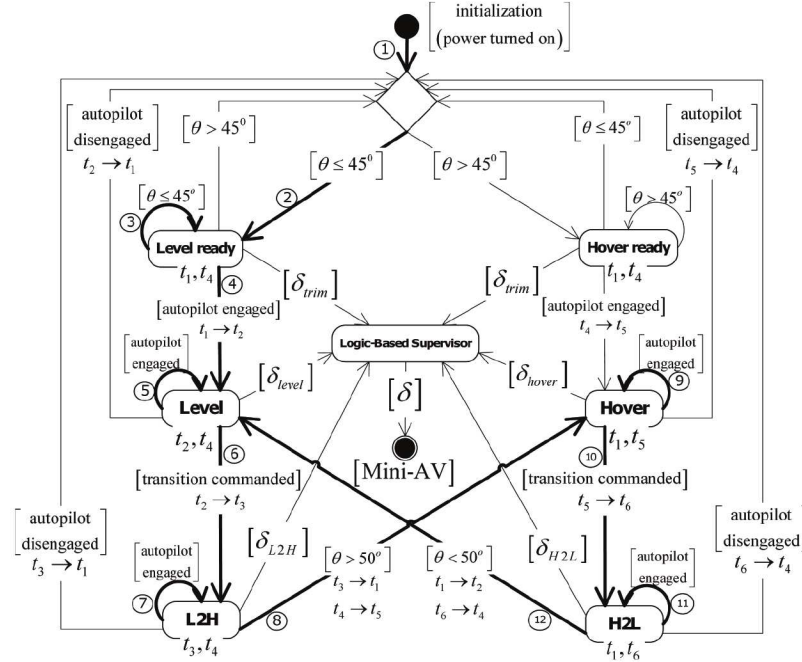


Fig. 2. Block diagram of the logic-based supervisor

TABLE I
TRANSFER FUNCTION FOR LONGITUDINAL LEVEL FLIGHT CONTROL

	$\delta_e [deg]$
$\theta [deg]$	$\frac{-2.11(0.22s+1)}{s(0.0082s^2+0.0543s+1)}$

TABLE II
TRANSFER FUNCTIONS FOR LATERAL LEVEL FLIGHT CONTROL

	$\delta_a [deg]$	$\delta_r [deg]$
$p [deg/s]$	$\frac{-4.61e^{-0.17s}}{(0.05s+1)}$	$\frac{0.18se^{-0.17s}}{(0.06s+1)(1.71s+1)}$
$r [deg/s]$	$\frac{-4.05(-0.18s+1)e^{-0.17s}}{(0.08s+1)(0.81s+1)}$	$\frac{0.38(10s+1)e^{-0.17s}}{(0.05s+1)(1.5s+1)}$
$\phi [deg]$	$\frac{-6.35e^{-0.17s}}{s(0.04s+1)}$	$\frac{-0.05se^{-0.17s}}{(0.07s+1)(1.55s+1)}$

For the longitudinal closed-loop control (Fig. 4), the pitch angle is the variable of interest. A PI with anti-windup is used to design a pitch-attitude-hold controller [10]. The throttle is left open-loop (δ_t^{level}). For lateral closed-loop control (Fig. 3), cascaded loops are used [10]. The first inner loop controls the roll rate, whereas the intermediate inner loop controls the roll angle. The error on the roll angle is converted to a roll rate set point by a proportional gain. Finally, the outer loop controls the yaw angle or the heading set point. Using an approximation of the sum of the forces along the body axis during a coordinated turn [12], the error on the yaw angle is converted to a roll angle set point by a PI regulator. A yaw damper is also introduced [10] in the lateral controller because, during a steady turn, $r \neq 0$. A high-pass filter (HPF), which has zero steady-state gain and unity high-frequency gain, is implemented to ensure that a desired rudder deflection was commanded. To ensure good

handling qualities, the responses of the closed loops had to be quick, well damped, and without static error. To ensure the stability of the system, enough amplitude and phase margin are also required. Table III shows the transfer functions of the level-flight controller.

TABLE III
TRANSFER FUNCTIONS OF THE LEVEL-FLIGHT CONTROLLER

	$\phi [deg]$	$\theta [deg]$	$\psi [deg]$	$p [deg/s]$	$r [deg/s]$
P	-8			-0.08	0.25
PI		$\frac{-1(1.4s+1)}{1.4s}$	$\frac{1(3s+1)}{3s}$		

B. Hover Flight Mode

Because standard Euler angles have singularities at $\theta = \pm 90$ deg, quaternions are useful to represent the attitude of a vehicle which has to perform wide angle maneuvers. In our case, quaternions are used to calculate a deviation attitude from a quaternion reference attitude. This quaternion deviation is then transformed into angular position errors in each body axis. These angular position errors are similar to the vertical Euler angles and they are used to describe the vehicle's attitude in hovering. The identification of the model structures and controllers in hover flight for this Mini-AV was done by Bilodeau et al. [13], [14]. Simplification hypothesis of the hovering behavior were formulated, and a linearization of a nonlinear 6-DOF rigid body model was made to obtain a decoupled structure (Table IV). The attitude variables are obtained by integrating the angular rates.

Even if there are disturbances from the ailerons on the elevator and rudder, the couplings are neglected in the design. Cascaded loops are used to control each axis of the hovering

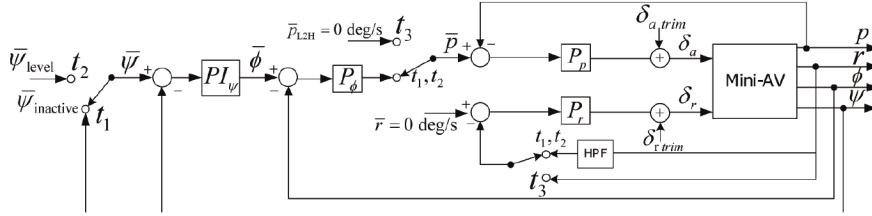


Fig. 3. Block diagram of the switching lateral controller

TABLE IV
TRANSFER FUNCTIONS FOR HOVERING CONTROL

	$\delta_a [deg]$	$\delta_e [deg]$	$\delta_r [deg]$
$p [deg/s]$	$\frac{-5.44e^{-0.17s}}{(0.67s+1)}$		
$q [deg/s]$		$\frac{-7.96se^{-0.17s}}{(5s-1)(0.08s+1)}$	
$r [deg/s]$			$\frac{2.46se^{-0.17s}}{(1.43s-1)(0.09s+1)}$

mode (Fig. 5). The ailerons, the elevator and the rudder are used to control the roll angle, the pitch angle, and the yaw angle respectively. Θ represents the attitude set points $[\bar{\phi}, \bar{\theta}, \bar{\psi}]$, Θ stands for the measured attitude $[\phi, \theta, \psi]$, Ω corresponds to the angular rates $[p, q, r]$, and δ represents the manipulated variables $[\delta_a, \delta_e, \delta_r]$. The inner loop has a proportional gain to accelerate the system. The external loop is used to respect amplitude and phase margin criteria. PI controllers are used to ensure that there is no static error. Tuning is made using a frequency methodology based on Nichols Chart. In the Y_B axis controller, $\bar{\theta}$ is set to 90 deg. In the X_B axis, $\bar{\phi}$ is set so that the belly of the Mini-AV faces a certain direction $\bar{\psi}_{level}$. In the Z_B axis controller, $\bar{\psi}$ is set to 0 deg. The throttle is used for altitude control in hovering (Fig. 6). Table V shows the transfer functions of the hovering controller.

TABLE V
TRANSFER FUNCTIONS OF THE HOVERING CONTROLLER

	$\phi [deg]$	$\theta [deg]$	$\psi [deg]$	$p [deg/s]$	$q [deg/s]$	$r [deg/s]$
P				-0.3	-0.18	0.16
PI	$\frac{-2.5(3s+1)}{3s}$	$\frac{-7(2.5s+1)}{2.5s}$	$\frac{-5(2s+1)}{2s}$			

C. Level to Hover Flight Mode (L2H mode)

The behavior of the platform during transition is extremely non-linear, making identification of a complete model tedious. Thus, with a priori knowledge of the behavior of the vehicle in both flying modes, a systematic approach is chosen to understand and develop the transition strategies. This methodology should be usable to implement transition strategies on other platforms.

First, the level to hover (L2H) control mode used the same control architecture and controller structures as the level-flight mode. The elevator is used to pitch up the vehicle to the hovering position. The ailerons and rudder keep the same functions used for level flight. The throttle is pitch scheduled.

The hypothesis is made that the TFs of the longitudinal and lateral modes can be used to approximate the behavior of the aircraft up to the stall angle. This stall angle is found experimentally to be $\theta = 65$ deg.

Consider a Mini-AV in nominal level flight (Fig. 2 and Fig. 4 : t_2 , $\delta_t level$ and $\bar{\theta}_{level}$) moving in a certain direction $\bar{\psi}_{level}$ (Fig. 2 and Fig. 3 : t_2). The following steps are performed when a transition is commanded by the user :

- 1) In the longitudinal controller (Fig. 2 and Fig. 4 : $t_2 \rightarrow t_3$), $\bar{\theta}$ is set to 90 deg. δ_t is specified open-loop via a pitch-angle schedule. Up to $\theta = 25$ deg, δ_t is increased (+10%) to allow the platform to rapidly increase the pitch angle. When θ reaches 25 deg, δ_t is decreased (-20%) to ensure a limit in altitude. In the lateral controller (Fig. 2 and Fig. 3 : $t_2 \rightarrow t_3$), the external loops are open and only the inner loop with set points of $\bar{p} = 0$ and $\bar{r} = 0$ is used.
- 2) The hover mode controller is switched on at $\theta = 50$ deg (Fig. 2 and Fig. 5 : $t_4 \rightarrow t_5$, Fig. 2, Fig. 3, and Fig. 4 : $t_3 \rightarrow t_1$). The hovering reference attitude is generated using the heading set point of the level-flight mode ; $\bar{\theta}$ is set to 90 deg, $\bar{\phi}$ is set to $\bar{\psi}_{level}$, and $\bar{\psi}$ is set to 0 deg. The altitude set point \bar{z} is chosen 30 centimeters above the current Mini-AV position. The altitude is controlled using the vision system (Fig. 2 and Fig. 6 : $t_4 \rightarrow t_5$).

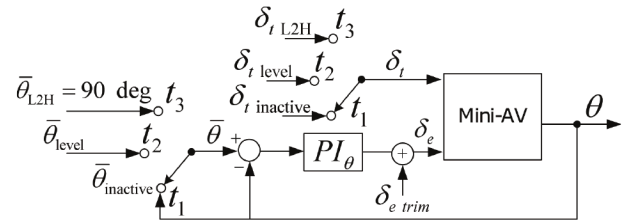


Fig. 4. Block diagram of the switching longitudinal controller

D. Hover to Level-Flight Mode (H2L mode)

This control mode used the same control architecture and controller structures as the hovering mode. The elevator is used to pitch down the vehicle to level flight. The ailerons and rudder keep the same functions as in hovering. The throttle is increased during the maneuver. The hypothesis is made that the model structures in hover flight can be used to approximate the behavior of the aircraft down to 50 deg of pitch.

Consider a Mini-AV stabilized in hovering mode (Fig. 2 and Fig. 6 : t_5 , $\delta_t hover$) with the belly of the Mini-AV facing a certain direction ψ_{level} (Fig. 2 and Fig. 5 : t_5). The following steps are performed when a transition is commanded by the user :

- 1) In the hovering controller (Fig. 2 and Fig. 5 : $t_5 \rightarrow t_6$), $\bar{\phi}$ is set to ψ_{level} , $\bar{\theta}$ is set to 0 deg, and $\bar{\psi}$ is set to 0 deg. The throttle is increased by 15% compared with $\delta_t hover$ to avoid the loss of altitude during the pitch maneuver (Fig. 2 and Fig. 6 : $t_5 \rightarrow t_6$). The altitude was left open-loop.
- 2) The level-flight mode controller is switched at $\theta = 50$ deg (Fig. 2, Fig. 3, and Fig. 4 : $t_1 \rightarrow t_2$, Fig. 2, Fig. 5 and Fig. 6 : $t_6 \rightarrow t_4$). The heading set point in level flight is generated using the quaternion reference of the hovering mode ; $\bar{\theta}$ is set to θ_{level} and $\bar{\psi}$ is set to ψ_{level} . Finally, δ_t is set to $\delta_t level$.

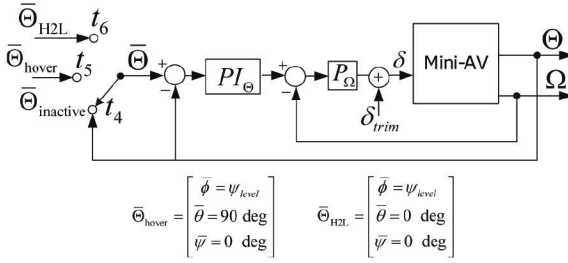


Fig. 5. Block diagram of the switching hovering controller

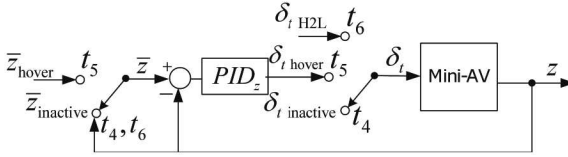


Fig. 6. Block diagram of the switching altitude controller

IV. TEST BED RESULTS

For the level to hovering transition trials, the vehicle took off in level-flight mode from a launch pad at a height of 1.5 meters. It could travel forward 8 meters before striking a safety net, or up 3 meters before striking the ceiling. A vision system consisting of seven cameras was located near the ceiling and was positioned to form a capture volume of 4-m x 4-m x 2-m above the safety net. During flight, the heading set point was set to the initial heading at take-off, and a pitch angle set point of 18 deg was chosen to ensure a nominally stable flight. A transition was commanded 1.2 seconds after take-off. Fig. 7 shows the results of the longitudinal variables and Fig. 8 shows the lateral variables. Standard Euler angles are used to represent the level-flight and the L2H modes, whereas the angular position errors, calculated with the quaternion deviation, are used to represent the attitude in the hovering and the H2L modes. From Fig. 7, it can be seen that the Mini-AV was in the L2H mode for less than 0.6 second. It took less than 3 seconds after a transition for the

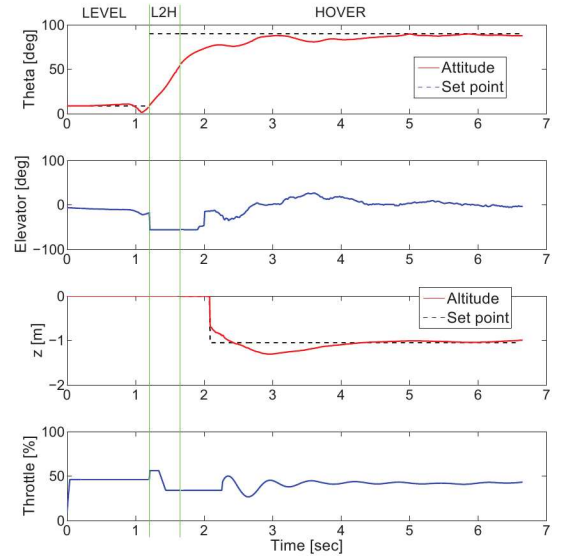


Fig. 7. Trial (level flight to hovering): θ , z , δ_e and δ_t

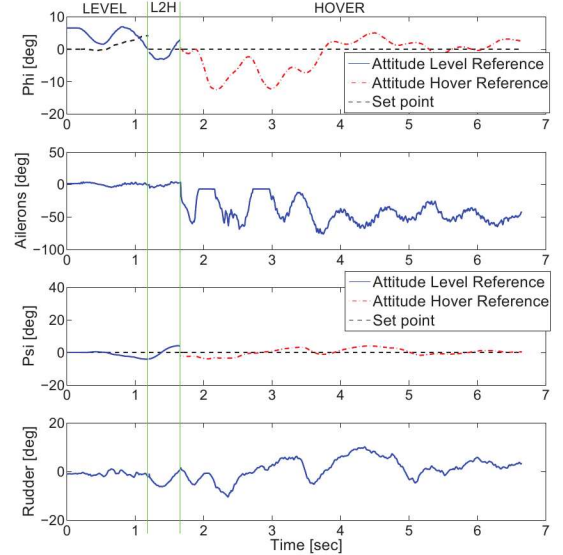


Fig. 8. Trial (level flight to hovering) : ϕ , ψ , δ_a and δ_r

system to settle around the reference altitude as compared to the 6 seconds settling time of Frank et al. [3]. The roll and yaw oscillated within ± 4 deg. An overshoot in altitude of 30-cm was observed, as compared to the 1.4-m oscillation of Frank et al. [3].

The results of the hovering to level transition trial are shown in Fig. 9 and Fig. 10. A transition was commanded manually in hovering mode when all the axes were stabilized. Fig. 9 shows the results of the longitudinal variables, while Fig. 10 shows the lateral variables. The Mini-AV was in the H2L mode for less than 0.6 second. The settling time to control the pitch angle and heading was less than 2 seconds, which is similar to the result of Frank et al. [3]. There was no significant loss or gain of altitude during the maneuver,

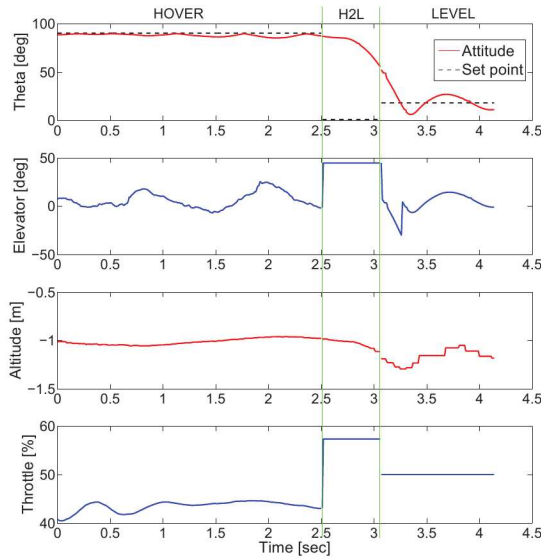


Fig. 9. Trial (hovering to level flight) : θ , z , δ_e and δ_t

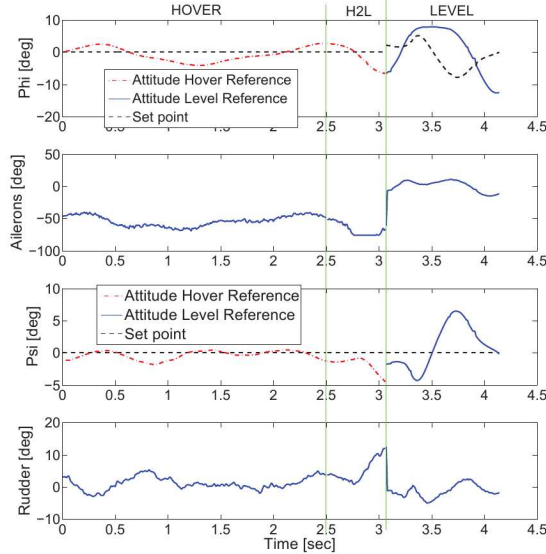


Fig. 10. Trial (hovering to level flight): ϕ , ψ , δ_a and δ_r

as compared to the 50-cm loss in altitude of Frank et al. [3].

V. CONCLUSIONS

This paper described a level-flight controller, a hovering controller, and the transition strategies used between level and hover flights, which rely on a logic-based supervisor for a fixed-wing Mini-AV. The logic-based supervisor allowed bumpless switching between the different modes. Controller tuning was accomplished using a method based on identified transfer functions that were determined by a linearization of a nonlinear 6-DOF rigid body model. A systematic approach was chosen to develop a logic-based switching supervisor. Experimental test results showed L2H and H2L transitions could be completed in less than 3

seconds using the developed strategy. Future work will focus on the implementation of altitude estimation methods with the objective of eliminating the need for a vision system to act as a feedback sensor for altitude control.

VI. ACKNOWLEDGMENT

The authors thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and "Le Fonds québécois de la recherche sur la nature et les technologies" (FQRNT). Their financial support has made this work possible. The authors also thank the Defence Research and Development Canada (DRDC) who has initiated and supported this project.

REFERENCES

- [1] Green, W.E. and P.Y. Oh, *A MAV That Flies Like an airplane and Hovers Like a Helicopter*, International Conference on Advanced Intelligent Mechatronics, Monterey, United states, 2005, 694-698.
- [2] Green, W.E., *A Multimodal Micro Air Vehicle for Autonomous Flight in Near-Earth Environments*, Thesis submitted to the faculty of Drexel University in partial fulfillment of the requirements for the degree of Ph. D., 2007.
- [3] Frank, A., J. Mcgrew, M. Valenti, D. Levine and J. P. How, *Hover, Transition, and level Flight Control design for a Single-propeller Indoor Airplane*, AIAA GNC, Hilton Head, United states, 2007, AIAA 2007-6318.
- [4] Johnson Eric N, A. Wu, J. C. Neidhoefer , S. K. Kannan and M.A. Turbe , *Flight-Test Results of Autonomous Airplane Transitions Between Steady-Level and Hovering Flight*, Proceedings of the AIAA Guidance, Navigation, and Control Conference, Monterey, CO, 2006.
- [5] Stone R.H., *Control Architecture for a Tail-Sitter Unmanned Air Vehicle*, 5th Asian Control Conference, 2004.
- [6] Anderson P. and R.H. Stone, *Predictive Guidance and Control for a Tail-Sitting Unmanned Aerial Vehicle*, Information, Decision and Control, 2007. IDC '07 (2007), pp. 148-153.
- [7] Stone R.H., P. Anderson and P. Gibbens, *Predictive Control for a Tail-Sitter UAV: Final Report (Part 2)*, AOARD, UNIT 45002, APO, AP, 96337-5002.
- [8] Knoebel, N.B., *Adaptive Quaternion Control of a Miniature Tail-sitter UAV*, Thesis submitted to the faculty of Brigham Young University in partial fulfillment of the requirements for the degree of Master, 2007.
- [9] Osborne S.R , *Transition Between Hover and Level Flight for a Tail-sitter UAV*, Thesis submitted to the faculty of Brigham Young University in partial fulfillment of the requirements for the degree of Master, 2007.
- [10] Etkin, B. and L. D. Reid, *Dynamics of Flight, Stability and Control*, John Wiley & Sons, Inc., 3rd ed., January 1996.
- [11] Morelli E. and V. Klein, *Aircraft System Identification : Theory and Practice*, American Institute of Aeronautics and Astronautics, AIAA Education Series, 2006.
- [12] Blakelock J.H., *Automatic Control of Aircraft and Missiles*, Second Edition, Wiley, 1991.
- [13] Bilodeau, P.-R., E. Poulin, F. Wong, E. Gagnon and A. Desbiens, *Control of a Hovering Mini Fixed Wing Aerial Vehicle*, AIAA Conference on Guidance, Navigation and Control, Chicago, 2009.
- [14] Bilodeau, P.-R., *Commande d'un mini véhicule aérien sans pilote en vol stationnaire pour l'opération en milieu restreint*, Thesis submitted to the faculty of Université Laval in partial fulfillment of the requirements for the degree of MSc., 2009.
- [15] Kita K., A. Konno, and M. Uchiyama, *Hovering Control of a Tail-Sitter VTOL Aerial Robot*, Journal of Robotics and Mechatronics, Vol.21 No.2, 2009
- [16] Stevens B.L. and F.L. Lewis, *Aircraft Control and Simulation*, 2nd edition, John Wiley & sons Inc, New Jersey, 2003.