

## TABLE DES MATIÈRES

	Page
INTRODUCTION .....	1
CHAPITRE 1 REVUE DE LITTERATURE .....	5
1.1 MOPS (Multi-Objective Parametric Synthesis).....	5
1.2 CONDUIT (Control Designer's Unified Interface).....	6
1.3 Travaux de Eicher Low et William L.Garrard.....	8
1.4 Travaux de Kimon P. Valavanis .....	10
1.5 Travaux effectués au sein de l'ÉTS .....	12
1.6 Conclusion .....	13
CHAPITRE 2 MODÉLISATION DE LA DYNAMIQUE DE VOL DE L'HÉLICOPTÈRE .....	15
2.1 Généralités sur l'hélicoptère .....	15
2.1.1 Principe de propulsion .....	15
2.1.2 Commandes du vol .....	16
2.1.3 Rotor principal .....	16
2.1.4 Rotor de la queue .....	17
2.2 Équations de mouvement d'un hélicoptère.....	17
2.2.1 Matrice de rotation .....	19
2.2.2 Matrice de transformation.....	19
2.2.3 Équation du moment angulaire et de l'accélération rectiligne.....	20
2.2.4 Récapitulatif des équations de mouvement de l'hélicoptère.....	21
2.3 Forces et moments s'exerçant sur l'hélicoptère.....	22
2.3.1 Forces .....	22
2.3.2 Moments .....	24
2.4 Équations de la poussée et du battement.....	26
2.4.1 Poussée générée par le rotor principal .....	26
2.4.2 Poussée générée par le rotor de la queue .....	27
2.4.3 Battement .....	27
2.5 Modèle d'état .....	28
CHAPITRE 3 LES QUALITÉS DE VOL DE L'HÉLICOPTÈRE .....	31
3.1 Introduction.....	31
3.2 Critères des spécifications relatives aux qualités de vol.....	35
3.2.1 Critère de la bande passante.....	35
3.2.2 Types de réponses .....	37
3.3 Exigences des spécifications pour la manœuvre <i>Hover</i> et basses vitesses .....	40
3.3.1 Changement d'attitude à petites amplitudes (tangage et roulis) .....	40
3.3.2 Changement d'attitude à amplitudes modérées (tangage et roulis) .....	43
3.3.3 Changement d'attitude à larges amplitudes .....	44
3.3.4 Couplage Interaxe .....	44

3.4	Conclusion .....	48
CHAPITRE 4 AFFECTATION DE LA STRUCTURE MODALE .....		49
4.1	Généralités sur les systèmes linéaires .....	49
4.2	Théorème de Moore .....	54
4.3	Liberté procurée par l'affectation de la structure modale .....	58
4.4	Sous espace vectoriel contenant les vecteurs propres réalisables .....	59
4.5	Projection des vecteurs propres désirés .....	60
4.6	Calcul des gains .....	62
4.6.1	Cas où les valeurs propres sont distinctes .....	62
4.6.2	Cas où les valeurs propres sont multiples .....	62
CHAPITRE 5 CONCEPTION DU CONTRÔLEUR POUR HÉLICOPTÈRE .....		69
5.1	Conception de la boucle interne du contrôleur .....	72
5.2	Conception de la boucle externe du contrôleur .....	87
5.3	Conclusion .....	100
CHAPITRE 6 IMPLÉMENTATION DES LOIS DE CONTRÔLE SUR HELISIM .....		103
6.1	Environnement d'implémentation et de simulation .....	103
6.2	Implémentation des lois de contrôle pour l'hélicoptère BELL-412 .....	108
6.3	Implémentation de la loi de contrôle de l'axe latéral uniquement .....	111
6.4	Implémentation de la loi de contrôle de l'axe latéral et l'axe de lacet .....	113
CONCLUSION .....		115
ANNEXE I	CODE MATLAB DE CALCUL DES MATRICES K ET H .....	117
ANNEXE II	CODE VISUAL STUDIO POUR L'IMPLEMENTATION DE HELISIM .....	123
LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES .....		145

## **LISTE DES TABLEAUX**

	Page	
Tableau 1.1	Jumelage optimal pour le Yamaha R-50 .....	12
Tableau 3.1	Types de réponses requis pour la manœuvre <i>Hover</i> .....	33
Tableau 3.2	Exigences pour le Changement d'attitude à larges amplitudes .....	45
Tableau 5.1	Liste des vecteurs propres désirés .....	77
Tableau 5.2	Liste des vecteurs propres réalisables .....	77



## LISTE DES FIGURES

	Page	
Figure 1.1	Séparation entre la structure modèle/contrôleur et le réglage des paramètres de la conception .....	7
Figure 1.2	Canal tangage.....	9
Figure 1.3	Architecture du système de contrôle en deux boucles.....	9
Figure 1.4	Contrôleur en cascade (boucle interne et boucle externe) .....	10
Figure 1.5	Approche linéaire.....	11
Figure 1.6	L'hélicoptère vu comme un système carré .....	11
Figure 2.1	Configuration typique d'un hélicoptère .....	15
Figure 2.2	Les forces aérodynamiques agissant sur une pale.....	17
Figure 2.3	Repère relatif à l'hélicoptère.....	18
Figure 2.4	L'angle $\beta_{lc}$ entre HP (Hub Plane) et TPP dans la direction longitudinale .....	23
Figure 2.5	L'angle $\beta_{ls}$ entre HP et TPP dans la direction latérale.....	23
Figure 2.6	Signification de $h_m$ et $h_t$ .....	25
Figure 2.7	Signification de $l_t$ , $l_m$ et $y_m$ .....	25
Figure 2.8	Signification de $Q_{MR}$ .....	26
Figure 3.1	Définition des niveaux des qualités de vol .....	32
Figure 3.2	Définition de la Bande passante.....	35
Figure 3.3	Réponse à une entrée impulsionale de l'actuateur.....	38
Figure 3.4	Exigences pour des changements d'attitude à petites amplitudes .....	41
Figure 3.5	positionnement des pôles en fonction de $w_n$ et $\zeta$ .....	41

Figure 3.6	Limites sur le oscillations (tangage et roulis).....	42
Figure 3.7	Définition des paramètres du critère d'amplitudes modérées .....	42
Figure 3.8	Exigences pour changement d'attitude à amplitudes modérées.....	42
Figure 3.9	Exigences concernant le couplage lacet-collectif .....	46
Figure 3.10	Exigences concernant le couplage tangage-roulis .....	47
Figure 5.1	Boucle interne du contrôleur.....	72
Figure 5.2	Valeurs singulières de $F_{BIF}$ .....	80
Figure 5.3	Valeurs singulières des fonctions de transfert des quatre canaux.....	81
Figure 5.4	Valeurs singulières de $G(s) = \frac{4}{s+4}$ .....	81
Figure 5.5	Diagramme de Bode des fonctions de transfert des quatre canaux.....	83
Figure 5.6	Diagramme de Bode de la fonction de transfert $G(s) = \frac{4}{s+4}$ .....	84
Figure 5.7	Réponse du système ( $q$ , $u$ , $w$ et $\theta$ ) (en BO et en BIF) à une entrée échelon de 5 deg/s sur la consigne de la vitesse de roulis .....	85
Figure 5.8	Réponse du système ( $p$ , $r$ , $v$ et $\phi$ ) (en BO et en BIF) à une entrée échelon de 5 deg/s sur la consigne de la vitesse de roulis .....	86
Figure 5.9	Boucle externe du canal roulis.....	87
Figure 5.10	Diagramme de phase (axes tangage et roulis).....	89
Figure 5.11	Réponse du système ( $q$ , $u$ , $w$ et $\theta$ ) , en BIF, à une entrée échelon ( $p_c = 5$ deg/s) et à une entrée échelon ( $\phi_c = 5$ deg) avec BEF .....	91
Figure 5.12	Réponse du système ( $p$ , $r$ , $v$ et $\phi$ ) , en BIF, à une entrée échelon ( $p_c = 5$ deg/s) et à une entrée échelon ( $\phi_c = 5$ deg) avec BEF .....	92
Figure 5.13	Réponse du système ( $q$ , $u$ , $w$ et $\theta$ ) , en BIF, à une entrée échelon ( $q_c = 5$ deg/s) et à une entrée échelon ( $\theta_c = 5$ deg) avec BEF .....	93

Figure 5.14	Réponse du système ( $p$ , $r$ , $v$ et $phi$ ) , en BIF, à une entrée échelon ( $q_c = 5$ deg/s ) et à une entrée échelon ( $\theta_c = 5$ deg) avec BEF .....	94
Figure 5.15	Réponse du système ( $q$ , $u$ , $w$ et $theta$ ) , boucle interne fermée et boucle externe fermée, à une entrée échelon ( $r_c = 5$ deg/s) .....	96
Figure 5.16	Réponse du système ( $p$ , $r$ , $v$ et $phi$ ) , boucle interne fermée et boucle externe fermée, à une entrée échelon ( $r_c = 5$ deg/s) .....	97
Figure 5.17	Réponse du système ( $q$ , $u$ , $w$ et $theta$ ) , boucle interne fermée et boucle externe fermée, à une entrée échelon ( $w_c = 2$ m/s) .....	98
Figure 5.18	Réponse du système ( $p$ , $r$ , $v$ et $phi$ ) , boucle interne fermée et boucle externe fermée, à une entrée échelon ( $w_c = 2$ m/s) .....	99
Figure 6.1	Schéma SIMULINK (boucle externe fermée tangage et roulis).....	104
Figure 6.2	Schéma SIMULINK modifié pour implémentation.....	106
Figure 6.3	Correspondance entre les entrées de Simulink et ceux de Visual Studio .....	106
Figure 6.4	Génération de code à partir du Fichier Simulink .....	107
Figure 6.5	Implémentation de la loi de contrôle sur HELISIM (axe latéral) .....	107
Figure 6.6	Fenêtre de simulation de vol HLSIM SCENE .....	108
Figure 6.7	Évolution des angles $\phi$ et $\theta$ en fonction des consignes .....	109
Figure 6.8	Évolution de la position des actuateurs.....	110
Figure 6.9	Angles de déflexion commandés aux actuateurs .....	110
Figure 6.10	Évolution des angles $\phi$ et $\theta$ (implémentation de la loi de commande de l'axe latéral uniquement avec les autres axes pris par défaut) .....	112
Figure 6.11	Évolution d'autres variables (implémentation de la loi de commande de l'axe latéral uniquement avec les autres axes pris par défaut) .....	112
Figure 6.12	Évolution des angles $\phi$ et $\theta$ (implémentation des lois de commande de l'axe latéral et de lacet avec les autres axes pris par défaut).....	113
Figure 6.13	Évolution d'autres variables (implémentation des lois de commande de l'axe latéral et de lacet avec les autres axes pris par défaut).....	114



## **LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES**

ACAH	Attitude Command with Attitude Hold
ADS	Aeronautical Design Standard
AQ	Attitude Quickness
BF	Body-fixed Frame
CDG	Centre de gravité
CIFER	Comprehensive Identification from Frequency Response
CONDUIT	Control Designer's Unified Interface
DLR	Deutsches Zentrum für Luft- Und Raumfahrt
EF	Earth-fixed Frame
EA	Eingenstructure assignment
FSQP	Feasible Sequential Quadratic Programming
FORECAST	Full Order Rotorcraft Evaluation
GENHEL	General Helicopter Flight Dynamics
HQ	Handling Qualities
HP	Hub Plane
MOPS	Multi-Objective Parametric Synthesis
MTE	Mission Task Element
OFE	Operational Flight Envelopes
PD	Phase Delay
RC	Rate Command
RIPTIDE	Real-time Interactive Prototype Technology Integration/Development Environment

SF	Spatial frame
TPP	Tip path plane
TRCPH	Translational Rate Command with Position Hold
UAV	Unmanned Aerial Vehicle
UCE	Usable Cue environment

## LISTE DES SYMBOLES ET UNITÉS DE MESURE

### **Vitesse angulaire rad/s (radians/s)**

$p$	Vitesse angulaire de roulis (établie dans BF)
$q$	Vitesse angulaire de tangage (établie dans BF)
$r$	Vitesse angulaire de glissement (établie dans BF)
$\dot{\phi}$	Vitesse angulaire de roulis d'Euler (établie dans SF)
$\dot{\theta}$	Vitesse angulaire de tangage d'Euler (établie dans SF)
$\dot{\psi}$	Vitesse angulaire de glissement d'Euler (établie dans SF)
$\Omega$	Vitesse angulaire du rotor principal

### **Vitesse de translation m/s (mètre/s)**

$u$	Vitesse longitudinale (relative à l'air)
$v$	Vitesse latérale (relative à l'air)
$w$	Vitesse verticale (relative à l'air)
$w_b$	Vitesse de la pale du rotor principal par rapport à l'air
$w_i$	Vitesse de l'air induite à travers la TPP

### **Angles rad (radians)**

$\phi$	Angle de roulis d'Euler
$\theta$	Angle de tangage d'Euler
$\psi$	Angle de glissement d'Euler
$i$	Angle d'incidence de la pale
$\beta_{lc}$	Angle de battement longitudinal
$\beta_{ls}$	Angle de battement latéral
$u_{coll}$	Entrée de contrôle du collectif
$u_{lat}$	Entrée de contrôle latérale
$u_{long}$	Entrée de contrôle longitudinale

### **Longueur m (mètre)**

$c$	Corde de la pale
$h_m$	Distance entre le CDG et le rotor principal (au long de l'axe Z de BF)
$h_t$	Distance entre le CDG et le rotor de la queue (au long de l'axe Z de BF)
$l_m$	Distance entre le CDG et le rotor principal (au long de l'axe X de BF)
$l_t$	Distance entre le CDG et le rotor de la queue (au long de l'axe X de BF)
$y_m$	Distance entre le CDG et le rotor principal (au long de l'axe Y de BF)
$R$	Rayon du rotor principal

**Forces N (Newton)**

$f_x^B$	Composante de la force agissant sur l'hélicoptère au long de l'axe X de BF
$f_y^B$	Composante de la force agissant sur l'hélicoptère au long de l'axe Y de BF
$f_z^B$	Composante de la force agissant sur l'hélicoptère au long de l'axe Z de BF
$Q_{MR}$	Traînée du rotor principal
$T_{MR}$	Poussée du rotor principal
$T_{TR}$	Poussée du rotor de la queue
$u_{TR}$	Entrée de contrôle du rotor de la queue

**Moments N.m (newton-mètre)**

$L$	Moment agissant sur l'hélicoptère par rapport à l'axe X
$M$	Moment agissant sur l'hélicoptère par rapport à l'axe Y
$N$	Moment agissant sur l'hélicoptère par rapport à l'axe Z

**Masse kg (kilogramme)**

$m$	Masse de l'hélicoptère
-----	------------------------

**Accélération m/s<sup>2</sup> (mètre par seconde au carré)**

$g$	Accélération de la pesanteur
-----	------------------------------

**Masse volumique kg/m<sup>3</sup> (kilogramme par mètre cube)**

$\rho$	Masse volumique de l'air
--------	--------------------------

## INTRODUCTION

Le contrôle des aéronefs modernes est une tâche complexe pour les ingénieurs étant donné l'aspect multidisciplinaire qui caractérise ce domaine : qualités de vol, modélisation de la dynamique de l'appareil, architecture des lois de commande, dynamique des senseurs et des actuateurs, tests au sol et en vol et considération des critères de performance etc.

En effet, dès la phase de conception du système de contrôle, il est essentiel de considérer les caractéristiques désirées en ce qui concerne la réponse du système, et ce, dans le but de respecter un ensemble de critères de performance.

L'objectif ultime de tout système de contrôle est, en effet, de mettre à disposition du pilote des moyens de contrôler l'aéronef en assurant la sécurité et l'efficacité sur toute l'enveloppe de vol ce qui signifie la satisfaction des qualités de vol.

Dans le cas des hélicoptères, l'ADS-33, (*Aeronautical design standard*), (Anon, 2000) développé par l'armée américaine, constitue un référentiel élaboré pour les spécifications à respecter (comportement du système) afin de garantir la satisfaction des qualités de vol requises.

Les méthodes classiques de design des systèmes de contrôle pour les hélicoptères reposent sur les étapes citées ci-dessous :

- Une analyse des modes propres de l'aéronef en boucle ouverte;
- Choix d'une structure de contrôle et conception d'une loi de commande permettant d'améliorer la stabilité de l'appareil;
- Des simulations non linéaires sont effectuées afin de vérifier les caractéristiques de la réponse dynamique de l'aéronef;
- Des retouches de gains sont faites;
- Les différents découplages sont traités;

- Des boucles locales de stabilisation sont réalisées.

Suite à ce processus, la vérification, a posteriori, de la satisfaction des spécifications (ADS-33) est menée. Ce qui conduit, inéluctablement, à des situations d'essais et erreurs causant des retombées néfastes sur la durée et le coût du processus de design du système de contrôle. Dans le cadre de la présente recherche, notre objectif est de procéder de manière inverse aux approches traditionnelles.

Notre point de départ pour le design d'un système de contrôle de vol pour hélicoptère sera le choix d'un ensemble cohérent de spécifications à respecter, tirées de l'ADS-33. Ensuite, la synthèse ainsi que l'optimisation de lois de commande seront effectuées afin que la réponse du système, en boucle fermée, respecte ces spécifications.

Pour ce faire, l'objectif de la recherche bibliographique sera une analyse de la littérature en rapport avec la conception des systèmes de contrôle pour hélicoptères tout en privilégiant les structures de commande dont l'aspect conceptuel va nous permettre de trouver des passerelles entre les spécifications à respecter et les paramètres des lois de contrôle à mettre en œuvre.

Étant donné que plusieurs sous-spécifications des normes de l'ADS-33 sont présentées sous forme de gabarit dans le plan complexe, la performance du système de contrôle sera évaluée, pour chaque axe en utilisant les critères du domaine fréquentiel en particulier la bande passante et le retard de phase.

L'objectif principal de ce travail de maîtrise est donc d'intégrer un ensemble cohérent de qualités de vol désirées, reflétées par des critères dans le domaine fréquentiel, dans la synthèse des lois de commande pour hélicoptères, permettant de répondre aux multiples exigences de ces qualités de vol de façon globale et sûre.

La conception d'un tel système de contrôle de vol pour hélicoptères, capable de respecter les qualités de vol et les marges de stabilité, stipulées par l'ADS-33, assure une faible charge de travail pour le pilote, une plus grande efficacité et une sécurité accrue pendant les différentes opérations et ce pour toutes les conditions météorologiques et de visibilité.

Pour atteindre notre objectif, nous allons utiliser le modèle linéaire de la dynamique de l'hélicoptère considéré, en l'occurrence le BELL-412. Ce modèle linéaire sera déduit, pour la condition de vol (*surplace*) ou *Hover*, du logiciel HELISIM (voir chapitre 6). Par ailleurs, l'ADS-33 indique les réponses types recommandées ainsi que les performances à atteindre pour cette condition de vol. L'optimisation se traduit dès lors par la détermination d'une matrice de compensation et d'une matrice de retour d'état (constituant le contrôleur), ce qui amène le comportement du système, en boucle fermée, à se rapprocher le plus possible du comportement désiré et qui est reflété par le choix de valeurs et de vecteurs propres désirés (les vecteurs réalisables seront calculés en utilisant les techniques d'affectation modale).

Le premier chapitre sera dédié à la revue de littérature pour présenter les travaux antérieurs en rapport avec l'intégration des spécifications de l'ADS-33 dans le processus de design du système de contrôle pour hélicoptères. Dans le deuxième chapitre, la modélisation de la dynamique de l'hélicoptère sera traitée tout en expliquant comment le modèle linéaire peut être déduit. Par ailleurs, le choix et l'analyse des spécifications contenues dans l'ADS-33 seront effectués dans le troisième chapitre. Ensuite, le quatrième chapitre sera une introduction aux techniques d'affectation de la structure modale. Dans le cinquième chapitre, la structure du contrôleur et le calcul des matrices qui le constituent seront traités, en outre des simulations sur MATLAB seront menées pour vérifier le respect par le système de contrôle des qualités de vol requises. Et finalement, le chapitre 6 sera consacré au logiciel HELISIM et aux aspects de l'implémentation.



## CHAPITRE 1

### REVUE DE LITTERATURE

L'intérêt pour le développement d'outils permettant l'optimisation des lois de contrôle pour les aéronefs en rapport avec les performances à atteindre et les spécifications à respecter s'est manifesté dans les années mille neuf cent quatre-vingt, en parallèle, en Allemagne et aux États Unis. Dans ce qui suit, nous allons présenter les travaux les plus marquants dans ce domaine.

#### 1.1 MOPS (Multi-Objective Parametric Synthesis)

Du côté allemand, au sein du DLR (*Deutsches Zentrum für Luft- Und Raumfahrt*), Centre Allemand pour l'Aéronautique et l'Aérospatiale, Kreisselmeir et Steinhauser ont proposé, en 1979, d'utiliser une structure d'optimisation basée sur des inégalités « Min-Max » ainsi qu'un resserrement progressif des objectifs du design. Ce travail précurseur s'est concrétisé par un article dans '*International Journal of Control*', (Kreisselmeier et Steinhauser, 1983), contenant une explication détaillée de l'approche de design proposée ainsi qu'une étude de cas du système de contrôle de vol du F-4C. Ce travail a évolué pour donner un outil appelé MOPS, (Grübel et Joos, 1997), utilisé notamment dans le cadre du programme *Eurofighter* (Moritz et Osterhuber, 2006).

MOPS est un environnement logiciel permettant l'assistance à la conception des lois de commande en aidant l'ingénieur à traduire correctement son problème de conception sous forme d'optimisation multi-objectif. Le problème de conception étant la minimisation simultanée de plusieurs critères de performance en présence de différentes contraintes, cet ensemble d'exigences est traduit en termes d'inégalités de type « Min-Max » prises en compte dans un processus d'optimisation utilisant des solveurs à base de gradient (*gradient-based solvers*). L'optimisation par détection des frontières de Pareto est également supportée par MOPS, ce qui en fait un outil puissant et interactif puisqu'il affiche les valeurs normalisées des critères pendant les étapes intermédiaires de l'optimisation (itérations).

Ainsi, le concepteur sera à même de déterminer les conflits entre les critères nécessitant des compromis durant le processus de conception.

## 1.2 CONDUIT (Control Designer's Unified Interface)

Aux États Unis, l'expérience a commencé par une thèse de doctorat de Nye à l'université du Berkeley (Nye, 1984) et qui a mené à l'élaboration d'un outil appelé *Delight Package*, suite à quoi, Fan et al. de l'université du Maryland ont développé CONSOL-OPTCAD en 1991, et qui est un FSQP (*Feasible Sequential Quadratic Programming*) : outil logiciel à base d'un algorithme d'optimisation permettant de résoudre des problèmes de conception (dans le domaine du contrôle entre autres) en traitant des fonctions sous forme d'objectifs ou de contraintes. L'armée américaine s'est intéressée dès lors à ces travaux en nouant un partenariat de recherche avec l'Université du Maryland ce qui a conduit au développement de l'outil de conception et d'optimisation des lois de pilotage CONDUIT.

L'approche de CONDUIT a été résumée par Mark B. Tischler, Christina M. Ivler et M. Hossein Mansur (Tischler et al., 2008). Cette approche repose sur CONSOL-OPTCAD dans le processus d'optimisation en offrant une interface interactive pour le designer lui évitant ainsi de créer des fichiers de définition du problème et des contraintes sur MATLAB, tout en lui permettant de choisir le modèle ainsi que la structure du contrôleur. De ce fait, lors de l'optimisation, les paramètres qui seront réglés pour satisfaire les contraintes vont dépendre de cette structure (voir la figure 1.1).

Ces paramètres peuvent être les gains du régulateur comme ils peuvent être les caractéristiques du modèle de commande (amortissement et fréquence naturelle).

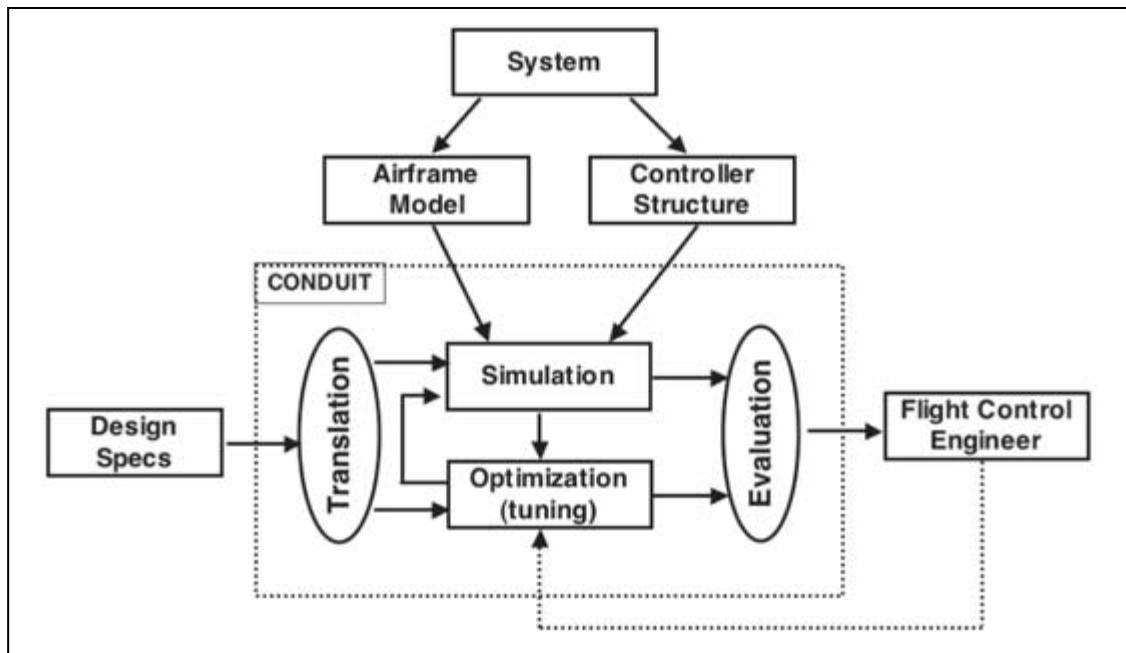


Figure 1.1 Séparation entre la structure modèle/contrôleur et le réglage des paramètres de la conception  
Tirée de Tischler et al. (2008, p. 3)

CONDUIT est un outil intéressant, cependant, il n'est pas disponible à l'achat puisqu'il est la propriété de l'armée américaine. Néanmoins, la connaissance des méthodes utilisées dans son développement permet d'aborder ce sujet de recherche en s'appuyant sur une référence déjà établie et qui a fait ses preuves.

Ceci étant, l'utilisation de CONDUIT requiert plusieurs logiciels à l'appui, en l'occurrence :

- CIFER (Comprehensive Identification from Frequency Response);
- GenHel (General Helicopter Flight Dynamics Simulation);
- RIPTIDE (Real-time Interactive Prototype Technology Integration/Development);
- FORECAST (Full Order Rotorcraft Evaluation Code and Simulation Tool).

Par conséquent, il est clair que le développement d'une solution, qui repose sur cette approche, est très nécessiteux en temps de programmation pouvant s'étaler sur plusieurs années.

### 1.3 Travaux de Eicher Low et William L.Garrard

Dans un article paru dans *Journal of Guidance, Control And Dynamics* en 1993, une méthodologie visant le design des lois de contrôle pour hélicoptère dans le but d'améliorer les qualités de vol (HQ) a été présentée (LOW et GARRARD, 1993).

La procédure proposée utilise les techniques de placement de structure propre (*Eingenstructure assignment*) pour la conception de lois de contrôle qui visent le découplage des variables  $p$ ,  $q$ ,  $r$  (vitesses angulaires de roulis, tangage et lacet) et  $w$  (vitesse verticale) afin qu'ils suivent, indépendamment, leurs consignes respectives (poursuite de vitesse). Cette méthode procure aussi des bandes passantes appropriées dans tous les canaux et stabilise les boucles ouvertes en basses fréquences.

Après fermeture des boucles, les réponses des vitesses angulaires et de la vitesse verticale, aux commandes, sont approximées par quatre systèmes découplés de premier ordre.

Différentes architectures de contrôle, comme ACAH (*Attitude Command with Attitude Hold*), peuvent être facilement réalisées, par simple retour d'état en plus d'un gain, greffés sur l'entrée et sortie du canal choisi (voir la figure 1.2).

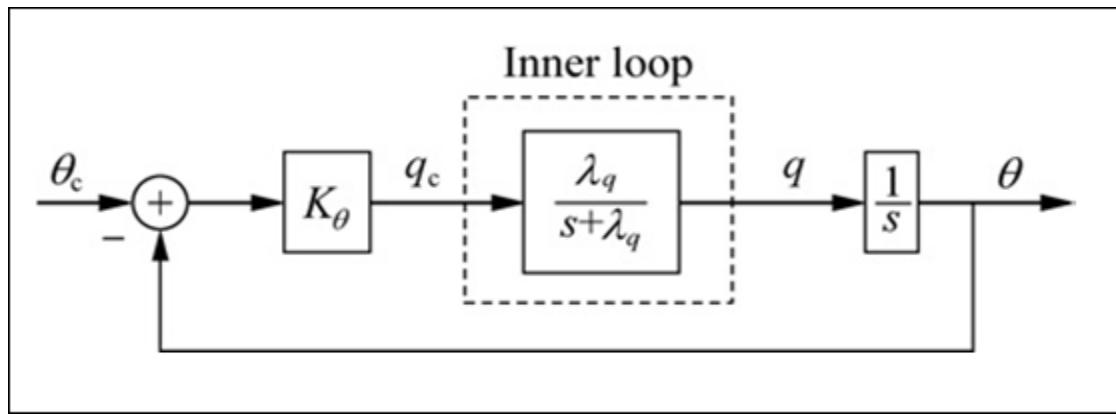


Figure 1.2 Canal tangage

Tirée de Song Yanguo (2009, p. 253)

La méthodologie proposée par Eicher low et William L.Garrard a été appliquée récemment par les chercheurs Song Yanguo et Wang Huanjin de l'Université Nanjing d'aéronautique en chine (Yanguo et Huanjin, 2009) dans la conception d'un système de contrôle pour hélicoptère miniature et les résultats obtenus ont confirmé la crédibilité et l'efficacité du système de contrôle de vol élaboré avec cette approche.

La figure 1.3 illustre le schéma global de l'implantation du système de contrôle.

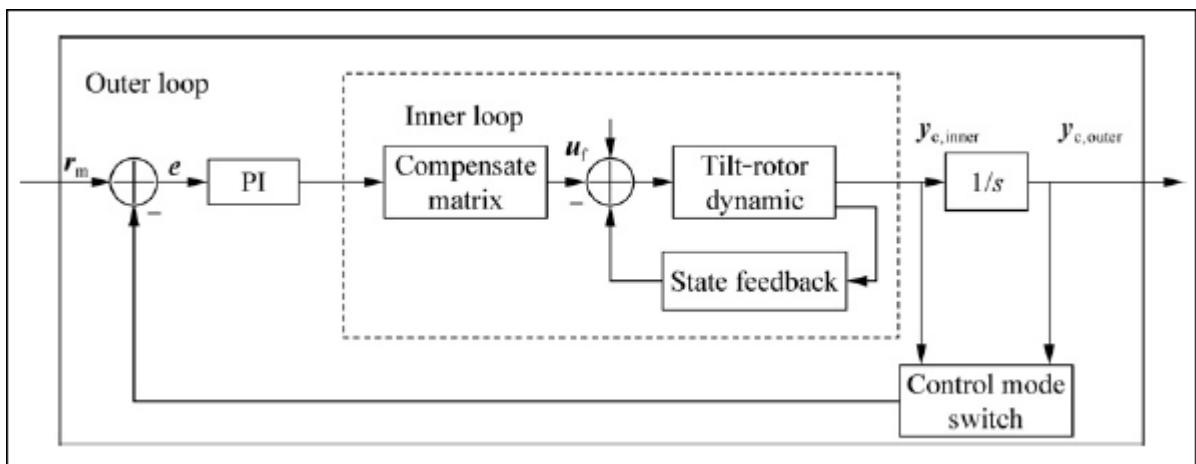


Figure 1.3 Architecture du système de contrôle en deux boucles

Tirée de Song Yanguo (2009, p. 252)

## 1.4 Travaux de Kimon P. Valavanis

Cet auteur a apporté des contributions importantes au domaine du contrôle des UAV (*Unmanned Aerial Vehicle*), en particulier les hélicoptères miniatures. Ainsi, dans son ouvrage (Valavanis, 2007), l'état de l'art de cette discipline a été dressé.

La structure de contrôle, mise en exergue par l'auteur, se caractérise par la décomposition du contrôleur en deux boucles : boucle interne et boucle externe (voir figure 1.4). Cette architecture en cascade, s'est révélée appropriée dans le cas des hélicoptères étant donné le haut degré de couplage entre les états de ces systèmes et leur instabilité en boucle ouverte.

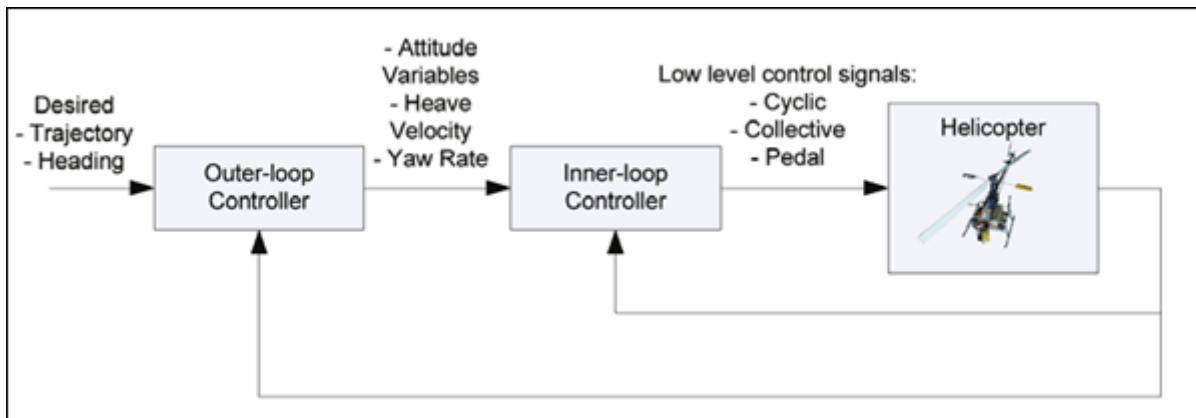


Figure 1.4 Contrôleur en cascade (boucle interne et boucle externe)

Tirée de Kimon P. Valavanis (2007, p. 88)

En effet, la boucle interne assure la stabilisation de l'hélicoptère ainsi que le découplage des variables d'états commandant l'attitude de l'appareil, alors que la boucle externe est utilisée pour le guidage, en générant les vitesses de consigne aux entrées de la boucle interne.

La dynamique de l'hélicoptère étant non linéaire, il convient de noter que les techniques reposant sur le contrôle linéaire nécessitent l'utilisation de plusieurs modèles linéarisés autour de points d'opération (régime de vol) permettant ainsi d'approximer l'évolution du système réel à travers l'espace d'état sur toute l'enveloppe de vol ce qui se traduit par le

besoin de changer les paramètres du contrôleur, dans le cas où la même structure de contrôle est préservée, on parle de technique d'échelonnement des gains ou *gain-scheduling*. Dans le cas, moins rencontré en pratique, où l'utilisation de contrôleurs de différentes structures est requise, des techniques de mélange sont alors mises en œuvre afin d'assurer des transitions en douceur d'une architecture vers une autre sans perturber le système « *Gain blending* » (voir figure 1.5).

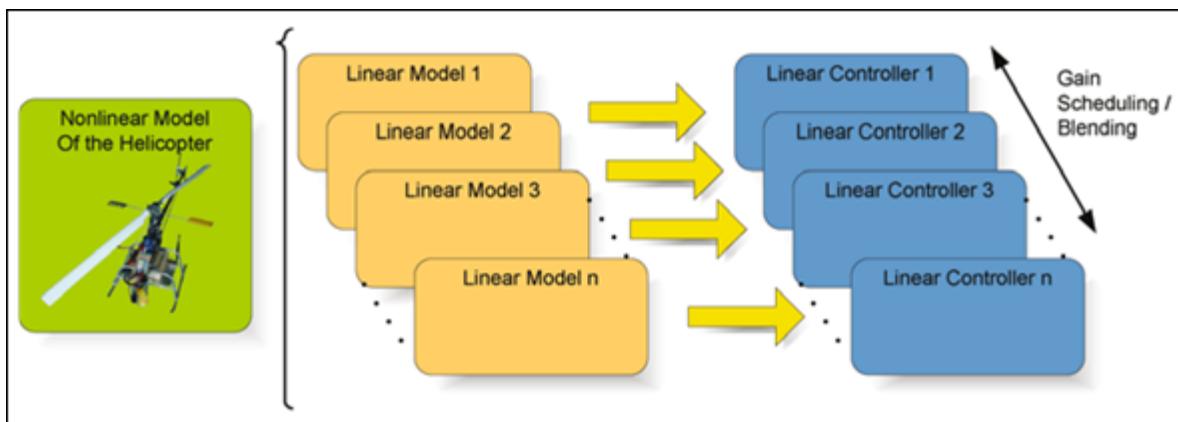


Figure 1.5 Approche linéaire

En matière de couplage, et dans le même ouvrage cité plus haut, une représentation de l'hélicoptère comme un système carré, avec quatre entrées et quatre sorties, a été faite pour illustrer le jumelage « *pairing* » entre les entrées et les sorties du système (voir figure 1.6).

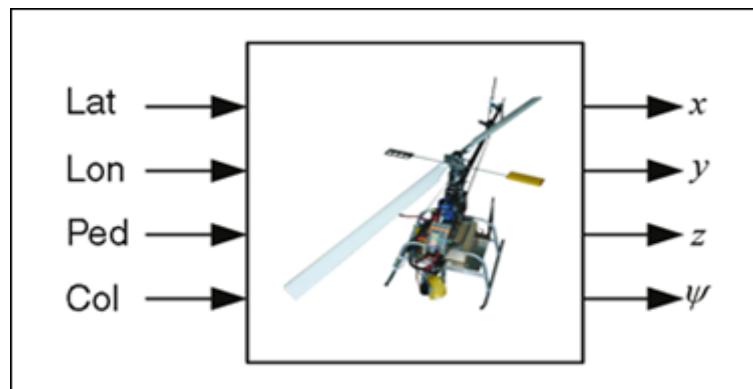


Figure 1.6 L'hélicoptère vu comme un système carré  
(exemple de jumelage)

Ainsi, l'auteur a effectué une comparaison entre les jumelages les plus utilisés en pratique, dans l'objectif de déterminer le plus avantageux, pour le cas de l'appareil Yamaha R-50 en condition de vol *Hover*. La combinaison d'entrées-sorties la plus optimale, en termes de découplage entre canaux, selon cette analyse est illustrée dans le tableau 1.1.

Tableau 1.1 Jumelage optimal pour le Yamaha R-50  
en *hover*

Entrée	Sortie correspondante
<b>Ulat</b> (Entrée cyclique latérale)	Angle de roulis $\Phi$
<b>Ulon</b> (Entrée cyclique longitudinale)	Angle de tangage $\theta$
<b>Uped</b> (Entrée pédales)	Vitesse angulaire de glissement $r$
<b>Ucoll</b> (Entrée collective)	Vitesse verticale $w$

## 1.5 Travaux effectués au sein de l'ÉTS

Sous la direction de Madame Ouassima Akhrif, deux mémoires de maîtrise, relevant du domaine du contrôle des aéronefs et réalisés à l'ÉTS ont été étudiés dans le cadre de la présente recherche bibliographique vue leur pertinence par rapport à notre sujet.

Le mémoire de maîtrise de Raymond Abi-Elias (Abi-Elias, 1999), quoique destiné aux avions à voilure fixe, présente l'avantage de la clarté et explique en profondeur les techniques d'affectation de la structure modale. Ces techniques seront utilisées pour le calcul des gains de notre contrôleur pour hélicoptère. Ainsi, ce mémoire constituera un document de référence qui sera sollicité régulièrement au cours de ce travail.

Un autre mémoire de maîtrise, ayant fait l'objet des travaux de Madani (Madani, 2007) a retenu tout aussi bien notre intérêt puisqu'il traite la commande active des hélicoptères et nous permet de cerner l'aspect implantation des lois de commande.

## 1.6 Conclusion

La méthodologie décrite précédemment dans le paragraphe 1.2 est très pertinente par rapport à notre objectif de recherche dans le sens où elle rend aisée l'intégration des nombreuses spécifications contenues dans l'ADS-33. Cette approche qui considère l'aspect global de la dynamique de l'hélicoptère est très avantageuse dans le sens où elle traite le comportement de l'hélicoptère dans son ensemble et présente l'avantage de la compacité de l'étude et de fait, l'optimisation de la structure du contrôleur vise l'atteinte des objectifs de stabilité et de performance, stipulées par l'ADS-33 notamment, sur tous les axes en même temps. Nous allons donc nous baser sur cette configuration pour la détermination de la loi de contrôle dans ce travail de recherche.



## CHAPITRE 2

### MODÉLISATION DE LA DYNAMIQUE DE VOL DE L'HÉLICOPTÈRE

#### 2.1 Généralités sur l'hélicoptère

L'histoire de l'hélicoptère a commencé au début du vingtième siècle, en revanche, cet aéronef n'a montré son utilité, militaire notamment, qu'à partir de la seconde guerre mondiale. Dès lors, l'utilisation de l'hélicoptère a connu son essor, en particulier au sein de l'armée, des services de police et de douane ainsi que dans le domaine civil. Des apports technologiques (propulsion par turbomoteurs et pilote automatique entre autres) ont amélioré l'efficacité et la sécurité de cet appareil.

##### 2.1.1 Principe de propulsion

Le mouvement de l'hélicoptère est généré, en général, par deux rotors (le rotor principal et le rotor installé sur la queue de l'appareil) comme illustré sur la figure 2.1.

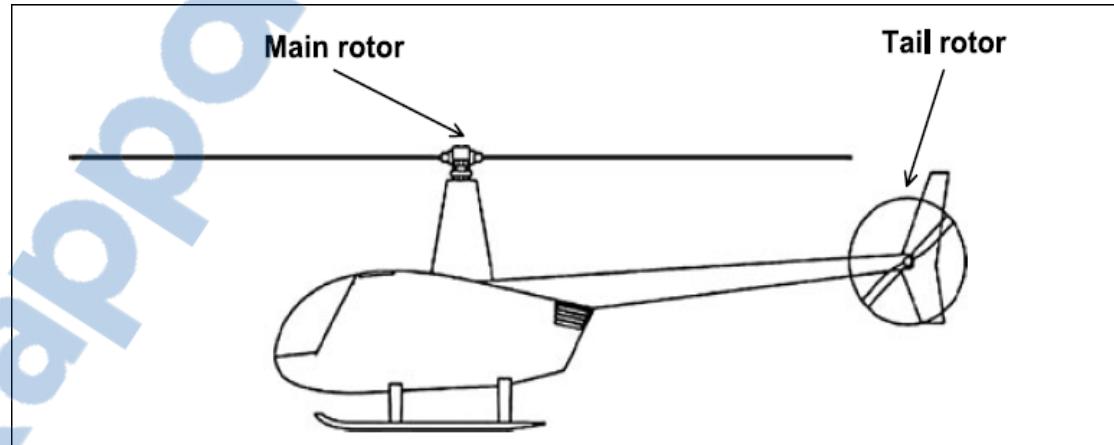


Figure 2.1 Configuration typique d'un hélicoptère

Le rotor principal produit la poussée nécessaire pour le déplacement vertical de l'hélicoptère tandis que le rotor de la queue compense le moment produit par le premier et contrôle la

direction de l'appareil. D'autre part, le changement des angles d'attitude du fuselage de l'hélicoptère conduit à l'inclinaison du rotor principal ce qui produit les forces de propulsion à même d'assurer le mouvement longitudinal et latéral.

### 2.1.2 Commandes du vol

Il existe quatre commandes associées au pilotage de l'hélicoptère, le vecteur entrée de contrôle est défini par  $\delta = [\delta_{coll}, \delta_{lat}, \delta_{long}, \delta_{TR}]^T$ , avec  $\delta_{coll}$  et  $\delta_{TR}$  (palonnier) sont les entrées de contrôle du rotor principal et du rotor de la queue respectivement, ces deux entrées contrôlent la poussée des rotors par un changement uniforme de l'angle d'incidence de toutes les pales en même temps. Ainsi  $\delta_{coll}$  commande les montées et les descentes de l'hélicoptère tandis que  $\delta_{TR}$  commande le déplacement de l'appareil autour de son axe de lacet. Les deux autres commandes  $\delta_{lat}$ ,  $\delta_{long}$  sont les commandes du cyclique de l'hélicoptère (accessibles au pilote à l'aide du manche vertical), ces deux dernières commandes agissent sur l'angle d'incidence de toutes les pales du rotor principal permettant le mouvement de l'appareil sur ses axes de roulis et tangage respectivement.

### 2.1.3 Rotor principal

En général, l'entraînement des pales du rotor principal s'effectue à l'aide d'une boîte de transmission. Par ailleurs, le rotor tourne toujours à une vitesse angulaire constante, ce qui fait que le seul moyen d'agir sur son comportement reste le changement de l'angle d'incidence des pales, angle formé entre le vent relatif et la corde de la pale, comme illustré sur la figure 2.2.

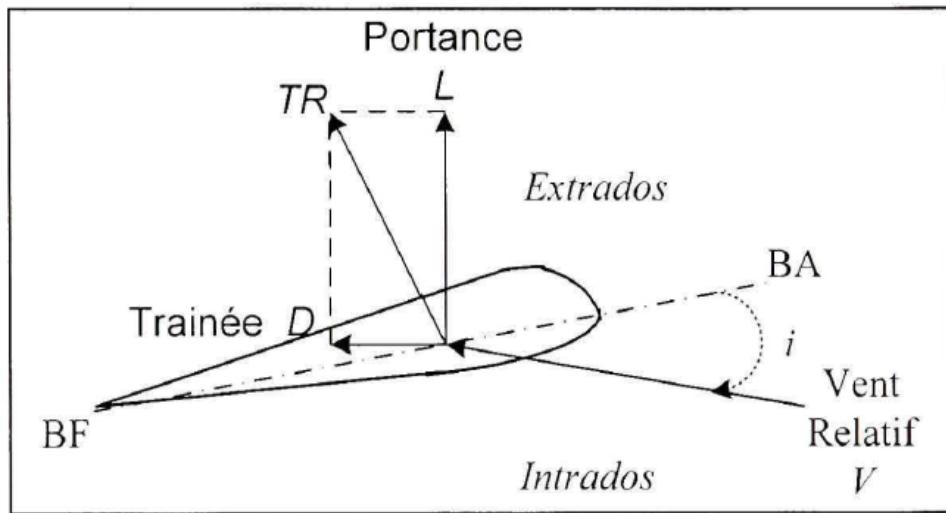


Figure 2.2 Les forces aérodynamiques agissant sur une pale  
Tirée de Shawn Coyle (1996, p. 9)

Où, TR est la force aérodynamique engendrée, L est la portance, D la traînée, i l'angle d'incidence de la pale, BA le bord d'attaque et BF le bord de fuite. Par ailleurs, les forces L, D et TR dépendent de l'incidence i.

#### 2.1.4 Rotor de la queue

Le rotor de la queue, ou rotor anti-couple, permet de contrecarrer le couple de réaction, créé par la rotation du rotor principal et tendant à faire tourner la cellule de l'appareil sur elle-même. En outre, ce rotor anti-couple contrôle le mouvement de l'hélicoptère sur son axe de lacet.

#### 2.2 Équations de mouvement d'un hélicoptère

Pour modéliser la dynamique d'un hélicoptère, on le considère comme étant un corps rigide ayant six degrés de liberté. La déduction des équations du mouvement fait appel aux repères suivants :

- Le repère inertiel (relatif à la terre et noté EF) ayant pour origine un point arbitrairement choisi sur la surface de la terre (l'aéroport de décollage par exemple) et ses vecteurs unitaires sont dirigés typiquement vers le nord, l'est et le centre la terre respectivement;
- Le repère relatif à l'hélicoptère (*Body Frame*) et noté BF, ayant pour origine le centre de gravité de l'appareil noté BF comme illustré ci-dessous :

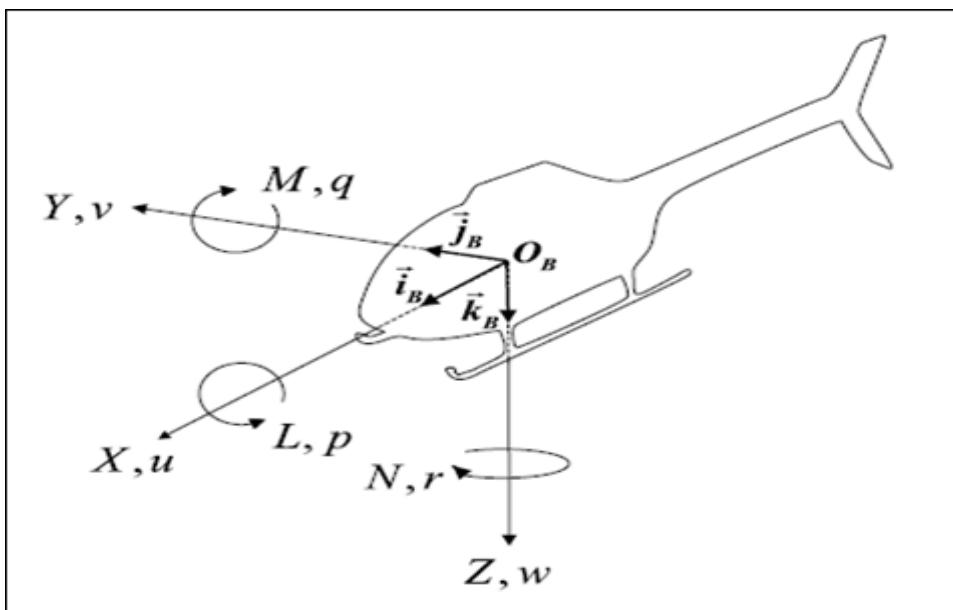


Figure 2.3 Repère relatif à l'hélicoptère

Enfin, on définit le repère spatial SF comme étant le repère orienté comme EF et ayant le même centre que BF. La vitesse de translation de l'hélicoptère, par rapport à l'air, est définie dans le repère BF par  $v^B = [u \ v \ w]$ , les vitesses de rotation par rapport aux axes du BF sont définies par  $w^B = [p \ q \ r]^T$  et les angles d'Euler sont notées  $\Theta = [\phi \ \theta \ \Psi]^T$ , ces dernières caractérisent les angles entre le BF et le SF.

En outre, la somme des forces, établie dans BF, agissant sur l'appareil est notée  $F^B = [f_x^B \ f_y^B \ f_z^B]^T$ . D'autre part, la somme des moments, exprimés dans BF, s'exerçant sur l'hélicoptère autour de son centre de gravité est notée  $\tau^B = [L \ M \ N]^T$ .

### 2.2.1 Matrice de rotation

La matrice de rotation est définie par :

$$R_{sb}(\Theta) = \begin{pmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{pmatrix} \quad (2.1)$$

Cette matrice permet d'exprimer un vecteur dans le référentiel SF à partir de ses coordonnées dans le référentiel BF.

Ainsi, si la vitesse par rapport au repère inertiel est :  $V = \begin{bmatrix} v_x^I \\ v_y^I \\ v_z^I \end{bmatrix}$

Alors,  $V = R_{sb}(\Theta) \begin{bmatrix} u \\ v \\ w \end{bmatrix}$

La matrice  $R_{bs}(\Theta) = R_{sb}^{-1}(\Theta) = R_{sb}^T(\Theta)$  (puisque la matrice  $R_{sb}(\Theta)$  est orthonormale (Hald et al., 2005)) permet d'effectuer l'opération inverse.

### 2.2.2 Matrice de transformation

La relation entre les deux vecteurs  $w^B$  et  $\Theta$  est comme suit (Hald et al., 2005) :

$$w^B = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{pmatrix} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (2.2)$$

Ou encore :

$$\overset{\square}{w^B} = P_{bs}(\Theta) \cdot \overset{\square}{\Theta} \quad (2.3)$$

La matrice  $P_{bs}(\Theta)$  est la matrice de transformation du repère SF vers le repère BF.

L'inversion de cette matrice donne

$$P_{sb}(\Theta) = P_{bs}^{-1}(\Theta) = \begin{pmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & \sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{pmatrix} \quad (2.4)$$

L'équation (2.3) donne alors :

$$\overset{\square}{\Theta} = P_{sb}(\Theta) \cdot \overset{\square}{w} \quad (2.5)$$

### 2.2.3 Équation du moment angulaire et de l'accélération rectiligne

L'équation du moment par rapport au centre de gravité est la suivante (Hald et al., 2005) :

$$\overset{\square}{H} = \left\{ \frac{dH}{dt} \right\} + \overset{\square}{w} \times H \quad (2.6)$$

Avec  $H$  le vecteur moment angulaire et  $\overset{\square}{H} = \overset{\square}{\tau^B}$ . Par ailleurs, nous avons :

$$H = I \cdot \overset{\square}{w} \quad (2.7)$$

$I$  étant la matrice d'inertie. Pour simplifier, cette matrice est supposée composée juste des éléments diagonaux :

$$I = \begin{pmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{pmatrix} \quad (2.8)$$

Le développement de l'équation (2.6) en remplaçant  $H$  par  $I.w$  et en supposant  $\frac{dI}{dt} = 0$  (l'hélicoptère est considéré comme un corps rigide) conduit à :

$$\overset{\square}{w} = I^{-1}(\tau^B - w \times (I.w)) \quad (2.9)$$

D'autre part, l'expression de l'accélération rectiligne est la suivante (Hald et al., 2005) :

$$\overset{\square}{V}^B = \frac{1}{m} F^B - w \times V^B \quad (2.10)$$

$m$  étant la masse de l'hélicoptère et  $V^B$  est un vecteur, établi dans BF, et contenant les vitesses de translation de l'hélicoptère par rapport à EF.

#### 2.2.4 Récapitulatif des équations de mouvement de l'hélicoptère

L'équation (2.5) donne :

$$\overset{\square}{\Theta} = \begin{pmatrix} \overset{\square}{\phi} \\ \overset{\square}{\theta} \\ \overset{\square}{\psi} \end{pmatrix} = \begin{bmatrix} p + \sin \phi \cdot \tan \theta \cdot q + \cos \phi \cdot \tan \theta \cdot r \\ \cos \phi \cdot q - \sin \phi \cdot r \\ \frac{\sin \phi}{\cos \theta} q + \frac{\cos \phi}{\cos \theta} r \end{bmatrix} \quad (2.11)$$

L'équation (2.9) conduit à :

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \frac{(I_{yy} - I_{zz}).q.r + L}{I_{xx}} \\ -\frac{(I_{xx} - I_{zz}).p.r - M}{I_{yy}} \\ \frac{(I_{xx} - I_{yy}).p.q + N}{I_{zz}} \end{bmatrix} \quad (2.12)$$

Et finalement l'équation (2.10) donne :

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \frac{f_x^B}{m} + v.r - w.q \\ \frac{f_y^B}{m} - u.r + w.p \\ \frac{f_z^B}{m} + u.q - v.p \end{bmatrix} \quad (2.13)$$

## 2.3 Forces et moments s'exerçant sur l'hélicoptère

### 2.3.1 Forces

Les forces qui s'exercent sur l'hélicoptère sont générées par la poussée des deux rotors en plus de la force de gravité. Le mouvement des pales du rotor principal de l'hélicoptère s'effectue dans un plan noté TPP (*Tip path plane*) qui est défini par deux angles  $\beta_{lc}$  et  $\beta_{ls}$  dans les directions longitudinale et latérale respectivement comme illustré sur les figures 2.4 et 2.5 ( $T_{MR}$  désigne la poussée du rotor principal). Ainsi, la force s'exerçant sur l'hélicoptère et causée par la poussée du rotor principal s'écrit dans le repère BF :

$$F_{MR}^B = \begin{bmatrix} f_{x,MR}^B \\ f_{y,MR}^B \\ f_{z,MR}^B \end{bmatrix} = \begin{bmatrix} -T_{MR} \sin \beta_{lc} \\ T_{MR} \sin \beta_{ls} \\ -T_{MR} \cos \beta_{lc} \cos \beta_{ls} \end{bmatrix} \quad (2.14)$$

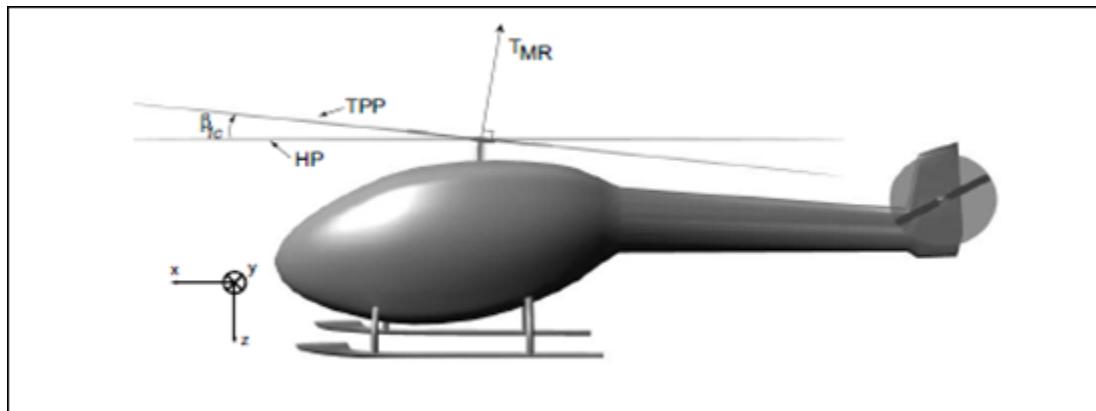


Figure 2.4 L'angle  $\beta_{lc}$  entre HP (Hub Plane) et TPP dans la direction longitudinale

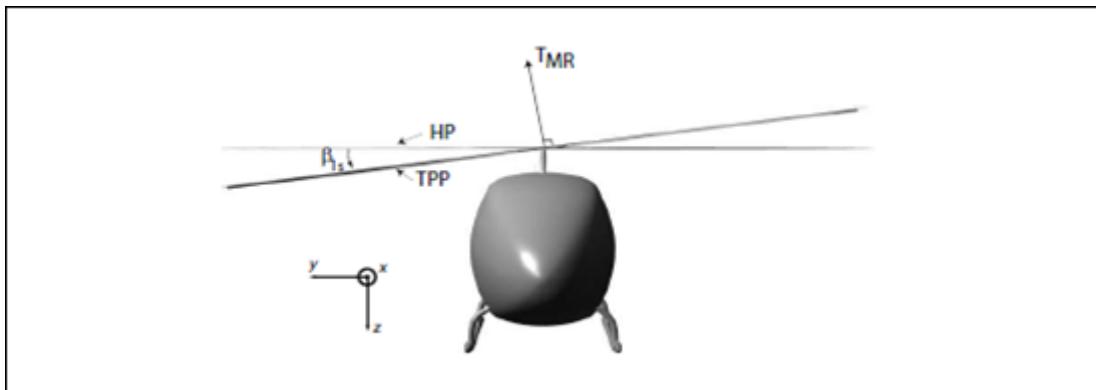


Figure 2.5 L'angle  $\beta_{ls}$  entre HP et TPP dans la direction latérale

Par ailleurs, le rotor de la queue génère une poussée dans la direction (y) de BF uniquement, ce qui donne :

$$F_{TR}^B = \begin{bmatrix} f_{x,TR}^B \\ f_{y,TR}^B \\ f_{z,TR}^B \end{bmatrix} = \begin{bmatrix} 0 \\ T_{TR} \\ 0 \end{bmatrix} \quad (2.15)$$

D'autre part, l'expression de la force de gravité s'exerçant sur l'hélicoptère, exprimée dans le repère spatial (SF), est la suivante :

$$F_g^S = \begin{bmatrix} f_{x,g}^S \\ f_{y,g}^S \\ f_{z,g}^S \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (2.16)$$

Ce qui donne :

$$F_g^B = R_{bs}(\Theta) \cdot \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = \begin{bmatrix} -\sin \theta \cdot mg \\ \sin \phi \cdot \cos \theta \cdot mg \\ \cos \phi \cdot \cos \theta \cdot mg \end{bmatrix} \quad (2.17)$$

En conclusion, la résultante des forces s'exerçant sur l'hélicoptère, par rapport au repère BF s'écrit :

$$F^B = \begin{bmatrix} -T_{MR} \sin \beta_{lc} - \sin \theta \cdot mg \\ T_{MR} \sin \beta_{ls} + T_{TR} + \sin \phi \cdot \cos \theta \cdot mg \\ -T_{MR} \cos \beta_{lc} \cos \beta_{ls} + \cos \phi \cdot \cos \theta \cdot mg \end{bmatrix} = \begin{bmatrix} F_x^B \\ F_y^B \\ F_z^B \end{bmatrix} \quad (2.18)$$

### 2.3.2 Moments

Les moments qui s'appliquent sur l'hélicoptère sont générés par le rotor principal, le rotor de la queue ainsi que par la traînée sur le rotor principal. Les grandeurs utilisées dans le calcul des moments sont illustrées sur les figures correspondantes.

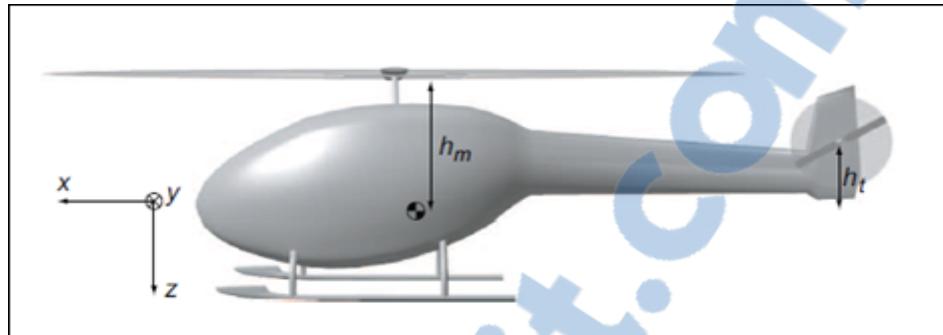


Figure 2.6 Signification de  $h_m$  et  $h_t$

Dans cette vue de côté de l'hélicoptère (Figure 2.6), sur l'axe z,  $h_m$  est la distance entre le CDG (centre de gravité) et le rotor principal alors que  $h_t$  représente la distance entre le CDG et le rotor de la queue.

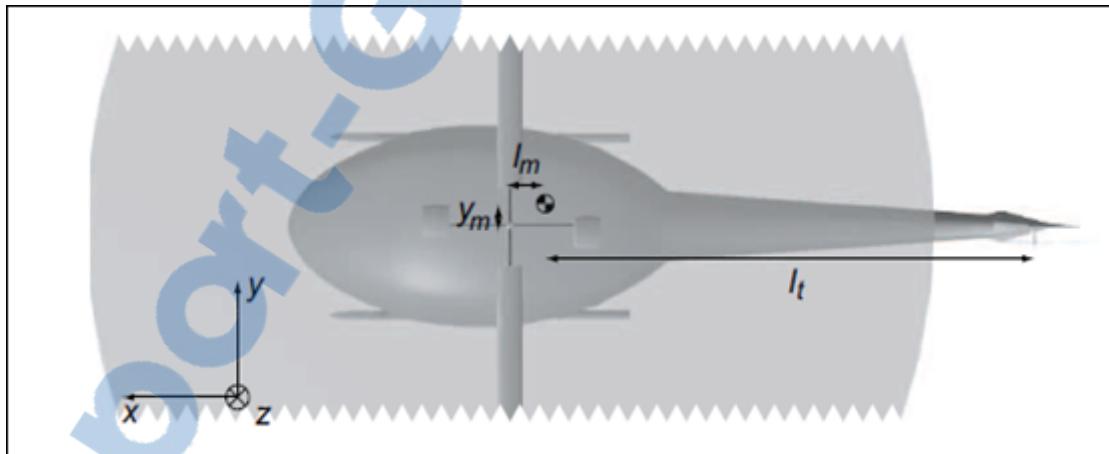


Figure 2.7 Signification de  $l_t$ ,  $l_m$  et  $y_m$

Dans la vue d'en haut de l'hélicoptère (Figure 2.7), sur l'axe x,  $l_t$  est la distance entre le CDG et le rotor de la queue et  $l_m$  la distance entre le CDG et le rotor principal. Par rapport à l'axe y,  $y_m$  est la distance entre CDG et le rotor principal. Ainsi, la somme des moments agissant sur l'hélicoptère s'écrit (Hald et al., 2005) :

$$\tau^B = \begin{bmatrix} f_{y,MR}^B \cdot h_m - f_{z,MR}^B \cdot y_m + f_{y,TR}^B \cdot h_t + Q_{MR} \cdot \sin \beta_{1c} \\ -f_{x,MR}^B \cdot h_m - f_{z,MR}^B \cdot l_m - Q_{MR} \cdot \sin \beta_{1s} \\ f_{x,MR}^B \cdot y_m + f_{y,MR}^B \cdot l_m - f_{y,TR}^B \cdot l_t + Q_{MR} \cdot \cos \beta_{1c} \cdot \cos \beta_{1s} \end{bmatrix} = \begin{bmatrix} L \\ M \\ N \end{bmatrix} \quad (2.19)$$

$Q_{MR}$  étant la traînée générée par la rotation du rotor principal et qui dépend de  $T_{MR}$ . La traînée résultante sur les pales du rotor principal se traduit par un moment agissant sur l'hélicoptère.

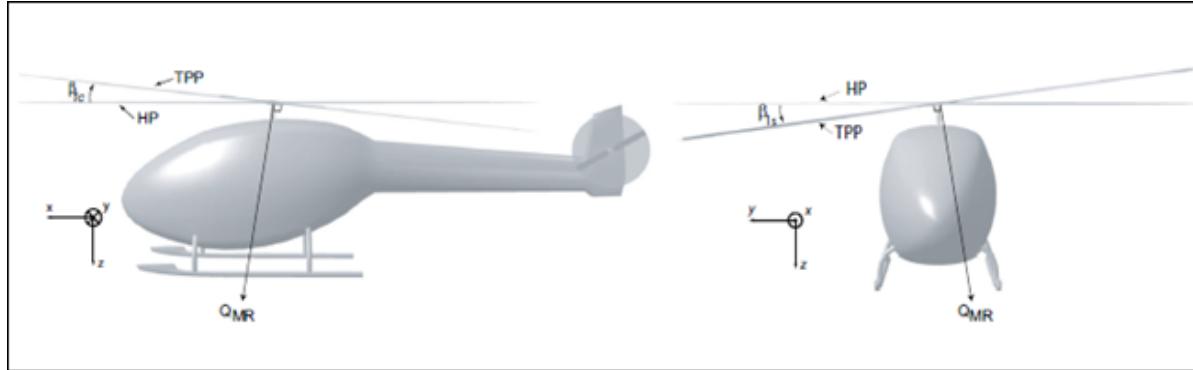


Figure 2.8 Signification de  $Q_{MR}$

## 2.4 Équations de la poussée et du battement

### 2.4.1 Poussée générée par le rotor principal

L'expression de la poussée du rotor principal (Hald et al., 2005) :

$$T_{MR} = (w_b - v_i) \cdot \frac{\rho \cdot \Omega \cdot R^2 \cdot a \cdot B \cdot c}{4} \quad (2.20)$$

Avec  $\rho$  la masse volumique de l'air,  $\Omega$  la vitesse angulaire du rotor,  $R$  le rayon du rotor,  $a$  est la pente de la courbe de montée,  $B$  le nombre de pales,  $c$  la corde de la pale,  $w_b$  la vitesse de la pale du rotor principal par rapport à l'air et  $v_i$  la vitesse de l'air induite à travers la TPP.

#### 2.4.2 Poussée générée par le rotor de la queue

La poussée générée par le rotor de la queue s'écrit (Hald et al., 2005) :

$$T_{TR} = \frac{f_{y,MR}^B \cdot l_m + f_{x,MR}^B \cdot y_m + Q_{MR} \cdot \cos \beta_{lc} \cdot \cos \beta_{ls}}{l_t} + \delta_{TR} \quad (2.21)$$

#### 2.4.3 Battement

En notant  $A_{MR}$  et  $B_{MR}$  la contribution des entrées du cyclique au tangage des pales, un modèle décrivant les expressions des angles  $\beta_{lc}(t)$  et  $\beta_{ls}(t)$  est obtenu par approximation empirique, pour un rotor vacillant à compensation par charnière (Hald et al., 2005) :

$$\begin{aligned} \beta_{lc}(t) = & \frac{3.06 \cdot 10^{-7} \cdot (-3.26 \cdot 10^6 \cdot B_{MR} \pi^2 - 816.97 \cdot v^2(t) \cdot B_{MR} - 3275.88 \cdot u(t) \cdot v_i)}{\pi^2} \\ & - \frac{3.06 \cdot 10^{-7} \cdot (2456.91 \cdot u^2(t) \cdot B_{MR} - 1637.94 \cdot A_{MR} \cdot v(t) \cdot u(t) + 1.13 \cdot 10^5 \cdot \pi \cdot p(t))}{\pi^2} \\ & - \frac{3.06 \cdot 10^{-7} \cdot (4.67 \cdot 10^5 \cdot \pi \cdot q(t) + 1.95 \cdot 10^5 \cdot \pi \cdot \delta_{coll}(t) \cdot u(t))}{\pi^2} \end{aligned} \quad (2.22)$$

$$\begin{aligned}
\beta_{1s}(t) = & -\frac{3.06 \cdot 10^{-7} \cdot (1.94 \cdot 10^5 \pi \cdot \delta_{coll}(t) \cdot v(t) - 3.26 \cdot 10^6 \pi^2 \cdot A_{MR})}{\pi^2} \\
& - \frac{3.06 \cdot 10^{-7} \cdot (1637.94 \cdot B_{MR} \cdot u(t) \cdot v(t) \cdot v_i + 4.67 \cdot 10^5 \pi \cdot p(t) - 3275.88 \cdot v_i)}{\pi^2} \quad (2.23) \\
& - \frac{3.06 \cdot 10^{-7} \cdot (2456.91 \cdot A_{MR} \cdot v^2(t) - 818.97 \cdot u^2(t) \cdot A_{MR} + 1.12 \cdot 10^5 \pi \cdot q(t))}{\pi^2}
\end{aligned}$$

## 2.5 Modèle d'état

Les trois équations (2.11), (2.12) et (2.13) décrivent la position ainsi que le mouvement de translation du centre de gravité de l'hélicoptère par rapport au repère EF. Par ailleurs, les angles d'attitude et la vitesse angulaire sont également décrits. Ces équations contiennent les forces et moments agissant sur l'hélicoptère.

Ceci étant, les équations (2.18) et (2.19) donnent l'expression des forces et moments en fonction des poussées du rotor principal et du rotor de la queue ainsi que des angles de battement du rotor principal. Finalement, les équations (2.20), (2.21), (2.22) et (2.23) donnent les expressions des poussées et des angles de battement en fonction des entrées du système. Par ailleurs, et pour les besoins d'analyse de stabilité et de contrôle, ces équations sont souvent linéarisées autour d'une condition de vol en considérant des petites perturbations autour de ce point d'équilibre. En effet, L'hélicoptère est considéré comme possédant deux composantes :

- un mouvement moyen qui représente l'équilibre;
- un mouvement dynamique représentant les perturbations au mouvement moyen.

Par ailleurs, lors d'une linéarisation, les perturbations sont toujours considérées comme petites. Ainsi, chaque variable d'état est séparée en deux composantes :

- une composante à l'équilibre,
- une composante représentant les perturbations.

La vitesse de tangage, par exemple, est exprimée comme suit :  $q = q_0 + \Delta q$ . Pour la condition de vol *Hover*, et pour simplifier, ce point d'équilibre est choisi de sorte que les angles d'Euler, les vitesses de translation et de rotation ainsi que les angles de battement ont tous une valeur nulle.

Les équations exprimant  $u, v, w, p, q, r, \phi, \theta$  et  $\psi$  sont linéarisées en utilisant une approximation de Taylor. Celles exprimant  $\beta_{ls}$  et  $\beta_{lc}$  sont linéarisées avec une méthode dite «boîte noire». Finalement, les expressions de  $Q_{MR}$  et  $T_{MR}$ , ayant un aspect récursif, sont linéarisées numériquement.

L'objet de ce chapitre étant d'expliquer comment la modélisation de la dynamique est effectuée analytiquement, sans pour autant utiliser cette méthode pour notre application puisque le modèle que nous allons utiliser sera fourni par le logiciel Helisim (voir chapitre 5), les détails de la linéarisation ne seront pas traités dans ce travail pour des raisons de brièveté (se référer à (Hald et al., 2005) pour de plus amples précisions).

La linéarisation des équations citées précédemment permet une représentation matricielle de la dynamique du système. Cette représentation est dite : modèle d'état et se présente sous la forme :

$$\Delta \overset{\circ}{x} = A \Delta x + B \Delta \delta \quad (2.24)$$

Avec  $x = [q \ u \ w \ \theta \ p \ r \ v \ \phi]^T$  et  $\delta = [\delta_{coll}, \delta_{lat}, \delta_{long}, \delta_{TR}]^T$ .  $\Delta x$  et  $\Delta \delta$ , nommés respectivement vecteur d'état et vecteur d'entrée représentent la perturbation du vecteur  $x$  et l'écart du vecteur  $\delta$  par rapport au point d'équilibre et les matrices  $A$  et  $B$ , respectivement appelées matrice des paramètres et matrice d'entrée sont considérées invariantes dans le temps.

La représentation générale inclut un vecteur de sortie ( $\Delta y$ ), une matrice de sortie  $C$  et une matrice de couplage entrées-sorties  $D$  définis comme suit :

$$\Delta y = C \cdot \Delta x + D \cdot \Delta \delta \quad (2.25)$$

## CHAPITRE 3

### LES QUALITÉS DE VOL DE L'HÉLICOPTÈRE

#### 3.1 Introduction

Le rôle des hélicoptères modernes ayant été élargi grâce aux avancements technologiques, les qualités de vol reflètent les demandes variées des différentes missions ce qui explique les efforts fournis par l'armée américaine afin d'établir des spécifications orientées missions (contenues dans l'ADS-33).

Il est à noter que la spécification des qualités de vol, notées HQ (*Handling Qualities*), n'est pas aisée, et pour cause, cette démarche implique la quantification de la charge du pilote pendant les missions requises. À ce sujet, l'échelle Cooper Harper constitue la base pour toutes les spécifications relatives aux qualités de vol (voir figure 3.1). Dans les spécifications relatives aux qualités de vol contenues dans ADS-33, on trouve plusieurs abréviations spécifiques qu'il convient de préciser dès maintenant afin d'éviter la confusion du lecteur :

- MTE (*Mission Task Element*) est relatif à la mission globale définie par l'utilisateur de l'hélicoptère. Cette mission globale est décomposée en éléments (tâches). Des exemples de ces éléments sont : *Hover*, *Landing* et *Pirouette* etc.
- OFE (*Operational Flight Envelope*) est l'enveloppe de vol qui définit l'enveloppe de la vitesse, facteur de charge, altitude, taux de montée, angle de dérapage et autres paramètres opérationnels et qui déterminent les conditions dans lesquelles l'hélicoptère peut réussir sa mission.
- UCE (*Usable Cue Environment*) est déterminé par la combinaison de l'information issue de l'environnement opérationnel et des moyens d'aide à la vision embarqués sur l'hélicoptère.
- *Hover* est la condition de vol stationnaire (vitesses linéaires et angulaires nulles).

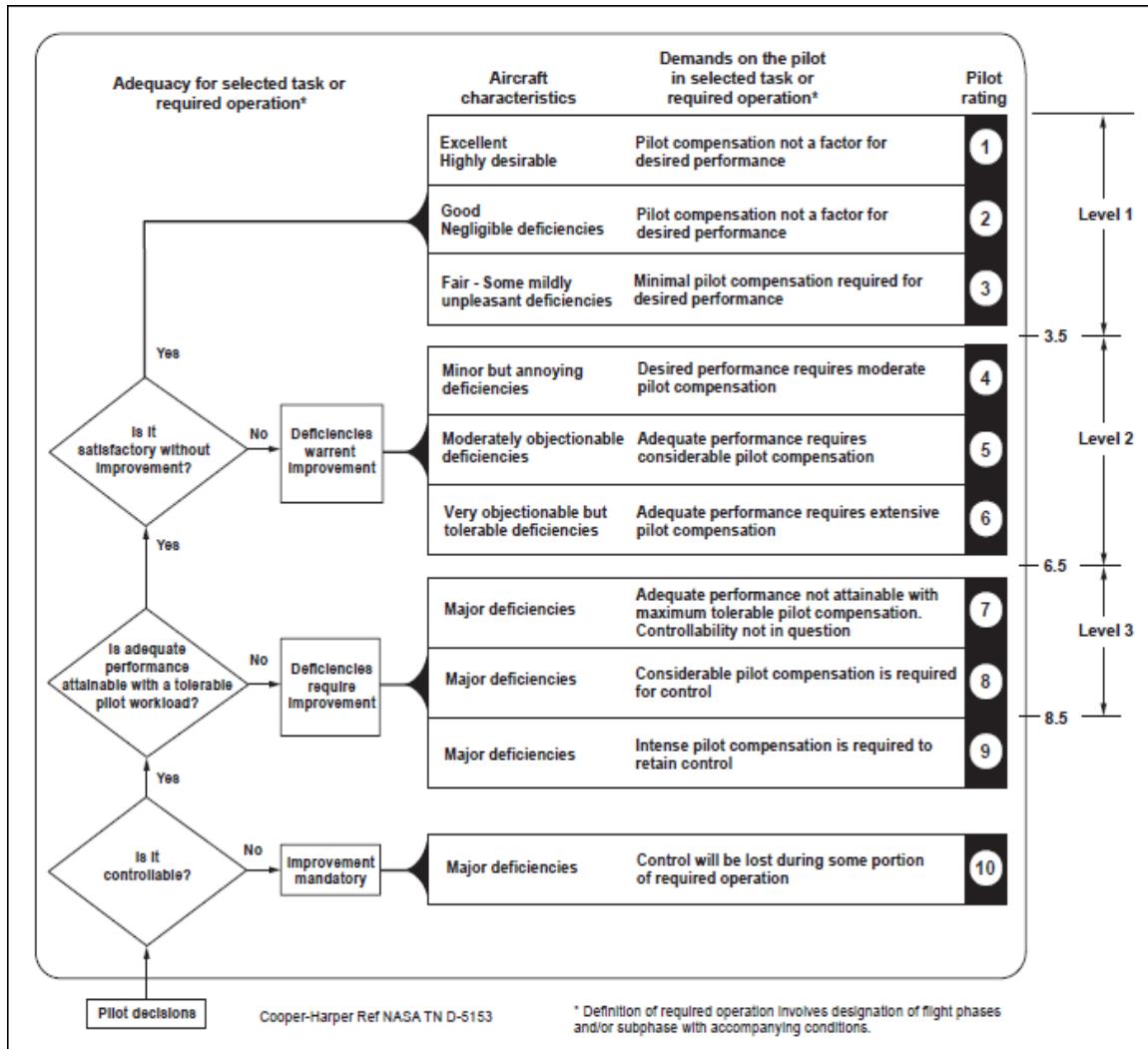


Figure 3.1 Définition des niveaux des qualités de vol  
Tirée de ADS-33E-PRF (2000, p. 73)

Les architectures de contrôle suivantes, appelées types de réponses (*response types*), sont également évoquées régulièrement dans ce travail :

- ACAH (*Attitude Command Attitude Hold*) permet la commande et le maintien d'attitude (angle de tangage ou de roulis par exemple).
- *Rate* ou *RC* (*Rate Command*) permet la commande de la vitesse angulaire (vitesse angulaire de roulis ou tangage par exemple).

- RCDH (*Rate Command Direction Hold*) permet la commande de la vitesse angulaire avec maintien de direction (cap).
- RCHH (*Rate Command Height Hold*) permet la commande de la vitesse angulaire avec maintien d'altitude.
- TRC (*Translational Rate Command*) permet la commande de la vitesse de translation (longitudinale par exemple).
- PH (*position Hold*) désigne le maintien de la position de l'hélicoptère.

Le tableau (3.1) illustre l'utilisation des MTE et UCE pour déterminer les types de réponses requis, pour la manœuvre *Hover* et les basses vitesses. La colonne de gauche du tableau montre les MTE. Les UCE sont classées de niveau 1 à niveau 3 et deux niveaux HQ sont considérés. On constate que pour tous les MTE, selon le niveau UCE et HQ, le ou les types de réponses requis sont indiqués. Par ailleurs, des types de réponses additionnels pour des MTE spécifiques sont mentionnés.

Tableau 3.1 Types de réponses requis pour la manœuvre *Hover*  
et basses vitesses – près de la terre  
Tiré de ADS-33E-PRF (2000, p. 67)

MTE	UCE = 1		UCE = 2		UCE = 3	
	Level 1	Level 2	Level 1	Level 2	Level 1	Level 2
Required Response-Type for all MTEs. Additional requirements for specific MTEs are given below.	RATE	RATE	ACAH	RATE + RCDH	TRC+RCDH + RCHH+PH	ACAH
Hover			RCDH + RCHH			RCDH + RCHH
Landing			RCDH			RCDH
Slope landing			RCDH			RCDH
Hovering turn			RCHH			RCHH
Pirouette			RCHH			RCHH
Vertical Maneuver			RCDH			RCDH
Depart/Abort			RCDH + RCHH			RCDH + RCHH
Lateral Reposition			RCDH + RCHH			RCDH + RCHH
Slalom	NA	NA	RCHH			RCHH
Vertical remask			RCDH			RCDH
Acceleration and deceleration			RCDH + RCHH			RCDH + RCHH
Sidestep			RCDH + RCHH			RCDH + RCHH
Turn to target			RCDH + RCHH			RCDH + RCHH
Divided attention required	RCDH + RCHH + PH		RCDH + RCHH			RCDH + RCHH

Un type de réponses adapté à une MTE, peut se révéler inapproprié pour une autre. L'exemple classique est la nécessité de ACAH pendant le *Hover* et l'atterrissement quand UCE=2 pour atteindre le niveau 1 HQ. En effet, l'expérience a montré que ces manœuvres, conjuguées à des conditions de visibilité médiocres peuvent occasionner des accidents, particulièrement en atterrissage, à cause de la difficulté qu'éprouvent les pilotes, dans de pareils cas de figure, à stabiliser l'hélicoptère. Par contre, il est fortement déconseillé d'utiliser l'ACAH lors de la manœuvre agressive *quickstop* parce qu'il conduit à l'instabilité de l'appareil.

Pour les hélicoptères, le concept d'UCE revêt une importance cruciale, les moyens d'aide à la visibilité embarqués, vision infra-rouge, radar, vision de nuit, permettent le vol pendant la nuit ou encore des mauvaises conditions météo ; en revanche, leur efficacité se trouve limitée sous certaines conditions.

Trois niveaux d'UCE sont définis, le niveau 1 permet des manœuvres agressives et/ou précises alors que le niveau 3 ne permet que des manœuvres plus simples. À part MTE et UCE, les exigences à satisfaire par l'hélicoptère dépendent de l'attention du pilote ainsi que de la gamme des vitesses.

Ainsi, la spécification concernant l'attention du pilote définit deux niveaux (voir dernière ligne du tableau 3.1), le cas où le pilote a une attention complète sur la MTE et le cas où il est engagé dans d'autres activités (opérations tactiques dans le domaine militaire par exemple), la spécification est plus exigeante dans ce deuxième cas.

Par ailleurs, l'ADS-33 stipule que pour des vitesses inférieures à 45 nœuds (*Hover* et basses vitesses), la dynamique de l'hélicoptère est considérée comme pendant la manœuvre *Hover* avec une dynamique similaire pour le tangage et le roulis et des réponses désirables de premier ordre pour la vitesse angulaire de lacet et la vitesse verticale; et pour des vitesses supérieures à 45 nœuds, la dynamique de l'hélicoptère est supposée proche de celle des avions à voilure fixe.

Les spécifications relatives aux qualités de vol étant très longues (Anon, 2000), seules celles qui concernent la condition de vol (*Hover* et basses vitesses) et en rapport avec la conception du système de contrôle pour hélicoptères seront exposées dans ce chapitre.

### 3.2 Critères des spécifications relatives aux qualités de vol

#### 3.2.1 Critère de la bande passante

Le critère de la bande passante décrit la capacité du système à suivre une gamme de fréquences appliquées à son entrée. Il existe deux définitions de la fréquence reflétant la bande passante (marge de gain  $w_{BW_{gain}}$  et marge de phase  $w_{BW_{phase}}$ ) comme illustré sur la figure 3.2.

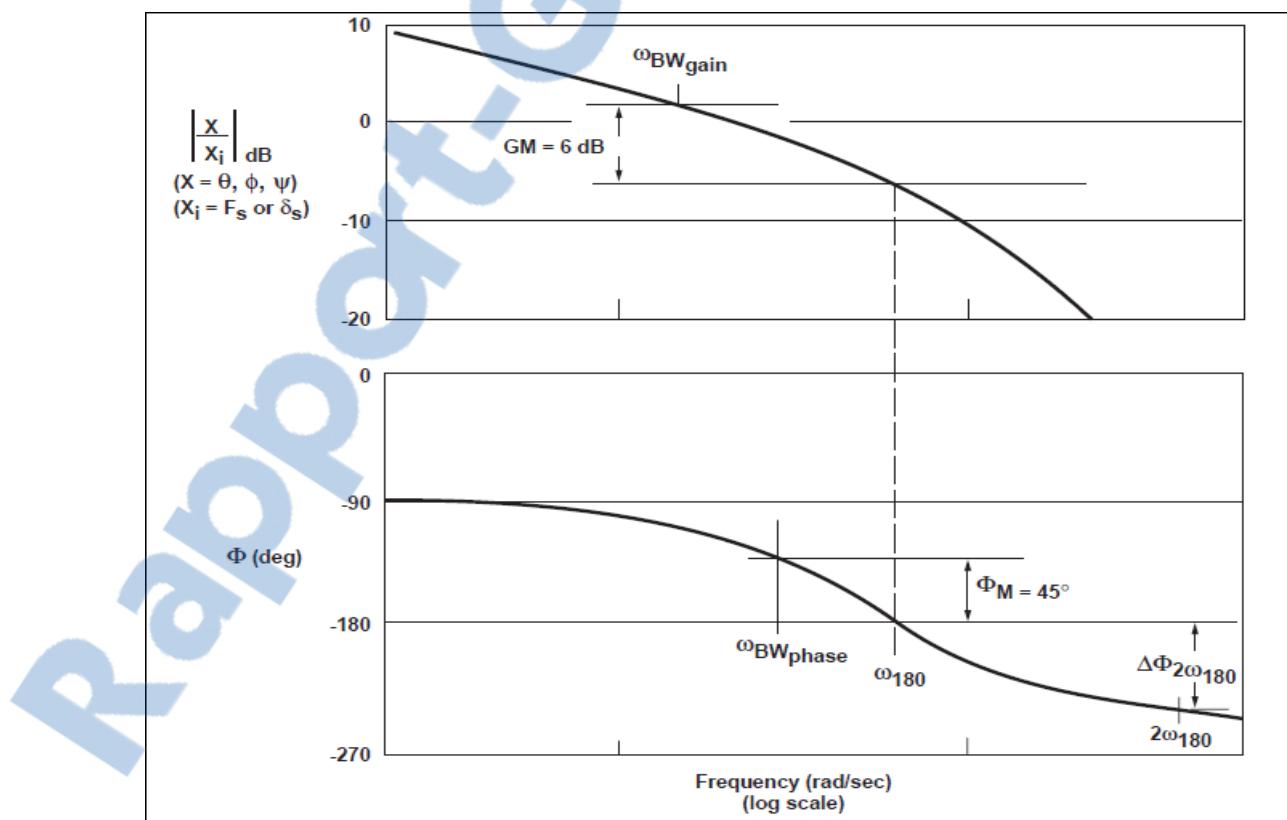


Figure 3.2 Définition de la Bande passante  
Tirée de ADS-33E-PRF (2000, p. 76)

Ces deux grandeurs sont définies pour un système de second ordre et décrivent les marges qui permettent d'augmenter le gain par le pilote ou avoir un retard de phase sans causer une instabilité pour le système.

La définition de la bande passante dans l'ADS-33 est différente de celle contenue dans les livres classiques de contrôle. Pour calculer  $w_{BW_{gain}}$ , le gain correspondant à  $w_{180}$  doit être déterminé ( $w_{180}$  étant la fréquence pour laquelle la phase est  $-180^\circ$ ). La fréquence à laquelle le gain est de 6 DB supérieur au gain à la fréquence  $w_{180}$  représente  $w_{BW_{gain}}$ . Une bande passante  $w_{BW_{gain}}$  faible rend le système instable parce que des changements dans le gain du pilote causent une réduction rapide dans la marge de phase.

En outre,  $w_{BW_{phase}}$  est la fréquence à laquelle la phase est  $-135^\circ$  (de  $45^\circ$  supérieure à celle de la fréquence  $w_{180}$ ). La détermination de cette grandeur peut être effectuée à l'aide du diagramme de Bode (phase) comme indiqué sur la figure 3.2.

Pour ACAH, la bande passante n'inclut pas de limite pour la marge de gain (elle est égale à la marge de phase). Par ailleurs, pour les réponses types RC, la bande passante est la plus petite de la marge de gain et de la marge de phase.

Il est à noter que pour les systèmes de premier ordre, la bande passante et la valeur propre sont identiques, alors que pour les systèmes de deuxième ordre, elle est donnée par l'équation suivante (Garrard et Low, 1990) :

$$w_{BW} = w_n (\xi + \sqrt{\xi^2 + 1}) \quad (3.1)$$

$w_n$  étant la fréquence naturelle et  $\xi$  l'amortissement du système.

Les valeurs courantes choisies, permettant d'atteindre le niveau 1 pour les cas de petits changements dans l'attitude (tangage et roulis), sont  $w_{BW} = 4$  rad/s,  $w_n = 2.071$  rad/s et  $\xi = 0.707$ .

Le paramètre de retard de phase est une mesure de la pente à laquelle la phase baisse au-delà de la fréquence  $-180^\circ$ . Un retard de phase de grande valeur indique qu'il existe une petite marge entre la poursuite normale effectuée par le pilote à  $45^\circ$  de marge de phase et l'instabilité. Ceci étant, un hélicoptère avec un grand retard de phase est sujet au phénomène PIO (*Pilot Induced Oscillations*). Le retard de phase est défini par l'équation (3.2) tirée de ADS-33E-PRF (2000, p. 76).

$$\tau_p = \frac{\Delta\Phi_{2w_{180}}}{57.3(2w_{180})} \quad (3.2)$$

Avec,  $\Delta\Phi_{2w_{180}}$  est la différence de phase entre  $-180^\circ$  et la phase à  $2w_{180}$  (le double de la fréquence à laquelle l'angle de phase est  $-180^\circ$ ). Si la phase est non linéaire entre  $w_{180}$  et  $2w_{180}$ , la méthode des moindres carrés est utilisée pour estimer  $\tau_p$ .

### 3.2.2 Types de réponses

Les types de réponses doivent respecter des critères précis. Pour ACAH, une entrée constante de contrôle doit provoquer un déplacement proportionnel d'attitude. Par ailleurs, lors de l'application d'une impulsion à l'actuateur, l'attitude doit retourner à 10 % de son pic dans moins de 20 secondes pour UCE=1, et dans moins de 10 secondes pour UCE supérieur à 1 (voir figure 3.3). L'attitude de roulis doit retourner dans moins de 10 secondes pour toutes les valeurs d'UCE.

Le type de réponses ACAH est requis pour la manœuvre *Hover* quand les conditions de visibilité sont médiocres ou encore quand l'attention du pilote est divisée avec d'autres tâches.

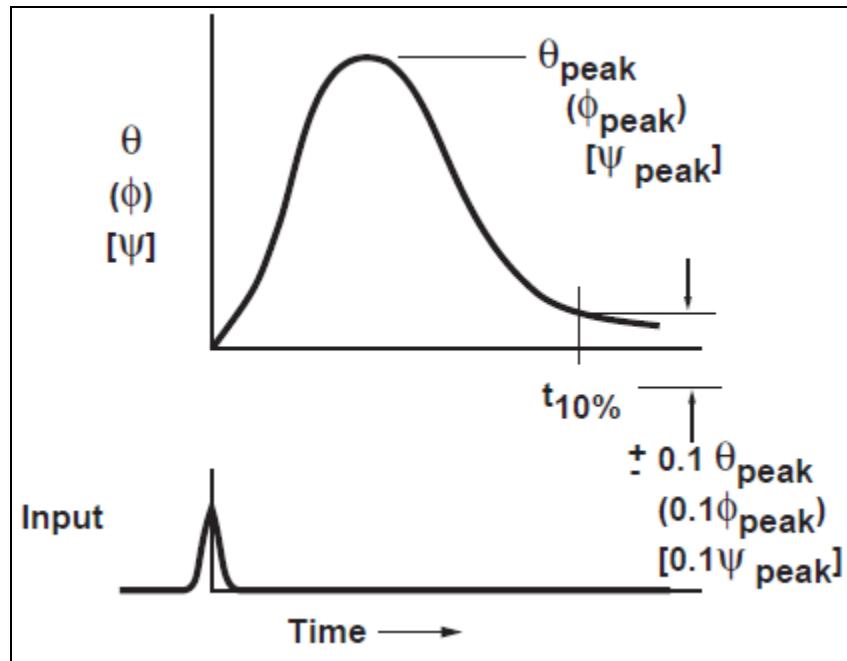


Figure 3.3 Réponse à une entrée impulsionnelle de l'actuateur  
Figure tirée de ADS-33E-PRF (2000, p. 74)

Les types de réponses ne conduisent pas au même degré de stabilisation pour les hélicoptères. La classification des combinaisons rencontrées dans la pratique, de la moins à la plus stabilisante, est donnée dans l'ADS-33 (page 8) :

1. RATE;
2. RATE + RCDH (*rate command direction hold*) + RCHH (*rate command heading hold*) + PH (*position hold*);
3. ACAH+RCDH;
4. ACAH+RCDH+RCHH;
5. ACAH+RCDH+RCHH+PH;
6. TRC (*translational rate command*) +RCDH+RCHH+PH.

Les fonctions de transfert désirées (Garrard et Low, 1990) pour ACAH (pour le roulis et le tangage) sont :

$$\frac{\phi}{\phi_c} = \frac{w_\phi^2}{s^2 + 2\xi w_\phi s + w_\phi^2} \quad (3.3)$$

$$\frac{\theta}{\theta_c} = \frac{w_\theta^2}{s^2 + 2\xi w_\theta s + w_\theta^2} \quad (3.4)$$

Pour RC (*rate command*), les fonctions de transfert désirées sont :

$$\frac{w}{w_c} = \frac{\lambda_w}{s + \lambda_w} \quad (3.5)$$

$$\frac{p}{p_c} = \frac{\lambda_p}{s + \lambda_p} \quad (3.6)$$

$$\frac{q}{q_c} = \frac{\lambda_q}{s + \lambda_q} \quad (3.7)$$

$$\frac{r}{r_c} = \frac{\lambda_r}{s + \lambda_r} \quad (3.8)$$

Les bandes passantes de la vitesse verticale, la vitesse de roulis, la vitesse de tangage et la vitesse de lacet sont respectivement  $\lambda_w$ ,  $\lambda_p$ ,  $\lambda_q$  et  $\lambda_r$ . Ces valeurs propres sont choisies égales à 4 rad/s pour satisfaire les HQ (*Handling Qualities*).

Pour TRCPH (*translational rate command with position hold*), concernant u et v, des fonctions de transfert de second ordre, similaires à (3.3) et (3.4) sont également désirées.

### 3.3 Exigences des spécifications pour la manœuvre *Hover* et basses vitesses

Les exigences pour le tangage et roulis sont les mêmes, par contre le canal lacet a des exigences à part. Par ailleurs, ces exigences sont divisées en trois parties :

- Changement d'attitude à petites amplitudes;
- Changement d'attitude à amplitudes modérées;
- Changement d'attitude à larges amplitudes.

#### 3.3.1 Changement d'attitude à petites amplitudes (tangage et roulis)

Les critères des réponses pour les petites amplitudes sont divisés en deux volets :

- Hautes fréquences (supérieures à la bande passante définie en 3.2.1);
- Fréquences inférieures à  $w_{BW}$ .

Pour les hautes fréquences, le critère de la bande passante est utilisé (comme illustré sur la figure 3.4). Comme on peut le voir sur cette figure, la bande passante requise pour maintenir un niveau 1 des HQ augmente sensiblement quand le retard de phase dépasse 0.17 secondes.

Par ailleurs, les hélicoptères montrent des réponses (basse fréquence) pour les vitesses  $u$  et  $v$  analogues à la réponse phugoides des avions à voilure fixe. Elles sont appelées réponses de moyen terme. La spécification des HQ pour ces réponses est donnée en termes de fréquence et d'amortissement comme illustré sur la figure 3.5. Cette spécification concerne les systèmes de deuxième ordre dont l'expression des pôles est la suivante :

$$P_{1,2} = -\xi w_n \pm w_n \sqrt{\xi^2 - 1} \quad (3.9)$$

La nature des pôles (complexe, réel, stable ou instable) dépend donc de la valeur de  $\xi$ .

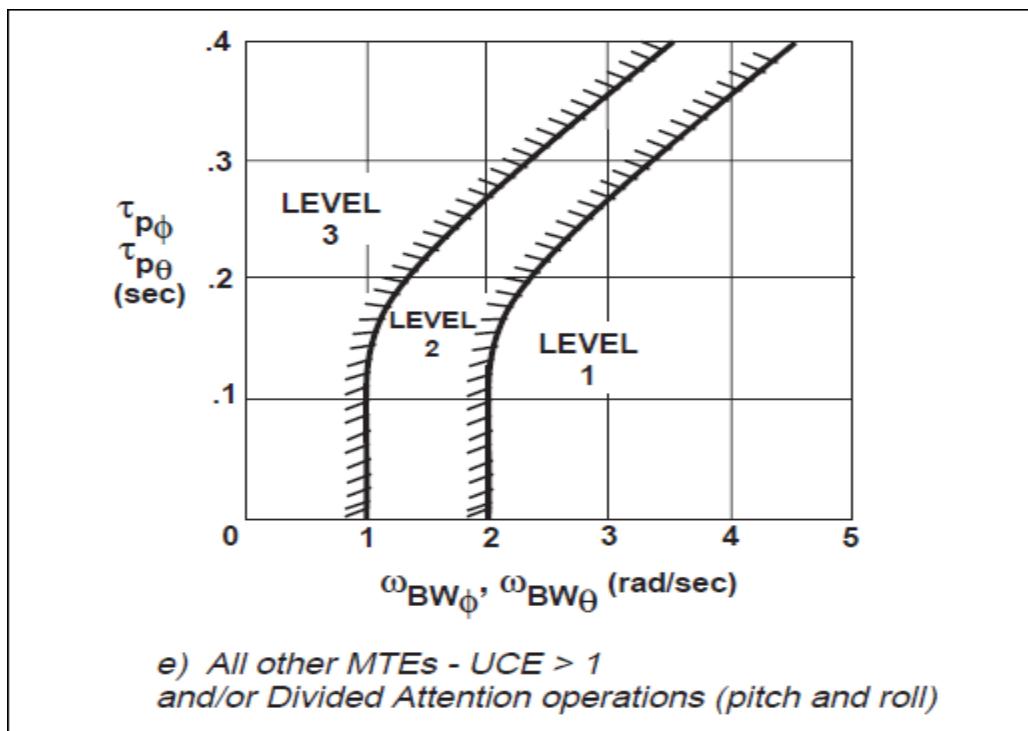


Figure 3.4 Exigences pour des changements d'attitude à petites amplitudes  
Figure tirée de ADS-33E-PRF (2000, p. 75)

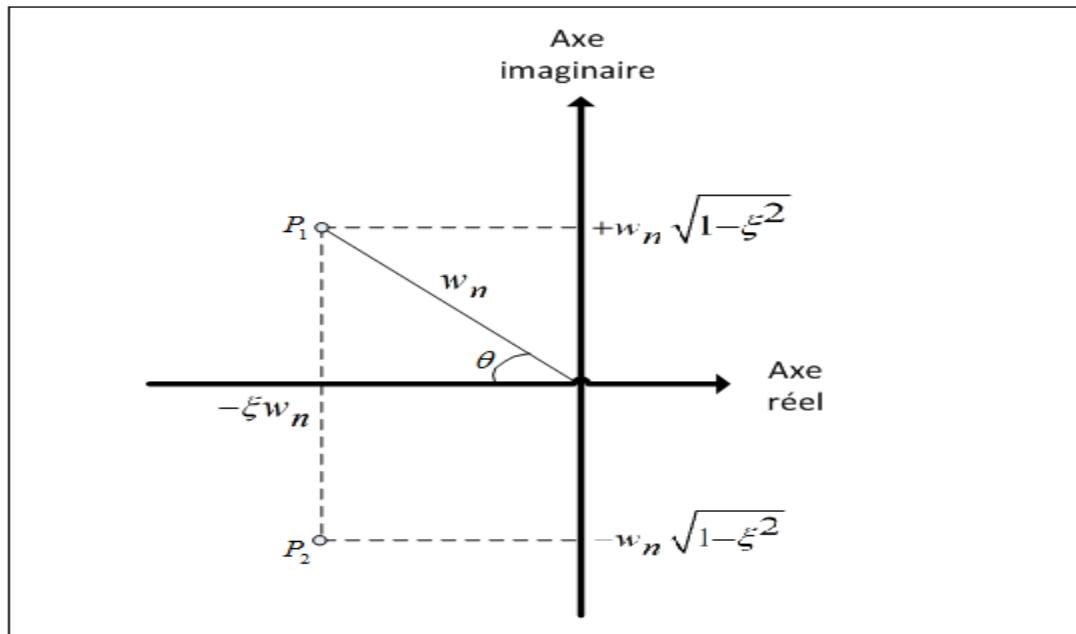


Figure 3.5 positionnement des pôles en fonction de  $w_n$  et  $\xi$

Sur la figure 3.5, les lignes iso-amortissement sont des droites partant de l'origine avec un angle  $\theta$  tel que  $\cos \theta = \xi$ .

Pour les opérations à attention complète du pilote (*fully attended operations*), le niveau 1 des HQ nécessite d'avoir un amortissement d'au moins 0.35 excepté pour les fréquences inférieures à 0.5 rad/s où des réponses instables sont acceptables pour le niveau 1 (un amortissement minimum de -0.20 est requis) comme indiqué sur la figure 3.6.

Par ailleurs, pour les opérations à attention divisée du pilote (*Divided attended operations*), les limites de la figure 3.5 doivent être respectées, sauf que pour le niveau 1, l'amortissement ne doit pas être inférieur à 0.35 et ce pour toutes les fréquences.

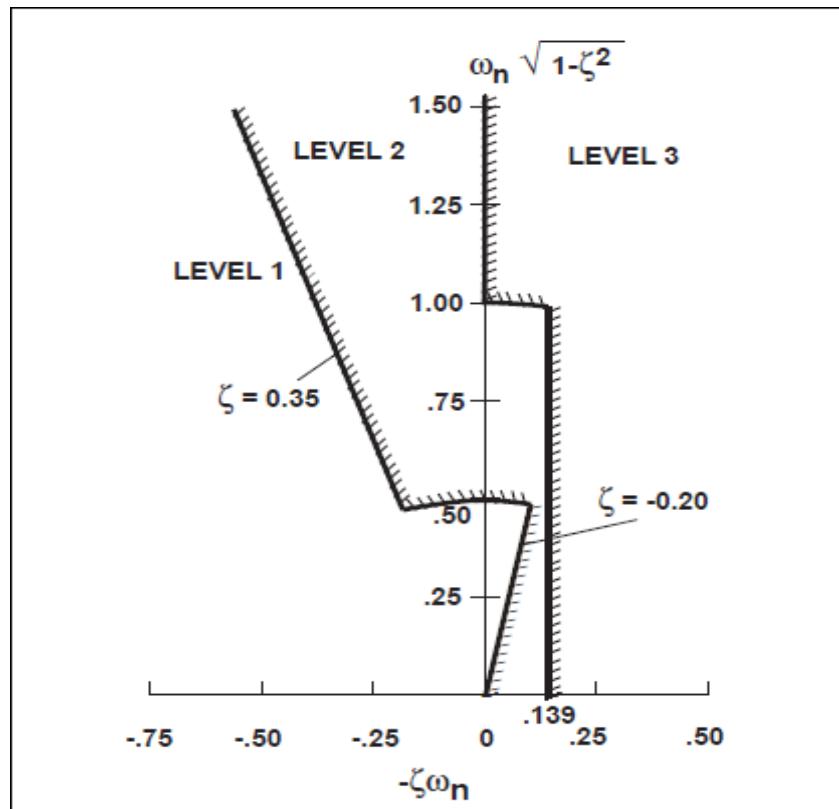


Figure 3.6 Limites sur le oscillations (tangage et roulis) pour la manoeuvre *hover* et basses vitesses  
Tirée de ADS-33E-PRF (2000, p. 77)

### 3.3.2 Changement d'attitude à amplitudes modérées (tangage et roulis)

L'évaluation des changements d'attitude à amplitudes modérées requiert des simulations. Les changements d'attitude requis pour la conformité avec cette spécification doivent varier de 5 à 30 degrés pour le tangage et de 10 à 60 degrés pour le roulis. Quand un changement d'attitude est appliqué, l'écart maximal et minimal d'attitude (par rapport à l'attitude de départ) ainsi que la vitesse angulaire maximale correspondante à l'axe concerné et enregistrée lors de ce changement sont déterminés (voir figure 3.7).

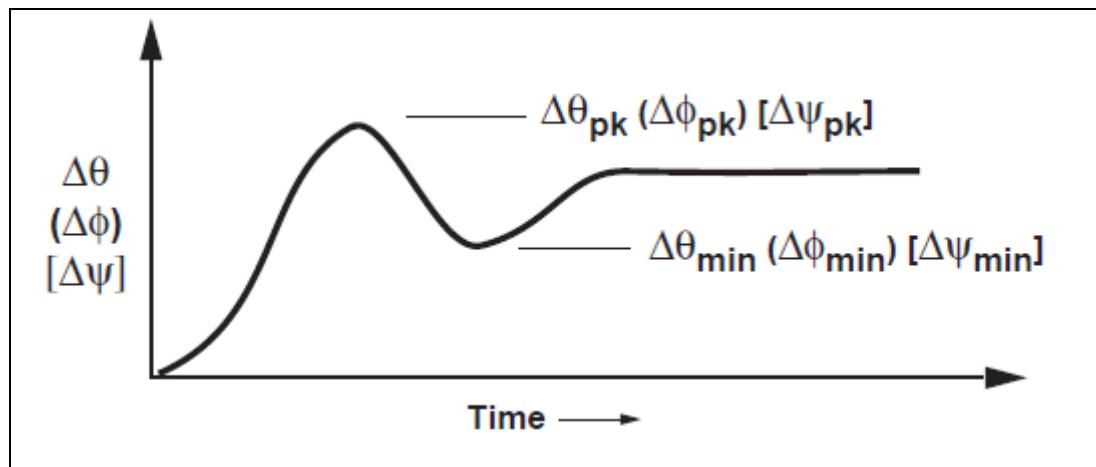


Figure 3.7 Définition des paramètres du critère d'amplitudes modérées  
Tirée de ADS-33E-PRF (2000, p. 78)

Après la détermination des paramètres requis pour chaque changement d'attitude, la vérification du respect de la spécification se fait en se reportant à la figure 3.8. On remarque

que les exigences sont différentes pour le tangage et le roulis. Le paramètre  $\frac{q_{pk}}{\Delta\theta_{pk}} \left( \frac{p_{pk}}{\Delta\phi_{pk}} \right)$

est le rapport entre la vitesse maximale de tangage (roulis) atteinte lors du changement d'une attitude de tangage (roulis) d'équilibre à une autre, et l'écart maximal d'attitude (voir figure 3.8).

Par ailleurs, la satisfaction de cette spécification n'est pas requise pour les types de réponses désignées comme applicables uniquement à UCE = 2 ou 3.

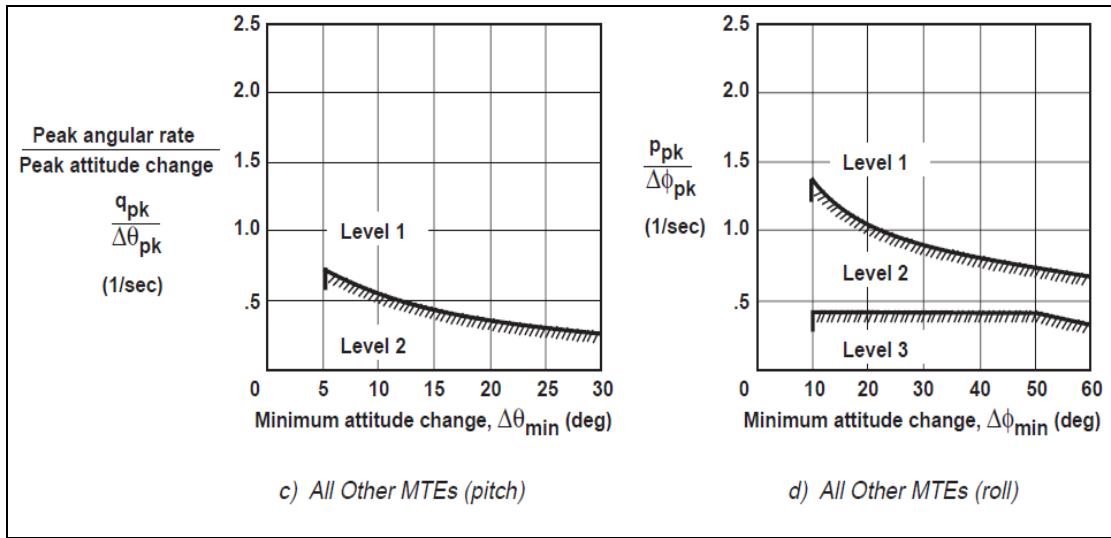


Figure 3.8 Exigences pour changement d'attitude à amplitudes modérées

(tangage et roulis) pour la manœuvre *hover* et basses vitesses

Tirée de ADS-33E-PRF (2000, p. 78)

### 3.3.3 Changement d'attitude à larges amplitudes

Les exigences pour les larges amplitudes sont spécifiées en fonction de la vitesse minimale que l'hélicoptère doit être en mesure d'atteindre pour les types de réponses *Rate Response* et d'attitude minimale qui doit être atteinte pour les types de réponses *Attitude Response*. Par ailleurs, trois niveaux d'agressivité des manœuvres sont précisés dans le tableau 3.2.

Il est à noter que l'évaluation des critères relatifs au changement d'attitude à amplitudes larges et modérées requièrent des analyses non linéaires ainsi que des simulations.

### 3.3.4 Couplage Interaxe

Les entrées de contrôle dédiées à réaliser une réponse sur un axe ne doivent pas être des sources de réponses indésirables sur d'autres axes, ce phénomène est crucial surtout pour les manœuvres agressives. La spécification concernant le couplage entre l'axe vertical et l'axe de lacet stipule que la réponse de la vitesse de lacet à une entrée échelon brusque sur le

collectif ne doit pas dépasser les limites spécifiées dans la figure 3.9. Pour déterminer les paramètres requis pour vérifier le respect de cette spécification, les graphes représentant l'évolution de la vitesse angulaire de lacet ( $r$ ) et de la vitesse verticale ( $h$ ) sont tracés après application d'une entrée échelon sur le collectif à  $t=0$ . Les mesures de ( $r$ ) après une seconde ( $r(1)$ ) et trois secondes ( $r(3)$ ) ainsi que la mesure de ( $h$ ) après trois secondes ( $h(3)$ ) sont alors déterminés.

Tableau 3.2 Exigences pour le Changement d'attitude à larges amplitudes  
Tiré de ADS-33E-PRF (2000, p. 68)

AGILITY CATEGORY MTE	RATE RESPONSE-TYPES						ATTITUDE COMMAND RESPONSE-TYPES			
	ACHIEVABLE ANGULAR RATE (deg/sec)						ACHIEVABLE ANGLE (deg)			
	LEVEL 1			LEVELS 2 AND 3			LEVEL 1		LEVELS 2 AND 3	
	Pitch	Roll	Yaw	Pitch	Roll	Yaw	Pitch	Roll	Pitch	Roll
<u>Limited Agility</u> Hover Landing Slope Landing	±6	±21	±9.5	±3	±15	±5	±15	±15	±7	±10
<u>Moderate Agility</u> Hovering Turn Pirouette Vertical Maneuver Depart/Abort Lateral Reposition Slalom	±13	±50	±22	±6	±21	±9.5	+20 -30	±60	±13	±30
<u>Aggressive Agility</u> Vertical Remask Acceleration Deceleration Sidestep <u>Target Acquisition and Track</u> Turn to Target	±30	±50	±60	±13	±50	±22	±30	±60	+20 -30	±30

La vérification du respect de la spécification (voir figure 3.9) requiert la détermination de deux autres paramètres :

- $r_1$  est le premier pic de la vitesse angulaire de lacet avant 3 secondes ou  $r(1)$  si aucun pic ne survient avant 3 secondes;
- $r_3$  est défini comme suit :

$$r_3 = r(3) - r_1 \text{ si } r_1 > 0, \text{ ou } r_1 - r(3) \text{ pour } r_1 < 0.$$

La spécification montre que pour avoir un niveau 1 HQ,  $\left| \frac{r_1}{h(3)} \right|$  doit être inférieur à 0.65 et  $\left| \frac{r_3}{h(3)} \right|$  doit être inclus entre -0.15 et 0.2.

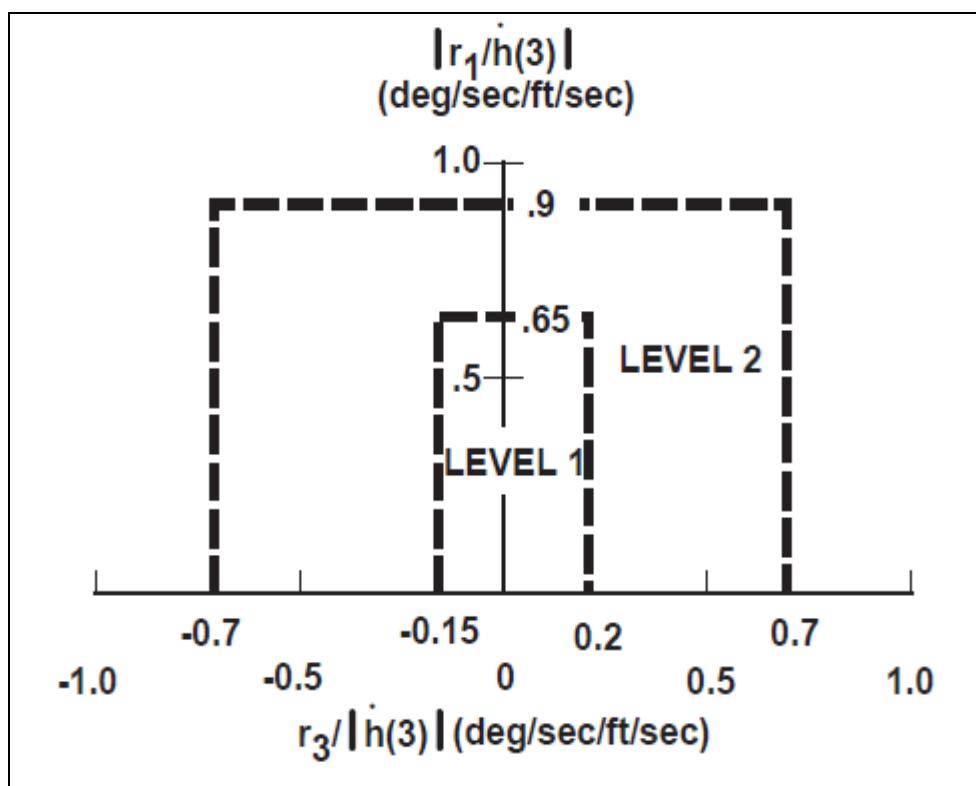


Figure 3.7 Exigences concernant le couplage lacet-collectif  
Tirée de ADS-33E-PRF (2000, p. 80)

La spécification relative au couplage entre les axes tangage et roulis est divisée en deux volets.

Pour les manœuvres à agilité agressive (voir tableau 3.2), le rapport entre le pic à partir de l'équilibre (*trim*) (pendant les 4 premières secondes) de l'attitude de l'axe dont la réponse est non désirée (*off-axis*) sur la mesure de l'attitude de l'axe désiré (à 4 secondes à partir de l'équilibre)  $\frac{\Delta\theta_{pk}}{\Delta\phi_4}$  ( $\frac{\Delta\phi_{pk}}{\Delta\theta_4}$ ), suivant une entrée latérale (longitudinale) brusque de type échelon, ne doit pas dépasser  $\pm 0.25$  pour le niveau 1 HQ et  $\pm 0.60$  pour le niveau 2.

Pour la MTE (*Target Acquisition and Tracking*), la spécification (voir figure 3.10) concerne les rapports moyens des vitesses angulaires ( $q/p$  quand l'axe roulis est désiré et  $p/q$  dans le cas contraire).

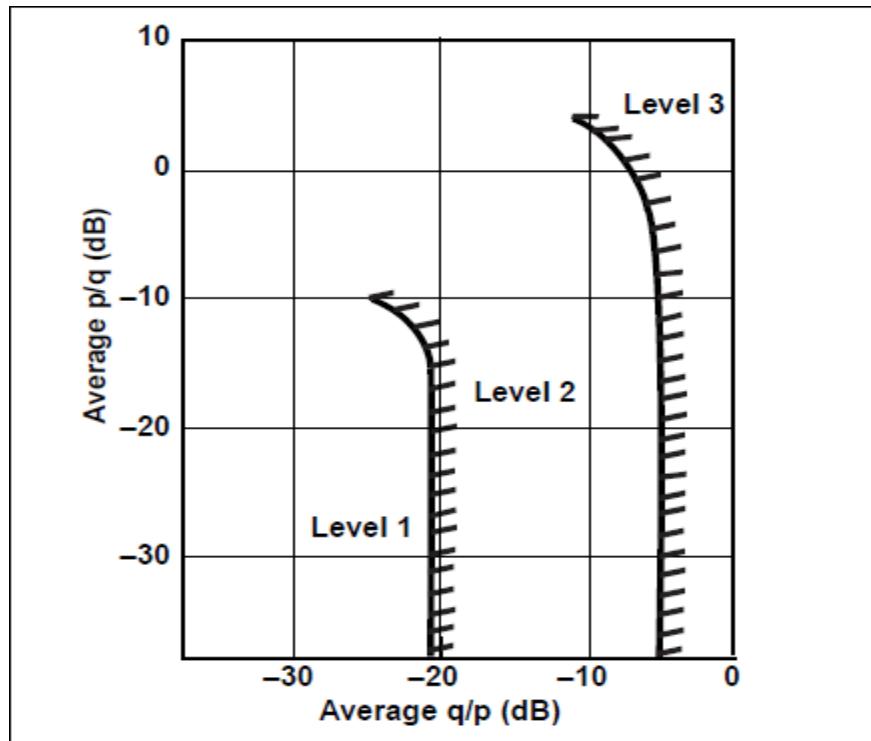


Figure 3.8 Exigences concernant le couplage tangage-roulis  
(Agilité agressive)

Tirée de ADS-33E-PRF (2000, p. 81)

### 3.4 Conclusion

Pour ce travail, nous allons choisir un ensemble de ces HQ pour les intégrer au design du système de contrôle pour hélicoptère. Puisque les spécifications à respecter sont exprimées en fonction des types de réponses, notre choix s'est porté sur ACAH (*Attitude Command Attitude Hold*) et RC (*Rate Command*). Ainsi, notre système de contrôle sera constitué d'une boucle interne dédiée à la réalisation de RC sur les quatre canaux (vertical, tangage, roulis et directionnel), tandis que deux boucles externes viseront la réalisation de ACAH sur les axes tangage et roulis. La procédure de conception de la boucle interne repose sur l'utilisation de l'affectation de la structure modale qui permet de découplage des quatre canaux (intégration du critère de couplage Interaxe). Par ailleurs, le critère de la bande passante sera également pris en compte dans toutes les fonctions de transfert désirées (réflétant les types de réponses considérées), une bande passante de 4 rad/s sera choisie pour s'assurer d'atteindre le niveau 1 HQ (voir figure 3.4).

## CHAPITRE 4

### AFFECTATION DE LA STRUCTURE MODALE

Dans ce chapitre, une introduction aux techniques d'affectation de la structure modale (*eigenstructure assignment* ou EA) sera faite, en mettant l'accent sur la notion de retour d'état qui servira de base pour l'élaboration ultérieure du contrôleur pour hélicoptère, les concepts de valeurs propres et vecteurs propres seront expliqués.

#### 4.1 Généralités sur les systèmes linéaires

Les systèmes reflétant la dynamique de l'hélicoptère sont, en réalité, non-linéaires, toutefois, ces systèmes sont linéarisés pour des fins de conception du contrôleur. En général, nous considérons donc un système linéaire invariant dans le temps (LTI) décrit par les deux équations suivantes :

$$\square \\ x(t) = Ax(t) + Bu(t) \quad (4.1)$$

$$y(t) = Cx(t) \quad (4.2)$$

Avec  $x \in R^n$ ,  $u \in R^m$  et  $y \in R^r$ ; ces trois vecteurs sont appelés vecteur d'état, d'entrée et de sortie respectivement. Par ailleurs, on note la condition initiale  $x(t=0) = x_0$ . Les matrices  $A$  de dimension  $(n \times n)$ ,  $B$  de dimension  $(n \times m)$  et  $C$  de dimension  $(r \times n)$  sont des matrices constantes à coefficients réels avec  $\text{rang}(B) = m \neq 0$  et  $\text{rang}(C) = r \neq 0$ . Les notions de commandabilité et observabilité sont définies comme suit :

Le système décrit par (4.1) et (4.2) est commandable si et seulement si

$$\text{rang}[B \ AB \ \dots \ A^{n-1}B] = n \quad (4.3)$$

Ce même système est observable si et seulement si

$$\text{rang}[C \ A^T C^T \ \dots \ (A^T)^{n-1} C^T] = n \quad (4.4)$$

Dans ce qui suit, on suppose que  $(A, B)$  est commandable et  $(A, C)$  est observable. Les pôles du système sont les valeurs propres de la matrice  $A$  définies comme les solutions de l'équation caractéristique (d'ordre  $n$ ) :

$$\det(\lambda I - A) = 0 \quad (4.5)$$

Supposons que les valeurs propres de la matrice  $A$ , notées  $\{\lambda_i\}_{i=1}^n$ , sont distinctes. Les vecteurs propres à droite, notés  $\{v_i\}_{i=1}^n$ , vérifient l'équation (4.6) pour  $i = 1, 2, \dots, n$ .

$$Av_i = \lambda_i v_i \quad (4.6)$$

La matrice modale  $M$  est définie à partir des vecteurs propres, à droite, comme étant :

$$M = [v_1 \ v_2 \ \dots \ v_n] \quad (4.7)$$

La matrice  $M$  est non singulière étant donné l'indépendance linéaire des vecteurs propres  $\{v_i\}_{i=1}^n$ . Faisons un changement de variables :

$$x = Mz \quad (4.8)$$

On a alors  $z = M^{-1}x$  et  $z(0) = z_0 = M^{-1}x_0$ . L'équation (4.1) peut alors être réécrite sous la forme :  $\square$   $x = Ax + Bu = AMz + Bu$   $\square$  ce qui donne  $Mz = AMz + Bu$  ou encore :

$$\overset{\square}{z}(t) = M^{-1} A M z(t) + M^{-1} B u(t) \quad (4.9)$$

$M$  étant la matrice modale de  $A$ , et puisque cette dernière a des valeurs propres  $\{\lambda_i\}_{i=1}^n$  distinctes, nous avons l'égalité suivante :

$$M^{-1} A M = \Lambda = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_n] \quad (4.10)$$

En supposant l'entrée  $u(t)$  nulle, l'équation (4.9) devient :

$$\overset{\square}{z}(t) = \Lambda z(t) \quad (4.11)$$

Par ailleurs, si on fait l'approximation de  $z(t)$  par sa série de Taylor, on trouve :

$$z(t) = z_0 + \overset{\square}{z}_0 t + z_0^{(2)} \frac{t^2}{2!} + \dots + z_0^{(k)} \frac{t^k}{k!} + \dots = \left[ I + t\Lambda + \frac{t^2}{2!} \Lambda^2 + \dots + \frac{t^k}{k!} \Lambda^k + \dots \right] z_0 \quad (4.12)$$

$\overset{\square}{z}_0, z_0^{(2)}, z_0^{(k)} \dots$  sont les dérivées successives de  $z$  à  $t=0$ . Le calcul de ces dérivées découle de l'équation (4.11) et on a alors, pour tout  $k \in \mathbb{N}$  (nombres entiers) :

$$z_0^{(k)} = \Lambda^k z_0 \quad (4.13)$$

Le terme  $[I + t\Lambda + \frac{t^2}{2!} \Lambda^2 + \dots + \frac{t^k}{k!} \Lambda^k + \dots]$  désigne  $e^{\Lambda t}$  et nous avons :

$$e^{\Lambda t} = \text{diag}[e^{(\lambda_1 t)}, e^{(\lambda_2 t)}, \dots, e^{(\lambda_n t)}] \quad (4.14)$$

Ainsi, la solution de l'équation (4.11) est :

$$z(t) = e^{\Lambda t} z_0 \quad (4.15)$$

L'équation (4.15) peut être réécrite sous la forme :

$$x(t) = e^{\Lambda t} x_0 = M e^{\Lambda t} M^{-1} x_0 \quad (4.16)$$

Pour résoudre l'équation (4.16), on commence par calculer  $M e^{\Lambda t}$  :

$$M e^{\Lambda t} = [v_1 \ v_2 \ \dots \ v_n] \text{diag}[e^{(\lambda_1 t)}, e^{(\lambda_2 t)}, \dots, e^{(\lambda_n t)}] = [v_1 e^{(\lambda_1 t)}; v_2 e^{(\lambda_2 t)}; \dots; v_n e^{(\lambda_n t)}] \quad (4.17)$$

D'autre part, les lignes de la matrice inverse de la matrice modale ( $M^{-1}$ ) sont les vecteurs propres à gauche, notés  $\{w_i\}_{i=1}^n$ , et qui vérifient l'équation (4.18) pour  $i = 1, 2, \dots, n$ .

$$w_i^T A = \lambda_i w_i^T \quad (4.18)$$

L'équation (4.16) devient :

$$x(t) = \{\exp(\lambda_1 t) v_1; \exp(\lambda_2 t) v_2; \dots; \exp(\lambda_n t) v_n\} \begin{bmatrix} w_1^T \\ w_2^T \\ \vdots \\ w_n^T \end{bmatrix} x_0$$

Ce qui donne :

$$x(t) = \sum_{i=1}^n e^{\lambda_i t} v_i w_i^T x_0 \quad (4.19)$$

En considérant l'entrée  $u(t)$ , la solution du système décrit par l'équation d'état (4.1) est :

$$x(t) = e^{At} x_0 + \int_0^t e^{A(t-s)} B u(s) ds \quad (4.20)$$

L'équation (4.20) peut être réécrite comme suit :

$$x(t) = \sum_{i=1}^n v_i e^{\lambda_i t} w_i^T x_0 + \sum_{i=1}^n v_i w_i^T B \int_0^t e^{\lambda_i(t-s)} u(s) ds \quad (4.21)$$

Par ailleurs, l'expression de  $y(t)$  se déduit de l'équation (4.2) :

$$y(t) = \sum_{i=1}^n C v_i e^{\lambda_i t} w_i^T x_0 + \sum_{i=1}^n C v_i w_i^T B \int_0^t e^{\lambda_i(t-s)} u(s) ds \quad (4.22)$$

En conclusion, la réponse du système dépend de quatre paramètres :

- La condition initiale et qui influe uniquement sur la réponse libre du système en déterminant le niveau auquel chaque mode va participer;
- Les valeurs propres qui fixent le taux de variation de la réponse du système;
- Les vecteurs propres qui déterminent l'allure de la réponse (en influant sur l'interaction entre les modes dynamiques du système avec les variables d'état);
- L'entrée du système  $u(s)$ .

Il est donc évident que lors de l'utilisation d'un retour d'état ou de sortie pour améliorer la réponse du système (voir les paragraphes suivants), un choix judicieux des valeurs et vecteurs propres en boucle fermée s'impose.

## 4.2 Théorème de Moore

Un des plus importants outils de la théorie de contrôle moderne est le placement des valeurs propres d'un système et qui permet la conception d'un contrôleur par retour d'état ou de sortie en assignant un ensemble de valeurs propres désirées.

Par ailleurs, pour les systèmes à multiples entrées et multiples sorties (MIMO), le gain de rétroaction permettant d'assigner les valeurs propres n'est pas unique, ce qui laisse une marge de manœuvre au concepteur pour le placement des vecteurs propres également dans le but d'améliorer les performances du système en boucle fermée.

Pour atteindre les performances désirées, une loi linéaire de retour d'état ( $u = -Kx$ ) ou de retour de sortie ( $u = -Ky$ ) est appliquée au système. Avec  $u = -Kx$ ,  $K \in R^{m \times n}$ , l'équation d'état du système en boucle fermée devient :

$$\overset{\square}{x} = (A - BK)x \quad (4.23)$$

Le problème qui se pose dès lors s'énonce comme suit :

Étant donné un ensemble de nombres complexes  $\{\lambda_i^d\}$   $i = 1, 2, \dots, n$  avec, si  $\lambda_j^d$ , appartenant à cet ensemble, est un nombre complexe, son conjugué appartient aussi à cet ensemble, et étant donné un ensemble de vecteurs correspondants à ces valeurs  $\{v_i^d\}$   $i = 1, 2, \dots, n$  vérifiant : si un vecteur est complexe, son conjugué appartient aussi à cet ensemble, l'objectif est de trouver une matrice  $K$  telle que les valeurs propres de  $A - BK$  soient  $\{\lambda_i^d\}$  ayant comme vecteurs propres correspondants  $\{v_i^d\}$ .

Moore a été le premier à identifier la liberté offerte par le retour d'état au-delà du placement des valeurs propres du système dans le cas où ces dernières, en boucle fermée, sont

distinctes (Moore, 1976). Ainsi, Moore considère les systèmes définis par l'équation (4.1) et déduit les conditions nécessaires et suffisantes pour l'existence d'une matrice  $K$  permettant de placer les valeurs et vecteurs propres là où c'est désiré. En outre, il propose une procédure permettant de calculer cette matrice. Pour ce faire, les matrices  $S_\lambda$  et  $R_\lambda$  sont définies comme suit :

$$S_\lambda \triangleq [\lambda I - A \mid B] \quad (4.24)$$

$$R_\lambda = \begin{bmatrix} N_\lambda \\ M_\lambda \end{bmatrix} \quad (4.25)$$

Les colonnes de  $R_\lambda$  forment une base pour l'espace nul (ou encore le noyau) de  $S_\lambda$ .

Théorème (Moore, 1976) :

Soit  $\{\lambda_i\}_{i=1}^{i=n}$  un ensemble de nombres complexes distincts, avec si  $\lambda_j$  est un nombre complexe, son conjugué appartient aussi à cet ensemble. Il existe une matrice réelle  $K (m \times n)$  telle que :

$$(A - BK) v_i = \lambda_i v_i \quad i = 1, 2, \dots, n \quad (4.26)$$

Si et seulement si, pour chaque  $i$  :

- (i)  $\{v_i\}_{i=1}^{i=n}$  est un ensemble linéairement indépendant de  $C^n$  (l'espace des vecteurs complexes de dimension  $n$ );
- (ii)  $v_i = v_j^*$  quand  $\lambda_i = \lambda_j^*$ ;
- (iii)  $v_i \in \text{span}\{N_{\lambda_i}\}$ , le sous espace vectoriel engendré par  $N_{\lambda_i}$ .

Par ailleurs, si  $K$  existe et  $\text{rang}(B) = m$ , alors  $K$  est unique. Ainsi, si les trois conditions (i), (ii) et (iii) sont satisfaites, il existe un vecteur  $z_i$  (puisque  $v_i \in \text{span}\{N_{\lambda_i}\}$ ) tel que :

$$v_i = N_{\lambda_i} z_i \quad (4.27)$$

D'autre part, et d'après la définition de  $S_{\lambda_i}$  et  $R_{\lambda_i}$ , nous avons :

$$(\lambda_i I - A)N_{\lambda_i} + BM_{\lambda_i} = 0 \quad (4.28)$$

Ce qui donne, en multipliant l'équation précédente par  $z_i$  :

$$(\lambda_i I - A)N_{\lambda_i} z_i + BM_{\lambda_i} z_i = 0 \quad (4.29)$$

S'il existe une matrice  $K$  vérifiant :

$$M_{\lambda_i} z_i = K v_i \quad (4.30)$$

Alors, il s'ensuit que :

$$[\lambda_i I - (A - BK)]v_i = 0 \quad (4.31)$$

Ainsi,

$$K [v_1 \ v_2 \ \dots \ v_n] = [M_{\lambda_1} z_1 \ M_{\lambda_2} z_2 \ \dots \ M_{\lambda_n} z_n] \quad (4.32)$$

Dans le cas où les  $\lambda_i$  sont réels,  $v_i$  et  $z_i$  sont eux aussi réels et la matrice des vecteurs propres est non singulière. Par conséquent,

$$K = [M_{\lambda_1} z_1 \ M_{\lambda_2} z_2 \ \dots \ M_{\lambda_n} z_n] [v_1 \ v_2 \ \dots \ v_n]^{-1} \quad (4.33)$$

Si les valeurs propres désirées sont complexes, une modification de l'équation précédente s'impose puisque la matrice  $K$  ne peut contenir que des éléments réels. Supposons que  $\lambda_1$  est complexe, alors  $\lambda_2 = \lambda_1^*$  doit appartenir à l'ensemble des valeurs propres comme indiqué dans le théorème de Moore et la condition (ii) nous donne  $v_1 = v_2^*$ , ce qui implique :  $z_1 = z_2^*$ . Ainsi, en supposant que les autres valeurs propres sont réelles, l'équation (4.31) peut être réécrite sous la forme :

$$K [v_{1R} + jv_{1I}, \ v_{1R} - jv_{1I}, \ v_3, \ \dots, \ v_n] = [T_{1R} + jT_{1I}, \ T_{1R} - jT_{1I}, \ T_3, \ \dots, \ T_n] \quad (4.34)$$

Avec  $T_i = M_{\lambda_i} z_i$ . On multiplie, par la droite, les deux termes de l'équation (4.34) par la matrice de dimension  $(n \times n)$  suivante :

$$P = \begin{bmatrix} 1/2 & -j/2 & & \\ 1/2 & j/2 & & \\ & & I & \end{bmatrix} \quad (4.35)$$

$I$  est la matrice identité de rang  $(n-2)$  et les autres éléments de la matrice  $P$  sont nuls. Le résultat de cette multiplication est l'équation suivante :

$$K [v_{1R} \ v_{1I} \ v_3 \ \dots \ v_n] = [T_{1R} \ T_{1I} \ T_3 \ \dots \ T_n] \quad (4.36)$$

Puisque les vecteurs  $\{v_i\}_{i=1}^n$  sont indépendants, La matrice  $[v_{1R} \ v_{1I} \ v_3 \ \dots \ v_n]$  est non singulière, ce qui fait que  $K$  peut être calculé, à l'aide de l'équation précédente.

En résumé, avec ce théorème, Moore caractérise la classe des vecteurs propres en boucle fermée que l'on peut affecter avec un ensemble de valeurs propres distinctes en appliquant le retour d'état. Par ailleurs, une procédure pour le calcul de la matrice  $K$  a été donnée en commençant par trouver une base pour l'espace nul des  $S_\lambda$ . Néanmoins, dans le cas général, un vecteur propre désiré peut ne pas résider dans le sous espace prescrit et ne peut donc être réalisé.

#### 4.3 Liberté procurée par l'affectation de la structure modale

La liberté dont dispose le concepteur en utilisant l'affectation de la structure modale a fait l'objet de travaux de Srinathkumar (Srinathkumar, 1978) pour les systèmes avec retour d'état et avec retour de sortie.

Dans le cas où un retour d'état est appliqué afin de placer la structure propre, pour un système commandable,  $n$  valeurs propres et un maximum de  $n \times m$  entrées des vecteurs propres peuvent être choisies arbitrairement. Par ailleurs,  $m$  entrées au maximum peuvent être spécifiées pour chaque vecteur propre. Et dans le cas d'un retour de sortie, pour un système commandable et observable, avec des matrices  $B$  et  $C$  de rang maximum,  $\max(m, r)$  valeurs propres en boucle fermée peuvent être choisies et  $\max(m, r)$  vecteurs propres sont partiellement affectables avec  $\min(m, r)$  entrées, arbitrairement choisies, dans chaque vecteur.

En général, plus de  $\min(m, r)$  entrées de chaque vecteur propre sont à affecter afin d'atteindre les performances requises pour un système donné, ce qui amène au choix d'un vecteur qui est l'approximation du vecteur désiré selon un critère déterminé. Le choix se porte alors sur la projection orthogonale du vecteur désiré sur un sous espace (la projection du vecteur désiré est appelée vecteur réalisable), ce sous espace vectoriel est calculé en fonction des matrices  $A$ ,  $B$  et de la valeur propre désirée en boucle fermée. La projection sur le sous espace prescrit permet de choisir un vecteur réalisable (le plus proche au sens des moindres carrés du vecteur propre désiré).

#### 4.4 Sous espace vectoriel contenant les vecteurs propres réalisables

Il existe plusieurs méthodes pour le calcul du sous espace vectoriel dans lequel résident les vecteurs propres réalisables pour des matrices  $A$ ,  $B$  et une valeur propre désirée en boucle fermée données.

La première méthode a été développée par Moore et présentée au paragraphe (4.2). En effet, les vecteurs réalisables appartiennent au sous espace engendré par  $N_{\lambda_i}$  ( $v_i \in \text{span}\{N_{\lambda_i}\}$ ).

Une autre façon pour faire ce calcul se base sur l'équation d'état du système en boucle fermée (4.23). Ainsi, pour chaque valeur propre désirée  $\lambda_i$ , les vecteurs propres, en boucle fermée satisfont l'équation (4.26). Cette dernière équation donne :

$$(\lambda_i I - A)v_i + BKv_i = 0 \quad (4.37)$$

Ce qui conduit à :

$$v_i = -(\lambda_i I - A)^{-1} BKv_i \quad (4.38)$$

En posant  $n_i = -Kv_i$  et  $E_i = (\lambda_i I - A)^{-1} B$ , L'équation (4.37) implique :

$$v_i = E_i n_i \quad (4.39)$$

$$n_i = -Kv_i \quad (4.40)$$

L'équation (4.39) montre que chaque vecteur propre  $v_i$  doit être une combinaison linéaire des colonnes de la matrice  $E_i$  avec :

$$E_i = (\lambda_i I - A)^{-1} B \quad (4.41)$$

Le sous espace engendré par les colonnes de la matrice  $E_i$  est de dimension  $m$  (le rang de la matrice  $B$  et qui est aussi égal au nombre des variables de contrôle indépendantes).

Ainsi, le nombre de ces variables détermine la dimension du sous espace dans lequel les vecteurs réalisables doivent résider.

Par conséquent, si on choisit un vecteur résident dans le sous espace engendré par les colonnes de  $(\lambda_i I - A)^{-1} B$ , ce vecteur sera réalisé (toutes ses entrées seront affectées précisément).

#### 4.5 Projection des vecteurs propres désirés

En général, un vecteur propre désiré  $v_i^d$  peut ne pas résider dans le sous espace engendré par les colonnes de  $E_i = (\lambda_i I - A)^{-1} B$  et ne peut donc pas être réalisé. Le meilleur choix possible pour un vecteur réalisable est fait en projetant  $v_i^d$  sur  $E_i$ . Le vecteur ainsi obtenu est noté  $v_i^r$ .

L'équation (4.39) devient :

$$v_i^r = E_i n_i \quad (4.42)$$

Avec  $v_i^r \in R^m$ .

Pour faire cette projection, la valeur de  $n_i$  correspondante doit être déterminée.  $n_i$  est alors choisi de manière à minimiser une fonction de performance. Un choix possible de cette fonction de performance est donné par :

$$J_p = \|v_i^d - v_i^r\|^2 = \|v_i^d - E_i n_i\|^2 \quad (4.43)$$

Ainsi,  $n_i$  qui minimise  $J_p$  doit satisfaire :

$$\frac{dJ_p}{dn_i} = 0 \quad (4.44)$$

Par ailleurs,

$$\frac{dJ_p}{dn_i} = 2E_i^T(E_i n_i - v_i^d) \quad (4.45)$$

L'équation (4.44) nous donne :

$$n_i = (E_i^T E_i)^{-1} E_i^T v_i^d \quad (4.46)$$

Et à partir de l'équation (4.42), on déduit :

$$v_i^r = E_i (E_i^T E_i)^{-1} E_i^T v_i^d \quad (4.47)$$

D'une façon générale, la fonction de performance peut être exprimée comme suit :

$$J = (v_i^r - v_i^d)^* P_i (v_i^r - v_i^d) \quad (4.48)$$

Où l'indice ( $^*$ ) indique le nombre complexe transposé et  $P_i$  est une matrice symétrique définie positive dont les éléments peuvent être choisis pour pondérer la différence entre certains éléments, des vecteurs désirés et réalisables, plus que d'autres. Typiquement, cette matrice peut être choisie diagonale pour simplifier.

## 4.6 Calcul des gains

Le calcul des gains repose sur le calcul des vecteurs propres réalisables. Dans un premier temps nous allons calculer le gain de retour d'état dans le cas où les valeurs propres sont distinctes. Par la suite, on va traiter le cas, plus général, où les valeurs propres sont multiples, introduire le concept de vecteur propre généralisé, et enfin présenter deux méthodes pour le calcul du gain de retour d'état dans ce cas de figure.

### 4.6.1 Cas où les valeurs propres sont distinctes

Comme on a pu le voir dans le paragraphe précédent, le calcul des vecteurs réalisables  $v_i^r$  peut s'effectuer en projetant les vecteurs désirés  $v_i^d$  sur les sous espaces prescrits  $E_i$ . Par ailleurs, le calcul des vecteurs  $n_i$  peut être fait. Les équations qui permettent ces calculs sont (4.46) et (4.47). Par ailleurs, l'équation (4.40) permet d'exprimer les  $n_i$  en fonction des  $v_i^r$  et de la matrice des gains  $K$  comme suit :

$$n_i = -Kv_i^r \quad (4.49)$$

Cette dernière équation nous permet le calcul de la matrice  $K$  :

$$K = -[n_1 \ n_2 \ \dots \ n_n][v_1^r \ v_2^r \ \dots \ v_n^r]^{-1} \quad (4.50)$$

Si les valeurs propres désirées sont complexes, une méthode similaire à celle utilisée dans le paragraphe (4.2) peut être utilisée pour calculer  $K$ .

### 4.6.2 Cas où les valeurs propres sont multiples

Ce cas nous intéresse parce que, comme on va le voir ultérieurement lors du design du contrôleur pour hélicoptère, notre choix sera porté sur une valeur propre de multiplicité 4.

Les résultats du théorème de Moore, présenté dans le paragraphe (4.2), ont été étendus par un article (Klein et Moore, 1977) et dans lequel la classe des chaînes des vecteurs propres généralisés qui peuvent être réalisés, pour un ensemble de valeurs propres non distinctes, a été caractérisée.

Par ailleurs, les auteurs ont proposé un algorithme pour le calcul de la matrice de retour d'état qui donne les valeurs propres, en boucle fermée, sélectionnées ainsi que les chaînes des vecteurs propres généralisés correspondants. Ainsi, en considérant un système caractérisé par l'équation (4.23) ayant été obtenu en appliquant un retour d'état  $u = Kx$  sur le système caractérisé par l'équation (4.1) avec  $x \in R^n$  et  $u \in R^m$ , Chen (Chen, 1970) a montré que si le système, en boucle fermée, (la matrice  $A - BK$ ) contient  $k$  blocs d'ordres  $d_1, d_2, \dots, d_k$  dans sa forme canonique de Jordan, il existe  $k$  chaînes de vecteurs propres généralisés définies par :

$$(\lambda_i I - A + BK)v_{i1} = 0 \quad (4.51)$$

$$(\lambda_i I - A + BK)v_{ij} = v_{j-1}, \quad j = 2, 3, \dots, d_i \quad (4.52)$$

$k$  étant le nombre des valeurs propres distinctes et  $d_i$  est le nombre représentant la multiplicité de la valeur propre  $\lambda_i$ . Ainsi, la proposition suivante permet de caractériser la classe des chaînes de vecteurs propres généralisés réalisables :

Proposition de Klein et Moore (1977) :

Soit  $\{\lambda_i / i \in [1, 2, \dots, k], k \leq n\}$  un ensemble symétrique de nombres complexes et soit  $\{d_i / i \in [1, 2, \dots, k], k \leq n\}$  un ensemble de nombres entiers positifs satisfaisant  $\sum_{i=1}^k d_i = n$ .

Alors, il existe une matrice de retour d'état  $K$ , constituée d'éléments réels, telle que pour

chaque  $i \in [1, 2, \dots, k]$ , les équations (4.50) et (4.51) sont vérifiées si et seulement si les trois conditions suivantes sont satisfaites :

- Les vecteurs  $\{v_{ij}, j \in [1, \dots, d_i], i \in [1, \dots, k]\}$  appartenant à  $C^n$  sont linéairement indépendants;
- Pour chaque  $i \in \{1, \dots, k\}$ , il existe  $j \in \{1, \dots, k\}$  tels que  $d_i = d_j$ ,  $\lambda_i = \lambda_j^*$  et  $v_{il} = v_{jl}^*$  pour  $l = 1, 2, \dots, d_i$ ;
- Pour chaque  $i \in \{1, \dots, k\}$ , il existe un ensemble de vecteurs  $\{w_{ij}, j \in [1, \dots, d_i]\}$  satisfaisant :

$$[\lambda_i I - A, \quad B] \begin{bmatrix} v_{ij} \\ w_{ij} \end{bmatrix} = v_{ij-1} \quad (4.53)$$

Avec  $v_{i0} = 0$ . La matrice des gains  $K$  est alors calculée en considérant :

$$Kv_{ij} = w_{ij} \text{ pour tout } j \in \{1, \dots, d_i\} \text{ et tout } i \in \{1, \dots, k\} \quad (4.54)$$

Un algorithme de calcul de la matrice de retour d'état  $K$ , permettant d'aboutir à la configuration de Jordan (Chen, 1970) correspondante au choix des nombres entiers  $k, d_i, i \in \{1, \dots, k\}$  a également été proposé par les auteurs comme suit :

- Calcul des matrices de rang maximal :

$$P_i = \begin{bmatrix} P_{1i} \\ P_{2i} \end{bmatrix} \quad \text{et} \quad Q_i = \begin{bmatrix} Q_{1i} \\ Q_{2i} \end{bmatrix}$$

Ces deux matrices doivent satisfaire :

$$[\lambda_i I - A \quad B] \begin{bmatrix} P_{1i} \\ P_{2i} \end{bmatrix} = 0 \quad (4.55)$$

$$[\lambda_i I - A \quad B] \begin{bmatrix} Q_{1i} \\ Q_{2i} \end{bmatrix} = I \quad (4.56)$$

- Génération des chaînes de vecteurs propres :

$$v_{i1} = P_{1i} s_{i1}$$

$$v_{i2} = Q_{1i} v_{i1} + P_{1i} s_{i2}$$

$$v_{ij} = Q_{1i} v_{ij-1} + P_{1i} s_{ij}$$

$$w_{i1} = P_{2i} s_{i1}$$

$$w_{ij} = Q_{2i} v_{ij-1} + P_{2i} s_{ij}$$

Les vecteurs  $s_{ij}$  sont choisis de sorte que la condition 2) de la proposition soit satisfaite.

- Calcul de la matrice  $K$  satisfaisant  $Kv_{ij} = w_{ij}$  pour tout  $j \in \{1, \dots, d_i\}$  et tout  $i \in \{1, \dots, k\}$ .

De la même manière qu'au paragraphe (4.2), la matrice peut être déduite des vecteurs propres généralisés et des vecteurs  $w_{ij}$ . Les vecteurs  $s_{ij}$  doivent être choisis soigneusement afin de produire des vecteurs généralisés indépendants et à même d'amener le système, en boucle fermée, à atteindre les performances requises.

D'autres chercheurs se sont intéressés au calcul de la matrice de retour d'état dans le cas général où les valeurs propres ne sont pas distinctes. Ainsi, Garrard, Liebst et Farm (GARRARD, LIEBST et FARM, 1987) ont présenté une procédure de calcul. Cette approche est résumée ci-après.

En partant de l'équation du système en boucle fermée (4.23), L'équation (4.38) relie les vecteurs propres généralisés et les valeurs propres avec la matrice des gains  $K$ . Par ailleurs, l'équation (4.40) définit le paramètre  $n_i$  comme étant égal à  $-Kv_i$ . Ce paramètre est choisi de manière à minimiser l'indice de performance de l'équation (4.48). L'équation (4.40) devient :

$$K = -MV^{-1} \quad (4.57)$$

Avec  $M$  la matrice ayant pour colonnes les  $n_i$  et  $V$  est une matrice dont les colonnes sont les vecteurs  $v_i^r$ . Une fois les vecteurs  $n_i$  et  $v_i^r$  trouvés, la matrice  $K$  est calculée par l'équation (4.57). Par ailleurs, l'équation (4.38) peut être réécrite sous la forme :

$$(\lambda_i I - A)v_i^r = Bn_i \quad (4.58)$$

Pour une valeur propre fixée  $\lambda_i$ , il est désiré de minimiser  $J_i$  (défini par l'équation (4.48)) et qui est sujet à l'équation (4.58). Ainsi en combinant l'équation (4.58) à  $J_i$  avec les multiplicateurs de Lagrange  $\mu_i$ , on obtient :

$$\bar{J}_i = (v_i^r - v_i^d)^* P_i (v_i^r - v_i^d) + \mu_i^* [(\lambda_i I - A)v_i^r - Bn_i] \quad (4.59)$$

Le paramètre  $\bar{J}_i$  de l'équation (4.59) est minimisé quand :

$$\frac{\partial \bar{J}_i}{\partial n_i} = -\mu_i^* B = 0 \quad (4.60)$$

$$\frac{\partial \bar{J}_i}{\partial v_i^r} = (v_i^r - v_i^d)^* P_i + \mu_i^* (\lambda_i I - A) = 0 \quad (4.61)$$

Il en résulte, sous forme matricielle :

$$\begin{pmatrix} (\lambda_i - A) & -B & 0 \\ 0 & 0 & -B^T \\ P_i & 0 & (\lambda_i - A)^* \end{pmatrix} \begin{bmatrix} v_i^a \\ m_i \\ \mu_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ P_i v_i^d \end{bmatrix} \quad (4.62)$$

En posant :

$$L_i = \begin{pmatrix} (\lambda_i - A) & -B & 0 \\ 0 & 0 & -B^T \\ P_i & 0 & (\lambda_i - A)^* \end{pmatrix} \quad (4.63)$$

L'équation (4.61) devient :

$$L_i \begin{bmatrix} v_i^a \\ m_i \\ \mu_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ P_i v_i^d \end{bmatrix} \quad (4.64)$$

Les vecteurs  $v_i^r$  et  $m_i$  peuvent alors être déduits de l'équation :

$$\begin{bmatrix} v_i^a \\ m_i \\ \mu_i \end{bmatrix} = N_i^{-1} \begin{bmatrix} 0 \\ 0 \\ P_i v_i^d \end{bmatrix} \quad (4.65)$$

Finalement, le calcul de la matrice de retour d'état  $K$  peut être fait à partir de l'équation (4.57).

## CHAPITRE 5

### CONCEPTION DU CONTRÔLEUR POUR HÉLICOPTÈRE

Le but de ce chapitre est de proposer une structure de contrôle simple pour contrôler l'hélicoptère Bell 412 pendant la manœuvre *hover* tout en intégrant les spécifications relatives aux HQ et contenues dans l'ADS-33. La méthodologie qui sera présentée utilise les techniques d'affectation de la structure modale pour la conception de lois de contrôle qui permettent de réaliser directement les types de réponses désirés.

Pour atteindre cet objectif, les types de réponses désirés sont utilisés pour générer un modèle (sous forme d'espace d'état) ce qui permet d'identifier l'ensemble des valeurs et vecteurs propres désirés. Par la suite, les techniques d'affectation de la structure modale sont utilisées pour la conception de lois de contrôle qui donnent la meilleure approximation (au sens des moindres carrés) de la structure propre désirée.

La structure de contrôle est divisée en boucle interne et boucle externe. L'affectation de la structure modale est utilisée pour la conception de lois de contrôle pour la boucle interne permettant ainsi de découpler la vitesse verticale et les vitesses angulaires de roulis, tangage et lacet et procurer le type de réponses commande de vitesse (*Rate Command*) dans la boucle de contrôle de chaque canal. L'hélicoptère étant un système naturellement instable, le placement des vecteurs propres est utilisé pour stabiliser les modes instables et procurer des bandes passantes appropriées. Par ailleurs, le placement des vecteurs propres vise le découplage modal. En outre, une matrice de compensation est utilisée dans la boucle interne pour réduire le problème de couplage modes-entrées du système (voir figure 5.1).

Après fermeture des boucles internes, la réponse de l'hélicoptère aux entrées du pilote a la forme de quatre systèmes découplés de premier ordre, ce qui ouvre la voie à la conception de la boucle externe. Chaque canal de la boucle externe peut être traité séparément et une variété de types de réponses désirés peut alors être réalisée (ACAH en particulier) en utilisant des

techniques dédiées aux systèmes à une seule entrée et une seule sortie SISO (*Single Input Single Output*).

Le modèle mathématique, décrivant la dynamique de l'hélicoptère, et qui sera utilisé pour le design du contrôleur, a été fourni par le logiciel HELISIM (qui fera l'objet du chapitre 6). Ce modèle est écrit sous forme de variable d'état comme suit :

$$\overset{\square}{x}(t) = Ax(t) + Bu(t) \quad (5.1)$$

$$y(t) = Cx(t) \quad (5.2)$$

Notons que toutes les variables d'état sont des variations autour du point d'équilibre (voir paragraphe 2.5). Ce point d'équilibre a été choisi pour une altitude de 500 pieds et une valeur nulle pour toutes les variables d'état. Les matrices du système sont :

$$A = \begin{bmatrix} -0.570203 & 0.0126392 & 0.00621049 & 0.137234 & 0.467613 & -0.12547 & -0.0149969 & -0.00305424 \\ 0.257673 & -0.031165 & 0.0153012 & -10.3929 & -2.70343 & 0.378434 & 0.0214262 & -0.00980549 \\ 0.0578227 & 0.000373782 & -0.311831 & -0.712819 & 0.213162 & 0.940126 & 0.232868 & 0.461841 \\ 0.998976 & 0 & 0 & 0 & 0 & 0.0452493 & 0 & 0 \\ -2.13777 & -0.107229 & -0.0116698 & 0.0285303 & -3.82668 & 4.77314 & -0.281239 & 0.876043 \\ -0.102858 & -0.00948746 & 0.00334082 & 0.000953749 & 12.7014 & -30.9573 & 0.0807634 & 0.0571409 \\ -0.661691 & -0.0129566 & -0.00585714 & 0.0321259 & -12.4082 & 24.7438 & -0.124384 & 10.3798 \\ -0.00306561 & 0 & 0 & 0 & 1 & 0.0676799 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1.73733 & -0.0800566 & -0.25652 & -0.132318 \\ -2.64041 & 2.76947 & 0.343154 & 0.890595 \\ -0.197531 & -27.3584 & 0.0273133 & 0.371054 \\ 0 & 0 & 0 & 0 \\ 2.97338 & -3.17439 & 6.47515 & -2.01631 \\ 0.144659 & 1.95917 & 0.320913 & 1.45181 \\ 0.62182 & -0.421427 & 1.54279 & -1.28706 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Avec  $x \in \mathbb{R}^8$ ,  $u \in \mathbb{R}^4$  et  $y \in \mathbb{R}^4$ .

$$x = \begin{bmatrix} q \\ u \\ w \\ \theta \\ p \\ r \\ v \\ \phi \end{bmatrix} \begin{array}{l} \text{vitesse angulaire de tangage} \\ \text{vitesse longitudinale} \\ \text{vitesse verticale} \\ \text{angle de tangage} \\ \text{vitesse angulaire de roulis} \\ \text{vitesse angulaire de glissement} \\ \text{vitesse latérale} \\ \text{angle de roulement} \end{array}$$

$$u = \begin{bmatrix} \delta_{long} \\ \delta_{coll} \\ \delta_{lat} \\ \delta_{ped} \end{bmatrix} \begin{array}{l} \text{commande longitudinale du cyclique} \\ \text{entrée de contrôle du rotor principal} \\ \text{commande latérale du cyclique} \\ \text{entrée de contrôle du rotor de la queue} \end{array}$$

$$y = \begin{bmatrix} q \\ w \\ p \\ r \end{bmatrix} \begin{array}{l} \text{vitesse angulaire de tangage} \\ \text{vitesse verticale} \\ \text{vitesse angulaire de roulis} \\ \text{vitesse angulaire de glissement} \end{array}$$

La dynamique de l'hélicoptère Bell 412 dans cette condition de vol *hover* présente deux modes instables :

- Mode associé à la vitesse longitudinale  $u$  avec deux valeurs propres complexes conjuguées  $(0.0863082 \pm 0.602753j)$ . Ce mode a un amortissement de -0.14 et une fréquence 0.09rad/s.

- Mode associé à la vitesse latérale  $v$  avec deux valeurs propres  $(0.0372627 \pm 0.619949j)$ , un amortissement de  $-0.0599$  et une fréquence  $0.098$  rad/s.

La présence de ces deux modes instables montre la nécessité d'un contrôleur pour stabiliser l'hélicoptère et permettre la satisfaction des HQ.

### 5.1 Conception de la boucle interne du contrôleur

La partie interne du contrôleur est constituée de quatre boucles :

- Boucle reliant la vitesse angulaire de tangage à la commande longitudinale du cyclique;
- Boucle reliant la vitesse verticale à l'entrée de contrôle du rotor principal;
- Boucle reliant la vitesse angulaire de roulis à la commande latérale du cyclique;
- Boucle reliant la vitesse angulaire de glissement à l'entrée de contrôle du rotor de la queue.

Ces quatre boucles visent le contrôle des vitesses. Ainsi, leur entrées sont indiquées avec l'indice  $c$  relatif à la consigne (voir figure 5.1).

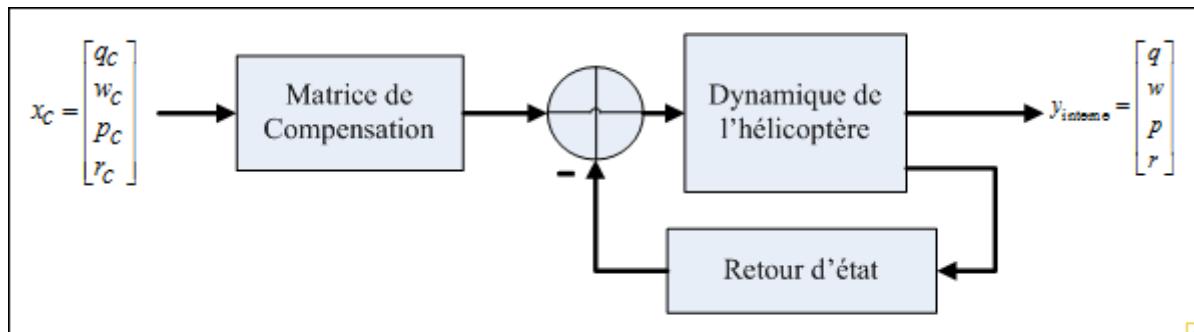


Figure 5.1 Boucle interne du contrôleur

La conception de cette partie interne du contrôleur est effectuée en calculant la matrice de retour d'état (affectation de la structure modale) et la matrice de compensation permettant au système de se comporter comme quatre sous-systèmes découplés de premier ordre. Pour atteindre les performances désirées, les pôles de ces quatre sous-systèmes sont choisis stables et offrant des bandes passantes suffisantes.

Les fonctions de transfert désirées (voir chapitre 3) entre les vitesses de consigne et les sorties correspondantes sont :

$$\frac{q}{q_c} = \frac{\lambda_q}{s + \lambda_q} \quad (5.3)$$

$$\frac{w}{w_c} = \frac{\lambda_w}{s + \lambda_w} \quad (5.4)$$

$$\frac{p}{p_c} = \frac{\lambda_p}{s + \lambda_p} \quad (5.5)$$

$$\frac{r}{r_c} = \frac{\lambda_r}{s + \lambda_r} \quad (5.6)$$

En outre (Yanguo et Huanjin, 2009), les fonctions de transfert désirées reliant la vitesse  $u$  à la consigne de la vitesse angulaire de tangage  $q_c$  ainsi que la vitesse  $v$  à la consigne de vitesse angulaire de roulis sont les suivantes :

$$\frac{u(s)}{q_c(s)} = \frac{\lambda_q}{s(s + \lambda_u)} \quad (5.7)$$

$$\frac{v(s)}{p_c(s)} = \frac{\lambda_p}{s(s + \lambda_v)} \quad (5.8)$$

$s$  étant l'opérateur de Laplace. L'équation (5.7) peut être réécrite sous la forme :

$$\frac{u(s)}{q(s)} = \frac{u(s)}{q_c(s)} \frac{q_c(s)}{q(s)} = \left(1 + \frac{\lambda_q}{s}\right) \left(\frac{1}{s + \lambda_u}\right) \quad (5.9)$$

En remplaçant  $\theta(s)$  par  $\frac{q(s)}{s}$ , l'équation (5.9) devient :

$$\ddot{u} = -\lambda_u u + q + \lambda_q \theta \quad (5.10)$$

De la même manière on obtient pour la vitesse latérale  $v$  :

$$\ddot{v} = -\lambda_v v + p + \lambda_p \phi \quad (5.11)$$

Les équations (5.3) à (5.11), exprimant les fonctions de transfert désirées, permettent de déduire le modèle idéal pour l'hélicoptère sous forme d'équation d'état (la boucle interne étant fermée) comme suit :

$$\ddot{x} = A_d x + B_d x_c \quad (5.12)$$

Avec :  $x_c = \begin{bmatrix} q_c \\ w_c \\ p_c \\ r_c \end{bmatrix}$  et  $x$  le vecteur d'état ayant été défini au début du chapitre.

L'expression des matrices  $A_d$  et  $B_d$  sont données dans les équations (5.13) et (5.14). Les valeurs de  $\lambda_w, \lambda_p, \lambda_q$  et  $\lambda_r$  sont déterminées en fonction des HQ, ainsi, une bande passante de 4 rad/s est choisie pour chaque canal, or les fonctions de transfert étant de premier ordre, la valeur des paramètres  $\lambda_i$  est exactement celle de la bande passante correspondante.

$$A_d = \begin{bmatrix} -\lambda_q & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -\lambda_u & 0 & \lambda_q & 0 & 0 & 0 & 0 \\ 0 & 0 & -\lambda_w & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\lambda_p & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\lambda_r & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -\lambda_v & \lambda_p \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (5.13)$$

Et

$$B_d = \begin{bmatrix} \lambda_q & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \lambda_w & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_p & 0 \\ 0 & 0 & 0 & \lambda_r \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.14)$$

A contrario, les valeurs qui seront choisies pour les paramètres  $\lambda_u$  et  $\lambda_v$  seront, bien sûr positives (raison de stabilité puisqu'ils constituent des valeurs propres désirées) mais faibles en raison de leur signification physique (ces deux valeurs propres sont associées aux forces de traînée), les travaux de Garrard et Low (Garrard et Low, 1990) ont montré que des valeurs respectives de 0.00199 et 0.00526 permettent au système d'avoir une bonne réponse en régime permanent (axe tangage et roulis) pour un hélicoptère typique. De ce fait, les valeurs choisies des différentes  $\lambda_i$  sont les suivantes :

$$\lambda_u = 0.00199$$

$$\lambda_v = 0.00526$$

$$\lambda_w = 4$$

$$\lambda_p = 4$$

$$\lambda_q = 4$$

$$\lambda_r = 4$$

La matrice  $A_d$  est maintenant déterminée, ce qui nous permet d'obtenir la configuration valeurs/vecteurs propres désirés. Cette configuration est obtenue à l'aide de Matlab. On tape la commande  $eig (A_d)$ , ce qui nous donne l'ensemble des valeurs et vecteurs propres désirés.

Liste des valeurs propres désirées :

$$\lambda_1 = -0.002$$

$$\lambda_2 = -0.0001$$

$$\lambda_3 = -4$$

$$\lambda_4 = -4$$

$$\lambda_5 = -0.0053$$

$$\lambda_6 = -0.0001$$

$$\lambda_7 = -4$$

$$\lambda_8 = -4$$

Tableau 5.1 Liste des vecteurs propres désirés

	$V_1^d$	$V_2^d$	$V_3^d$	$V_4^d$	$V_5^d$	$V_6^d$	$V_7^d$	$V_8^d$
$q$	0	0	0.9701	0	0	0	0	0
$u$	1	1	0	0	0	0	0	0
$w$	0	0	0	1	0	0	0	0
$\theta$	0	0.0005	-0.2425	0	0	0	0	0
$p$	0	0	0	0	0	0	0.9701	0
$r$	0	0	0	0	0	0	0	1
$v$	0	0	0	0	1	1	0	0
$\phi$	0	0	0	0	0	0.0013	-0.2425	0

La deuxième phase de la conception est le calcul des vecteurs propres réalisables à partir de ceux désirés pour en déduire la matrice de retour d'état en utilisant les techniques d'affectation de la structure modale (voir chapitre 4). La loi de commande considérée a la forme suivante :

$$u = -Kx \quad (5.15)$$

Le calcul de la matrice  $K$  est effectué à l'aide de Matlab (voir le code en annexe 1). Les vecteurs propres réalisables (*achievable*) ainsi que la matrice  $K$  sont donnés ci-après :

Tableau 5.2 Liste des vecteurs propres réalisables

	$V_1^a$	$V_2^a$	$V_3^a$	$V_4^a$	$V_5^a$	$V_6^a$	$V_7^a$	$V_8^a$
$q$	0	0	0.6845	-0.0291	0	0	0.0106	-0.0166
$u$	1	1	-0.4539	-0.0469	0	0	0.0221	-0.0281
$w$	0	0	-0.03	0.9945	0	0	0.0217	-0.0218
$\theta$	-0.0003	-0.0005	-0.1709	0.0075	-0.0003	-0.0003	-0.0042	-0.0032
$p$	0	0	0.0112	0.0211	0	0	0.8024	0.1604
$r$	-0.0001	-0.0001	-0.0153	-0.0218	0.0002	0.0002	0.1694	0.8327
$v$	0	0	0.0215	-0.0381	1	1	0.3356	-0.3298
$\phi$	-0.0011	-0.0011	-0.002	-0.0049	0.0028	0.0033	-0.204	-0.0569

$$K = \begin{bmatrix} -0.1882 & 0.0145 & -0.0358 & 0.0561 & 0.3213 & 0.0017 & -0.0175 & 0.0265 \\ 0.0054 & -0.0001 & -0.2850 & 0.0664 & 0.0059 & -0.0575 & 0.0016 & -0.0276 \\ -1.7348 & -0.0570 & 0.0761 & -0.0799 & -1.9289 & -0.2289 & -0.0443 & 0.1879 \\ 0.1913 & -0.0026 & 0.3102 & 0.0569 & 1.1859 & -10.8535 & 0.0672 & 0.3132 \end{bmatrix} \quad (5.16)$$

Par ailleurs, l'obtention de  $B_d$  nécessite l'utilisation d'une matrice de compensation  $H$  de sorte que :

$$B * H \approx B_d \quad (5.17)$$

La matrice  $B$  n'étant pas carrée, le calcul de la matrice  $H$  est effectué en calculant la matrice pseudo-inverse de  $B$ , ce qui donne :

$$H = (B^T * B)^{-1} * B^T * B_d \quad (5.18)$$

Ainsi, l'équation (5.18) donne :

$$H = \begin{bmatrix} 0.5759 & -0.0465 & 0.0915 & 0.0584 \\ -0.0006 & -0.3025 & -0.0002 & 0.0059 \\ -1.0147 & 0.0873 & 0.5267 & 0.3934 \\ -0.0568 & 0.2725 & 0.2756 & 2.5510 \end{bmatrix} \quad (5.19)$$

Le calcul de  $B * H$  nous donne :

$$B * H = \begin{bmatrix} 3.8642 & -0.0590 & -0.0069 & -0.0611 \\ -0.7020 & -0.2655 & -0.0533 & -0.0228 \\ -0.0590 & 3.9675 & 0.0000 & -0.0779 \\ 0.0000 & -0.0000 & -0.0000 & -0.0000 \\ -0.0069 & 0.0000 & 3.9983 & 0.0196 \\ -0.0611 & -0.0779 & 0.0196 & 3.5905 \\ -0.1577 & -0.2214 & 0.0596 & -1.2082 \\ -0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix} \quad (5.20)$$

On constate que la méthode utilisée pour calculer la matrice  $H$  a donné un bon résultat puisque le produit  $B * H$  est très proche de  $B_d$ .

Pour étudier le comportement du système, après fermeture de la boucle interne (voir figure 5.1) et en considérant les deux matrices  $K$  et  $H$  données par les équations (5.16) et (5.19), on va s'intéresser aux valeurs singulières de la matrice  $F_{BIF}$  (l'indice  $BIF$  pour boucle interne fermée). Cette matrice est définie par :

$$\begin{bmatrix} q \\ w \\ p \\ r \end{bmatrix} = F_{BIF} \begin{bmatrix} q_c \\ w_c \\ p_c \\ r_c \end{bmatrix} \quad (5.21)$$

Ou encore par la formule :

$$F_{BIF}(s) = C(I_s - A + BK)^{-1} BH \quad (5.22)$$

Pour étudier le comportement du système en boucle fermée, on va s'intéresser aux valeurs singulières de la matrice  $F_{BIF}$  (cette méthode est équivalente à celle utilisant le diagramme de

Bode pour les systèmes à simple entrée et simple sortie (SISO)). La figure 5.2 illustre les valeurs singulières de la matrice  $F_{BIF}$ , alors que la figure 5.3 montre les valeurs singulières des fonctions de transfert de la boucle de chaque canal. Pour les fréquences d'intérêt (0.1 à 10 rad/s), les figures 5.2 et 5.3 sont très proches ce qui montre que le système, en boucle fermée se comporte essentiellement comme quatre sous-systèmes découplés (correspondant aux quatre canaux). Par ailleurs, les valeurs singulières de chaque canal, sont celles d'un système de premier ordre de bande passante 4 rad/s (voir figure 5.4).

Par conséquent, la loi de commande présentée permet au système, après fermeture de la boucle interne de se rapprocher des fonctions de transfert désirées pour chaque canal. Par ailleurs, étant donné que les fréquences pour lesquelles le découplage n'est pas atteint sont faibles (inférieures à 0.1 rad/s), le pilote sera en mesure de compenser pour les couplages entre canaux.

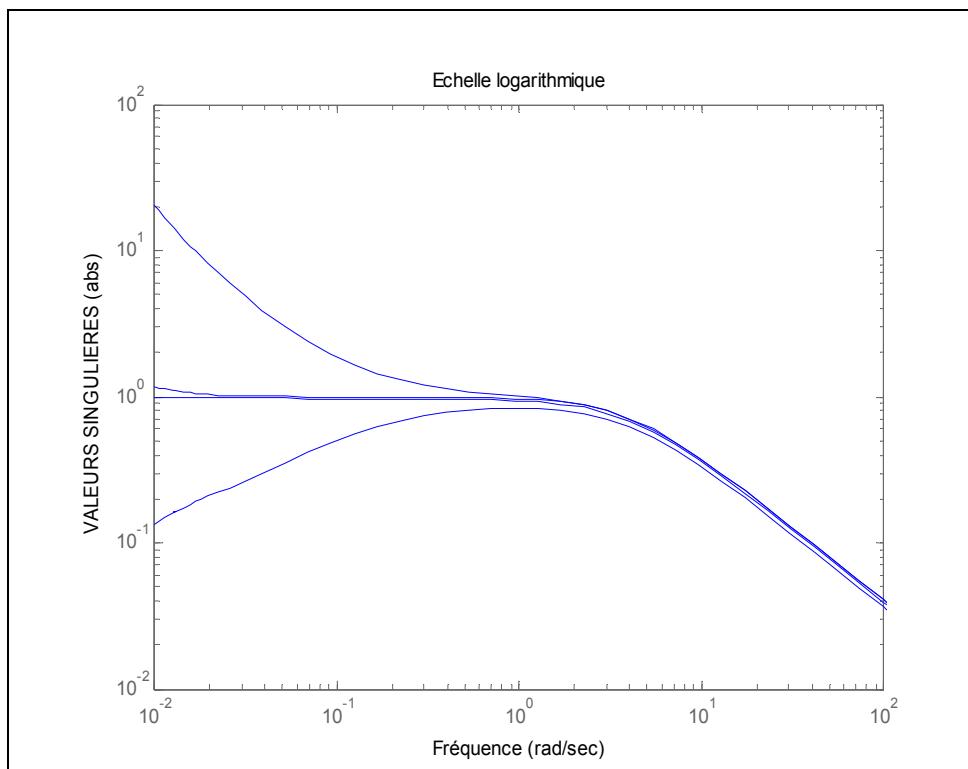


Figure 5.2 Valeurs singulières de  $F_{BIF}$

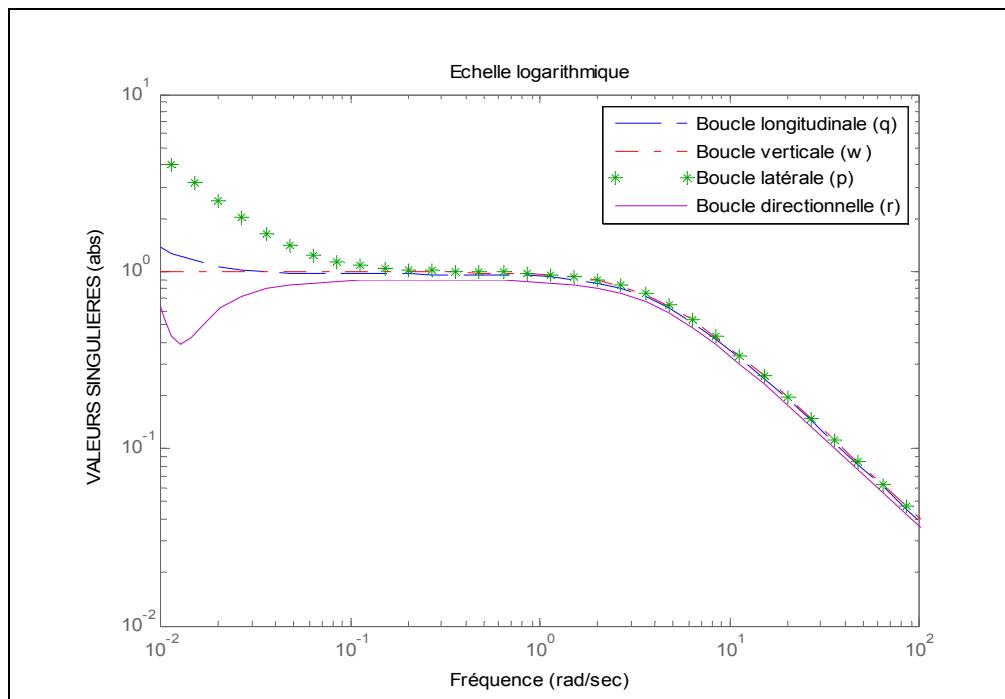


Figure 5.3 Valeurs singulières des fonctions de transfert des quatre canaux

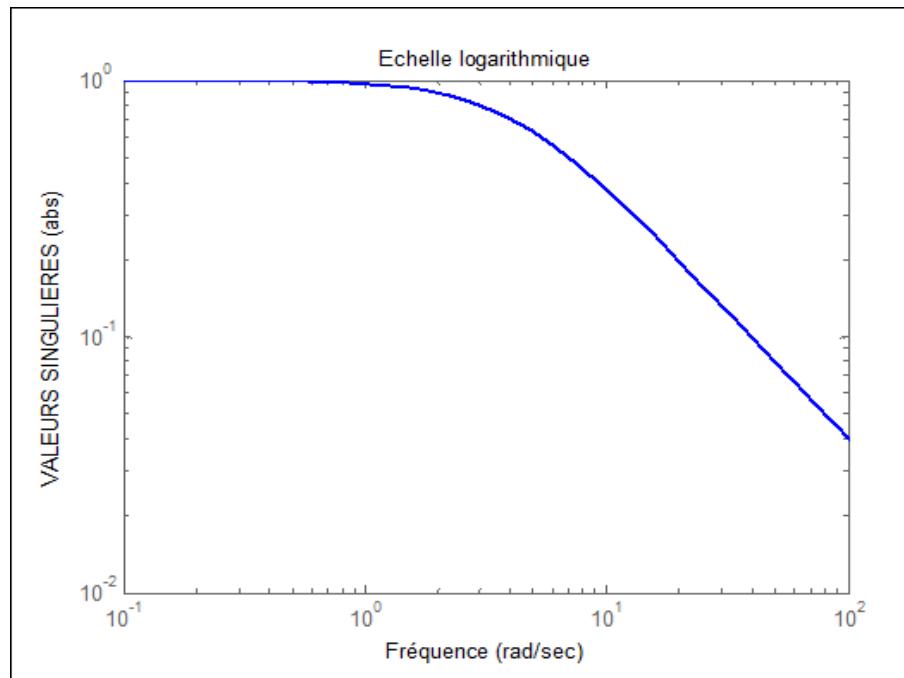


Figure 5.4 Valeurs singulières de  $G(s) = \frac{4}{s+4}$

Une autre façon de vérifier le comportement du système consiste en le tracé du diagramme de Bode des différentes fonctions de transfert. Comme on peut le constater sur la figure 5.5, la réponse du système pour les quatre canaux, pour les fréquences d'intérêt (0.1 à 10 rad/s), est celle d'un système de premier ordre et de bande passante 4 rad/s (voir la figure 5.6 qui

illustre le diagramme de Bode de la fonction de transfert  $\frac{4}{s+4}$ ).

En outre, les éléments non diagonaux de la matrice  $F_{BIF}$  et qui caractérisent le couplage entre les différents canaux sont négligeables.

Pour l'analyse dans le domaine temporel, la réponse du système (évolution des états du système en fonction du temps) à une entrée échelon de 5 deg/s appliquée au canal roulis est illustrée sur les figures 5.7 et 5.8. Les réponses du système en boucle ouverte et en boucle interne fermée ont été tracées sur les mêmes graphiques pour des fins de comparaison.

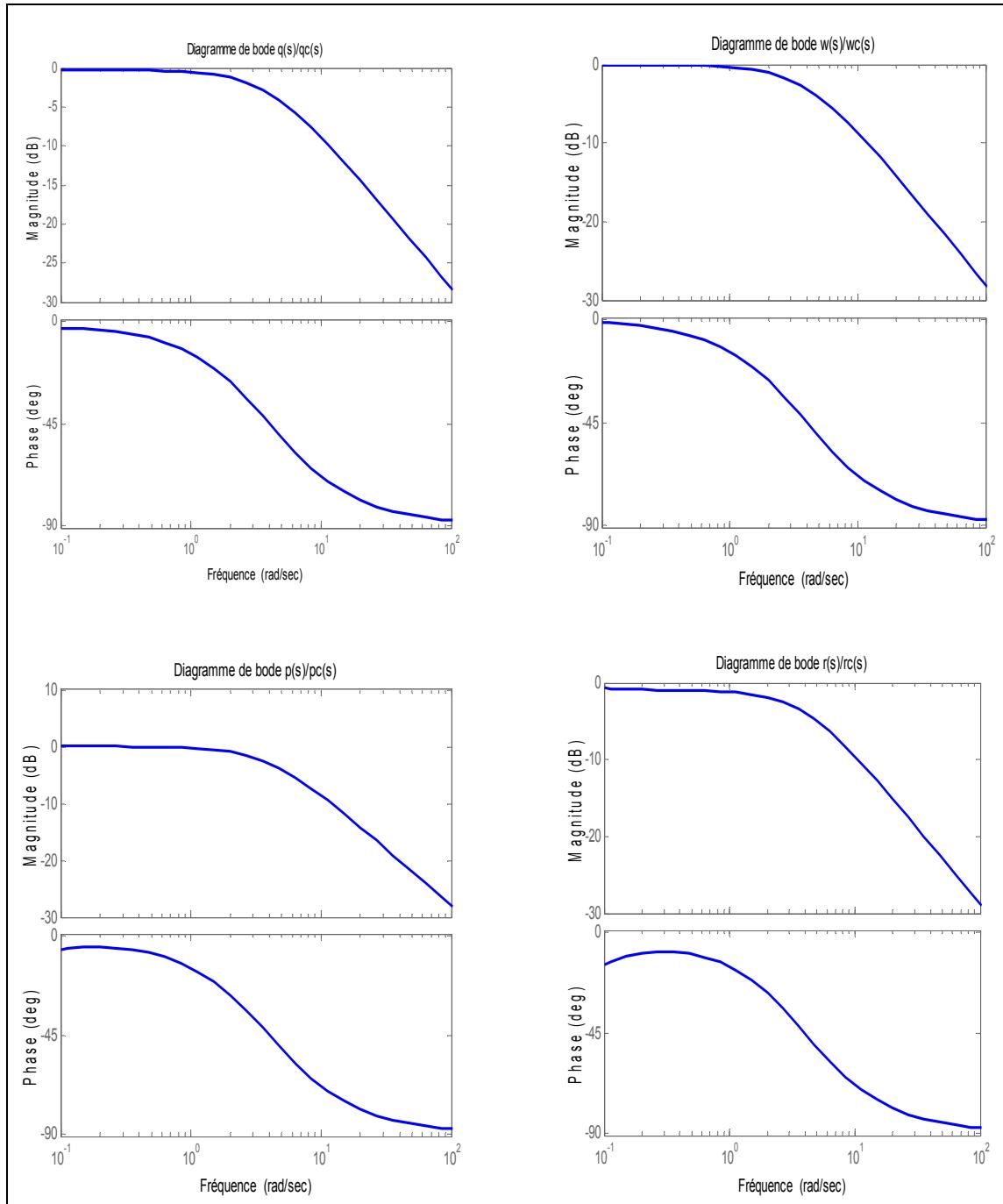


Figure 5.5 Diagramme de Bode des fonctions de transfert des quatre canaux (boucle interne fermée)

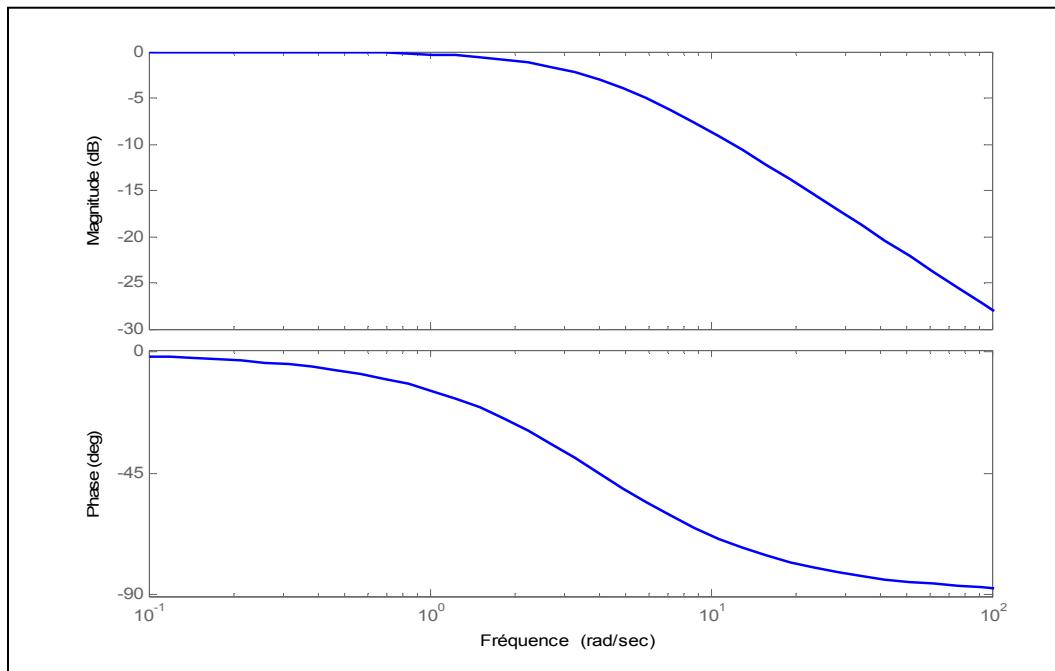


Figure 5.6 Diagramme de Bode de la fonction de transfert  $G(s) = \frac{4}{s+4}$

Les figures 5.7 et 5.8 montrent que les modes longitudinal, vertical et directionnel du système en boucle interne fermée ont été découplés du mode latéral quoique la vitesse de lacet s'éloigne légèrement de la valeur 0, cette divergence va être atténuée après fermeture de la boucle externe (voir paragraphe 5.2). Par ailleurs, la réponse de la vitesse de roulis est celle d'un système de premier ordre avec une bande passante de 4 rad/s, l'angle de roulis augmente linéairement avec le temps ; tandis que la vitesse latérale augmente avec une allure quadratique comme prédit par l'équation (5.8).

Le caractère instable de la boucle ouverte est également illustré par les deux figures 5.7 et 5.8. L'analyse de la réponse aux sollicitations externes des autres canaux conduit à des conclusions similaires.

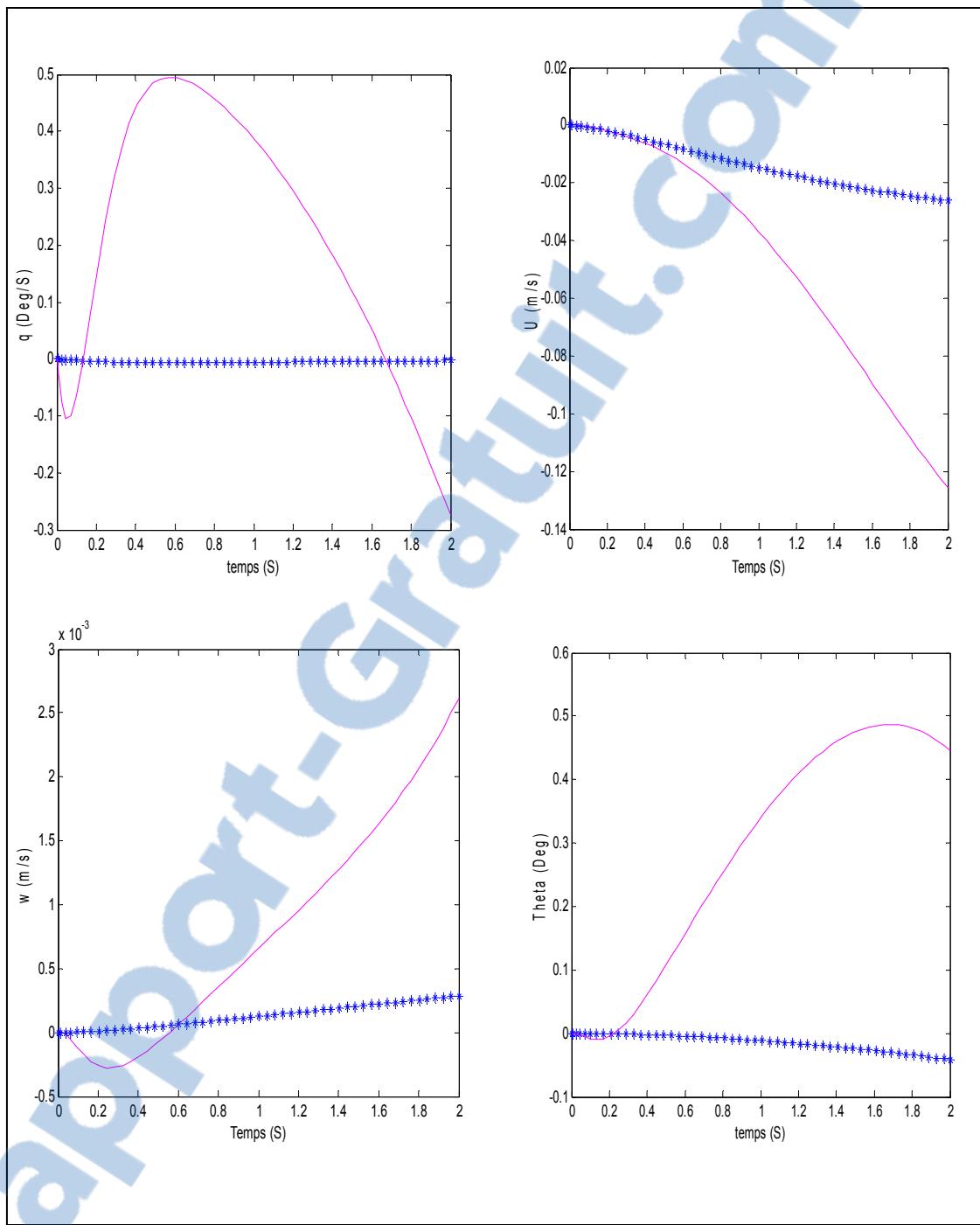


Figure 5.7 Réponse du système ( $q$ ,  $u$ ,  $w$  et  $\theta$ ) (en BO et en BIF) à une entrée échelon de 5 deg/s sur la consigne de la vitesse de roulis

— Boucle ouverte (BO)  
\*\*\*\*\* Boucle interne fermée (BIF)

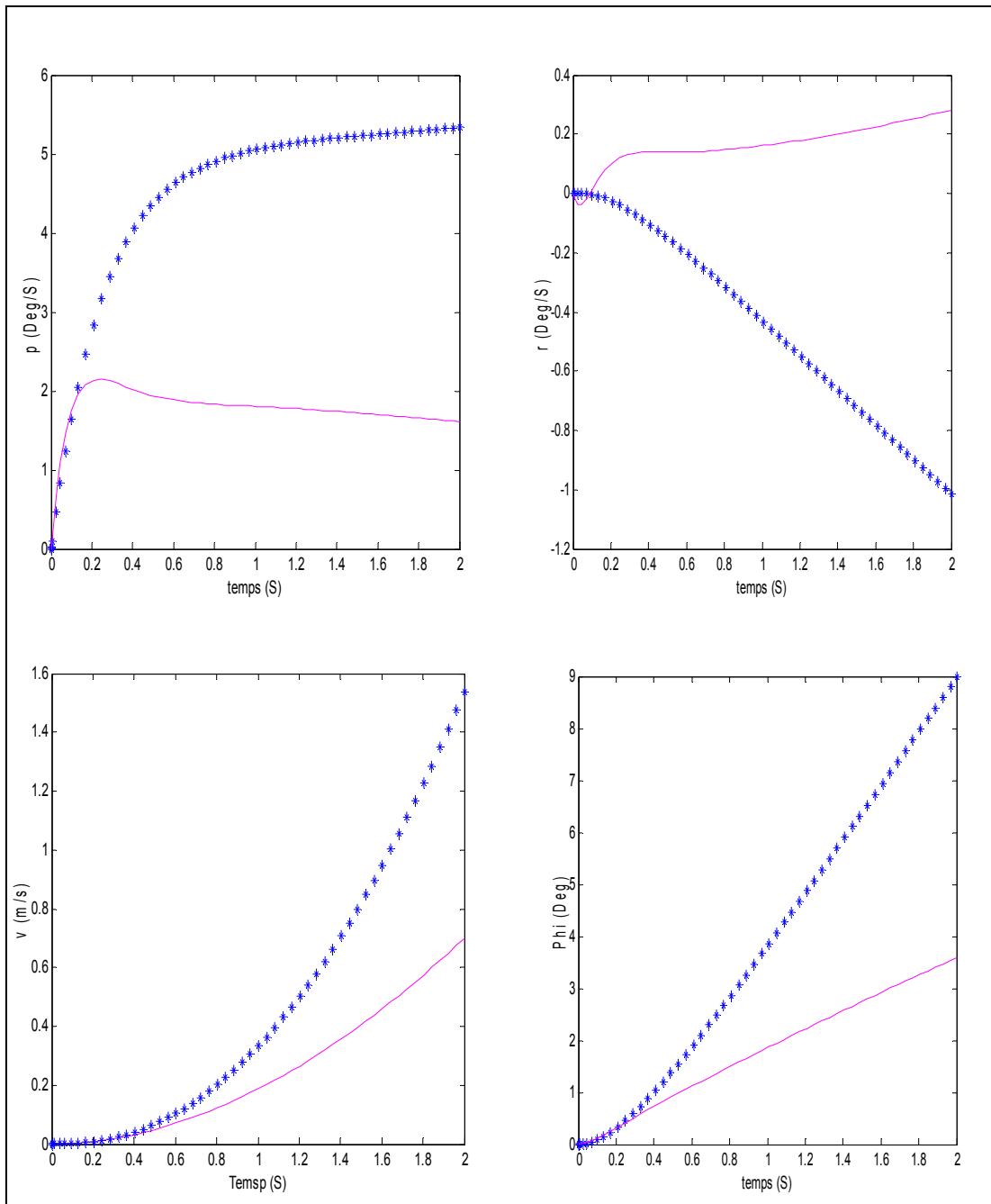


Figure 5.8 Réponse du système ( $p$ ,  $r$ ,  $v$  et  $\phi$ ) (en BO et en BIF) à une entrée échelon de 5 deg/s sur la consigne de la vitesse de roulis

— Boucle ouverte (BO)  
 \*\*\*\*\* Boucle interne fermée (BIF)

## 5.2 Conception de la boucle externe du contrôleur

La conception de la boucle interne a permis de stabiliser le système et de réaliser des types de réponses (commande de vitesse) sur les quatre canaux (longitudinal, vertical, latéral et directionnel). Avec la boucle interne fermée, le comportement du système est proche de celui de quatre axes découplés présentant des types de réponses de premier ordre et les équations (5.3) à (5.8) reflétant les fonctions de transfert désirées sont approximativement réalisées. Ceci ouvre la voie à la conception de la boucle externe permettant de réaliser le type de réponses ACAH (*Attitude Command Attitude Hold*) en particulier.

Le schéma de réalisation du type de réponse ACAH pour le canal roulis est illustré sur la figure 5.9.

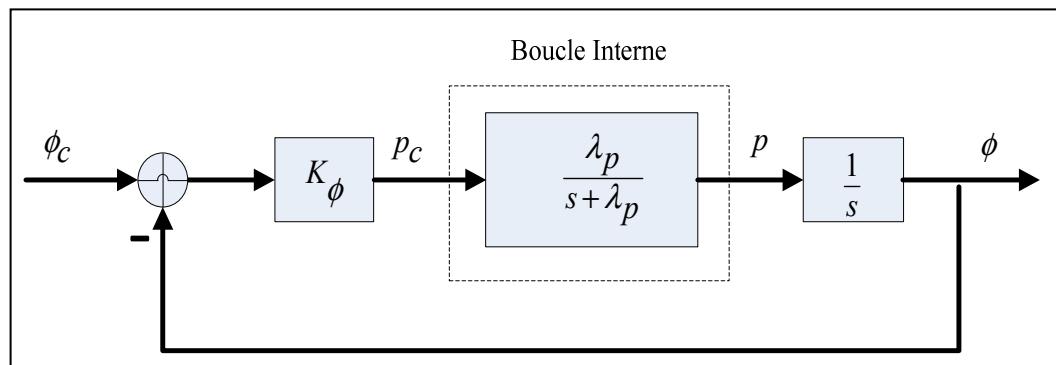


Figure 5.9 Boucle externe du canal roulis

L'intégrateur qui apparaît sur la figure ci-dessus peut ne pas être utilisé puisque l'angle  $\phi$  est mesurable à l'aide d'un capteur. Une boucle analogue à celle illustrée par la figure 5.9 peut être réalisée pour le canal de tangage également. L'expression de  $\frac{\phi}{\phi_c}$  , en considérant une fonction de transfert idéale entre  $p_c$  et  $p$  , est donnée par l'équation (5.23) :

$$\frac{\phi}{\phi_c} = \frac{K_\phi \lambda_p}{(s^2 + \lambda_p s + K_\phi \lambda_p)} \quad (5.23)$$

Pour le canal tangage :

$$\frac{\theta}{\theta_c} = \frac{K_\theta \lambda_q}{(s^2 + \lambda_q s + K_\theta \lambda_q)} \quad (5.24)$$

Si on met  $\frac{\phi}{\phi_c}$  et  $\frac{\theta}{\theta_c}$  sous la forme typique d'une fonction de transfert de deuxième ordre :

$$\frac{\phi}{\phi_c} = \frac{w_{n\phi}^2}{s^2 + 2\xi w_{n\phi} s + w_{n\phi}^2} \quad (5.25)$$

$$\frac{\theta}{\theta_c} = \frac{w_{n\theta}^2}{s^2 + 2\xi w_{n\theta} s + w_{n\theta}^2} \quad (5.26)$$

Un choix judicieux de  $K_\phi = 2$  permet d'égaliser (5.23) et (5.25). La valeur de  $\lambda_p$  ayant déjà été choisie égale à quatre lors du design de la boucle interne du contrôleur, les valeurs de  $\xi$  et  $w_{n\phi}$  peuvent alors être déduites :  $\xi = 0.707$  et  $w_{n\phi} = 2.83 \text{ rad/s}$ . Le choix de  $K_\theta = 2$  conduit aux mêmes valeurs de la fréquence naturelle et de l'amortissement pour le canal tangage également.

Pour ACAH, le critère de la bande passante présenté au chapitre 3 et tiré d'ADS-33 précise que la bande passante à prendre en compte est  $w_{BW_{phase}}$  (marge de phase). Pour les modèles idéaux décrits par les équations (5.23) à (5.26), la bande passante est calculée à l'aide de l'équation (3.1) et qui est reprise ci-dessous :

$$w_{BW} = w_n (\xi + \sqrt{\xi^2 + 1}) \quad (5.27)$$

Pour les axes tangage et roulis, on obtient  $w_{BW} = 5.47 \text{ rad/s}$ . Par ailleurs, le diagramme de phase des fonctions de transfert (5.25) et (5.26) après avoir remplacé  $w_{n\phi, \theta}$  et  $\xi$  par leur

valeurs (voir figure 5.10) montre que le paramètre  $\tau_p$  (équation 5.28) est nul pour l'axe tangage et roulis, ce qui assure un niveau 1 des HQ.

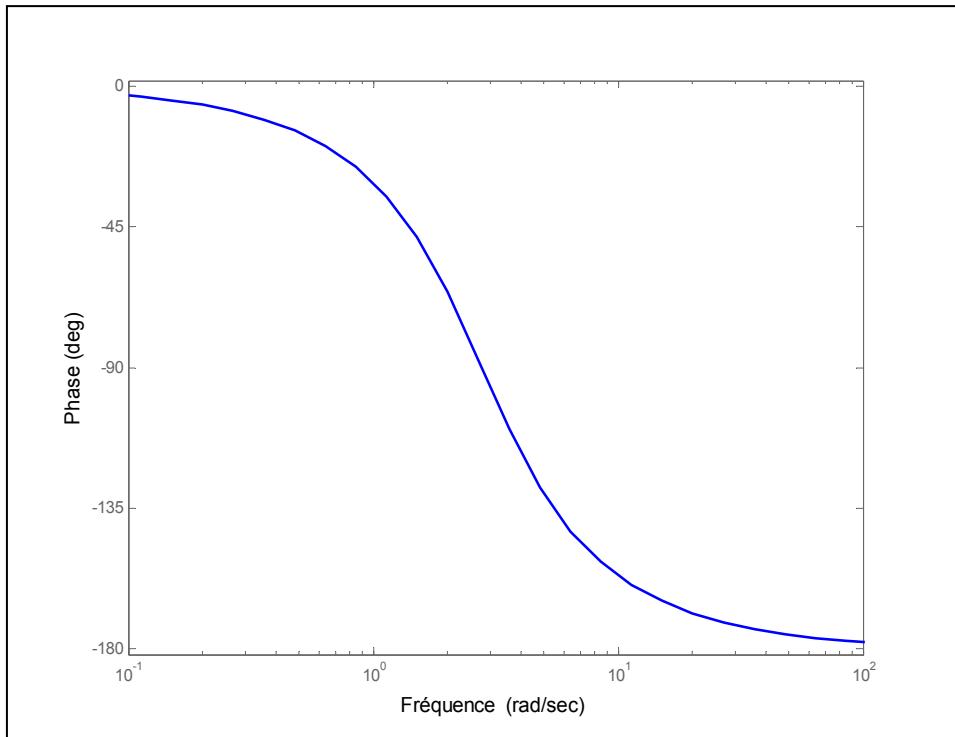


Figure 5.10 Diagramme de phase (axes tangage et roulis)

L'équation (3.2) est reprise ci-dessous :

$$\tau_p = \frac{\Delta\Phi_{2w_{180}}}{57.3(2w_{180})} \quad (5.28)$$

Maintenant qu'on a choisi  $K_{\phi,\theta} = 2$  en se basant sur les équations (5.3) et (5.5) des fonctions de transfert idéales, nous allons vérifier le respect du critère de la bande passante pour les boucles externes (tangage et roulis) en considérant les réponses réelles du système. Les diagrammes de phase de  $\frac{\phi}{\phi_c}$  ( $\frac{\theta}{\theta_c}$ ), en considérant les fonctions de transfert entre  $p_c$  et  $p$  ( $q_c$  et  $q$ ) obtenues après fermeture de la boucle interne, sont identiques à celui de la figure

5.10 et le niveau 1 des HQ est satisfait avec une bande passante de 5.47 rad/s pour les deux canaux.

Les figures 5.11et 5.12 illustrent la réponse du système à une entrée échelon de 5 deg/s sur la consigne de la vitesse de roulis (boucle interne fermée) et à une entrée échelon de 5 degrés sur la consigne de l'angle de roulis (boucle interne et externe fermées), les deux réponses ont été tracées sur les mêmes graphiques pour des fins de comparaison. On constate que la boucle interne permet la poursuite de vitesse (système de premier ordre) alors que la boucle externe permet la poursuite d'attitude (système de deuxième ordre). En outre, on constate que le découplage des autres modes a été amélioré après fermeture de la boucle externe. La vitesse de lacet converge vers une faible valeur comme a été prédit dans le paragraphe 5.1.

Pour une consigne de l'angle de roulis de 20 degrés, le ratio  $\frac{P_{pk}}{\Delta\phi_{pk}}$  est de 1.23 ce qui satisfait

le niveau 1 HQ du critère *attitude quickness*. Aussi, le ratio  $\frac{q_{pk}}{\Delta\theta_{pk}}$  , pour une consigne de

l'angle de tangage de 5 degrés, est égale à 1.21 ce qui satisfait le niveau 1 HQ également (voir la section 3.3.2 du chapitre 3).

En outre, les figures 5.11 et 5.12 montrent que le ratio  $\frac{\Delta\theta_{pk}}{\Delta\phi_4}$  est égal à -0.0022, valeur

inclus entre -0.25 et 0.25 ce qui assure le niveau 1 HQ en matière de couplage roulis-

tangage. De la même manière  $\frac{\Delta\phi_{pk}}{\Delta\theta_4}$  est calculé égale à 0.006 et le niveau 1 HQ est satisfait

pour le couplage tangage-roulis (voir la section 3.3.4 du chapitre 3).

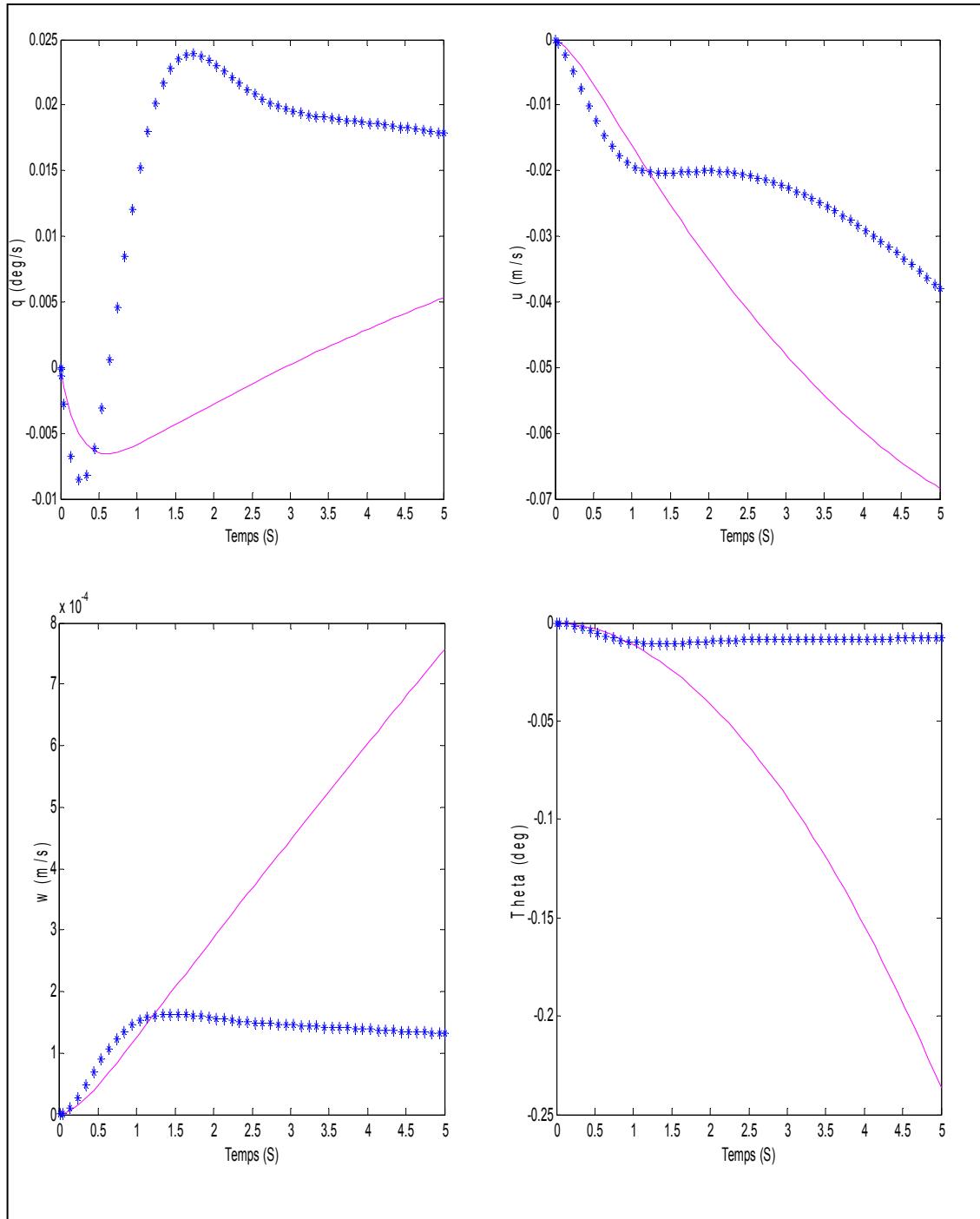


Figure 5.11 Réponse du système ( $q$ ,  $u$ ,  $w$  et  $\theta$ ), en BIF, à une entrée échelon ( $p_c = 5$  deg/s) et à une entrée échelon ( $\phi_c = 5$  deg) avec BEF

— Boucle interne fermée (BIF)  
 ----- Boucle externe fermée (BEF)

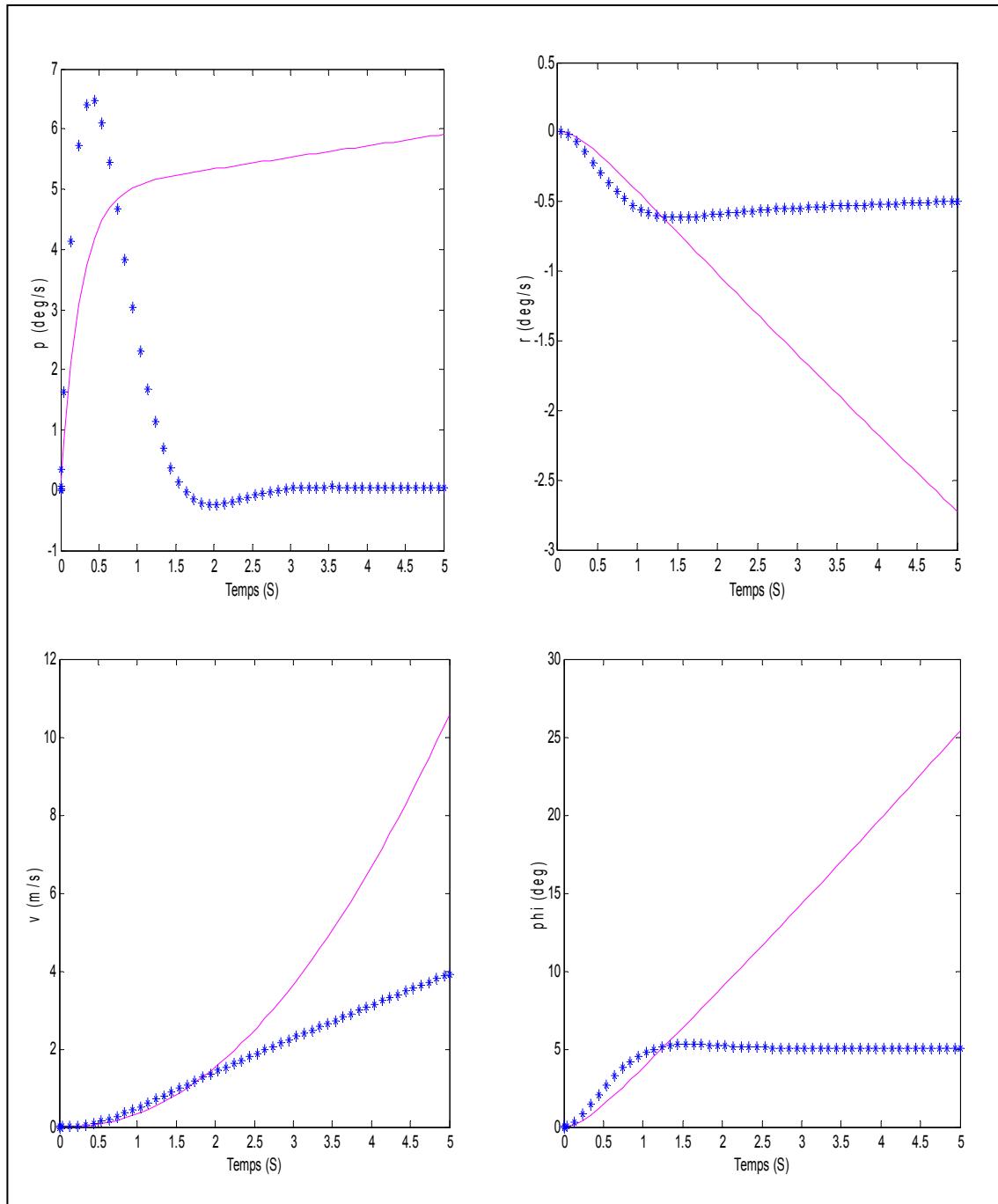


Figure 5.12 Réponse du système ( $p$ ,  $r$ ,  $v$  et  $\phi$ ), en BIF, à une entrée échelon ( $p_c = 5$  deg/s) et à une entrée échelon ( $\phi_c = 5$  deg) avec BEF

— Boucle interne fermée (BIF)  
\*\*\*\*\* Boucle externe fermée (BEF)

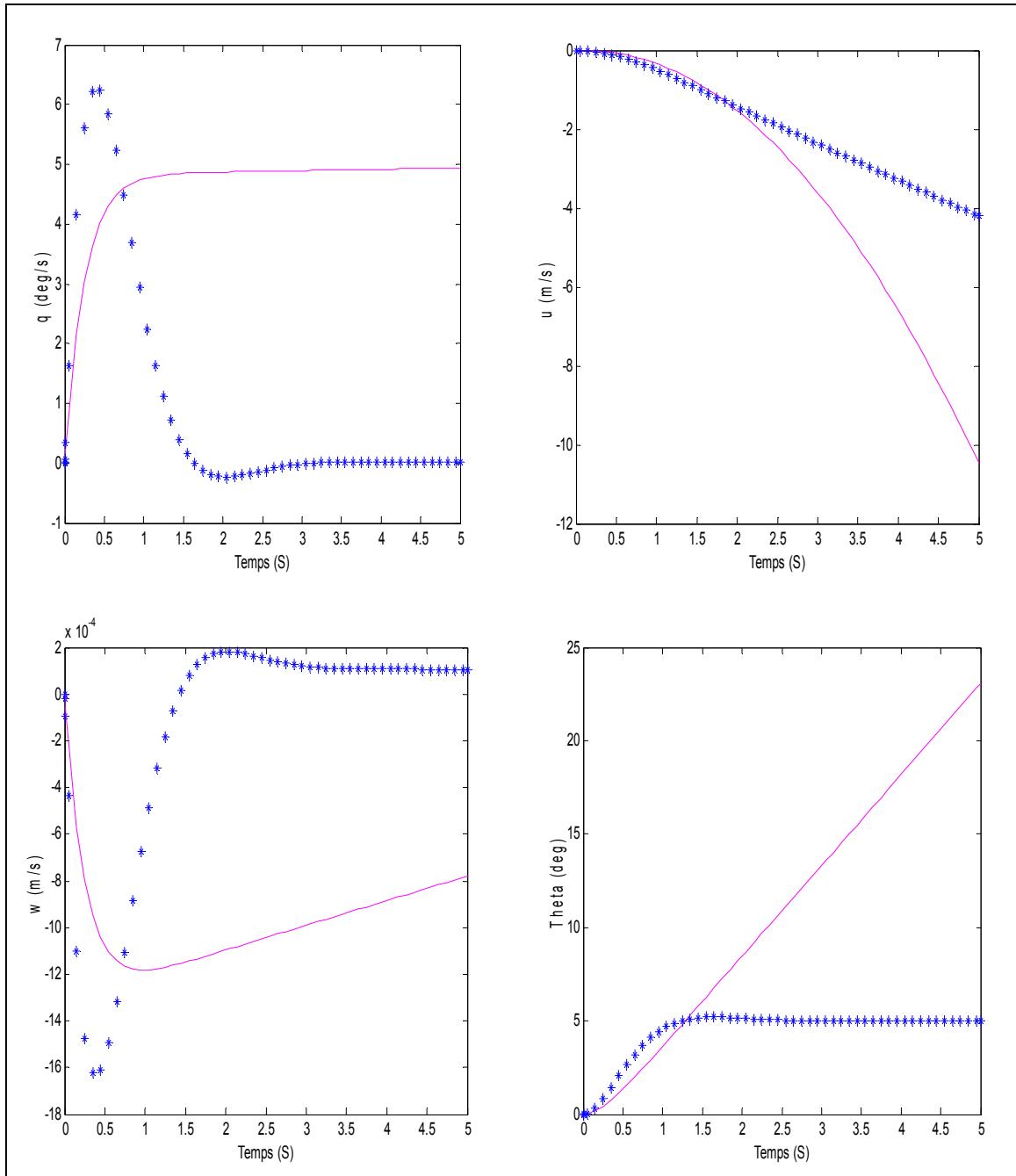


Figure 5.13 Réponse du système ( $q$ ,  $u$ ,  $w$  et  $\theta$ ), en BIF, à une entrée échelon ( $q_c = 5$  deg/s) et à une entrée échelon ( $\theta_c = 5$  deg) avec BEF

— Boucle interne fermée (BIF)  
 \*\*\*\*\* Boucle externe fermée (BEF)

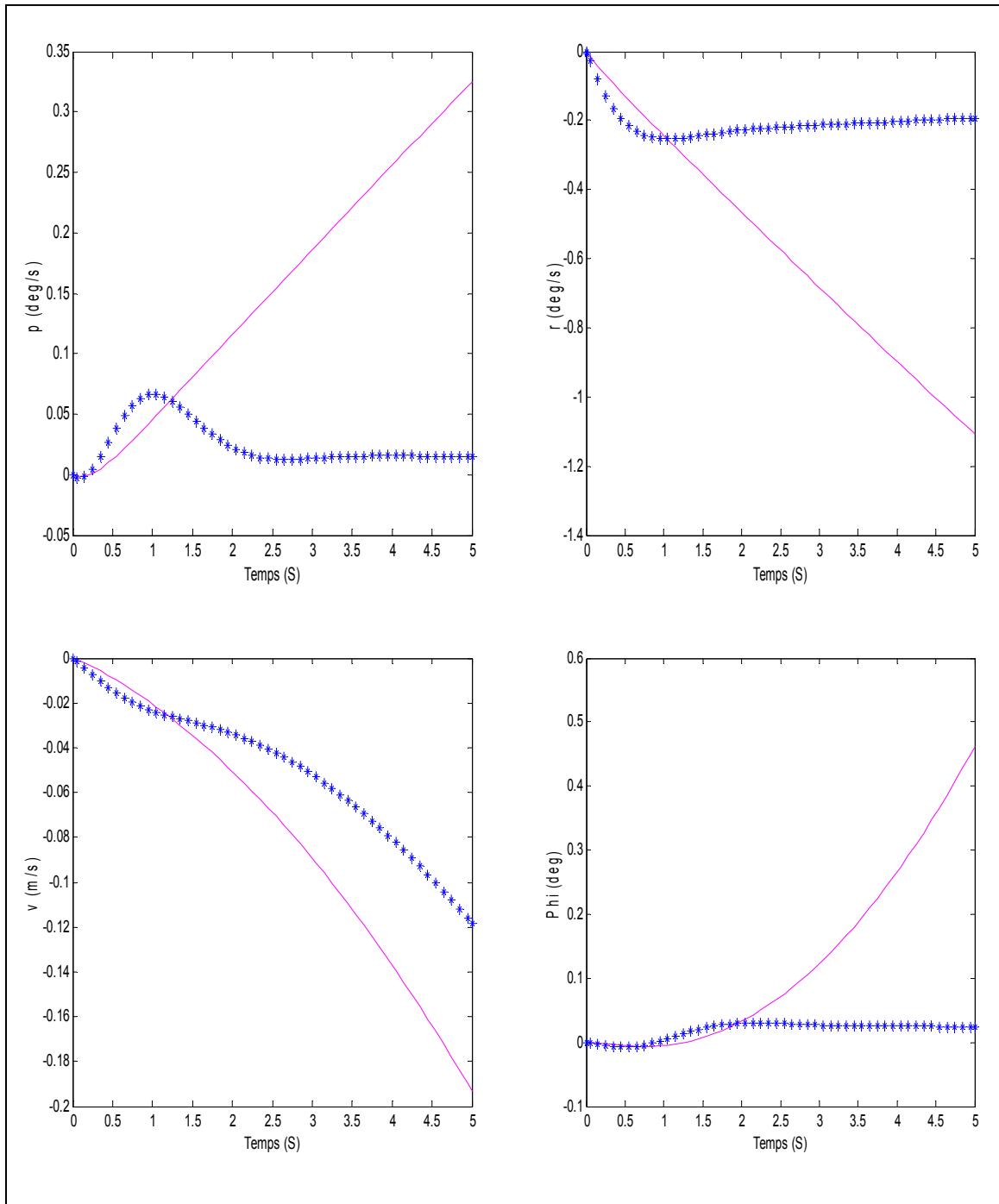


Figure 5.14 Réponse du système ( $p$ ,  $r$ ,  $v$  et  $\phi$ ), en BIF, à une entrée échelon ( $q_c = 5 \text{ deg/s}$ ) et à une entrée échelon ( $\theta_c = 5 \text{ deg}$ ) avec BEF

— Boucle interne fermée (BIF)  
 \*\*\*\*\* Boucle externe fermée (BEF)

Les réponses du système aux entrées de l'axe longitudinal ont été reportées sur les figures 5.13 et 5.14. Comme pour l'axe latéral, on remarque que le découplage a été amélioré après fermeture de la boucle externe. La boucle interne produit une poursuite de vitesse et la boucle externe conduit à la poursuite de l'angle de tangage. On remarque que la vitesse de tangage enregistre un pic, lors de la poursuite d'attitude, et revient ensuite à une valeur nulle après stabilisation de l'angle de tangage sur sa consigne, le même phénomène est constaté pour l'axe latéral.

L'évolution de la vitesse linéaire latérale  $v$  en fonction de la consigne  $\phi_c$  est approximativement celle de la fonction de transfert désirée (5.29), reliant ces deux paramètres et déduite des équations (5.5), (5.8) et (5.23). De la même manière, l'équation (5.30) donne  $u$  en fonction de  $\theta_c$  :

$$\frac{v(s)}{\phi_c(s)} = \frac{k_\phi \lambda_p (s + \lambda_p)}{(s + \lambda_v)(s^2 + \lambda_p s + k_\phi \lambda_p)} \quad (5.29)$$

$$\frac{u(s)}{\theta_c(s)} = \frac{k_\theta \lambda_q (s + \lambda_q)}{(s + \lambda_u)(s^2 + \lambda_q s + k_\theta \lambda_q)} \quad (5.30)$$

Par ailleurs, Les figures 5.15 et 5.16 illustrent l'évolution des états du système après application d'une entrée échelon de 5 deg/s sur la consigne de la vitesse de lacet, et les figures 5.17 et 5.18 montrent les réponses à une entrée échelon de 2 m/s sur la consigne de la vitesse verticale.

Les figures 5.15 et 5.16 montrent que la réponse de la vitesse de lacet est de premier ordre comme prédit par les fonctions de transfert désirées, la vitesse verticale a été bien découplée et les modes longitudinal et latéral également, quoiqu'une légère compensation du pilote sur ces deux axes permettrait d'améliorer la stabilité de l'hélicoptère.

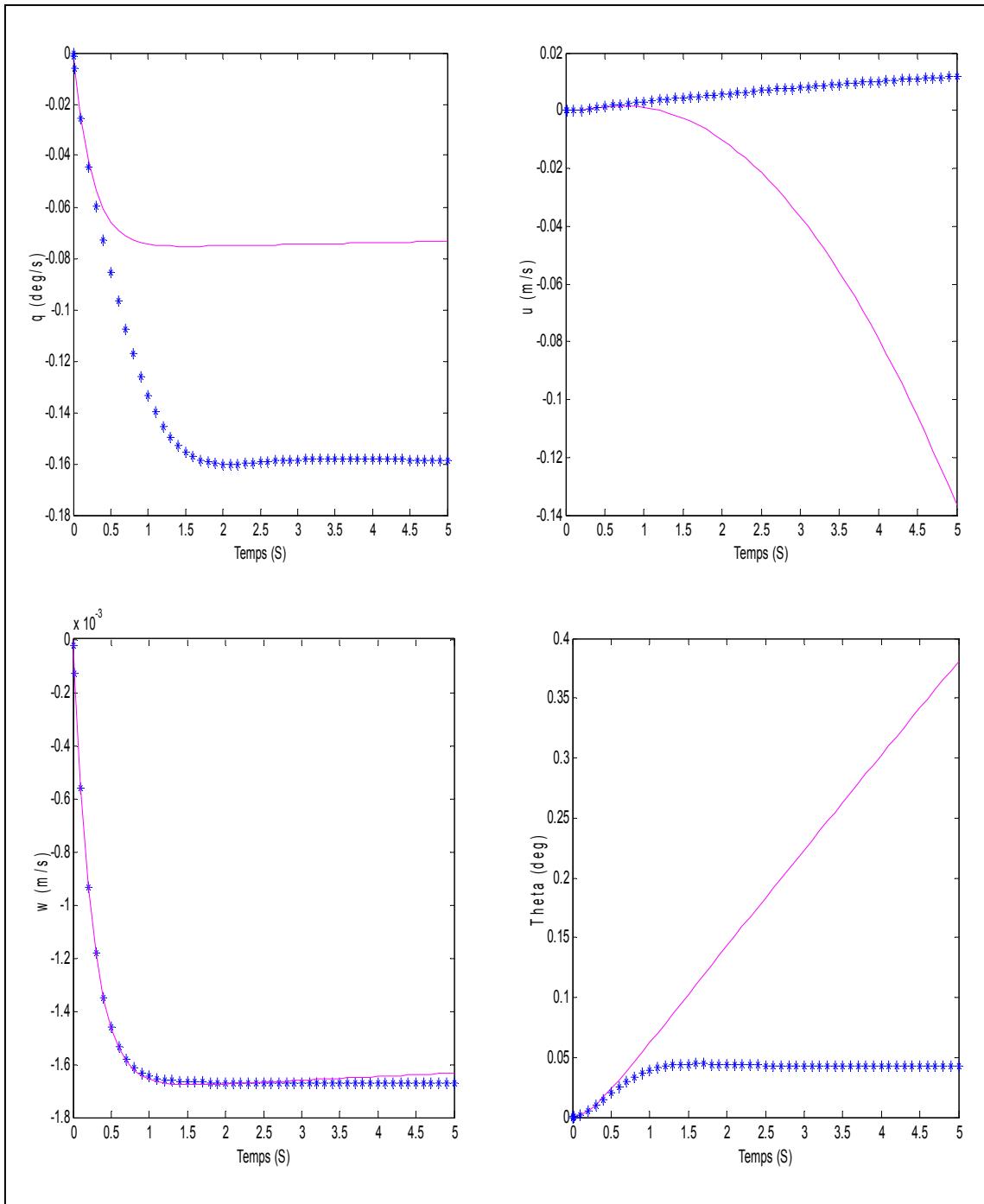


Figure 5.15 Réponse du système ( $q$ ,  $u$ ,  $w$  et  $\theta$ ), boucle interne fermée et boucle externe fermée, à une entrée échelon ( $r_c = 5$  deg/s)

— Boucle interne fermée  
\*\*\*\*\* Boucle externe fermée

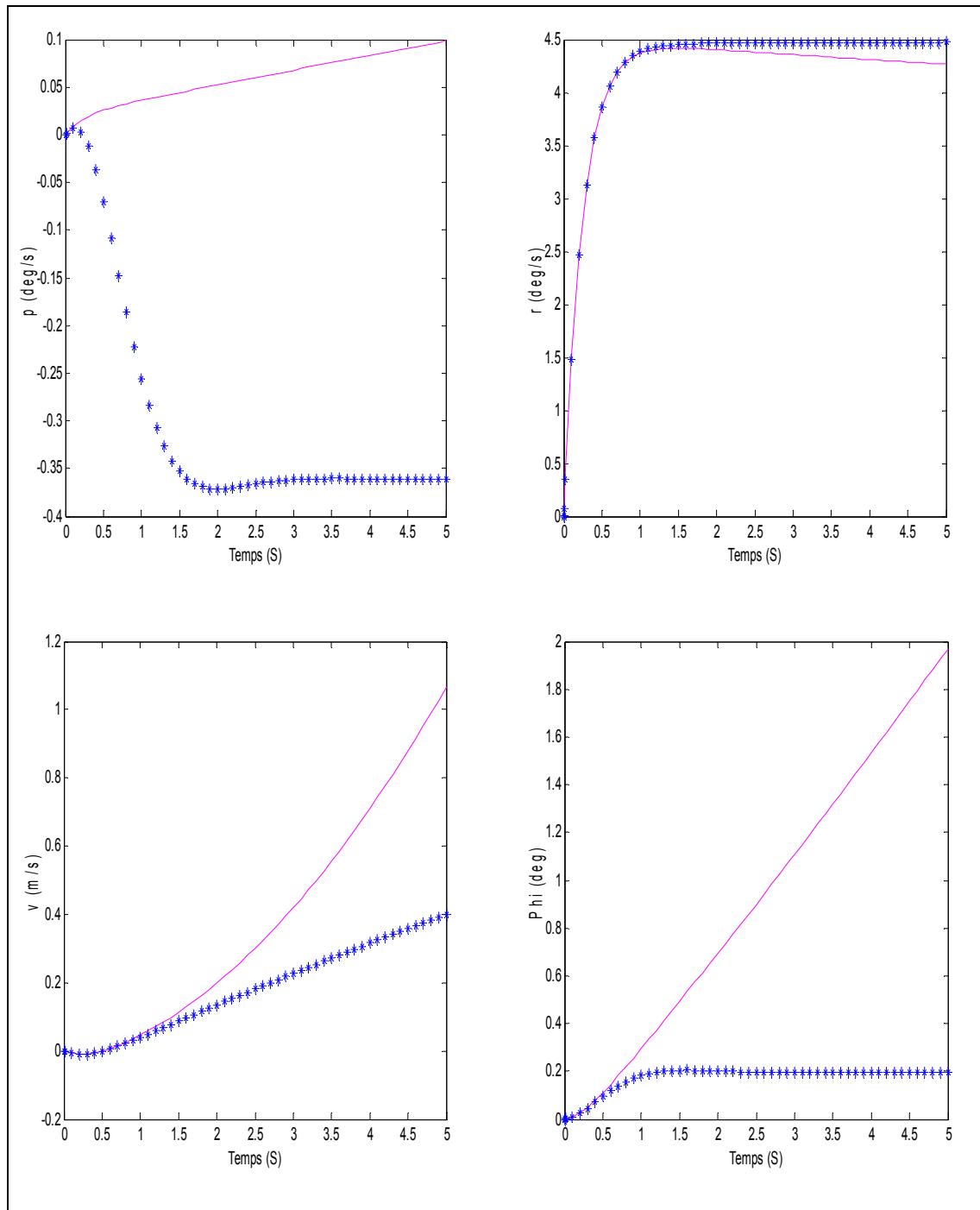


Figure 5.16 Réponse du système ( $p$ ,  $r$ ,  $v$  et  $\phi$ ), boucle interne fermée et boucle externe fermée, à une entrée échelon ( $r_c = 5$  deg/s)

— Boucle interne fermée  
 \*\*\*\*\* Boucle externe fermée

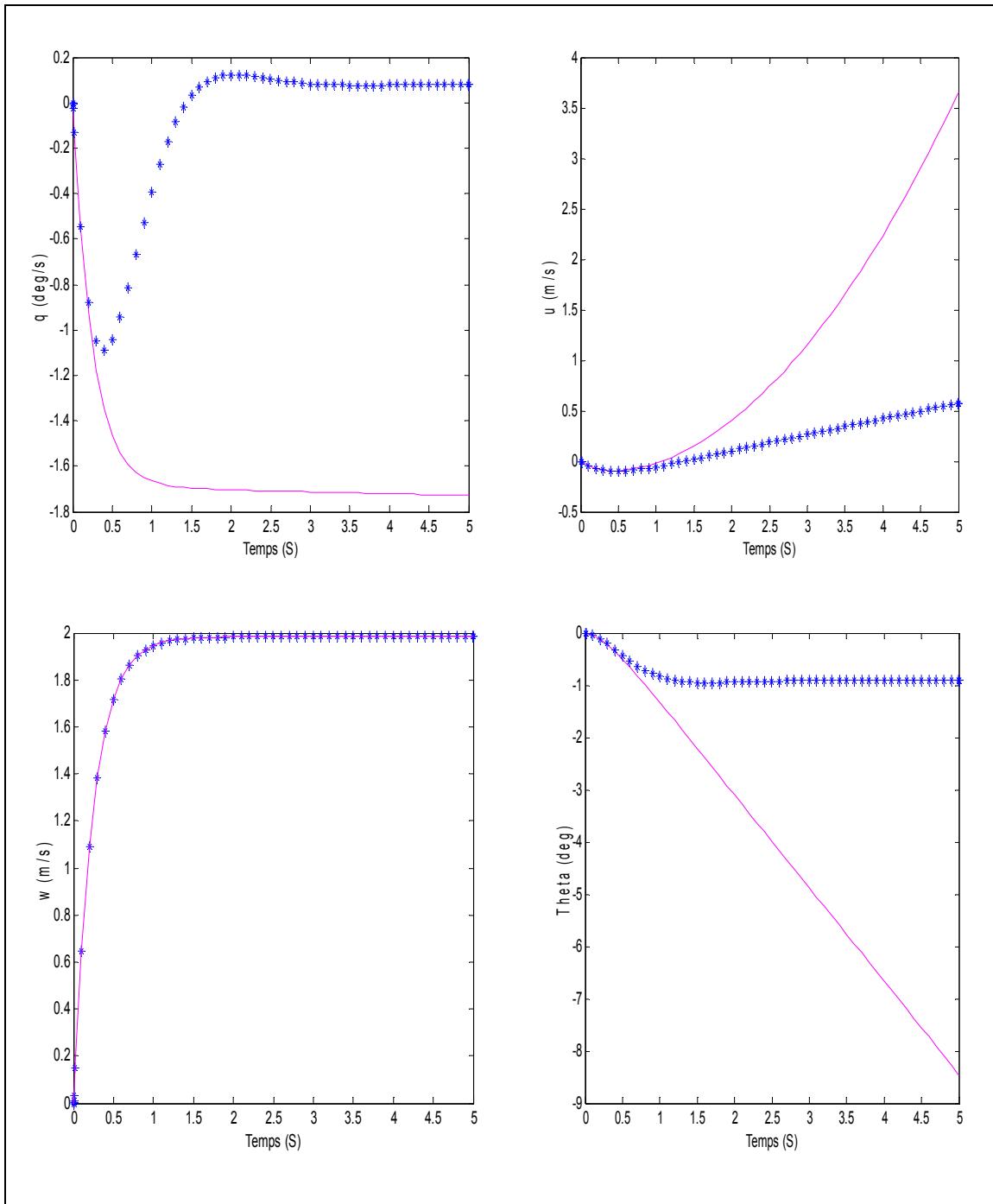


Figure 5.17 Réponse du système ( $q$ ,  $u$ ,  $w$  et  $\theta$ ) , boucle interne fermée et boucle externe fermée, à une entrée échelon ( $w_c = 2$  m/s)

— Boucle interne fermée  
 \*\*\*\*\* Boucle externe fermé

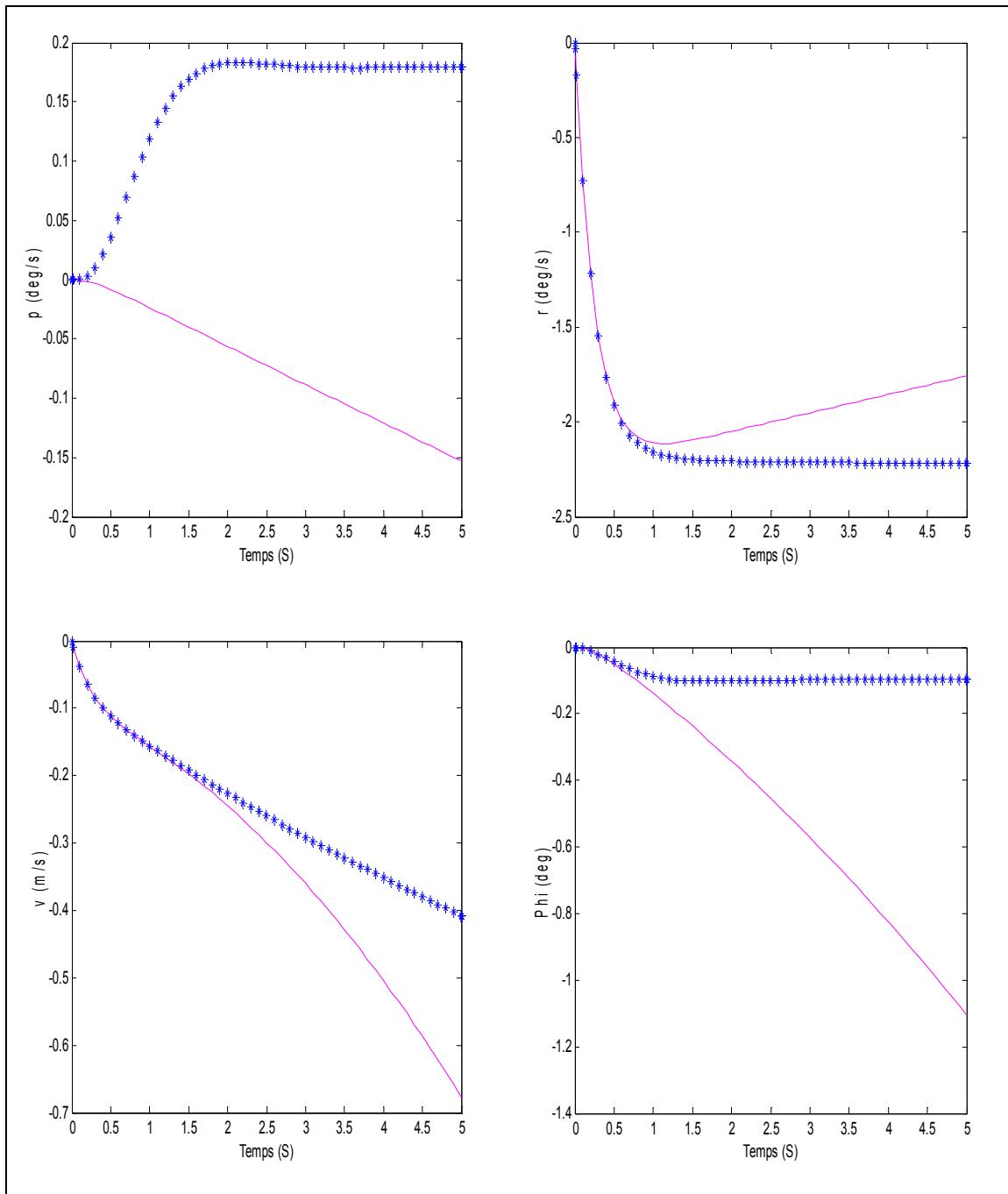


Figure 5.18 Réponse du système ( $p$ ,  $r$ ,  $v$  et  $\phi$ ), boucle interne fermée et boucle externe fermée, à une entrée échelon ( $w_c = 2$  m/s)

— Boucle interne fermée  
 \*\*\*\*\* Boucle externe fermée

Comme pour l'axe latéral et longitudinal, on remarque une amélioration du découplage après fermeture de la boucle externe. En outre, les figures 5.15 et 5.16 montrent que la réponse de la vitesse verticale est de premier ordre. Par ailleurs, le découplage des axes longitudinal et latéral du mode vertical est illustré par les faibles valeurs des états concernés, par contre, on constate que le mode directionnel a été influencé par l'application d'une entrée échelon sur le collectif. L'explication de ce phénomène vient du fait que le changement effectué sur le collectif a créé un couple (rotor principal), ce qui a provoqué un moment sur l'axe de lacet. Le contrôleur a produit une compensation sur la commande du rotor de la queue pour maintenir l'équilibre de l'appareil, ce qui a conduit à une vitesse latérale non négligeable.

Cette dynamique reliant l'axe vertical à l'axe de lacet justifie l'importance accordée à la spécification concernant le couplage entre ces deux axes. Comme a été expliqué au chapitre

3, les rapports  $\left| \frac{r_1}{h(3)} \right|$  et  $\left| \frac{r_3}{h(3)} \right|$  doivent être calculés. À partir des figures 5.17 et 5.18, on

trouve  $r_1 = -2.2 \text{ deg/s}$ ,  $r_3 = 0.1 \text{ deg/s}$   $\left| \frac{r_1}{h(3)} \right| = 2 \text{ m/s}$ . On a alors  $\left| \frac{r_1}{h(3)} \right| = 0.33$   $\left| \frac{r_3}{h(3)} \right| = 0.015$ .

Par ailleurs  $\left| \frac{r_1}{h(3)} \right|$  doit être inférieur à 0.65 et  $\left| \frac{r_3}{h(3)} \right|$  doit être inclus entre -0.15 et 0.2 pour

avoir le niveau 1 HQ. Ces conditions sont donc satisfaites.

### 5.3 Conclusion

Les résultats obtenus montrent que le contrôleur a été en mesure d'amener le système, en boucle fermée, à respecter les spécifications stipulées par l'ADS-33 (critère de la bande passante et du délai de phase, *attitude quickness* et couplage entre axes) ce qui était notre objectif de départ. En outre, la fermeture de la boucle interne a produit quatre canaux découplés de premier ordre, ce qui correspond aux fonctions de transfert désirées pour les

types de réponses *Rate Command*, tandis que les fonctions de transfert des deux boucles externes (tangage et roulis) présentent une allure de second ordre (comme prévu par les fonctions de transfert désirées pour ACAH).



## CHAPITRE 6

### IMPLÉMÉTATION DES LOIS DE CONTRÔLE SUR HELISIM

Le logiciel HELISIM est un outil professionnel qui intègre la dynamique complète de différents hélicoptères (BELL-412, AH-1, UH-60 etc.). En outre, il permet la linéarisation de la dynamique de l'hélicoptère choisi pour une condition de vol à spécifier par le concepteur grâce à la fonction *HsimLinearization*, ce qui donne l'accès aux matrices du modèle linéaire.

Après avoir conçu la loi de commande à appliquer à l'hélicoptère dans l'environnement MATLAB SIMULINK, le concepteur peut l'implémenter sur HELISIM à l'aide du logiciel de développement Microsoft Visual Studio. Finalement, des simulations de vol peuvent être effectuées sur HELISIM pour analyser le comportement de l'hélicoptère.

Nous allons maintenant montrer les étapes générales de l'implémentation de notre loi de contrôle (boucle externe fermée pour le canal tangage et roulis).

#### 6.1 Environnement d'implémentation et de simulation

L'hélicoptère faisant l'objet de notre étude est le BELL-412. Comme a été expliqué dans le chapitre 5, les matrices  $K$  et  $H$  sont calculées à partir des matrices  $A$  et  $B$  données par HELISIM pour une condition de vol *hover* (le code MATLAB est donné en annexe 1).

Le schéma SIMULINK correspondant à la loi de commande est illustré sur la figure 6.1.

Pour implémenter la loi de commande sur HELISIM, le schéma SIMULINK doit être modifié en remplaçant le modèle linéaire de la figure 6.1 par la dynamique de l'hélicoptère (intégrée sur HELISIM) et en précisant les entrées et sorties du système en accord avec le code Microsoft Visual Studio et HELISIM. La figure 6.2 montre le schéma SIMULINK modifié avec les différentes entrées sorties.

Les entrées du schéma 6.2 sont reprises dans le code du logiciel Microsoft Visual Studio comme illustré par la figure 6.3. Les sorties sont également prises en compte.

L'étape suivante est la génération d'un code utilisable par Microsoft Visual Studio à partir du schéma Simulink. Cette opération s'effectue à partir de la fenêtre Simulink en cliquant sur *Tools/Code Generation/Options* et ensuite en appuyant sur l'onglet *Build* de la fenêtre illustrée sur la figure 6.4.

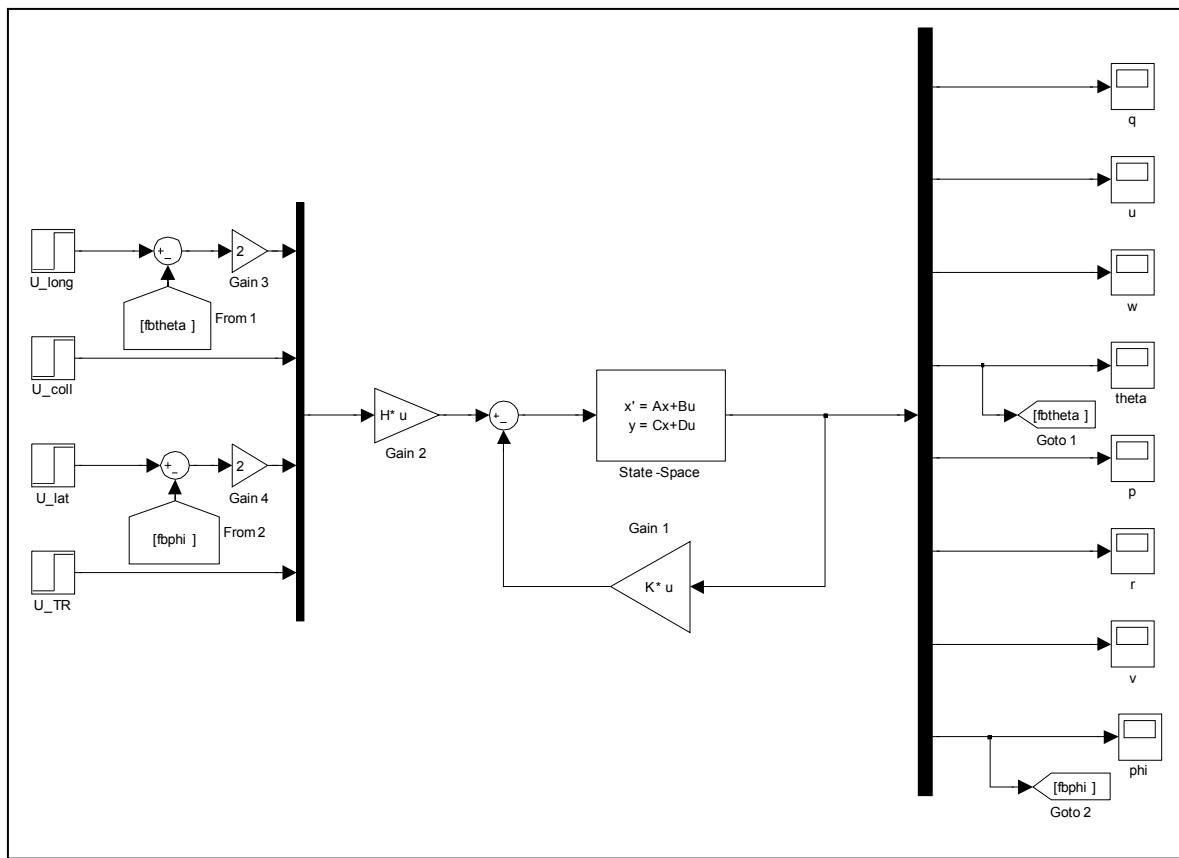


Figure 6.1 Schéma SIMULINK (boucle externe fermée tangage et roulis)

Le code Microsoft Visual Studio du projet est donné en annexe 2. Ce code est par la suite compilé à partir de l'onglet *Build/Build Solution*. Ce qui procure à HELISIM un code contenant tous les détails sur la loi de contrôle à implémenter. Il est à noter que les opérations

de génération de code doivent être effectuées an ayant les trois logiciels ouverts en même temps.

HELISIM contient deux volets : *Modeler* et *Run-Time*.

L'onglet *Modeler* permet d'implémenter les lois de commande. Sur la figure 6.5, la surface de contrôle (mr\_a1\_primary) correspondante à l'axe latéral de l'hélicoptère est activée, ensuite le choix de la loi de contrôle est effectué dans la zone *Simulation Model*. Dans ce cas de figure, cette loi est contenue dans le fichier « SurfLateral (Surf\_contr\_surf\_) » issu du code généré à partir de SIMULINK et Microsoft Visual Studio pour cet axe. De la même façon, les lois de commande des autres axes peuvent être implémentées. L'axe longitudinal correspond à la surface de contrôle (mr\_b1\_primary), l'axe de lacet correspond à (tr\_primary) et l'axe vertical correspond à (mr\_collective).

Après avoir spécifié la loi de commande que HELISIM prendra en compte pour chaque axe, l'onglet *Run-Time* permet le lancement de la simulation. En outre, HELISIM donne accès à l'évolution des états du système. Ainsi, différents tracés de courbes peuvent être réalisés. Par ailleurs, une fenêtre (HLSIM SCENE) est consacrée à la simulation de vol de l'hélicoptère (voir figure 6.6), permettant ainsi de visualiser le comportement de l'appareil et d'analyser sa stabilité.

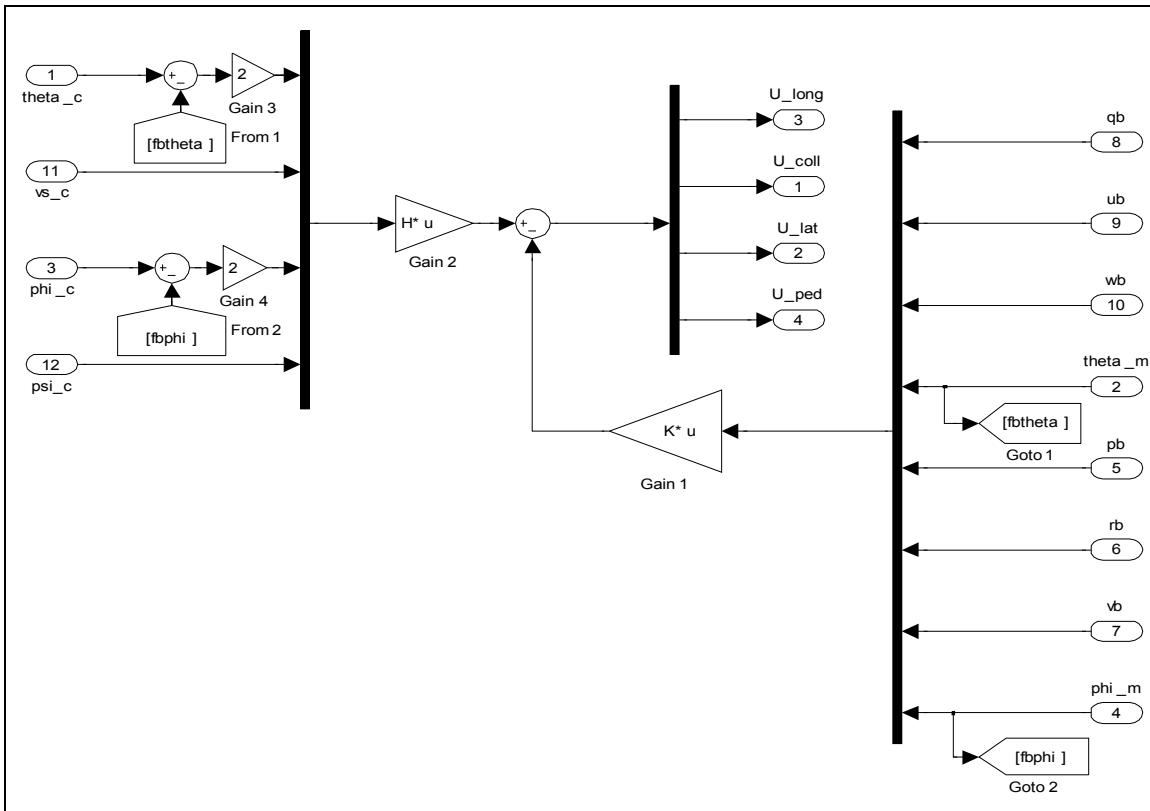


Figure 6.2 Schéma SIMULINK modifié pour implémentation

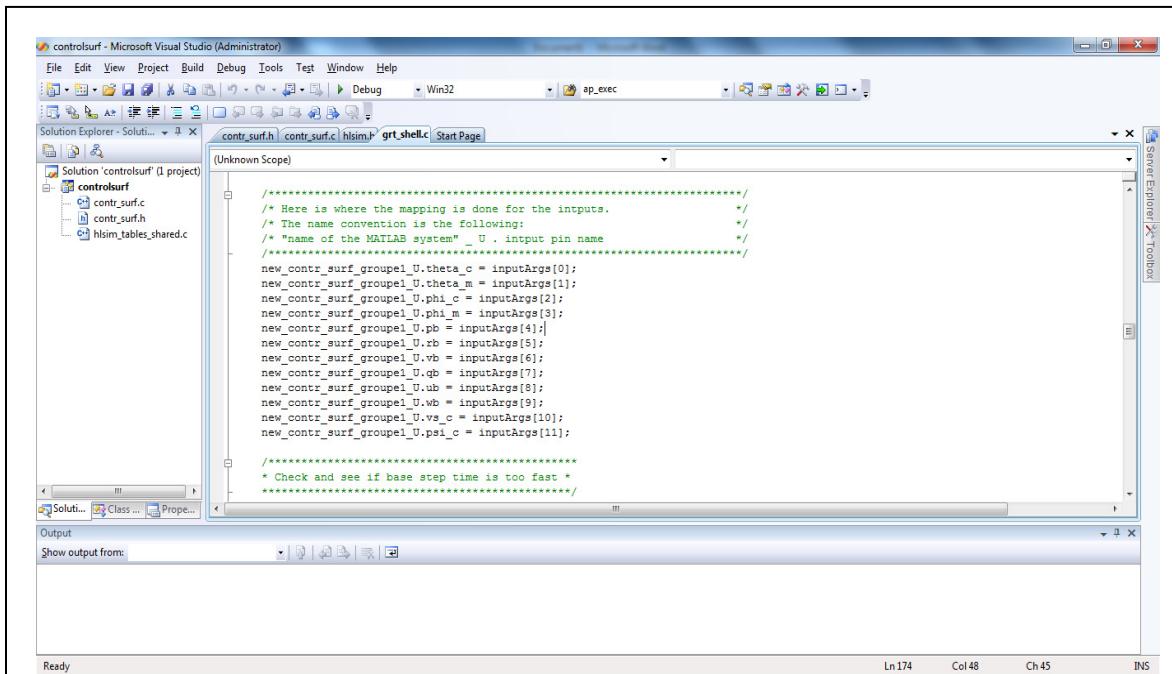


Figure 6.3 Correspondance entre les entrées de Simulink et ceux de Visual Studio

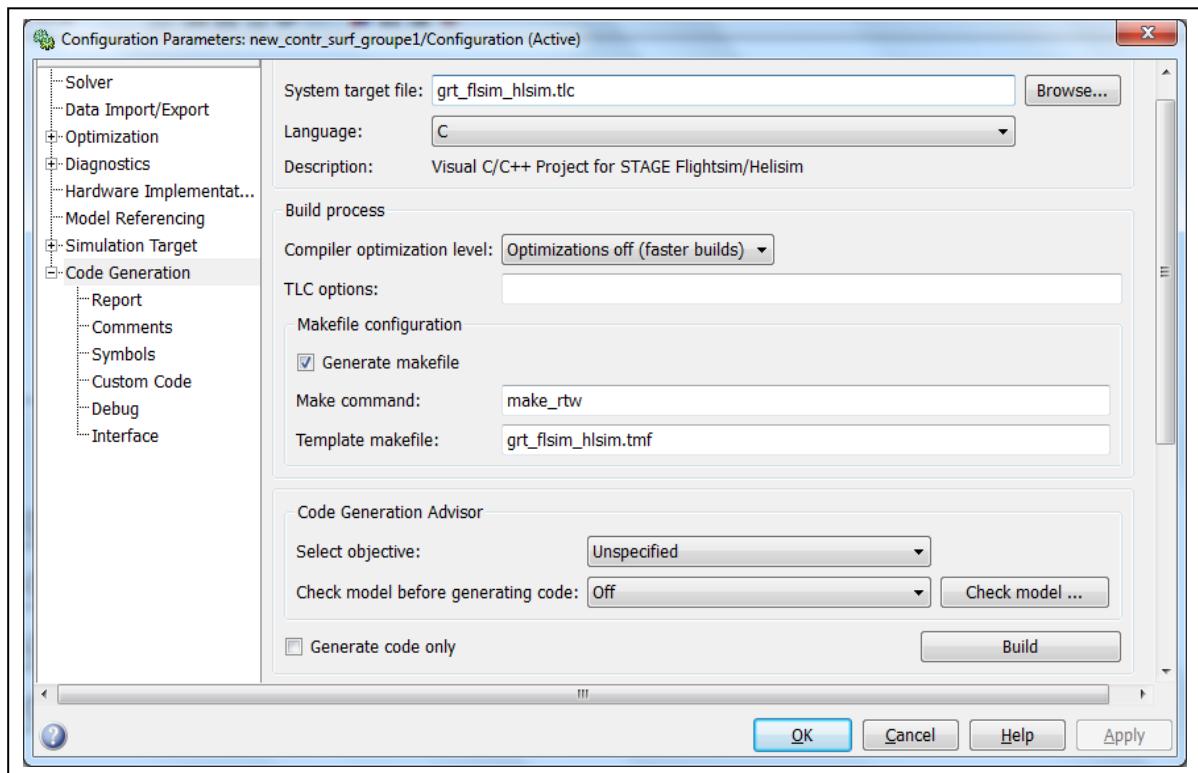


Figure 6.4 Génération de code à partir du Fichier Simulink

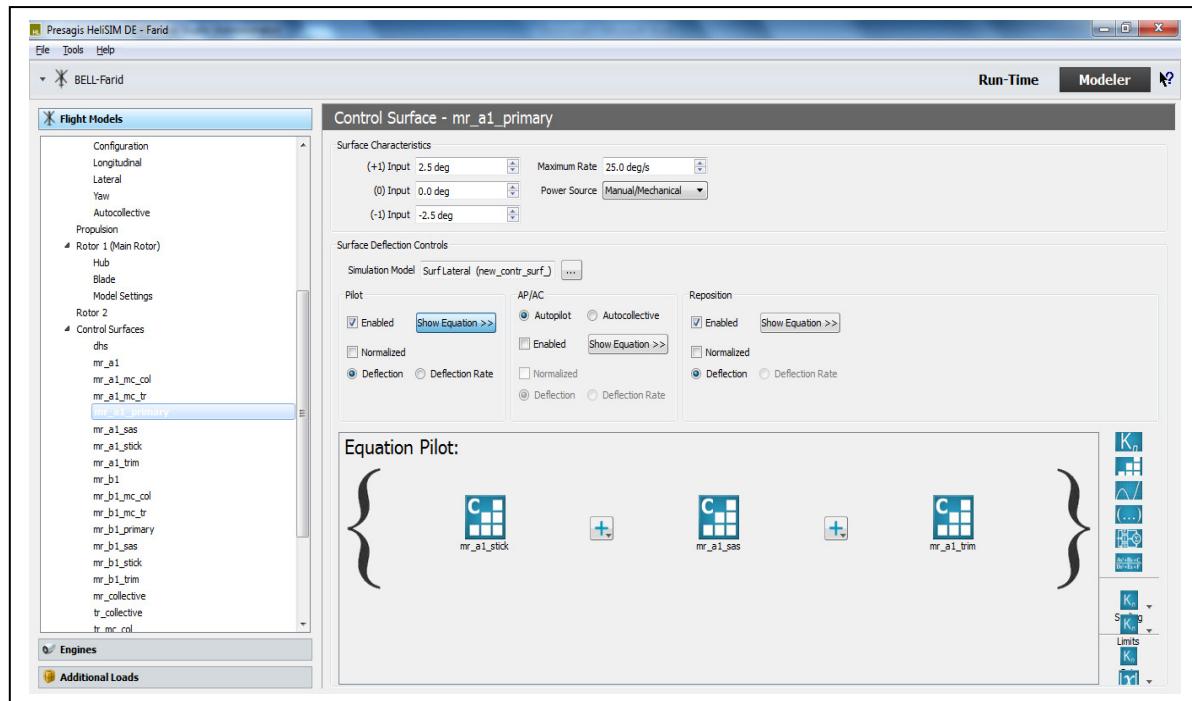


Figure 6.5 Implémentation de la loi de contrôle sur HELISIM (axe latéral)

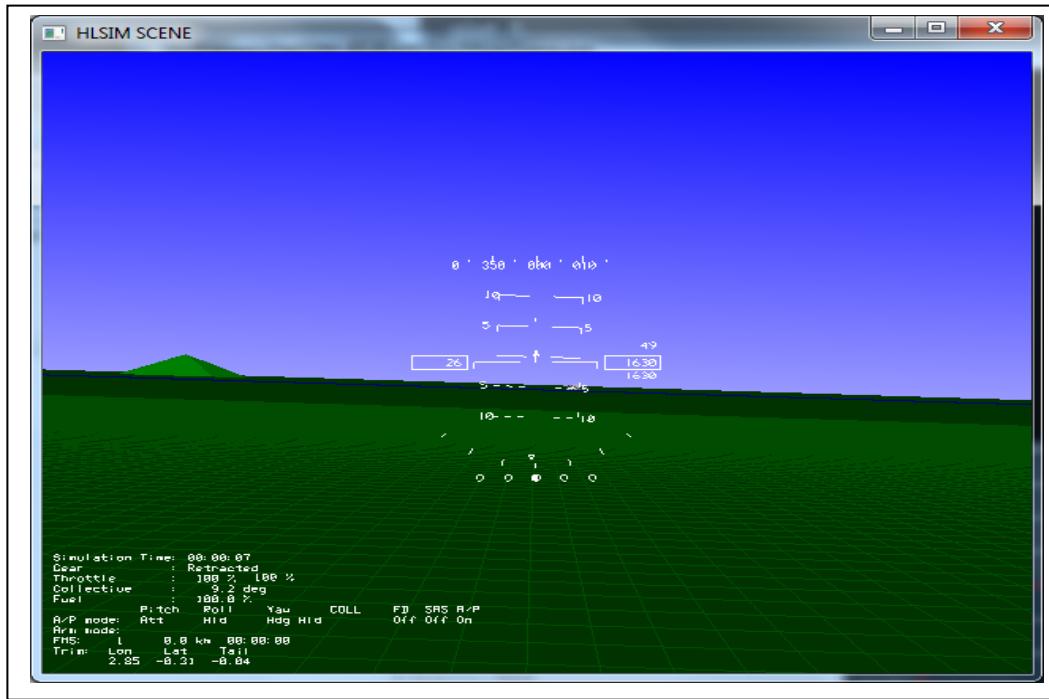


Figure 6.6 Fenêtre de simulation de vol HLSIM SCENE

## 6.2 Implémentation des lois de contrôle pour l'hélicoptère BELL-412

L'implémentation des lois de contrôle correspondantes aux quatre axes a conduit à un système instable. La figure 6.7 illustre l'évolution des angles  $\phi$  et  $\theta$  pour des consignes respectives  $\phi_c$  et  $\theta_c$  de 4.6 et -2 degrés. Le caractère instable est constaté. En effet, en l'absence d'un pilote, en mesure de compenser l'augmentation des vitesses linéaires  $u$  et  $v$  suite à l'application des consignes ( $\phi_c$  et  $\theta_c$ ) selon les équations (5.29) et (5.30), les conditions de vol de l'hélicoptère ont changé (la condition de vol *Hover* n'est plus respectée puisque les vitesses linéaires ne sont plus nulles). Ainsi, les lois de contrôle, calculées en fonction des matrices  $A$  et  $B$ , données par HELISIM pour la condition *Hover*, sont devenues obsolètes conduisant à l'instabilité du système.

Par ailleurs, sur la figure 6.8, on peut observer la position des actuateurs pour cette simulation de vol. On remarque que les actuateurs sont tout le temps en saturation. Les intervalles de fonctionnement pour les actuateurs du BELL-412 sont : (entre -2.5 et 2.5

degrés) pour l'axe latéral, (entre -8 et 8 degrés) pour l'axe longitudinal, (entre 0 et 10 degrés) pour l'axe vertical (collectif) et (entre -14.9 et 14.9 degrés) pour l'axe du lacet.

De retour à Matlab, les angles demandés aux actuateurs (efforts de contrôle requis) sont illustrés sur la figure 6.9. Cette simulation a été faite pour  $\phi_c$  et  $\theta_c$  de 4.6 et -2 degrés respectivement. On constate que les efforts de contrôle requis ont de grandes valeurs et ne restent pas dans les intervalles de fonctionnement normal des actuateurs.

En outre, on remarque que dès la première itération de la simulation, l'angle de déflexion demandé à l'actuateur de l'axe latéral est (-12 degrés) ce qui provoque sa saturation. L'actuateur de l'axe vertical se trouve également saturé puisqu'il est commandé par un angle négatif alors qu'il ne fonctionne qu'avec des angles positifs. Ces saturations conduisent à l'instabilité de l'hélicoptère et au changement de la condition de vol, ce qui rend, une fois encore, la loi de commande obsolète puisqu'elle a été conçue pour une condition de vol initiale donnée (des matrices  $A$  et  $B$  déterminées).



Figure 6.7 Évolution des angles  $\phi$  et  $\theta$  en fonction des consignes

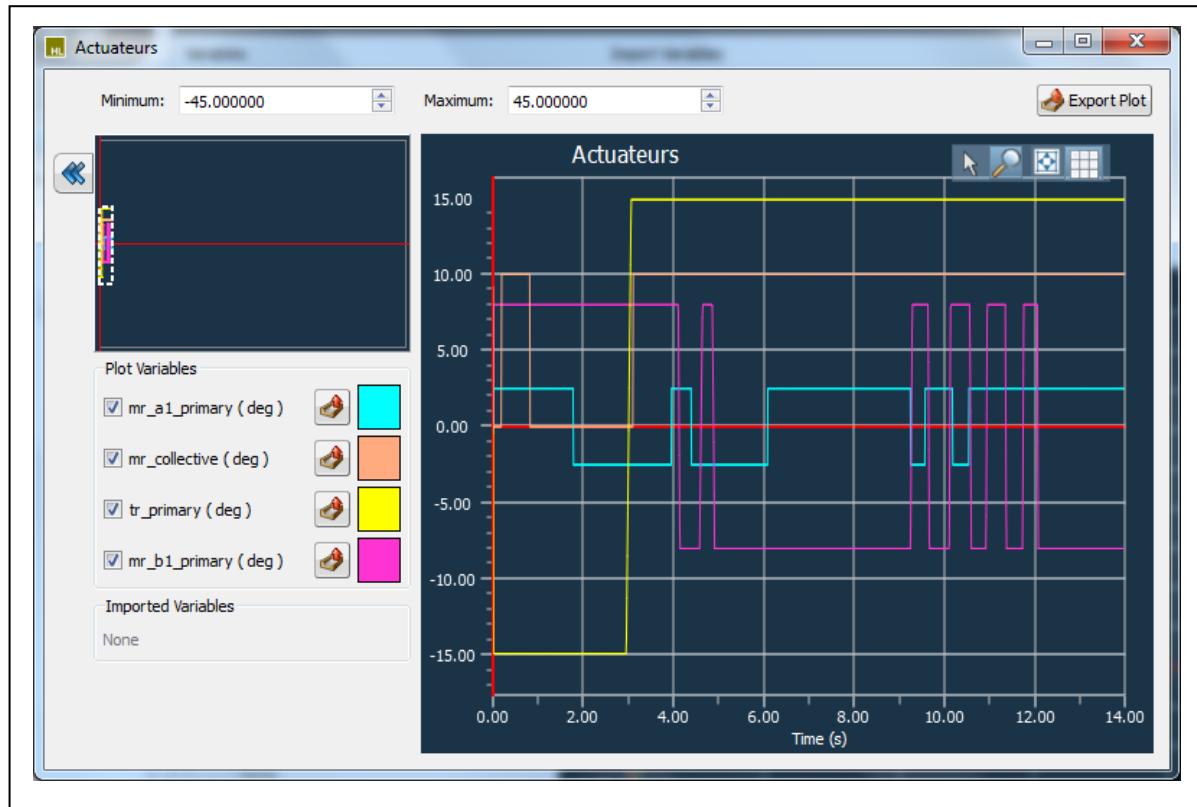


Figure 6.8 Évolution de la position des actuateurs

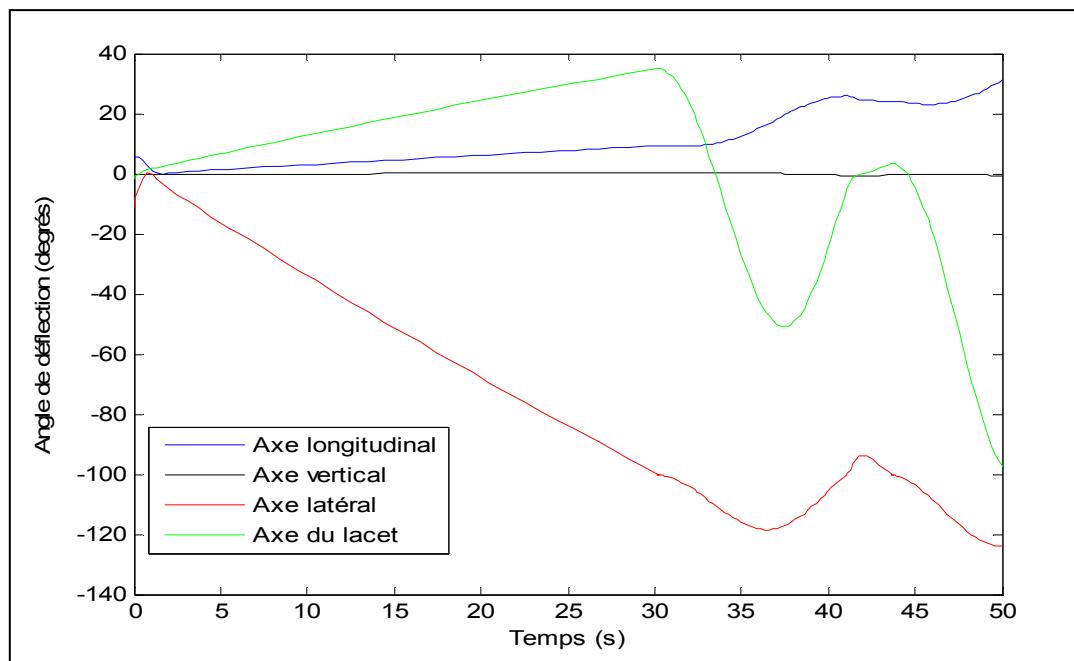


Figure 6.9 Angles de déflexion commandés aux actuateurs

En conclusion, on déduit que les deux causes de l'instabilité de l'hélicoptère sont :

- L'absence de compensation contre l'augmentation des vitesses linéaires  $u$  et  $v$  ;
- La saturation des actuateurs à cause de l'agressivité des lois de contrôle implémentées et qui envoient des commandes aux actuateurs au-delà de leur zone de fonctionnement normal, ce qui conduit à leur saturation.

La simulation sur Matlab a donné de bons résultats puisqu'aucune limitation n'a été imposée en ce qui concerne le fonctionnement des actuateurs et que la dynamique de l'hélicoptère n'a pas été modifiée (les matrices  $A$  et  $B$  ont été considérées constantes tout au long de la simulation).

### 6.3        **Implémentation de la loi de contrôle de l'axe latéral uniquement**

L'implémentation de la loi de commande de l'axe latéral uniquement, en gardant les lois de commande des autres axes par défaut (ceux d'un contrôleur, existant sur HELISIM, découpé par axe et à base de PID) a donné les résultats de la figure 6.10. Le contrôleur existant permet de maintenir les vitesses de translation nulles.

On constate que l'angle  $\phi$  converge vers une valeur différente de  $\phi_c$  (erreur en régime permanent). La figure 6.11 illustre l'évolution d'autres variables pertinentes (l'angle de lacet, la vitesse angulaire de lacet et la vitesse verticale). Par contre, on constate que le contrôleur conduit à une vitesse verticale de petite valeur (l'hélicoptère quoique stable gagne en altitude). Un ajustement des gains correspondants à l'axe latéral pourrait améliorer ces résultats tout en permettant d'annuler la vitesse verticale.

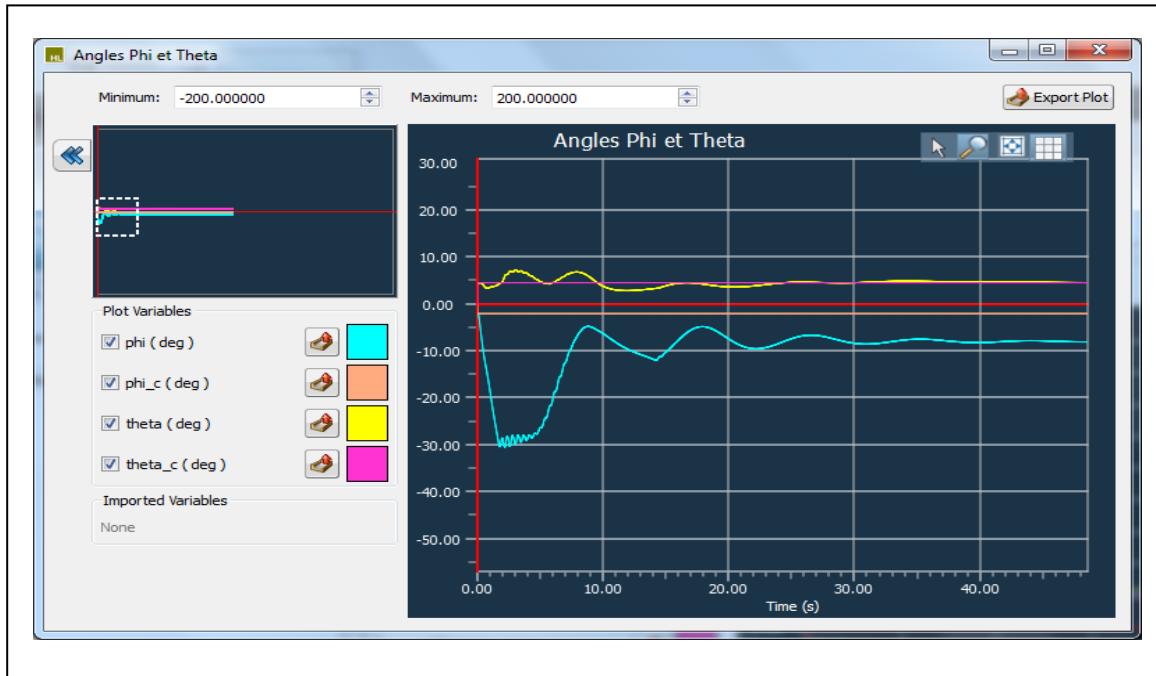


Figure 6.10 Évolution des angles  $\phi$  et  $\theta$  (implémentation de la loi de commande de l'axe latéral uniquement avec les autres axes pris par défaut)



Figure 6.11 Évolution d'autres variables (implémentation de la loi de commande de l'axe latéral uniquement avec les autres axes pris par défaut)

#### 6.4 Implémentation de la loi de contrôle de l'axe latéral et l'axe de lacet

En implémentant la loi de contrôle de l'axe lacet, en plus de celle de l'axe latéral, et en laissant les deux autres axes contrôlés par le contrôleur existant, on aboutit aux résultats des figures 6.12 et 6.13. On constate que l'erreur, en régime permanent entre  $\phi$  et  $\phi_c$  a diminué. Là encore, un ajustement des gains serait en mesure d'améliorer la convergence des angles ainsi que la stabilité de l'hélicoptère.

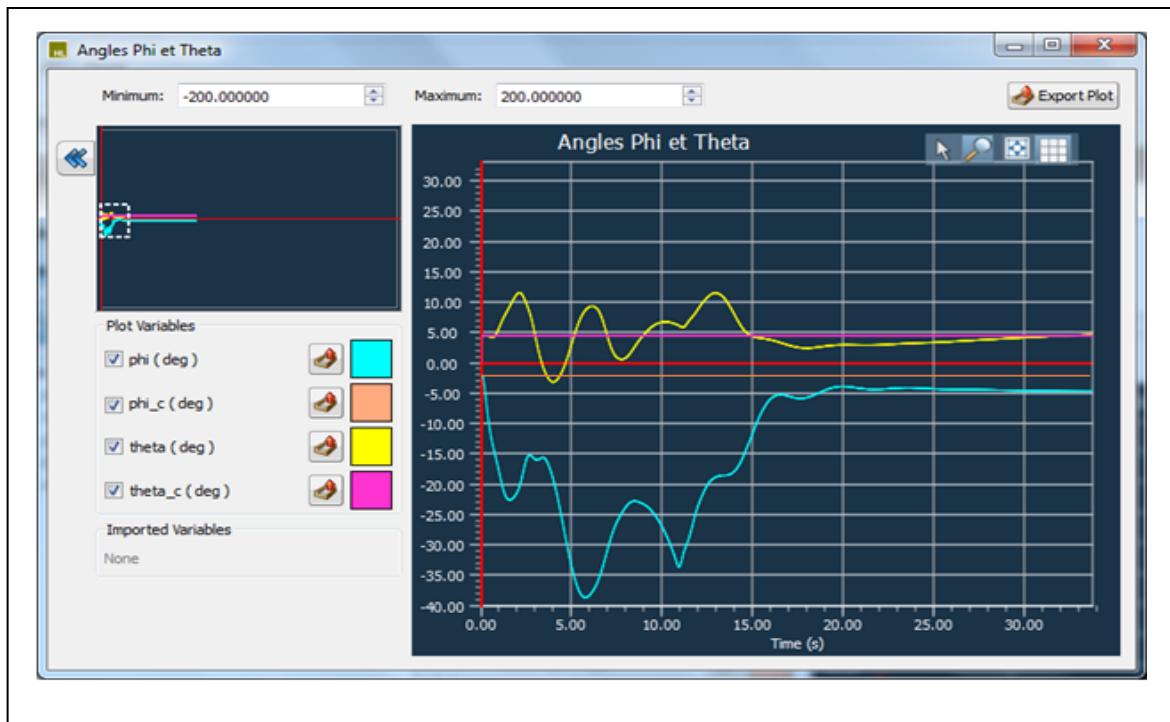


Figure 6.12 Évolution des angles  $\phi$  et  $\theta$  (implémentation des lois de commande de l'axe latéral et de lacet avec les autres axes pris par défaut)



Figure 6.13 Évolution d'autres variables (implémentation des lois de commande de l'axe latéral et de lacet avec les autres axes pris par défaut)

La convergence constatée lors de l'implémentation des lois de contrôle de l'axe latéral et de lacet en gardant les deux autres axes contrôlés par le contrôleur existant montrent que le système tend à se stabiliser, ce qui laisse présager que le contrôleur élaboré dans le chapitre 5 est capable de stabiliser l'hélicoptère si les problèmes de saturation des actuateurs et de maintien des vitesses linéaires nulles sont résolus.

## CONCLUSION

L'objectif principal de ce travail de recherche a été l'intégration des spécifications contenues dans l'ADS-33 (relatives aux qualités de vol requises) dans le processus de conception du système de contrôle pour hélicoptères. Le premier chapitre a dressé l'état de l'art des travaux en relation avec notre sujet, en particulier l'outil de conception et d'optimisation des lois de contrôle pour aéronefs (CONDUIT) a été présenté. Ce logiciel a été développé par l'armée américaine ; cet effort de développement illustre bien l'intérêt porté par les ingénieurs à l'échelle mondiale aux outils permettant d'intégrer les critères à respecter par les systèmes de contrôle dans la conception permettant de réduire les coûts. Par ailleurs, les travaux de William Garrard et d'Eicher low ont été mis en lumière. Ces travaux concernent l'utilisation des techniques d'affectation de la structure propre dans la conception des lois de commande pour hélicoptères.

Le deuxième chapitre a été dédié à la modélisation de la dynamique de l'hélicoptère, décrivant les forces et moments régissant le comportement de l'appareil en relation avec les différents organes (rotor principal, rotor de la queue, commandes de vol etc.). En outre, une méthode pour l'élaboration du modèle d'état a été présentée.

Dans le chapitre 3, une analyse des critères de performance a été faite pour la condition de vol *hover* et basses vitesses, ce qui nous a permis de déduire des fonctions de transfert désirées, de premier et de deuxième ordre, entre les entrées et sorties des différents canaux, dépendamment du type de réponses considéré.

Par la suite, les techniques d'affectation de la structure modale ont été introduites au chapitre 4, ces techniques ont été utilisées dans le chapitre 5 dans le calcul d'une matrice de retour d'état qui amène le système, en boucle fermée, à se comporter comme 4 canaux découplés et présentant des types de réponses de premier ordre (poursuite de vitesse), se rapprochant ainsi des fonctions de transfert désirées déduites des spécifications à respecter. Le calcul de la matrice de retour d'état commence par l'identification des valeurs et vecteurs propres désirés

issus d'un modèle idéal (sous forme d'équation d'état) généré à l'aide des fonctions de transfert désirées. Par ailleurs, une matrice de pré-compensation est également utilisée dans l'objectif d'éliminer les couplages entre les modes et les entrées. Le contrôleur proposé se compose de deux boucles, une boucle interne permettant la poursuite de vitesse et constituée à l'aide des deux matrices citées ci-haut et une boucle externe qui donne la possibilité de réaliser différents types de réponses (poursuite d'attitude en particulier).

Les simulations sur MATLAB, pour la dynamique de l'hélicoptère BELL-412, linéarisée pour une condition de vol *hover* (les matrices du modèle linéaire ont été calculées à l'aide de HELISIM), ont montré que le contrôleur élaboré à l'aide de cette méthode, permet au système en boucle fermée de respecter les spécifications requises (critère de la bande passante et du délai de phase, « attitude quickness » et couplage entre axes).

Dans le chapitre 6, l'implémentation des lois de contrôle développées dans le chapitre 5 a montré que le système n'est pas stable. Cette instabilité a prouvé que l'utilisation des types de réponses RC (commande de vitesse) et ACAH (commande d'attitude) uniquement n'est pas en mesure de stabiliser l'hélicoptère sans l'intervention du pilote pour annuler les vitesses linéaires de translation latérale et longitudinale. Par ailleurs, l'implémentation a permis de constater que les matrices de linéarisation données par HELISIM ne prennent pas en considération la dynamique des actuateurs. Ainsi, les lois de commande calculées avec ces matrices ont conduit à la saturation des actuateurs, ce qui a déstabilisé le système.

En conclusion, on recommande pour un travail de recherche futur, de chercher un moyen d'intégrer les limitations imposées par la saturation des actuateurs dans le processus de design ce qui permettra d'éviter cet écueil lors de l'implémentation. En outre, il est fortement recommandé pour effectuer une simulation sur HELISIM, pour la condition de vol *hover*, d'ajouter des types de réponses TRCPH (commande de vitesse linéaire avec maintien de position) permettant ainsi de maintenir l'hélicoptère dans cette condition de vol pour que le contrôleur soit en mesure de stabiliser l'appareil.

## ANNEXE I

### CODE MATLAB DE CALCUL DES MATRICES K ET H

```
clear
clc
Ad=[-4    0    0    0    0    0    0    0
      1 -0.00199 0    4    0    0    0    0
      0    0    -4    0    0    0    0    0
      1    0    0    0    0    0    0    0
      0    0    0    0    -4   0    0    0
      0    0    0    0    0    -4   0    0
      0    0    0    0    1    0   -0.00526 4
      0    0    0    0    1    0    0    0];
Bd=[4    0    0    0
      0    0    0    0
      0    4    0    0
      0    0    0    0
      0    0    4    0
      0    0    0    4
      0    0    0    0
      0    0    0    0];
[V, D]=eig(Ad);

A=[ -3.39816   0.125984   -0.009209   0.32197   3.38128   0.264369   -0.05101   -0.007470
     0.05722  -0.0256279   0.0163051  -10.6913  -0.690285   0.0560331   0.012913   0.000176
    -0.0723746  -0.000302421  -0.262329  -0.871104  -0.128251   0.750805  -0.020065   0.36892
     0.999373   0           0           0           0           0.035409   0           0
    -13.2514   -0.196484   -0.0156287   0.0387373  -14.0592   11.8281  -0.55627   1.27973
     1.45101   0.0140363   0.00348395  -0.00362807  2.49767  -20.4294   0.149822  -0.185666
    -0.772243  -0.00265147  0.00196474   0.0310787  -0.588983   7.89209  -0.0914927 10.6856
    -0.00285563   0           0           0           1           0.08059   0           0];
B=[ 5.11933   -0.851467   -0.902355   0
     -1.06859   1.06618    0.0847531   0
    -0.0953663  -13.0973   0.0117856   0
     0           0           0           0
    10.9978   -0.961971   6.31677   -1.21587
    -0.984915   1.606     -0.584557   1.51645
    0.279773   0.307194   0.343826  -0.533734
     0           0           0           0];

```

% % % CALCUL DE LA MATRICE DE COMPENSATION H % % %

H=pinv(B)\*Bd;

% % % CALCUL DU VECTEUR REALISABLE V1a ET DE m1 % % %

I1=-0.00199\*eye(8);

A1=I1-A;

A1t=transp(A1);

A0=zeros(8);

Bt=transp(B);

L10=[0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0];

L20=[0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0];

L30=[0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0];

P11=[A1 -B];

P21=[P11 A0];

P31=[L10 L20];

P41=[P31 -Bt];

P51=[eye(8) L30];

P61=[P51 A1t];

Lag1=[P21;P41;P61];

V1d=transp([0 1 0 0 0 0 0 0]);

VL1=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]);

z1=inv(Lag1)\*VL1;

V1a=[z1(1,1);z1(2,1);z1(3,1);z1(4,1);z1(5,1);z1(6,1);z1(7,1);z1(8,1)];

m1=[z1(9,1);z1(10,1);z1(11,1);z1(12,1)];

% % % CALCUL DU VECTEUR REALISABLE V2a ET DE m2 % % %

I2=-0.00526\*eye(8);

A2=I2-A;

A2t=transp(A2);

```

P12=[A2 -B];
P22=[P12 A0];
P32=[L10 L20];
P42=[P32 -Bt];
P52=[eye(8) L30];
P62=[P52 A2t];
Lag2=[P22;P42;P62];
V2d=transp([0 0 0 0 0 0 1 0]);
VL2=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0]);
z2=inv(Lag2)*VL2;
V2a=[z2(1,1);z2(2,1);z2(3,1);z2(4,1);z2(5,1);z2(6,1);z2(7,1);z2(8,1)];
m2=[z2(9,1);z2(10,1);z2(11,1);z2(12,1)];

```

% % % CALCUL DU VECTEUR REALISABLE V3a ET DE m3 % % %

```

I3=-4*eye(8);
A3=I3-A;
A3t=transp(A3);
P13=[A3 -B];
P23=[P13 A0];
P33=[L10 L20];
P43=[P33 -Bt];
P53=[eye(8) L30];
P63=[P53 A3t];
Lag3=[P23;P43;P63];
V3d=transp([0.9701 0 0 -0.2425 0 0 0 0]);
VL3=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0.9701 0 0 -0.2425 0 0 0 0]);
z3=inv(Lag3)*VL3;
V3a=[z3(1,1);z3(2,1);z3(3,1);z3(4,1);z3(5,1);z3(6,1);z3(7,1);z3(8,1)];
m3=[z3(9,1);z3(10,1);z3(11,1);z3(12,1)];

```

% % % CALCUL DU VECTEUR REALISABLE V4a ET DE m4 % % %

```

I4=-0.0001*eye(8);
A4=I4-A;
A4t=transp(A4);
P14=[A4 -B];
P24=[P14 A0];
P34=[L10 L20];
P44=[P34 -Bt];
P54=[eye(8) L30];
P64=[P54 A4t];
Lag4=[P24;P44;P64];
V4d=transp([0 1 0 0.0005 0 0 0 0]);
VL4=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0.0005 0 0 0 0]);

```

```

z4=inv(Lag4)*VL4;
V4a=[z4(1,1);z4(2,1);z4(3,1);z4(4,1);z4(5,1);z4(6,1);z4(7,1);z4(8,1)];
m4=[z4(9,1);z4(10,1);z4(11,1);z4(12,1)];

```

% % % CALCUL DU VECTEUR REALISABLE V5a ET DE m5 % % %

```

I5=-4*eye(8);
A5=I5-A;
A5t=transp(A5);
P15=[A5 -B];
P25=[P15 A0];
P35=[L10 L20];
P45=[P35 -Bt];
P55=[eye(8) L30];
P65=[P55 A5t];
Lag5=[P25;P45;P65];
V5d=transp([0 0 1 0 0 0 0 0]);
VL5=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0]);
z5=inv(Lag5)*VL5;
V5a=[z5(1,1);z5(2,1);z5(3,1);z5(4,1);z5(5,1);z5(6,1);z5(7,1);z5(8,1)];
m5=[z5(9,1);z5(10,1);z5(11,1);z5(12,1)];

```

% % % CALCUL DU VECTEUR REALISABLE V6a ET DE m6 % % %

```

I6=-0.0001*eye(8);
A6=I6-A;
A6t=transp(A6);
P16=[A6 -B];
P26=[P16 A0];
P36=[L10 L20];
P46=[P36 -Bt];
P56=[eye(8) L30];
P66=[P56 A6t];
Lag6=[P26;P46;P66];
V6d=transp([0 0 0 0 0 1 0.0013]);
VL6=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0.0013]);
z6=inv(Lag6)*VL6;
V6a=[z6(1,1);z6(2,1);z6(3,1);z6(4,1);z6(5,1);z6(6,1);z6(7,1);z6(8,1)];
m6=[z6(9,1);z6(10,1);z6(11,1);z6(12,1)];

```

% % % CALCUL DU VECTEUR REALISABLE V7a ET DE m7 % % %

```

I7=-4*eye(8);
A7=I7-A;
A7t=transp(A7);
P17=[A7 -B];

```

```

P27=[P17 A0];
P37=[L10 L20];
P47=[P37 -Bt];
P57=[eye(8) L30];
P67=[P57 A7t];
Lag7=[P27;P47;P67];
V7d=transp([0 0 0 0 0.9701 0 0 -0.2425]);
VL7=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0.9701 0 0 -0.2425]);
z7=inv(Lag7)*VL7;
V7a=[z7(1,1);z7(2,1);z7(3,1);z7(4,1);z7(5,1);z7(6,1);z7(7,1);z7(8,1)];
m7=[z7(9,1);z7(10,1);z7(11,1);z7(12,1)];

```

% % % CALCUL DU VECTEUR REALISABLE V8a ET DE m8 % % %

```

I8=-4*eye(8);
A8=I8-A;
A8t=transp(A8);
P18=[A8 -B];
P28=[P18 A0];
P38=[L10 L20];
P48=[P38 -Bt];
P58=[eye(8) L30];
P68=[P58 A8t];
Lag8=[P28;P48;P68];
V8d=transp([0 0 0 0 0 1 0 0]);
VL8=transp([0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0]);
z8=inv(Lag8)*VL8;
V8a=[z8(1,1);z8(2,1);z8(3,1);z8(4,1);z8(5,1);z8(6,1);z8(7,1);z8(8,1)];
m8=[z8(9,1);z8(10,1);z8(11,1);z8(12,1)];

```

% % % CALCUL DE LA MATRICE DE RETOUR D'ETAT K % % %

```

AV=[V1a V2a V3a V4a V5a V6a V7a V8a];
AW=[m1 m2 m3 m4 m5 m6 m7 m8];

```

```

K=-AW*inv(AV);

```



## ANNEXE II

### CODE VISUAL STUDIO POUR L'IMPLEMENTATION DE HELISIM

```
<*****  
/* FICHIERS INCLUS */  
*****  
  
#include "math.h"  
#include "hlsim.h"  
#include "hlsim_model.h"  
#include "sig_print.h"  
#include "contr_surf.h"  
  
*****  
/* FONCTIONS LOCALES */  
*****  
  
static int model_load_lat(Hlsim_Api_Subsys_Load_Data *load_data);  
static void model_unload_lat(void);  
static void model_repos_lat(int surface_num);  
static void pilot_exec_lat(int surface_num,  
                           double dt,  
                           int *normalized,  
                           int *cmd_type_rate,  
                           double *cmd_value);  
static void ap_exec_lat(int surface_num,  
                       double dt,  
                       int *normalized,  
                       int *cmd_type_rate,  
                       double *cmd_value);
```

```
static int model_load_long(Hlsim_Api_Subsys_Load_Data *load_data);
static void model_unload_long(void);
static void model_repos_long(int surface_num);
static void pilot_exec_long(int surface_num,
                           double dt,
                           int *normalized,
                           int *cmd_type_rate,
                           double *cmd_value);

static void ap_exec_long(int surface_num,
                        double dt,
                        int *normalized,
                        int *cmd_type_rate,
                        double *cmd_value);

static int model_load_ped(Hlsim_Api_Subsys_Load_Data *load_data);
static void model_unload_ped(void);
static void model_repos_ped(int surface_num);
static void pilot_exec_ped(int surface_num,
                           double dt,
                           int *normalized,
                           int *cmd_type_rate,
                           double *cmd_value);

static void ap_exec_ped(int surface_num,
                       double dt,
                       int *normalized,
                       int *cmd_type_rate,
                       double *cmd_value);

static int model_load_coll(Hlsim_Api_Subsys_Load_Data *load_data);
static void model_unload_coll(void);
```

```

static void model_repos_coll(int surface_num);
static void pilot_exec_coll(int    surface_num,
                            double dt,
                            int   *normalized,
                            int   *cmd_type_rate,
                            double *cmd_value);
static void ap_exec_coll(int    surface_num,
                        double dt,
                        int   *normalized,
                        int   *cmd_type_rate,
                        double *cmd_value);

/*****************/
/* LES TROIS FONCTIONS SUIVANTES SONT EXPORTEES A          */
/* PARTIR DE LA LIBRAIRIE GENEREE DE MATLAB                   */
/*****************/
__declspec (dllexport) char * initiateController(double stepsizefromSTAGE);
__declspec (dllexport) char * getControllerOutput(int nbrInputArgs, double* inputArgs,
int nbrOutputArgs, double* outputArgs);
__declspec (dllexport) char * performCleanup();

/*****************/
/* DONNEES LOCALES                                         */
/*****************/
static Hlsim_Subsys_Ctrl_Surf_Functions *hlsim_defaults_lat;
static Hlsim_Subsys_Ctrl_Surf_Functions *hlsim_defaults_lon;
static Hlsim_Subsys_Ctrl_Surf_Functions *hlsim_defaults_ped;
static Hlsim_Subsys_Ctrl_Surf_Functions *hlsim_defaults_coll;

```

```

/*-----*/
/* POINTEURS DE DONNEES */
/*-----*/
static Hlsim_Pilot_Controls*pil;
static Hlsim_Reposition_Data*repos;
static Hlsim_Static_General*gen;
static Hlsim_Api_Data_Pointer data_ptrs[] = {
{ HLSIM_GLOBAL_PILOT,&pil,HLSIM_API_DATA_GLOBAL },
{ HLSIM_GLOBAL_REPOSITION,&repos,HLSIM_API_DATA_GLOBAL },
{ HLSIM_STATIC_GENERAL,&gen,HLSIM_API_DATA_GLOBAL},
{ 0, 0, 0 }
};

/*-----*/
/* POINTEURS SYSTEME */
/*-----*/
static Hlsim_Flight_Control_Output **ctrl;
static Hlsim_Flight_Control_Input **ctrl_i;
static Hlsim_Eom_Output *eom;

static Hlsim_Api_Data_Pointer sys_ptrs[] = {
{HLSIM_SYSTEM_FLIGHT_CONTROL, &ctrl,
HLSIM_API_DATA_SYSTEM_OUTPUTS },
{ HLSIM_SYSTEM_FLIGHT_CONTROL, &ctrl_i,
HLSIM_API_DATA_SYSTEM_INPUTS },
{ HLSIM_SYSTEM_EOM, &eom,
HLSIM_API_DATA_SYSTEM_OUTPUTS },
{ 0, 0, 0 }
};

```

```

static Hlsim_Subsys_Ctrl_Surf_Functions subsys_func_long = {
    model_repos_long,
    pilot_exec_long,
    ap_exec_long,
    0
};

static Hlsim_Subsys_Ctrl_Surf_Functions subsys_func_lat = {
    model_repos_lat,
    pilot_exec_lat,
    ap_exec_lat,
    0
};

static Hlsim_Subsys_Ctrl_Surf_Functions subsys_func_ped = {
    model_repos_ped,
    pilot_exec_ped,
    ap_exec_ped,
    0
};

static Hlsim_Subsys_Ctrl_Surf_Functions subsys_func_coll = {
    model_repos_coll,
    pilot_exec_coll,
    ap_exec_coll,
    0
};

static double *theta_c;
static double *phi_c;
static double *vs_c;

```

```

static Hlsim_Flight_Control_Output *op_lat;
static Hlsim_Flight_Control_Output *op_lon;
static Hlsim_Flight_Control_Output *op_ped;
static Hlsim_Flight_Control_Output *op_coll;

/*****************/
/* DONNEES PUBLIQUES */
/*****************/

Hlsim_Api_Subsys contr_surf_model_lat = {
    /* maj_version_number */ 11,
    /* min_version_number */ 0,

    /* subsystem_name */ HLSIM_SUBSYSTEM_CTRL_SURF,
    /* model_name */ "Control Surf Lateral",

    /* init_global_data */ data_ptrs,
    /* init_system_data */ sys_ptrs,

    /* model_load */ model_load_lat,
    /* model_unload */ model_unload_lat,

    /* snapshot_restore */ 0,

    /* private_data */ 0,
    /* subsystem_functions */ &subsys_func_lat,
};

Hlsim_Api_Subsys contr_surf_model_long = {
    /* maj_version_number */ 11,

```

```
/* min_version_number */ 0,  
  
/* subsystem_name */ HLSIM_SUBSYSTEM_CTRL_SURF,  
/* model_name */ "Control Surf Longitudinal",  
  
/* init_global_data */ data_ptrs,  
/* init_system_data */ sys_ptrs,  
  
/* model_load */ model_load_long,  
/* model_unload */ model_unload_long,  
  
/* snapshot_restore */ 0,  
  
/* private_data */ 0,  
/* subsystem_functions */ &subsys_func_long,  
};  
Hlsim_Api_Subsys contr_surf_model_ped = {  
/* maj_version_number */ 11,  
/* min_version_number */ 0,  
  
/* subsystem_name */ HLSIM_SUBSYSTEM_CTRL_SURF,  
/* model_name */ "Control Surf Yaw",  
  
/* init_global_data */ data_ptrs,  
/* init_system_data */ sys_ptrs,  
  
/* model_load */ model_load_ped,  
/* model_unload */ model_unload_ped,  
  
/* snapshot_restore */ 0,
```

```

/* private_data      */ 0,
/* subsystem_functions */ &subsys_func_ped,
};

Hlsim_Api_Subsys contr_surf_model_coll = {
/* maj_version_number */ 11,
/* min_version_number */ 0,

/* subsystem_name */ HLSIM_SUBSYSTEM_CTRL_SURF,
/* model_name */ "Control Surf Collectif",

/* init_global_data */ data_ptrs,
/* init_system_data */ sys_ptrs,

/* model_load */ model_load_coll,
/* model_unload */ model_unload_coll,

/* snapshot_restore */ 0,

/* private_data */ 0,
/* subsystem_functions */ &subsys_func_coll,
};

static double input_value[12];
static double output_value[4];

/*****************/
/* CODE DES FONCTIONS LOCALES */
/*****************/

```

```

/*************
*          AXE LATERAL
*****
/*-----*/
/* model_load()          */
/*-----*/
static int model_load_lat(Hlsim_Api_Subsys_Load_Data *load_data)
{
    char *temp_error;

    (void)load_data;
    printf("\nSubSystem is loaded");
    printf("\nInitiate Matlab Controller from flsim");
    temp_error = initiateController(gen->basic_itime);
    printf("%s integration time set to:%f",temp_error,
    gen>basic_itime);

    hlsim_defaults_lat = load_data->default_subsystem_functions;
    theta_c = hlsim_get_param_pointer("theta_c",0);
    phi_c = hlsim_get_param_pointer("phi_c",0);
    vs_c = hlsim_get_param_pointer("vs_c",0);

    return(0);
}

/*-----*/
/* model_unload()          */
/*-----*/

```

```

static void model_unload_lat(void)
{
    /*****  

    /* Effectuer (the cleanup) de Simulink          */  

    *****/  

    //  performCleanup();  

    sig_print("model_unload(%s:%d)\n",__FILE__,__LINE__);  

}  

/*-----*/  

/* model_repos()                                */  

/*-----*/  

static void model_repos_lat(int surface_num)
{
    /*-----*/  

    /* Appel de la fonction par défaut HLSIM A/P */  

    /*-----*/  

    sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);  

    hlsim_defaults_lat->model_repos(surface_num);
}  

/*-----*/  

/* pilot_exec()                                */  

/*-----*/  

static void pilot_exec_lat(int    surface_num,
                           double dt,
                           int   *normalized,
                           int   *cmd_type_rate,
                           double *cmd_value)

```

```

{
/*-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/
sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_lat->pilot_exec(surface_num,
                                 dt,
                                 normalized,
                                 cmd_type_rate,
                                 cmd_value);
op_lat = ctrl[surface_num];
}

/*-----*/
/* ap_exec() */
/*-----*/
static void ap_exec_lat(int    surface_num,
                       double  dt,
                       int    *normalized,
                       int    *cmd_type_rate,
                       double *cmd_value)
{
    (void)surface_num;
    (void)dt;
    sig_print("Calling custom AP");

/*****************/
/* Envoi des entrées du pilote pour l'axe latéral */
/*****************/

```

```

input_value[0] = *theta_c * HLSIM_RTD;
input_value[1] = eom->ned.euler.y * HLSIM_RTD; //theta
input_value[2] = *phi_c * HLSIM_RTD;
input_value[3] = eom->ned.euler.x * HLSIM_RTD; //phi
input_value[4] = eom->body.w_n_b.x * HLSIM_RTD; //p
input_value[5] = eom->body.w_n_b.z * HLSIM_RTD; //r
input_value[6] = eom->body.v_abs_cg.y; //V
input_value[7] = eom->body.w_n_b.y * HLSIM_RTD; //q
input_value[8] = eom->body.v_abs_cg.x; //U
input_value[9] = eom->body.v_abs_cg.z; //W
input_value[10] = *vs_c * HLSIM_RTD;
input_value[11] = 0.0;

/*****************/
/* Cette fonction fait une seule itération */
/* Le nombre des entrées et des sorties doit être le même */
/* que pour le block Simulink */
/*****************/

getControllerOutput(12, input_value, 4, output_value);

/*****************/
/* Le "Mapping" doit être effectué ici pour la valeur */
/* reçue de Simulink */
/*****************/

printf("\nTheta c %f\n", input_value[0]);
printf("\nTheta m %f\n", input_value[1]);
printf("\nPhi c %f\n", input_value[2]);
printf("\nPhi m %f\n", input_value[3]);
printf("\nDelta a r %f\n", output_value[1]);

```

```
/*-----*/
/* Demander une déflexion de type (Commande normalisée) */
/*-----*/
    *normalized = 0; // Normalisé ? Si, oui 1
    *cmd_type_rate = 0;
    *cmd_value = output_value[1] * HLSIM_DTR;
}

/***** */
/*          AXE LONGITUDINAL */
/***** */

/*-----*/
/* model_load() */
/*-----*/
static int model_load_long(Hlsim_Api_Subsys_Load_Data *load_data)
{
    (void)load_data;
    printf("\nSubSystem is loaded");
    hlsim_defaults_lon = load_data->default_subsystem_functions;
    return(0);
}

/*-----*/
/* model_unload() */
/*-----*/
static void model_unload_long(void)
{
```

```

/*****************/
/* Effectuer "the cleanup" de Simulink */
/*****************/
sig_print("model_unload(%s:%d)\n",__FILE__,__LINE__);
}

/*
/*-----*/
/* model_repos() */
/*-----*/
static void model_repos_long(int surface_num)
{
/*-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/
sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_lon->model_repos(surface_num);
}

/*
/*-----*/
/* pilot_exec() */
/*-----*/
static void pilot_exec_long(int    surface_num,
                           double dt,
                           int   *normalized,
                           int   *cmd_type_rate,
                           double *cmd_value)
{
/*-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/

```

```

sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_lon->pilot_exec(surface_num,
        dt,
        normalized,
        cmd_type_rate,
        cmd_value);
op_lon = ctrl[surface_num];

}

/*-----
/* ap_exec()                               */
*-----
static void ap_exec_long(int    surface_num,
        double dt,
        int    *normalized,
        int    *cmd_type_rate,
        double *cmd_value)
{
    (void)surface_num;
    (void)dt;
    sig_print("Calling custom AP");

/*****************************************/
/* Le 'Mapping' doit être effectué ici pour la valeur          */
/* reçue de Simulink                                         */
/*****************************************/
printf("\nDelta e %f \n", output_value[2] * HLSIM_DTR);

```

```

/*
/* Demander une déflexion de type (Commande normalisée) */
/*
*normalized = 0; // Normalisé ? Si, oui 1
*cmd_type_rate = 0;
*cmd_value = -output_value[2] * HLSIM_DTR;
}

/***** AXE LACET *****/
/*
*-----*
/* model_load() */
*-----*
static int model_load_ped(Hlsim_Api_Subsys_Load_Data *load_data)
{
    (void)load_data;
    printf("\nSubSystem is loaded");
    hlsim_defaults_ped = load_data->default_subsystem_functions;
    return(0);
}

/*
*-----*
/* model_unload() */
*-----*
static void model_unload_ped(void)
{
    /*
    /* Effectuer "the cleanup" de Simulink
    */
}

```

```

    sig_print("model_unload(%s:%d)\n",__FILE__,__LINE__);
}

/*
 *-----*/
/* model_repos() */
/*-----*/
static void model_repos_ped(int surface_num)
{
/*
 *-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/
sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_ped->model_repos(surface_num);
}

/*
 *-----*/
/* pilot_exec() */
/*-----*/
static void pilot_exec_ped(int    surface_num,
                           double dt,
                           int   *normalized,
                           int   *cmd_type_rate,
                           double *cmd_value)
{
/*
 *-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/

```

```

sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_ped->pilot_exec(surface_num,
        dt,
        normalized,
        cmd_type_rate,
        cmd_value);
op_ped = ctrl[surface_num];
}

/*-----
/* ap_exec()                                     */
/*-----*/
static void ap_exec_ped(int    surface_num,
        double dt,
        int    *normalized,
        int    *cmd_type_rate,
        double *cmd_value)
{
    (void)surface_num;
    (void)dt;
    sig_print("Calling custom AP");

*****/* Le "Mapping" doit être effectué ici pour la valeur      */
/* reçue de Simulink                                         */
*****/printf("\nDelta e %f \n", output_value[3] * HLSIM_DTR);

```

```

/*-----*/
/* Demander une déflexion de type (Commande normalisée) */
/*-----*/
*normalized = 0; // Normalisé ? Si, oui 1
*cmd_type_rate = 0;
*cmd_value = output_value[3] * HLSIM_DTR;
}

/******/
/*          AXE COLLECTIF          */
/******/

/*-----*/
/* model_load()                      */
/*-----*/
static int model_load_coll(Hlsim_Api_Subsys_Load_Data *load_data)
{
    (void)load_data;
    printf("\nSubSystem is loaded");
    hlsim_defaults_coll = load_data->default_subsystem_functions;
    return(0);
}

/*-----*/
/* model_unload()                    */
/*-----*/
static void model_unload_coll(void)
{

```

```

/*****************/
/* Effectuer "the cleanup" de Simulink */
/*****************/
sig_print("model_unload(%s:%d)\n",__FILE__,__LINE__);
}

/*
-----*/
/* model_repos() */
/*-----*/
static void model_repos_coll(int surface_num)
{
/*
-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/
sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_coll->model_repos(surface_num);
}

/*
-----*/
/* pilot_exec() */
/*-----*/
static void pilot_exec_coll(int    surface_num,
                           double dt,
                           int   *normalized,
                           int   *cmd_type_rate,
                           double *cmd_value)
{
/*
-----*/
/* Appel de la fonction par défaut HLSIM A/P */
/*-----*/

```

```

sig_print("%s(%d) calling default autopilot\n",__FILE__,__LINE__);
hlsim_defaults_coll->pilot_exec(surface_num,
        dt,
        normalized,
        cmd_type_rate,
        cmd_value);
op_coll = ctrl[surface_num];

}

/*-----*/
/* ap_exec() */
/*-----*/
static void ap_exec_coll(int    surface_num,
        double dt,
        int    *normalized,
        int    *cmd_type_rate,
        double *cmd_value)
{
    (void)surface_num;
    (void)dt;
    sig_print("Calling custom AP");

/*****************/
/* Le 'Mapping' doit être effectué ici pour la valeur */
/* reçue de Simulink */
/*****************/
printf("\nDelta e %f\n", output_value[0] * HLSIM_DTR);

```

```
/*-----*/
/* Demander une déflexion de type (Commande normalisée) */
/*-----*/
*normalized = 0; // Normalisé ? Si, oui 1
*cmd_type_rate = 0;
*cmd_value = output_value[0] * HLSIM_DTR;
}
```

## LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

- Abi Elias, Raymond. 1999. « Commande Par Application de la Structure Modale Appliquée en Avionique ». Mémoire de Maîtrise en génie électrique, Montréal, École De Technologie Supérieure, 96 p.
- Madani, Sid Ahmed. 2007. « La Commande Active D'un Hélicoptère ». Mémoire de Maîtrise en génie électrique, Montréal, École De Technologie Supérieure, 88 p.
- Coyle, S. 1996. *The art and science of flying helicopters*, Iowa : Iowa State University Press, 225 p.
- Moritz, N., et R. Osterhuber. 2006. « Three-Stage Gradient-Based Optimization Scheme in Design of Feedback Gains within Eurofighter Primary Control Laws ». In *AIAA Guidance, Navigation, and Control Conference and Exhibit*. (Keystone Colorado, Aug. 21-24 2006).
- Anon, A.D.S. 2000. « 33E-PRF, Performance Specification, Handling Qualities Requirements for Military Rotorcraft ». *US Army AMCOM, Redstone, Alabama*, vol. 21.
- Chen, C.T. 1970. *Introduction to linear system theory*. Holt, Rinehart and Winston New York.
- Garrard, W., B. Liebst et J. Farm. 1987. « Eigenspace techniques for active flutter suppression ».
- Garrard, W.L., et E. Low. 1990. *Eigenspace design of helicopter flight control systems*. DTIC Document.
- Grübel, G., et H.D. Joos. 1997. « Multi-objective parameter synthesis (MOPS) ». *Robust Flight Control*, p. 13-21.
- Hald, U.B., M.V. Hesselbæk, J.T. Holmgaard, C.S. Jensen, S.L. Jakobsen, M. Siegumfeldt, A. la Cour-Harbo et J.S. Thomsen. 2005. « Autonomous Helicopter—Modelling and Control ». *Report, Aalborg university*.
- Klein, G., et B. Moore. 1977. « Eigenvalue-generalized eigenvector assignment with state feedback ». *Automatic Control, IEEE Transactions on*, vol. 22, n° 1, p. 140-141.
- Kreisselmeier, G., et R. Steinhauser. 1983. « Application of vector performance optimization to a robust control loop design for a fighter aircraft ». *International Journal of Control*, vol. 37, n° 2, p. 251-284.

- Low, E., et W. Garrard. 1993. « Design of Flight Control Systems to meet Rotorcraft Handling Qualities Specifications ». *Journal of Guidance, Control, and Dynamics*, vol. 16, n° 1, p. 69-78.
- Moore, BC. 1976. « On the flexibility offered by state feedback in multivariable systems beyond closed loop eigenvalue assignment ». *Automatic Control, IEEE Transactions on*, vol. 21, n° 5, p. 689-692.
- Nye, W.T. 1984. « Delight: An interactive system for optimization-based engineering design ». *Dissertation Abstracts International Part B: Science and Engineering[DISS. ABST. INT. PT. B- SCI. & ENG.]*, vol. 44, n° 8.
- Srinathkumar, S. 1978. « Eigenvalue/eigenvector assignment using output feedback ». *Automatic Control, IEEE Transactions on*, vol. 23, n° 1, p. 79-81.
- Tischler, M.B., C.M. Ivler, M.H. Mansur, K.K. Cheung, T. Berger et M. Berrios. 2008. « Handling-Qualities Optimization and Trade-offs in Rotorcraft Flight Control Design ». In., p. 4-6.
- Valavanis, K. 2007. *Advances in unmanned aerial vehicles: state of the art and the road to autonomy*, 33. Springer Verlag.
- Yanguo, S., et W. Huanjin. 2009. « Design of flight control system for a small unmanned tilt rotor aircraft ». *Chinese Journal of Aeronautics*, vol. 22, n° 3, p. 250-256.