

**ÉCOLE DE TECHNOLOGIE SUPÉRIEURE  
UNIVERSITÉ DU QUÉBEC**

**MÉMOIRE PRÉSENTÉ À  
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE**

**COMME EXIGENCE PARTIELLE  
À L'OBTENTION DE LA  
MAÎTRISE EN GÉNIE MÉCANIQUE  
M.ING.**

**PAR  
CHRISTIAN BELLEAU**

**CONCEPTION D'UN SYSTÈME DE CONTRÔLE POUR  
UN HÉLIICOPTÈRE À PROPULSION HUMAINE**

**MONTRÉAL, DÉCEMBRE 2000**

**@ droits réservés de Christian Belleau 2000**

**CE MÉMOIRE A ÉTÉ ÉVALUÉ  
PAR UN JURY COMPOSÉ DE :**

- **M. Louis Lamarche, directeur de mémoire**  
Département de génie mécanique à l'École de technologie supérieure
- **M. Christian Masson, professeur**  
Département de génie mécanique à l'École de technologie supérieure
- **M. Louis Thirion, président**  
TGI Technologies inc.

**IL A FAIT L'OBJET D'UNE PRÉSENTATION DEVANT JURY ET UN PUBLIC**

**À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE**

# **CONCEPTION D'UN SYSTÈME DE CONTRÔLE POUR UN HÉLICOPTÈRE À PROPULSION HUMAINE**

**Christian Belleau**

**(Sommaire)**

Depuis 1980, l'American Helicopter Society organise un concours qui consiste à faire voler un hélicoptère à propulsion humaine pendant une minute. Vingt ans plus tard, 17 équipes ont essayé de remporter le concours, mais aucune n'y est parvenu. Seulement deux de ces prototypes ont quitté le sol. En 1998, quatre membres du sous-marin Omer 3 ont commencé la conception d'un hélicoptère à propulsion humaine dans le but de participer à ce concours.

La plupart des équipes qui ont participé au concours ont eu des problèmes de stabilité en vol. Il était donc nécessaire de faire un modèle dynamique de l'hélicoptère pour confirmer la stabilité de celui-ci et s'assurer que les pales ne toucheront pas au sol. Ce modèle tient compte des effets dynamiques et aérodynamiques exercés sur l'hélicoptère.

La différence majeure entre le prototype conçu et tous les autres hélicoptères à propulsion humaine est qu'il possède un système de pale à pas variable contrôlé par un ordinateur. Ce type de système, qui a donné la victoire aux sous-marins Omer 2 et 3, est encore plus important dans le cas d'un hélicoptère. Il permet d'économiser l'énergie du pédaleur, de rendre le comportement en vol des pales identiques et de contrôler l'hélicoptère pour qu'il soit capable de respecter les règles du concours.

Étant donné les contraintes de masse, les composantes du système de contrôle électronique, principalement les moteurs et les contrôles en position de ceux-ci, doivent être allégé au maximum tout en gardant les mêmes caractéristiques.

## **AVANT-PROPOS**

Pour commencer, je tiens à remercier mon directeur de mémoire, M. Louis Lamarche, pour tout l'aide qu'il m'a apporté lors de l'analyse et de la programmation du modèle dynamique.

Après avoir fait le système de contrôle des sous-marins Omer 2 et 3 et l'injection électronique de la formule SAE 96, je voulais m'éloigner quelques temps des microprocesseurs, de l'électronique et de la programmation. Lorsque le projet Hélios débuta, je savais que je voulais apprendre le plus possible en ingénierie mécanique. Il est certain que les membres des projets Omer 2, Omer 3 et formule SAE 96 m'ont donné cette soif d'apprendre ce domaine, mais Hélios m'a permis de le mettre en pratique. Je tiens donc à remercier tous les membres de ces projets, et en particulier Simon Joncas et Daniel Côté qui ont toujours répondu à mes questions sur l'ingénierie mécanique, et il y en a eu beaucoup durant les trois dernières années. J'aimerais aussi remercier Louis Thirion qui m'a aidé plusieurs fois et appris plusieurs choses durant les cinq dernières années, lui aussi a répondu à beaucoup de questions.

Il est important de mentionner que l'ÉTS et ses employés, spécialement les techniciens en mécanique et les employés du département de l'immeuble et des équipements, ont grandement facilité notre travail sur Hélios.

Pour terminer, je veux remercier mes parents qui m'ont toujours encouragé durant mes études.

## TABLE DES MATIÈRES

	Page
SOMMAIRE .....	i
AVANT-PROPOS .....	ii
LISTE DES FIGURES.....	vi
LISTE DES ABRÉVIATIONS ET SIGLES .....	vii
INTRODUCTION.....	1
CHAPITRE 1 : Configuration générale de l'hélicoptère .....	4
CHAPITRE 2 : Modèle dynamique du vol de l'hélicoptère .....	9
2.1 Cinématique .....	9
2.2 Calcul de vitesse linéaire et angulaire.....	11
2.3 Dynamique.....	13
2.3.1 Accélération linéaire .....	13
2.3.2 Accélération angulaire .....	15
2.4 Inerties .....	17
2.4.1 Inertie des pales.....	17
2.4.2 Inertie de la nacelle et du pédaleur.....	20
2.4.3 Inertie de l'hélicoptère .....	21
2.5 Forces.....	22
2.5.1 Gravité.....	22
2.5.2 Forces aérodynamiques.....	22
2.6 Programmation du modèle.....	27
2.7 Modification du programme de calcul aérodynamique .....	29
2.8 Analyse des résultats.....	30

2.8.1	Force aérodynamique différente sur une des pales .....	30
2.8.2	Hélicoptère déséquilibré.....	31
CHAPITRE 3 : Système de contrôle.....		36
3.1	Banc d'essai .....	37
3.2	Algorithme de contrôle en vol ( boucle ouverte ).....	38
3.3	Algorithme de contrôle en vol ( boucle fermée ).....	41
CHAPITRE 4 : Système de contrôle en position des pales .....		46
CONCLUSION.....		50
RECOMMENDATIONS .....		52
BIBLIOGRAPHIE .....		53
ANNEXES		
A :	Fabrication des gabarits des moules .....	54
B :	Programmes Matlab pour le modèle dynamique .....	63
C :	Schéma électrique du contrôleur en position des moteur .....	81

## LISTE DES FIGURES

	Page
1.1 Configurations possibles de l'hélicoptère.....	4
2.1 Deux systèmes d'axes dans l'espace .....	9
2.2 Système d'axe d'une pale .....	18
2.3 Élément d'une pale .....	25
2.4 Modèle dynamique en SIMULINK.....	28
2.5 Exemple de l'appel d'une fonction MATLAB.....	28
2.6 Vitesse de rotation des pales et couple fourni par le pédaleur.....	32
2.7 Position linéaire et angulaire de l'hélicoptère par rapport au sol .....	32
2.8 Vitesse de rotation des pales et couple fourni par le pédaleur.....	33
2.9 Position linéaire et angulaire de l'hélicoptère par rapport au sol .....	33
3.1 Procédure de calibrage d'un rotor .....	39
3.2 Algorithme de contrôle en boucle ouverte .....	40
3.3 Modèle dynamique avec contre-réaction sur la vitesse de rotation de la nacelle.	42
3.4 Vitesse de rotation des pales et couple fourni par le pédaleur.....	43
3.5 Position linéaire et angulaire de l'hélicoptère par rapport au sol .....	43
4.1 Schéma-bloc du contrôleur de moteur.....	48
A.1 Profil unitaire DAEDALUS 31 .....	54
A.2 Profil unitaire DAEDALUS 31 avec axe de rotation à 33% .....	55
A.3 Profil DAEDALUS positionné à son angle de calage .....	55
A.4 Vue de dessus d'un profil .....	56
A.5 Vue de dessus de trois profils et d'un gabarit.....	57

## LISTE DES ABRÉVIATIONS ET DES SIGLES

$p_x$	Position en $x$ de l'hélicoptère par rapport au référentiel au sol, m
$p_y$	Position en $y$ de l'hélicoptère par rapport au référentiel au sol, m
$h$	Hauteur de l'hélicoptère par rapport au sol, m
$\phi$	Rotation autour de $x$ de l'hélicoptère par rapport au sol, rad
$\theta$	Rotation autour de $y$ de l'hélicoptère par rapport au sol, rad
$\psi$	Rotation autour de $z$ de l'hélicoptère par rapport au sol, rad
$u$	Vitesse de l'hélicoptère en $x$ ( $x$ du référentiel de l'hélicoptère), m/s
$v$	Vitesse de l'hélicoptère en $y$ ( $y$ du référentiel de l'hélicoptère), m/s
$w$	Vitesse de l'hélicoptère en $z$ ( $z$ du référentiel de l'hélicoptère), m/s
$p_x$	Vitesse de rotation de l'hélicoptère autour de $x$ ( $x$ du référentiel de l'hélicoptère), rad/s
$q_y$	Vitesse de rotation de l'hélicoptère autour de $y$ ( $y$ du référentiel de l'hélicoptère), rad/s
$r_z$	Vitesse de rotation de l'hélicoptère autour de $z$ ( $z$ du référentiel de l'hélicoptère), rad/s
$\gamma$	Position de la pale 1 par rapport à la nacelle, rad
$\omega$	Vitesse de rotation des pales, rad/s
$\alpha$	Angle de calage des pales, rad
$x_p$	Position du centre de pression de la pale en $x$ par rapport au centre de gravité de l'hélicoptère, m
$y_p$	Position du centre de pression de la pale en $x$ par rapport au centre de gravité de l'hélicoptère, m
$z_p$	Position du centre de pression de la pale en $x$ par rapport au centre de gravité de l'hélicoptère, m

## **INTRODUCTION**

Depuis maintenant vingt ans, l'American Helicopter Society (AHS) organise un concours dont l'objectif est de créer un hélicoptère à propulsion humaine. Au début de 1998, seulement 17 universités avaient essayé de remporter l'épreuve, aucune n'y arriva. Seulement deux universités ont réussi à faire décoller leurs engins, la première CalPoly avec leur prototype Da Vinci ont fait un vol de 7,1 secondes à quelques centimètres en 1991 et la deuxième est Nihon University au Japon avec YURI, qui lui a fait un vol d'environ 19 secondes à 75 cm en 1994. Il est bon de mentionner que le professeur Naito, qui a développé le YURI, en était à plus de 10 ans de travail dans le domaine des hélicoptères à propulsion humaine.

Après avoir conçu les sous-marins à propulsion humaine Omer 2 et 3, quatre membres de Omer ont décidé de prendre un défi de taille, débuter un projet d'hélicoptère à propulsion humaine à l'École de technologie supérieure. Ce projet, qui a une équipe composée de 4 étudiants à la maîtrise et de 8 étudiants au baccalauréat, se nomme Hélios. L'objectif initial était de concevoir et fabriquer le prototype en deux ans, malheureusement quelques difficultés ont ralenti le développement de l'hélicoptère.

Les critères pour gagner le concours sont de faire un hélicoptère à propulsion humaine qui fera un vol stable de une minute en restant dans un carré de 10\*10 mètres. Durant cette minute, l'engin doit aller momentanément à une altitude de trois mètres et la nacelle ne doit pas faire plus d'un tour sur l'axe des rotors. Il est interdit d'emmagasiner de l'énergie de quelque façon que ce soit pour faciliter le travail du pédaleur, il faut donc un entraînement direct entre les pales et le pédaleur. Il est aussi

interdit d'utiliser des gaz plus léger que l'air afin de diminuer le poids à lever par le pédaleur.

Ce qui a fait le succès des sous-marins Omer est le système de pas variable qui permet à l'hélice d'augmenter ou de diminuer son angle d'attaque. Le tout, contrôlé par ordinateur, rend le travail du pédaleur beaucoup plus facile car, au départ de la course, les angles d'attaques des pales sont mis au minimum afin de permettre au sous-marin d'accélérer avec le moins d'effort de la part du pédaleur alors qu'à la fin de la course, les pales sont mises à l'angle optimale afin d'obtenir la meilleure vitesse de pointe.

Ce système qui a permis d'augmenter les performances des sous-marins Omer devient presque essentiel dans le cas de l'hélicoptère. L'effort du pédaleur n'est plus de 30 secondes mais plutôt d'une minute, il est donc important d'éliminer les situations où le pédaleur travaille inutilement. De plus, les pales des sous-marins étaient courtes et n'avaient pas une grande poussée aérodynamique, on pouvait considérer qu'elles ne se déformaient pas. Les pales d'Hélios sont très grandes et la poussée qu'elles génèrent les obligent à se déformer, il est donc évident que chaque pale aura un comportement différent lorsqu'elle sera en fonction. Le système de pas variable servira à égaliser le comportement des pales en vol.

Le problème principal rencontré par les autres hélicoptères à propulsion humaine n'est pas d'obtenir assez de poussée aérodynamique pour pouvoir s'envoler, mais plutôt de rester stable dès le décollage. Les livres sur la dynamique de vol des hélicoptères discutent principalement des hélicoptères commerciaux avec un rotor principal et un rotor de queue. Les différences principales entre un modèle dynamique d'hélicoptère commercial et celui d'Hélios sont que ce dernier a deux rotors contre-rotatifs coaxiaux plutôt qu'un seul, qu'il ne possède pas de rotor de queue et qu'il n'avance jamais dans l'air. Par contre, l'approche utilisé pour déterminer les équations de dynamique de vol sera semblable. Lorsque le modèle sera fonctionnel, il permettra de connaître le

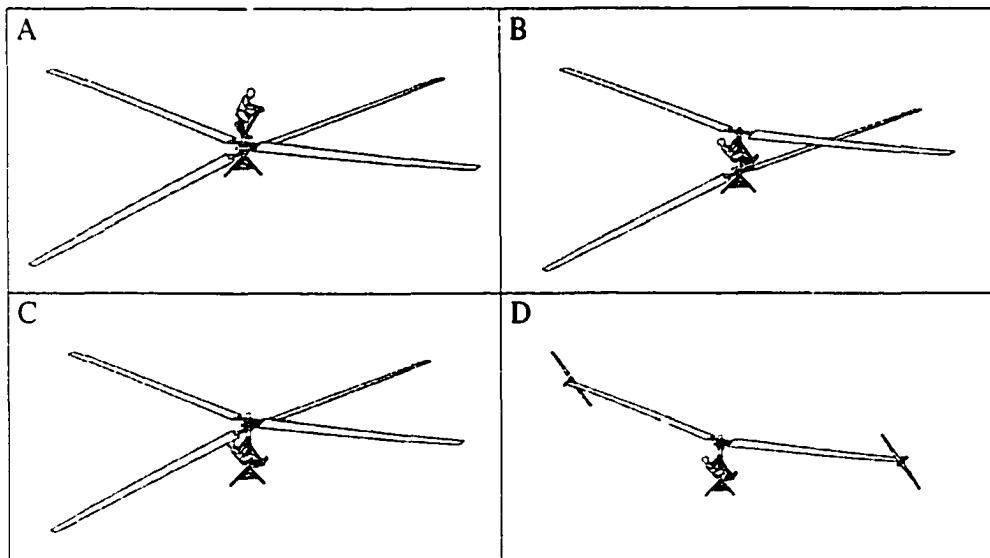
comportement de l'hélicoptère dans différentes situations, par exemple, une pale qui pousse plus que les trois autres. À ce point, le modèle pourra être utilisé pour y introduire des boucles de contre-réaction afin d'améliorer la stabilité et le contrôle du prototype. Il faut aussi déterminer si l'hélicoptère peut fonctionner seulement avec des boucles de contre-réaction ou s'il sera nécessaire de calibrer les rotors afin d'améliorer la stabilité.

En plus d'avoir fait le modèle dynamique au sein de l'équipe Hélios, j'ai aussi travaillé sur la conception aérodynamique, sur la conception et la fabrication des moules, sur la conception et la fabrication des pales, sur la programmation et l'usinage de pièces ayant des surfaces 3D complexes et sur la conception et la fabrication des différents systèmes de contrôles électroniques des bancs d'essais du projet Hélios. Certains de ces travaux ont été mis en annexe du mémoire car ils n'en constituent pas le sujet principal.

## CHAPITRE 1

### CONFIGURATION GÉNÉRALE DE L'HÉLICOPTÈRE

Avant de pouvoir faire un modèle dynamique, il faut décider quelle sera la forme de l'hélicoptère et quel sera son fonctionnement. Deux critères ont semblé plus importants que tout autre lors de la décision de la configuration générale : maximiser la poussée aérodynamique en laissant les hélices le plus près du sol et avoir la configuration la plus stable sans système de contrôle. En analysant les prototypes conçus par les autres équipes et par logique, il était possible de ressortir six configurations. On peut voir dans la figure 1.1, ces six configurations.



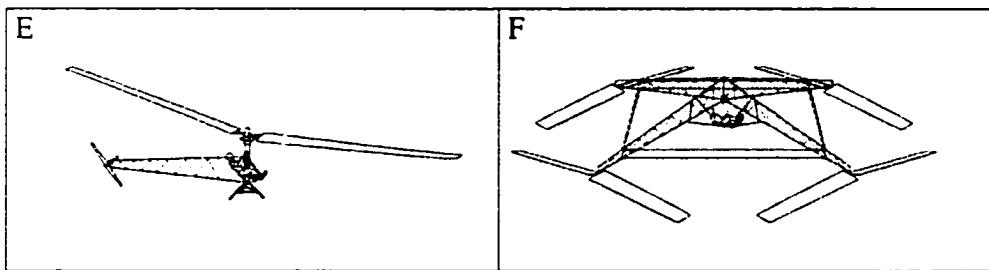


Figure 1.1 Configurations possibles de l'hélicoptère

Nous allons énumérer les différentes qualités et défauts de chaque configuration, ce qui nous permettra de déduire la configuration la plus intéressante. Il est important de se rappeler que dans chaque configuration, il doit y avoir un dispositif qui élimine le contre-couple généré par les pales. Il est à noter que les figures ne sont pas à l'échelle, car les pales sont tellement grandes que la nacelle aurait seulement l'air d'un point dans le centre de l'image.

La configuration A permet d'avoir la meilleure poussée aérodynamique car les hélices sont les plus basses, elles bénéficient donc plus de l'effet de sol et puisqu'elles sont contre-rotatives, le contre-couple de chaque hélice est éliminé par l'autre. Par contre, les hélices sont très près l'une de l'autre, donc si une des pales du bas se déforme plus qu'une des pales du haut, il y aura collision. Avec des hélices ayant un diamètre de 35 mètres, il est certain qu'il y aura contact si la distance entre les deux hélices est moins de 50 centimètres. Un point encore plus dangereux est que le centre de gravité est au-dessus de la poussée aérodynamique, on se retrouve donc avec un système instable.

La configuration B est aussi une configuration dont les hélices sont co-axiales et contre-rotatives comme en A. La distance entre les deux hélices est augmentée de beaucoup car la nacelle est positionnée entre les deux rotors. En ayant un rotor au-dessus de la nacelle, on peut s'assurer que le système devienne stable si le centre de

gravité est positionné assez bas et si la poussée du rotor du haut est plus importante que celle du bas. C'est la configuration choisie pour le projet Hélios.

La configuration C ressemble beaucoup à A, à l'exception des rotors qui sont au-dessus de la nacelle. Cela règle le problème de stabilité mais enlève de la poussée aérodynamique étant donné que les hélices sont moins près du sol. Le problème de collision entre les pales est identique à la configuration A.

La configuration D est celle qui a permis le premier vol d'un hélicoptère à propulsion humaine, le DaVinci. Son principal avantage est que la grande hélice ne génère pas de contre-couple sur la nacelle car elle est entraînée par deux petites hélices au bout de la grande hélice. Son principal inconvénient est son efficacité mécanique, les deux petites hélices ne transmettent pas toute la force du pédaleur en poussée car elles ont une traînée aérodynamique, et donc une perte d'énergie supplémentaire.

La configuration E est la configuration classique d'un hélicoptère commercial. Le rotor de queue, qui élimine le contre-couple du rotor principal, représente une perte d'énergie d'environ 10%, c'est peu lorsqu'on a une turbine de 1000 HP mais beaucoup lorsqu'on a moins de 1 HP. De plus, le rotor de queue oblige de faire une structure pour le soutenir, ce qui rajoute du poids inutile à l'hélicoptère.

La configuration F représente le deuxième hélicoptère qui a volé, le YURI. Ce modèle possède probablement le meilleur effet de sol et la meilleure stabilité. Par contre, la structure est complexe et lourde et l'efficacité mécanique diminue car les mécanismes pour propager l'énergie fournie par le pédaleur possèdent plusieurs roulements.

La configuration B semble être le meilleur choix car elle possède un bon compromis de tous les caractéristiques vues dans les autres configurations. On peut énumérer les principaux avantages :

- Un des deux rotors est positionné le plus bas possible,
- peut être stable,
- entraînement mécanique simple,
- structure simple et légère.

Après avoir décidé de la configuration de l'hélicoptère, il était possible de faire la conception aérodynamique. Il semblait plus judicieux de faire un programme d'aérodynamique en utilisant un langage de programmation plutôt que d'utiliser un programme commercial, ceci permettait de comprendre l'aérodynamique et de pouvoir adapter le programme à l'application qui n'est pas classique. Lorsque le programme était fonctionnel, il fut comparé aux résultats d'un autre programme afin d'en valider le fonctionnement. Pour terminer la conception aérodynamique, plus de 3 millions de pales ont été testées afin de trouver la meilleure pale pour chaque rotor. Les rotors retenus possèdent deux pales et ont une longueur de 17,5 mètres et ont une corde de départ de 1,2 mètres et de fin de 10 centimètres.

Lorsque la forme de la pale a été choisie, il fallait construire les moules. Puisque les résultats du programme d'aérodynamique étaient seulement l'angle du profil et la longueur de la corde selon le rayon à laquelle le profil était positionné, il fallait dessiner la pale ou du moins faire un programme qui dessine les gabarits pour faire les moules. La dernière solution est plus longue à planter, par contre lorsque le programme est fonctionnelle, il est possible de modifier la forme de la pale en un instant. En utilisant un système de dessin assisté par ordinateur (DAO), il fallait dessiner les deux pales, et si une modification devait être faite, il fallait recommencer complètement les dessins. Les gros programmes de DAO, comme CATIA, ou plus récemment les plus courant comme

Autocad, utilisent une approche paramétrique. Ce type de système permet de laisser les grandeurs du dessin en variable, mais le temps d'apprentissage du fonctionnement d'un tel système semblait trop long par rapport aux avantages qu'il fournissait. Le fonctionnement du programme qui dessine les gabarits des moules est expliqué en annexe de ce mémoire.

Par la suite, il a été possible de fabriquer les moules et les pales. Toute cette partie est expliquée en détail dans le mémoire de Simon Joncas « Conception et fabrication d'une aile rotative basse vitesse pour un hélicoptère à propulsion humaine ».

Maintenant que l'on connaît la forme de Hélios, il est possible de faire le modèle dynamique de l'hélicoptère.

## CHAPITRE 2

### MODÈLE DYNAMIQUE DU VOL DE L'HÉLICOPTÈRE

#### 2.1 Cinématique

Dans le modèle mathématique du vol de l'hélicoptère, ce qui nous intéresse est la position et surtout l'orientation de l'hélicoptère par rapport au sol. Il sera donc nécessaire de faire les calculs par rapport à deux référentiels, un référentiel fixe représentant le sol et l'autre mobile représentant l'hélicoptère.

Il faut trouver une série de transformation (translation et rotation) qui permettent de passer d'un système de coordonnées à l'autre. Les six transformations possibles sont trois translations et trois rotations autour de  $x$ ,  $y$  et  $z$ . La figure 2.1 représente une série de rotations possibles pour orienter un système d'axe par rapport à l'autre. Pour une explication plus détaillée sur les transformations entre deux systèmes d'axes , voir Etkin (1972) ou un ouvrage de dynamique de vol ou de robotique qui discute des angles d'Euler.

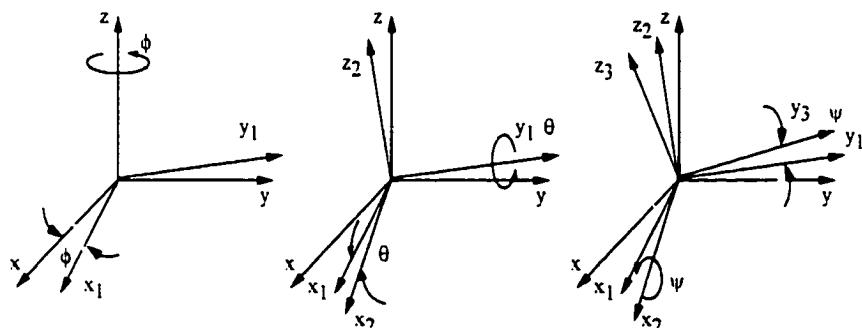


Figure 2.1 Deux systèmes d'axes dans l'espace

Voici les matrices de transformation en coordonnées homogènes (rotation et translation dans la même matrice). La première matrice tient compte des trois translations.

$$\begin{aligned}
 trans &= \begin{bmatrix} 1 & 0 & 0 & p_n \\ 0 & 1 & 0 & p_e \\ 0 & 0 & 1 & h \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 rot_z(\psi) &= \begin{bmatrix} c\psi & -s\psi & 0 & 0 \\ s\psi & c\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 rot_y(\theta) &= \begin{bmatrix} c\theta & 0 & s\theta & 0 \\ 0 & 1 & 0 & 0 \\ -s\theta & 0 & c\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 rot_x(\phi) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\phi & -s\phi & 0 \\ 0 & s\phi & c\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1}
 \end{aligned}$$

Afin de simplifier l'écriture  $c\phi$  signifie  $\cos(\phi)$ . Il est certain qu'avec une translation suivi d'une rotation autour de z, de y et de x, il sera possible d'aligner les deux systèmes de coordonnées.

$${}^0T_1 = trans(p_n, p_e, h) * rot_z(\psi) * rot_y(\theta) * rot_x(\phi) \tag{2.2}$$

où  ${}^0T_1$  représente la matrice de transformation qui permet de représenter le système de coordonnées 1 par rapport au système de coordonnées 0. La matrice suivante est le résultat final de la multiplication. La sous-matrice 3\*3 de gauche représente l'orientation alors que la 4<sup>ième</sup> colonne représente la position.

$${}^0T_1 = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi & p_n \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi & p_e \\ -s\theta & c\theta s\phi & c\theta c\phi & h \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

La sous-matrice d'orientation sera utilisée dans le modèle SIMULINK. Elle permettra de représenter la vitesse linéaire de l'hélicoptère par rapport au sol.

$${}^0R_1 = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (2.4) \text{ (équation I)}$$

En inversant la sous-matrice d'orientation, on aura l'orientation du système d'axe 0 par rapport à 1. Ce résultat sera utilisé dans les équations de cinématique.

$${}^1R_0 = \begin{bmatrix} c\psi c\theta & s\psi c\theta & -s\theta \\ c\psi s\theta s\phi - s\psi c\phi & s\psi s\theta s\phi + c\psi c\phi & c\theta s\phi \\ c\psi s\theta c\phi + s\psi s\phi & s\psi s\theta c\phi - c\psi s\phi & c\theta c\phi \end{bmatrix} \quad (2.5)$$

## 2.2 Calcul de vitesse linéaire et angulaire

Les résultats précédents permettent de positionner un système d'axe par rapport à l'autre, il faut maintenant trouver la vitesse linéaire et angulaire entre les deux systèmes d'axes. La vitesse linéaire relative  ${}^0\bar{v}_1$  peut se calculer avec l'équation suivante :

$${}^0\bar{v}_1 = {}^0\bar{v}_{trans_1} + {}^0\bar{\omega}_1 \times {}^0\bar{p}_1 \quad (2.6)$$

où  ${}^0\bar{p}_1$  est le vecteur position de la position du système 1 par rapport à 0 et  ${}^0\bar{\omega}_1$  est la vitesse angulaire du système 1 par rapport à 0. Pour simplifier les calculs dans la

partie dynamique, toutes les forces seront placées par rapport à l'hélicoptère, il sera donc utile d'avoir la vitesse linéaire entre le système 0 et 1 par rapport à 1.

$${}^1\bar{v}_{0,1} = {}^1R_0 {}^0\bar{v}_1 \quad (2.7)$$

Le calcul des vitesses angulaires est plus simple car la vitesse linéaire n'influence pas les vitesses angulaires. Donc  ${}^0\bar{\omega}_1$  sera seulement fonction de  $\dot{\phi}$ ,  $\dot{\theta}$  et  $\dot{\psi}$ .

$${}^0\bar{\omega}_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} + rot_z(\psi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + rot_z(\psi) rot_y(\theta) \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} \quad (2.8)$$

et le résultat final est

$${}^0\bar{\omega}_1 = \begin{bmatrix} c\psi c\theta & -s\psi & 0 \\ s\psi c\theta & c\psi & 0 \\ -s\theta & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (2.9)$$

Comme dans le cas de la vitesse linéaire, il sera intéressant d'avoir la vitesse angulaire entre 0 et 1 par rapport à 1.

$${}^1\bar{\omega}_{0,1} = {}^1R_0 {}^0\bar{\omega}_1 = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (2.10)$$

## 2.3 Dynamique

### 2.3.1 Accélération linéaire

Il reste seulement à trouver les équations d'accélération linéaire et angulaire pour concevoir le modèle dynamique. L'équation d'accélération linéaire se trouve facilement en dérivant la vitesse linéaire.

$$\begin{aligned} {}^1\bar{\boldsymbol{a}}_{0,I} &= {}^1\boldsymbol{R}_0 \frac{d}{dt}({}^0\bar{\boldsymbol{v}}_{0,I}) = {}^1\boldsymbol{R}_0 \frac{d}{dt}({}^0\boldsymbol{R}_I {}^1\bar{\boldsymbol{v}}_{0,I}) = {}^1\boldsymbol{R}_0 \left( {}^0\boldsymbol{R}_I {}^1\dot{\bar{\boldsymbol{v}}}_{0,I} + {}^0\dot{\boldsymbol{R}}_I {}^1\bar{\boldsymbol{v}}_{0,I} \right) \\ &= {}^1\boldsymbol{R}_0 \left( {}^0\boldsymbol{R}_I {}^1\dot{\bar{\boldsymbol{v}}}_{0,I} + {}^0\bar{\boldsymbol{\omega}}_{0,I} \times {}^0\boldsymbol{R}_I {}^1\bar{\boldsymbol{v}}_{0,I} \right) = \left( {}^1\dot{\bar{\boldsymbol{v}}}_{0,I} + {}^1\bar{\boldsymbol{\omega}}_{0,I} \times {}^1\bar{\boldsymbol{v}}_{0,I} \right) \end{aligned} \quad (2.11)$$

Tous les termes de l'équation 2.11 ont été calculés dans la partie cinématique et de vitesse linéaire et angulaire, il sera donc possible de coupler la partie dynamique et cinématique afin de trouver les positions linéaires et angulaires. Afin de simplifier les calculs, on définira des variables seulement pour la partie dynamique.

$${}^1\bar{\boldsymbol{v}}_{0,I} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad {}^1\bar{\boldsymbol{\omega}}_{0,I} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.12)$$

où  $u$ ,  $v$  et  $w$  représentent la vitesse de l'hélicoptère par rapport au sol exprimée dans le système d'axe de l'hélicoptère. Pour simplifier les équations, le système d'axe est positionné au centre de gravité de l'hélicoptère. Les variables  $p$ ,  $q$  et  $r$  représentent la vitesse de rotation de l'hélicoptère par rapport au sol autour des axes de roulis, de tangage et de lacet. On peut réécrire l'équation 2.11 avec les nouvelles variables.

$${}^1\bar{\boldsymbol{a}}_{0,I} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.13)$$

En utilisant la 2<sup>ème</sup> loi de Newton, il est possible de coupler les forces avec l'accélération.

$$\sum {}^1\bar{\mathbf{F}}_{0,I} = \frac{d}{dt}(m{}^1\bar{\mathbf{v}}_{0,I}) = \frac{dm}{dt}{}^1\bar{\mathbf{v}}_{0,I} + m \frac{d{}^1\bar{\mathbf{v}}_{0,I}}{dt} \quad (2.14)$$

Puisque la masse de l'hélicoptère ne varie pas avec le temps, il est possible de simplifier l'équation 2.14.

$$\sum {}^1\bar{\mathbf{F}}_{0,I} = m{}^1\bar{\mathbf{a}}_{0,I} \quad (2.15)$$

En utilisant l'équation 2.13 dans 2.15, on trouve l'équation finale représentant l'accélération linéaire de l'hélicoptère par rapport au sol dans le système d'axe de l'hélicoptère.

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = m * \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.16)$$

$$\begin{aligned} \dot{u} &= \frac{F_x}{m} + rv - qw \\ \dot{v} &= \frac{F_y}{m} - ru + pw \\ \dot{w} &= \frac{F_z}{m} + qu - pv \end{aligned} \quad (2.17) \text{ (équation II)}$$

Les équations 2.16 représentent l'accélération linéaire de l'hélicoptère par rapport au sol soumis à des forces  $F_x$ ,  $F_y$  et  $F_z$ . Ces forces ont comme référentiel l'hélicoptère. Les équations 2.17 sont utilisés directement dans le modèle SIMULINK.

### 2.3.2 Accélération angulaire

En utilisant la 2<sup>ième</sup> loi de Newton, il est possible de trouver l'accélération angulaire de l'hélicoptère par rapport au sol.

$$\begin{aligned} \sum {}^1\bar{M}_{0,I} &= {}^1R_0 \frac{d}{}({}^0\bar{h}_{0,I}) = {}^1R_0 \frac{d}{}({}^0R_1 {}^1\bar{h}_{0,I}) = {}^1R_0 \left( {}^0R_1 {}^1\dot{h}_{0,I} + {}^0\dot{R}_1 {}^1\bar{h}_{0,I} \right) \\ &= {}^1R_0 \left( {}^0R_1 {}^1\dot{h}_{0,I} + {}^0\bar{\omega}_{0,I} \times {}^0R_1 {}^1\bar{h}_{0,I} \right) = {}^1\dot{h}_{0,I} + {}^1\bar{\omega}_{0,I} \times {}^1\bar{h}_{0,I} \end{aligned} \quad (2.18)$$

où  ${}^1\bar{h}_{0,I}$  est le moment cinétique de l'hélicoptère par rapport au sol représenté dans le système d'axe de l'hélicoptère. Il est fonction de l'inertie et de la vitesse de rotation.

$${}^1\bar{h}_{0,I} = I {}^1\bar{\omega}_{0,I} \quad (2.19)$$

$$\frac{d}{} {}^1\bar{h}_{0,I} = \frac{dI}{} {}^1\bar{\omega}_{0,I} + I * \frac{d}{} {}^1\bar{\omega}_{0,I} \quad (2.20)$$

où  $I$  est la matrice d'inertie.

$$I = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{xy} & I_y & -I_{yz} \\ -I_{xz} & -I_{yz} & I_z \end{bmatrix} \quad (2.21)$$

$I_x$ ,  $I_y$  et  $I_z$  sont les inerties de masse autour de l'axe x, y et z de l'hélicoptère.  $I_{xy}$ ,  $I_{yz}$  et  $I_{xz}$  sont les produits d'inertie de masse.

Il y a deux points qui permettent de simplifier le modèle. Premièrement, si l'hélicoptère avait un seul rotor, il faudrait tenir compte de l'effet gyroscopique induit par celui-ci, mais dans notre cas, il y a deux rotors qui sont contre-rotatifs, l'effet gyroscopique est donc annulé, on n'a donc pas à en tenir compte. Deuxièmement, les

pales vont toujours se croiser au même endroit puisqu'il y a un couplage mécanique entre les deux rotors et la nacelle. Si, au départ, on oriente les pales par rapport à la nacelle en ligne avec l'axe des x ou des y de la nacelle, il y aura symétrie dans le plan x-z et y-z. Une symétrie dans le plan x-z élimine les produits d'inertie  $I_{xy}$  et  $I_{yz}$  et une symétrie dans le plan y-z élimine les produits d'inertie  $I_{xy}$  et  $I_{xz}$ . Il ne reste donc plus de produit d'inertie à tenir compte.

En utilisant les équations 2.18 à 2.21 et les simplifications que l'on vient de mentionner, il est possible de trouver les équations de dynamique angulaire.

$$\sum {}^1 \bar{M}_{0,I} = I' \ddot{\omega}_{0,I} + I' \dot{\bar{\omega}}_{0,I} + {}^1 \bar{\omega}_{0,I} \times I' \bar{\omega}_{0,I} \quad (2.22)$$

$$\sum {}^1 \bar{M}_{0,I} = \begin{bmatrix} L \\ M \\ N \end{bmatrix} \quad (2.23)$$

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = \begin{bmatrix} \dot{I}_x & 0 & 0 \\ 0 & \dot{I}_y & 0 \\ 0 & 0 & \dot{I}_z \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.24)$$

$$\begin{aligned} \dot{p} &= \frac{1}{I_x} (L - \dot{I}_x p + rq(I_y - I_z)) \\ \dot{q} &= \frac{1}{I_y} (M - \dot{I}_y q + pr(I_z - I_x)) \\ \dot{r} &= \frac{1}{I_z} (N - \dot{I}_z r + pq(I_x - I_y)) \end{aligned} \quad (2.25) \text{ (équation III)}$$

Les équations 2.25 représentent la dynamique angulaire de l'hélicoptère par rapport au sol lorsque celui-ci est soumis à trois moments L, M et N. Ces moments ont

comme référentiel l'hélicoptère. Ces équations sont utilisées dans SIMULINK pour représenter l'accélération angulaire de l'hélicoptère.

Pour compléter le modèle mathématiques, il reste à connaître l'orientation de l'hélicoptère par rapport au sol. On y arrive facilement en inversant l'équation 2.10.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & i\theta s\phi & i\theta c\phi \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.26) \text{ (équation IV)}$$

En utilisant les équations **I**, **II**, **III** et **IV**, le modèle peut être construit en utilisant SIMULINK. Il reste à trouver les forces appliquées sur l'hélicoptère et l'inertie de celui-ci.

## 2.4 Inerties

### 2.4.1 Inertie des pales

Pour faire le calcul des inerties, il est nécessaire de positionner les systèmes d'axes de la nacelle et des pales. Chaque pale aura un système d'axe qui sera positionné au bout de l'arbre (centre du rotor), le  $x_p$  pointera vers l'extrémité, le  $y_p$  vers la gauche et le  $z_p$  vers le haut comme montré à la figure 2.2.

Au départ, le système d'axe de l'hélicoptère est orienté exactement comme les systèmes d'axes des pales 1 et 3. Les systèmes d'axes des pales tournent autour de l'axe  $z$  de l'hélicoptère. Les pales 1 et 2 tournent dans le sens anti-horaire par rapport à la

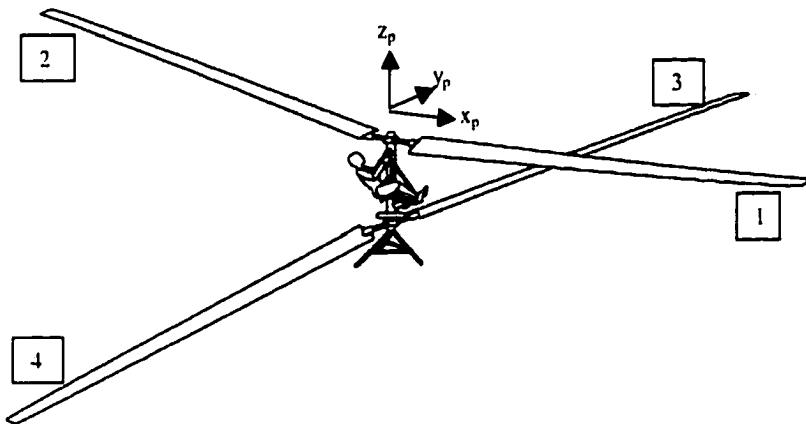


Figure 2.2 Système d'axe d'une pale

nacelle alors que les pales 3 et 4 tournent dans le sens horaire. L'angle qu'il y a entre le système d'axe de l'hélicoptère et le système d'axe de la pale 1 se nommera  $\gamma$ , et l'angle entre l'hélicoptère et la pale 3 sera  $-\gamma$ . La différence en z entre le système d'axe de l'hélicoptère et les systèmes d'axes des pales 1 et 2 s'appellera  $h_1$  et par rapport au pales 3 et 4,  $h_2$ .

On doit maintenant trouver les inerties par rapport aux référentiels des pales qu'on a positionnés par rapport à la nacelle. Une solution pour trouver l'inertie serait de modéliser les pales comme un volume rectangulaire, mais il serait difficile de s'assurer que les résultats représentent la réalité. Une solution plus précise est d'utiliser un système de dessin assisté par ordinateur qui peut trouver les inerties du volume qui est dessiné. La pale avait déjà été dessinée dans le logiciel d'élément finis ANSYS, il y donc été facile de trouver les inerties par rapport au référentiel qu'on a décidé plus tôt. Les résultats fournis par ce logiciel sont:

$$\begin{aligned} I_{px} &= 1,73 \text{ kg} \cdot \text{m}^2 \\ I_{py} &= 1139,3 \text{ kg} \cdot \text{m}^2 \\ I_{pz} &= 1140,8 \text{ kg} \cdot \text{m}^2 \end{aligned} \tag{2.27}$$

Il reste seulement à calculer les inerties par rapport au référentiel de l'hélicoptère, il faut donc considérer que les pales tournent et qu'elles ne sont pas à la même hauteur. En utilisant le théorème des axes parallèles, il sera possible de tenir compte de la différence de hauteur, mais il faut que les systèmes d'axes soient orientés dans la même direction pour utiliser ce théorème. Puisque les pales font seulement une rotation autour de l'axe des z de l'hélicoptère, il suffit de faire une rotation autour de z d'un angle  $\gamma$ .

$$Iph = rot_z(\gamma)Ip = \begin{bmatrix} c\gamma & -s\gamma & 0 \\ s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Ipx \\ Ipy \\ Ipz \end{bmatrix} \quad (2.28)$$

Après que la rotation a été faite, il est possible d'utiliser le théorème des axes parallèles.

$$\begin{aligned} Ih_x &= Iph_x + m(y^2 + z^2) \\ Ih_y &= Iph_y + m(z^2 + x^2) \\ Ih_z &= Iph_z + m(x^2 + y^2) \end{aligned} \quad (2.29)$$

Puisque les systèmes d'axes ont seulement une translation en z de  $h_1$  ou  $h_2$ , x et y valent donc 0 dans les équations 2.29. En combinant les équations 2.28 et 2.29, on obtient l'inertie des pales par rapport à l'hélicoptère.

$$\begin{aligned} Ih_x &= abs(Ip_x c\gamma - Ip_y s\gamma + mh^2) \\ Ih_y &= abs(Ip_x s\gamma + Ip_y c\gamma + mh^2) \\ Ih_z &= Ip_z \end{aligned} \quad (2.30)$$

où h vaut  $h_1$  pour les pales 1 et 2 et  $h_2$  pour les pales 3 et 4. Les valeurs absolues assurent que l'inertie sera toujours positive. L'angle  $\gamma$  est calculé entre la pale 1 et

l'hélicoptère, pour la pale 2 on utilise  $\gamma + \pi$ , pour la pale 3, on utilise  $-\gamma$  et pour la pale 4 on utilise  $-\gamma - \pi$ .

#### 2.4.2 Inertie de la nacelle et du pédaleur

Étant donné l'envergure des pales, il est évident que leur inertie sera beaucoup plus importante que celle de la nacelle et du pédaleur. Il sera donc acceptable de faire une approximation de l'inertie de l'ensemble nacelle-pédaleur. On modélisera cet ensemble par un volume rectangulaire ayant 1 mètre de long, 0.3 mètre de large et 0.5 mètre de haut, ce qui correspond environ à l'espace occupé par le pédaleur. Le centre du volume est au centre de gravité de l'hélicoptère. L'axe des x est dans la direction de la longueur, l'axe des y vers la gauche et l'axe des z vers le haut. Le calcul des inerties se fait avec les équations suivantes.

$$\begin{aligned} Inac_x &= \frac{1}{12}m(largeur^2 + hauteur^2) \\ Inac_y &= \frac{1}{12}m(hauteur^2 + longueur^2) \\ Inac_z &= \frac{1}{12}m(largeur^2 + longueur^2) \end{aligned} \quad (2.31)$$

En considérant que la masse du pédaleur est d'environ 70 kg et que la nacelle aura un masse de 20 kg, on trouve les inerties suivantes :

$$\begin{aligned} Inac_x &= 2.55 \text{ kg} \cdot \text{m}^2 \\ Inac_y &= 9.375 \text{ kg} \cdot \text{m}^2 \\ Inac_z &= 8.175 \text{ kg} \cdot \text{m}^2 \end{aligned} \quad (2.32)$$

### 2.4.3 Inertie de l'hélicoptère

Pour calculer l'inertie totale autour du centre de gravité de l'hélicoptère, il faut additionner l'inertie des quatre pales et de l'ensemble nacelle-pédaleur. Puisque l'inertie de la pale 1 sera pareille à l'inertie de la pale 2 et que l'inertie de la 3 sera pareille à l'inertie de la 4, on fera seulement les calculs des pales 1 et 3.

$$\begin{aligned} I_x &= 2 * Ih1_x + 2 * Ih3_x + Inac_x \\ I_y &= 2 * Ih1_y + 2 * Ih3_y + Inac_y \\ I_z &= 2 * Ih1_z + 2 * Ih3_z + Inac_z \end{aligned} \quad (2.33)$$

Les équations 2.33 servent à calculer les inerties nécessaires aux équations de dynamique en rotation. L'inertie de l'hélicoptère est fonction de  $\gamma$ , mais on a besoin de la dérivée de l'inertie par rapport au temps pour les équations de dynamique en rotation. On a donc besoin de  $\dot{\gamma}$  (angle entre la pale 1 et l'hélicoptère) en fonction du temps.

$$\frac{d\gamma}{dt} = \omega_{pale} \quad (2.34)$$

où  $\omega_{pale}$  est la vitesse de rotation des pales par rapport à la nacelle. Il reste à calculer la dérivée de l'inertie.

$$\begin{aligned} \dot{I}_x &= 2 * abs(-\omega(Ip_x \sin \gamma + Ip_y \cos \gamma)) + 2 * abs(-\omega(Ip_x \sin \gamma - Ip_y \cos \gamma)) \\ \dot{I}_y &= 2 * abs(-\omega(-Ip_x \cos \gamma + Ip_y \sin \gamma)) + 2 * abs(-\omega(Ip_x \cos \gamma - Ip_y \sin \gamma)) \\ \dot{I}_z &= 0 \end{aligned} \quad (2.35)$$

## 2.5 Forces

### 2.5.1 Gravité

La première force qui s'appliquera sur l'hélicoptère sera la gravité. Cette force a comme référentiel le sol, il faudra donc la transformer pour l'exprimer dans le référentiel de l'hélicoptère. Elle est dans la direction du z+ du système d'axe du sol.

$${}^0\bar{F}_g = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (2.36)$$

$${}^1\bar{F}_g = {}^1R_0 {}^0\bar{F}_g = mg \begin{bmatrix} -s\theta \\ c\theta s\phi \\ c\theta c\phi \end{bmatrix} \quad (2.37)$$

### 2.5.2 Forces aérodynamiques

Comme pour les inerties, il faut positionner un système d'axe pour faire le calcul des forces aérodynamiques. Ce nouveau système d'axe a la même orientation que celui de l'inertie pour les pales du haut (x vers le bout de pale, y à gauche (bord d'attaque) et z vers le haut) mais il est positionné au centre de pression. Pour les pales du bas, le système d'axe est positionné aussi au centre de pression mais son orientation diffère (x vers le bout de pale, y à droite (bord d'attaque) et z vers le bas). L'équation suivante montre les transformations pour passer du système d'axe 2 des pales du haut (centre de pression des pales) au système d'axe 1 (hélicoptère).

$${}^1T_2 = rot_z(\gamma)rot_x(\alpha)trans(xp, yp, zp) \quad (2.38)$$

$${}^1T_2 = \begin{bmatrix} c\gamma & -s\gamma\alpha & s\gamma\alpha & c\gamma \cdot xp - s\gamma\alpha \cdot yp + s\gamma\alpha \cdot zp \\ s\gamma & c\gamma\alpha & -c\gamma\alpha & s\gamma \cdot xp + c\gamma\alpha \cdot yp - c\gamma\alpha \cdot zp \\ 0 & s\alpha & c\alpha & s\alpha \cdot yp + c\alpha \cdot zp \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.39)$$

où  $\gamma$  est l'angle qu'a la pale 1 par rapport à la nacelle,  $\alpha$  est l'angle qu'a la pale par rapport à l'horizontale et  $x_p$ ,  $y_p$  et  $z_p$  est la position du centre de pression par rapport à la nacelle. Pour les pales du bas, seul la rotation autour de x change.

$${}^1T_2 = rot_z(\gamma)rot_x(\pi - \alpha)trans(xp, yp, zp) \quad (2.40)$$

$${}^1T_2 = \begin{bmatrix} c\gamma & s\gamma\alpha & s\gamma\alpha & c\gamma \cdot xp + s\gamma\alpha \cdot yp + s\gamma\alpha \cdot zp \\ s\gamma & -c\gamma\alpha & -c\gamma\alpha & s\gamma \cdot xp - c\gamma\alpha \cdot yp - c\gamma\alpha \cdot zp \\ 0 & s\alpha & -c\alpha & s\alpha \cdot yp - c\alpha \cdot zp \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.41)$$

Ce que l'on cherche est l'angle que le vent fait par rapport à la ligne de corde. Comme le montre l'équation suivante, on doit donc considérer la vitesse de l'hélicoptère et des pales ainsi que le vent terrestre.

$${}^w\bar{v}_t = {}^w\bar{v}_w + {}^0\bar{v}_2 \quad (2.42)$$

où le vent terrestre et la vitesse du système d'axe 2 peuvent être calculés avec les équations suivantes.

$${}^0\bar{v}_w = \begin{bmatrix} {}^0v_{wx} \\ {}^0v_{wy} \\ 0 \end{bmatrix} \quad (2.43)$$

$${}^0\bar{v}_2 = {}^0\bar{v}_1 + {}^0\bar{\omega}_1 \times ({}^0R_1^{-1} \bar{p}_2) + {}^0R_1 \cdot ({}^1\bar{\omega}_2 \times {}^1\bar{p}_2) \quad (2.44)$$

On peut utiliser les variables du système dans l'équation 2.44.

$$\begin{aligned} {}^0\bar{v}_2 &= {}^0R_1 \begin{bmatrix} u \\ v \\ w \end{bmatrix} + {}^0\bar{\omega}_1 \times \left( {}^0R_1 \begin{bmatrix} c\gamma \cdot xp - s\gamma c\alpha \cdot yp + s\gamma s\alpha \cdot zp \\ s\gamma \cdot xp + c\gamma c\alpha \cdot yp - c\gamma s\alpha \cdot zp \\ s\alpha \cdot yp + c\alpha \cdot zp \end{bmatrix} \right) + \dots \\ &\dots {}^0R_1 \left( \begin{bmatrix} 0 \\ 0 \\ \dot{\gamma} \end{bmatrix} \times \begin{bmatrix} c\gamma \cdot xp - s\gamma c\alpha \cdot yp + s\gamma s\alpha \cdot zp \\ s\gamma \cdot xp + c\gamma c\alpha \cdot yp - c\gamma s\alpha \cdot zp \\ s\alpha \cdot yp + c\alpha \cdot zp \end{bmatrix} \right) \end{aligned} \quad (2.45)$$

Étant donné que l'on cherche l'angle d'attaque du vent, il est nécessaire d'avoir le vent par rapport au système d'axe 2.

$${}^2\bar{v}_t = {}^2R_0 {}^0\bar{v}_t \quad (2.46)$$

Il est maintenant possible de trouver l'angle d'attaque du vent, les résultats seront différents pour les pales du haut et du bas.

$$\begin{aligned} (\alpha + \alpha_i)_{pale \ du \ haut} &= \tan^{-1} \frac{(-{}^2v_z)}{({}^2v_y)} \\ (\alpha + \alpha_i)_{pale \ du \ bas} &= \tan^{-1} \frac{(-{}^2v_z)}{(-{}^2v_y)} \end{aligned} \quad (2.47)$$

La figure 2.3 représente un élément de la pale où le vent est représenté par  $V_R$ .

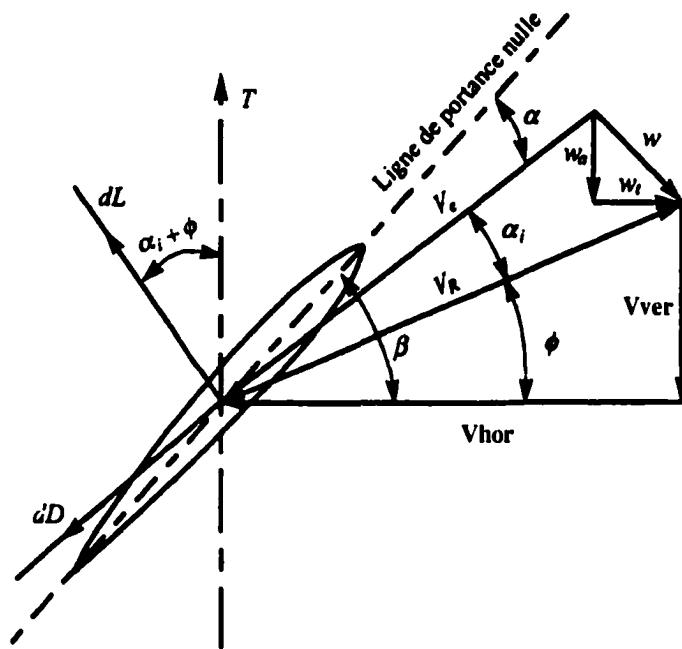


Figure 2.3 Élément de la pale

Pour connaître la poussée aérodynamique et la traînée aérodynamique, on doit connaître l'angle d'attaque du profil ( $\alpha$ ). Le résultat de l'équation 2.47 donne l'angle d'attaque ainsi que l'angle d'attaque induit, il faut donc calculer la valeur de l'angle d'attaque induit pour pouvoir connaître l'angle d'attaque. L'angle  $\beta$  représente l'angle que fait la ligne de portance nulle par rapport au plan de rotation du profil. Il est donc possible de déterminer  $\phi$ .

$$\phi = \beta - (\alpha + \alpha_i) \quad (2.48)$$

Il est possible de calculer  $V_R$  à l'aide des composantes  $v_{ry}$  et  $v_{rz}$  de l'équation 2.47.

$$V_R = \sqrt{v_{ry}^2 + v_{rz}^2} \quad (2.49)$$

Par trigonométrie, on peut trouver les composantes  $V_{ver}$  et  $V_{hor}$ .

$$\begin{aligned} V_{ver} &= Vr * \sin(\phi) \\ V_{hor} &= Vr * \cos(\phi) \end{aligned} \quad (2.50)$$

À ce point, il est possible de calculer  $w_a$ ,  $w_t$  et  $V_e$  puisqu'on a les composantes  $V_{ver}$  et  $V_{hor}$ . Le mémoire de Simon Joncas « Conception et fabrication d'une aile rotative basse vitesse pour un hélicoptère à propulsion humaine » explique complètement le calcul de  $w_a$ ,  $w_t$  et  $V_e$  hors point de conception. Le calcul de  $\alpha_i$  se fait avec l'équation suivante.

$$\alpha_i = \frac{\sqrt{w_a^2 + w_t^2}}{V_e} \quad (2.51)$$

Puisque la pale possède un vrillage, il faut faire ces calculs pour chaque élément de la pale car l'angle  $\beta$  varie selon le rayon où l'élément est positionné. Il est donc possible de connaître l'angle d'attaque ( $\alpha$ ) pour chaque élément, ce qui nous permet de trouver les coefficients de portance ( $C_l$ ) et les coefficients de traînée ( $C_d$ ). Pour terminer, il reste à calculer la poussée aérodynamique et la traînée aérodynamique, ce qui se fait en intégrant les  $dL$  et le  $dD$  de chaque élément.

Les programmes Matlab qui ont été nécessaires pour faire les calculs symboliques sont en annexe.

## 2.6 Programmation du modèle

Les équations du modèle sont non-linéaires, il sera donc nécessaire de trouver les résultats avec des résolutions numériques. La façon la plus simple d'implanter ce type de calcul est d'utiliser SIMULINK. Ce logiciel permet de faire les liens entre les quatre systèmes d'équations (cinématique linéaire et angulaire et dynamique linéaire et angulaire) et les forces aérodynamiques. Par contre, SIMULINK ne permet pas d'écrire facilement les équations qu'on a trouvées plus tôt, on fera donc appelle à MATLAB pour les écrire. Lorsque tous les blocs sont reliés entre eux, il reste six entrées à combler qui sont les forces en x, y et z et les moments autour de x, y et z appliqués sur l'hélicoptère. C'est un autre programme MATLAB qui va calculer ces forces et moments en tenant compte de la position et de la vitesse des pales ainsi que la vitesse de l'hélicoptère lors de la simulation. Les programmes MATLAB sont en annexe. On peut voir à la figure 2.3, le modèle dynamique en SIMULINK.

Chaque bloc dans le modèle contient un schéma ressemblant à la figure 2.4. Ce schéma permet de créer un vecteur pour passer les arguments au programme MATLAB et de séparer le vecteur de sortie.

Le modèle doit limiter le couple fourni par le pédaleur car les perturbations du système peuvent entraînées une augmentation du couple, ce qui ne représenterait pas la réalité puisque le pédaleur a une puissance limitée. La limite de couple agit sur la vitesse de rotation des hélices, si le couple est trop bas, la vitesse de rotation augmente et si le couple est trop haut, la vitesse de rotation diminue, ce qui représente bien la réalité. Pour être encore plus près du comportement réel, il faudrait limiter la vitesse de rotation minimum et maximum des hélices car le pédaleur a

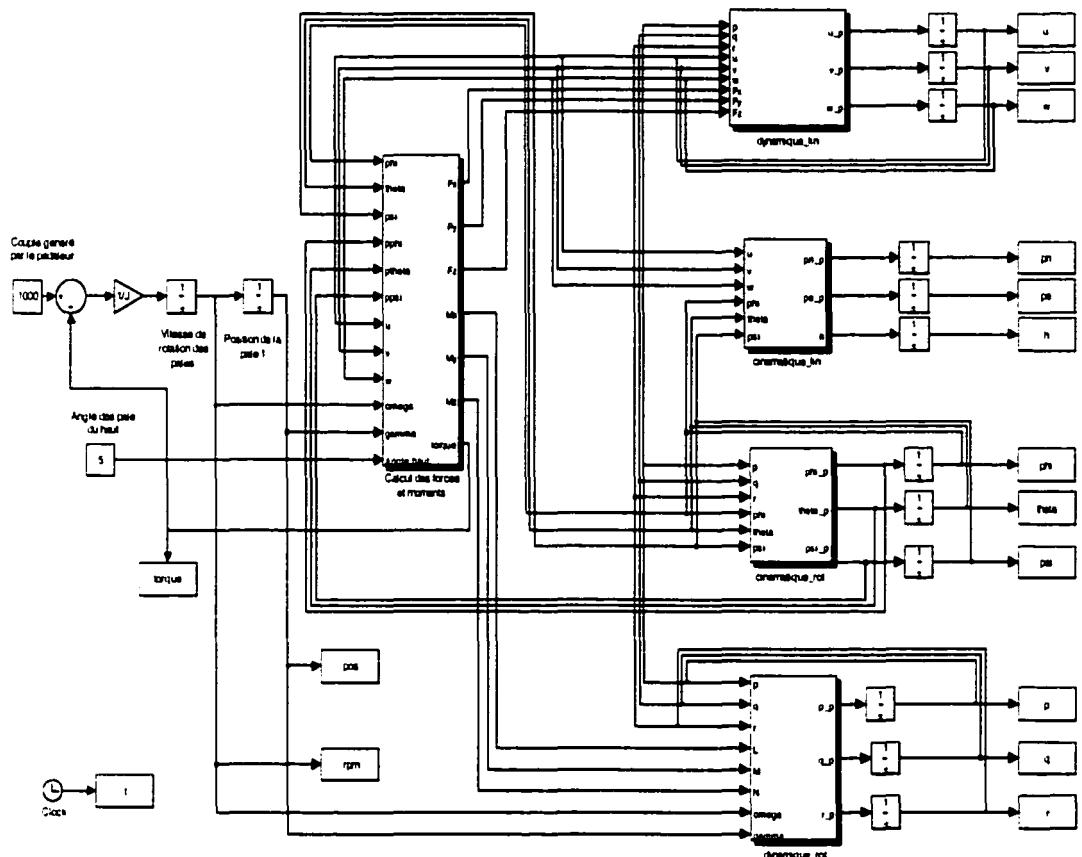


Figure 2.4 Modèle dynamique en SIMULINK

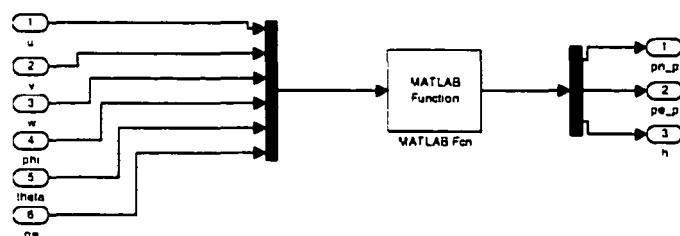


Figure 2.5 Exemple de l'appel d'une fonction MATLAB

une limite quant à la vitesse de rotation qu'il peut pédaler et à la force qu'il peut générer sur les pédales. Mais en considérant de petites perturbations, le modèle n'atteindra pas ces limites.

## 2.7 Modification du programme de calcul aérodynamique

Lors de la programmation du programme d'aérodynamique, il était nécessaire de faire une fonction qui permettait de calculer la poussée et la traînée aérodynamique lorsque la pale fonctionnait hors de ses paramètres de conception. Dans cette fonction, on doit trouver la valeur du  $C_l$  (coefficients de poussée), la valeur du  $C_d$  (coefficients de traînée) et l'angle d'attaque induit. Le problème est que le calcul du  $C_l$  et du  $C_d$  a besoin de l'angle d'attaque induit et l'angle d'attaque induit a besoin du  $C_l$  et du  $C_d$ . Voici l'algorithme de cette première version du programme.

fonction hors\_conception

```

angle1=0 ;
angle2=1 ;

tant que (angle2-angle1) > tolérance
    angle1 = angle2 ;
    [Cl,Cd] = Trouver_Cl_Cd(angle1) ;
    angle2 = Trouver_angle_induit(Cl,Cd)
fin

```

Ce programme montre seulement le calcul pour un élément de la pale, il doit être répété pour tous les éléments. Il est à noter que le programme appelé Trouver\_angle\_induit trouve la réponse de deux fonctions par résolution numérique, on a donc deux équations numériques imbriquées. Pour une pale contenant 21 éléments, ce

programme prend 8.33 secondes à s'exécuter sur un ordinateur donné. Dans notre modèle dynamique, on doit appeler quatre fois ce programme à chaque itération, ce qui fait environ 33 secondes de calcul par itération seulement pour ce programme. Il est donc impensable de l'utiliser, il faut le modifier pour diminuer beaucoup le temps de calcul nécessaire.

La modification est assez simple, on a seulement à calculer le  $C_l$  et le  $C_d$  en même temps que l'angle induit dans le programme Trouver\_angle\_induit et faire converger ce programme seulement sur l'angle induit. Le nouveau programme prend 0.39 seconde à s'exécuter sur le même ordinateur, ce qui rend le programme 21 fois plus rapide. La version finale de ce programme est en annexe.

## 2.8 Analyse des résultats

Dans un premier temps, il est important de s'assurer que le modèle de l'hélicoptère est stable lorsqu'il y a aucune perturbation. Par la suite, deux situations sont intéressantes à modéliser, lorsqu'une des pales a un angle supérieur et lorsque l'hélicoptère n'est pas à l'horizontale.

Dans la première situation, on veut savoir si l'hélicoptère reste stable même si une des poussées aérodynamiques est plus importantes que les autres. Il est évident qu'il y aura oscillation sans arrêt de la nacelle, mais il est important que l'amplitude de ces oscillations n'augmente pas avec le temps. Dans la deuxième situation, on veut savoir si l'hélicoptère peut revenir à l'horizontale par lui-même lorsqu'il est déséquilibré.

### 2.8.1 Force aérodynamique différente sur une des pales

Dans cette simulation, toutes les pales ont un angle d'attaque de cinq degrés sauf la pale 1 qui a un angle de 6,5 degrés. Le couple sera limité à 1000 Nm et on ne doit pas

dépasser une puissance de 740 Watts. Même si les pales ont été conçues pour un puissance de 600 Watts, le pédaleur est capable de fournir 740 Watts pendant 30 secondes. Le modèle actuel ne pouvant tenir compte de la perte de puissance du pédaleur dans le temps, on considérera que le pédaleur peut fournir une puissance de 740 Watts durant 1 minute. Les figures 2.6 et 2.7 montrent les résultats de la simulation.

En utilisant la figure 2.6, on peut calculer la puissance demandée au pédaleur, qui est de 730 Watts, la simulation est donc valide. Dans la figure 2.7, on peut voir que l'hélicoptère oscille autour de ses axes de roulis et de tangage, comme il était prévisible, mais que l'amplitude de l'oscillation se stabilise après un certain temps. La première oscillation est celle qui a la plus grande amplitude, ce qui est normal car la nacelle n'a aucune vitesse en rotation autour des axes de roulis et de tangage. Par la suite, la nacelle doit être ralentie avant de partir à tourner dans la direction opposée. La position linéaire en x et y se déplace au début lors de la plus grande des oscillations mais par la suite l'hélicoptère oscille autour du même endroit. La hauteur atteinte par l'hélicoptère n'est pas significative, il faudrait que le modèle aérodynamique tienne compte de la position en z de l'hélicoptère, ceci entraînerait une augmentation du couple nécessaire pour conserver la même poussée aérodynamique, ce qui limiterait la hauteur que l'hélicoptère pourrait atteindre. Les pales utilisées dans le modèle ont été conçues à trois mètres du sol, ce qui représente la position la plus élevée que l'hélicoptère doit atteindre.

### 2.8.2 Hélicoptère déséquilibré

Dans cette deuxième simulation, toutes les forces aérodynamiques sont identiques, par contre l'hélicoptère débute avec un angle de 1,5 degré autour de son axe de tangage ( $\theta$ ). Les contraintes en puissance sont les mêmes que pour la première simulation. Les figures 2.8 et 2.9 montrent les résultats de la simulation.

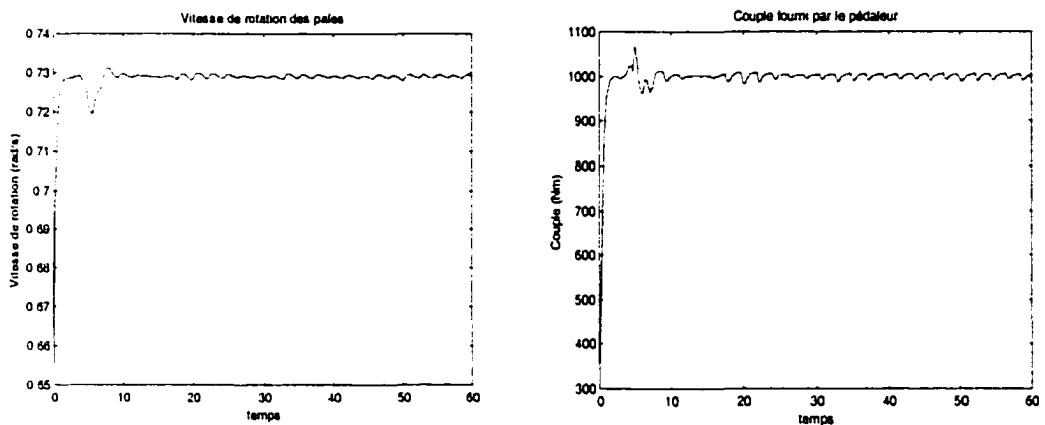


Figure 2.6 Vitesse de rotation des pales et couple fourni par le pédaleur

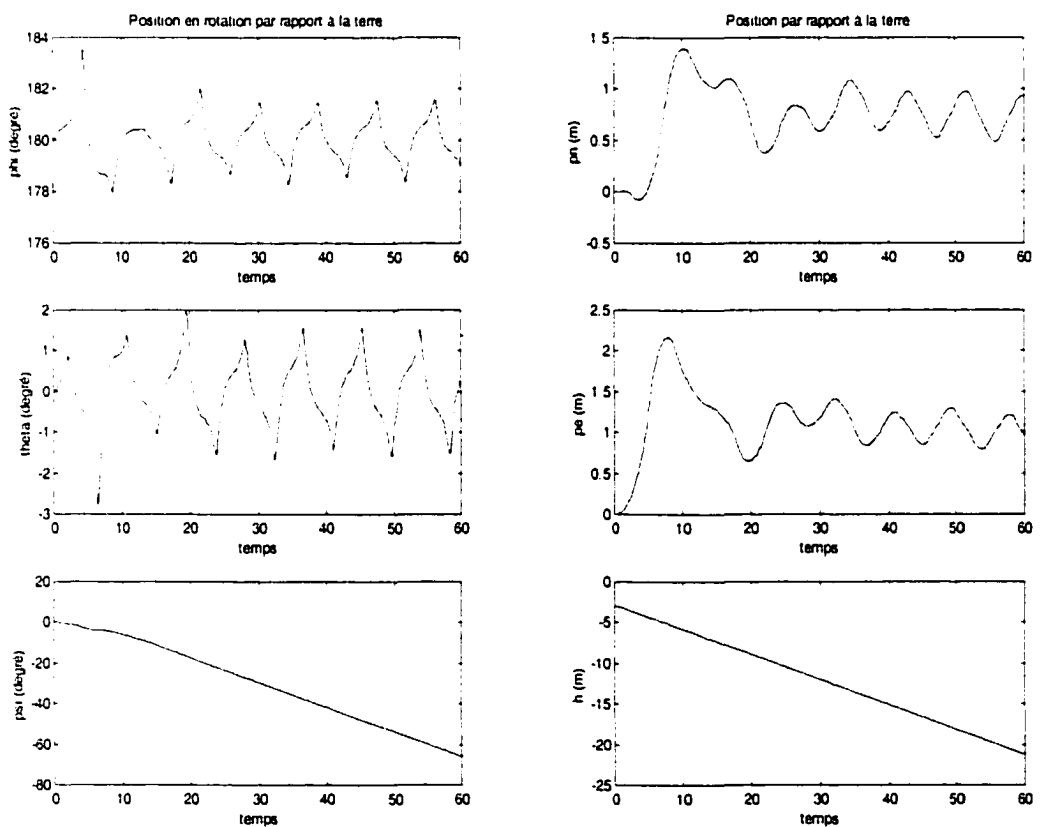


Figure 2.7 Position linéaire et angulaire de l'hélicoptère par rapport au sol

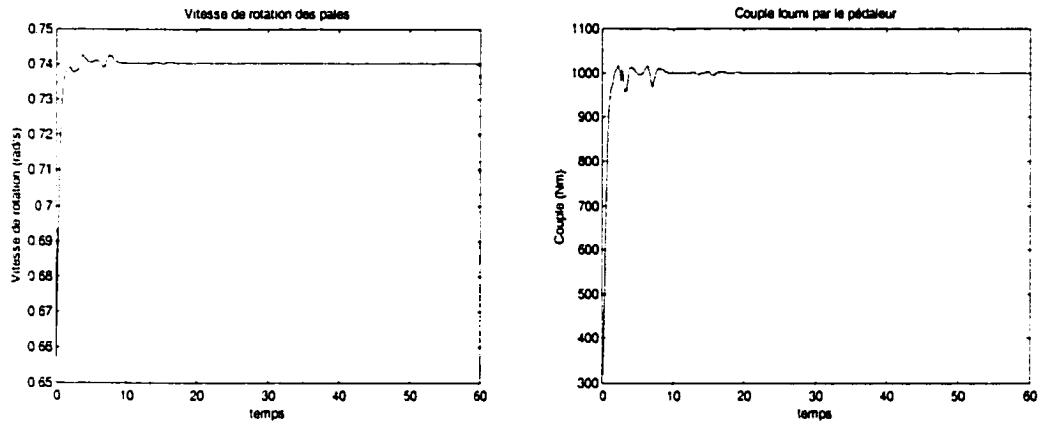


Figure 2.8 Vitesse de rotation des pales et couple fourni par le pédaleur

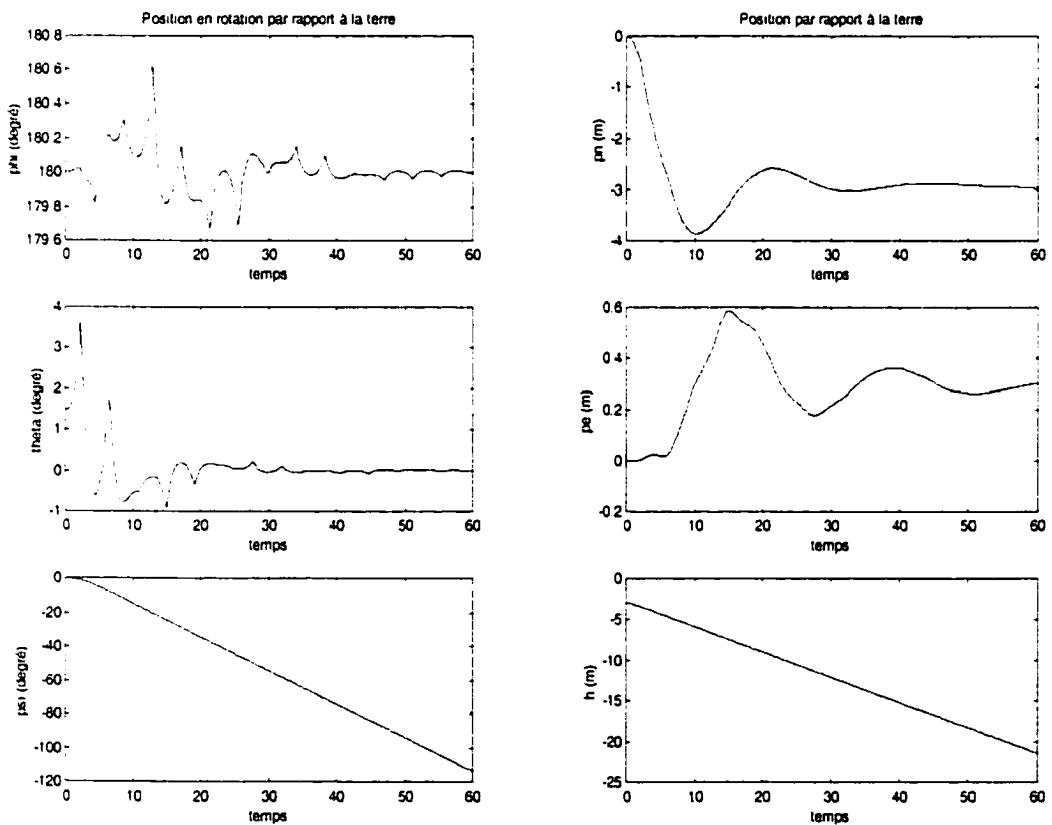


Figure 2.9 Position linéaire et angulaire de l'hélicoptère par rapport au sol

Comme dans la première simulation, la puissance demandée au pédaleur ne dépasse pas 740 Watts, la simulation est donc valide. D'un point de vue de stabilité, les graphiques de la position angulaire autour des axes de roulis et de tangage sont les plus intéressants. On peut voir qu'au départ l'hélicoptère a un angle de 1,5 degré autour de l'axe de tangage. Cet angle fait avancer l'hélicoptère en x assez rapidement puisque la poussée des pales a maintenant une composante horizontale, ce qui modifie les poussées aérodynamiques des quatre pales. Cela va augmenter la poussée générée par les pales 2 et 4 et diminuer la poussée générée par les pales 1 et 3 durant le premier demi-tour, l'hélicoptère va donc pivoter encore plus autour de l'axe de tangage jusqu'à environ 3,5 degrés. Par la suite, l'hélicoptère se stabilise rapidement, après 15 secondes elle oscille d'environ  $\pm 0,2$  degré autour des deux axes. Puisque le modèle ne tient pas compte de la contre-réaction qui permettra aux deux rotors de générer le même couple, la nacelle fait une rotation de presque un tiers de tour, ce qui ne dépasse pas la limite permise par la compétition.

En déplacement linéaire, l'hélicoptère parcourt près de 4 mètres en x, ce qui est proche de la limite permise de 5 mètres. Il revient ensuite se stabiliser à 3 mètres en x et 0,3 mètre en y. On peut voir qu'un faible angle autour de l'axe de roulis ou de tangage entraîne une grande vitesse linéaire, par contre, c'est cette vitesse linéaire qui permet à l'hélicoptère de se repositionner à l'horizontale.

On peut donc considérer que l'hélicoptère se stabilise par lui-même autour des axes de roulis ou de tangage mais qu'il est nécessaire de parcourir une grande distance linéaire pour que l'hélicoptère ait assez de vitesse pour modifier la force aérodynamiques que les pales exercent, et surtout pour que celles-ci soient à la bonne position. La seule limite qu'a cette façon de se stabiliser est que l'hélicoptère ne doit pas dépasser le périmètre de 10 mètres par 10 mètres.

La première simulation utilise trois pales identiques et la deuxième utilise quatre pales identiques, ce qui est impensable à obtenir en réalité étant donné l'envergure des pales. Il sera donc nécessaire d'avoir des systèmes de contrôles électroniques pour pouvoir obtenir des pales qui ont un comportement identique.

## CHAPITRE 3

### SYSTÈME DE CONTRÔLE

Les simulations ont permis de constater que l'hélicoptère est stable si les poussées aérodynamiques de chaque pale sont semblables. On doit maintenant vérifier si l'amplitude des oscillations n'est pas trop élevé et s'il y a un moyen de contrer ces oscillations. Étant donné que les pales sont très grandes ( 17.5 mètres ), une faible erreur autour des axes de roulis ou de tangage mène au contact des bouts de pales avec le sol. On peut calculer l'erreur admissible avec une simple équation trigonométrique :

$$erreur = \sin^{-1} \left( \frac{\text{hauteur de la pale basse}}{17.5} \right) \quad (3.1)$$

On peut voir qu'à 3 mètres, l'erreur possible est moins de 10°, et qu'à 1 mètre, une erreur de plus de 3,3° est fatale. Étant donné que l'énergie demandée au pédaleur est moins grande à basse altitude (< 1 mètre ) à cause de l'effet de sol, il est évident que la majeure partie du vol sera faite à cette hauteur. Puisque les pales se déforment beaucoup ( près de deux mètres lors de la poussée aérodynamique maximum ), le calcul précédent est pessimiste, mais il permet de faire le lien avec les résultats des simulations. Si on considère que nos simulations ont eu lieu à 1 mètre, les pales ne touchent pas au sol sauf lors de la première oscillation de la deuxième simulation. Il est clair que les perturbations des deux simulations ( 1,5 degré de plus d'angle d'attaque dans la première simulation et 1,5 degré d'angle autour de l'axe de tangage ) sont à la limite de permettre un contact entre le sol et les pales.

La grandeur des pales va créer un autre problème, la répétabilité de la fabrication. Il est évident qu'aucune pale ne sera pareille, même s'ils elles ont été faites dans le même moule avec la même procédure de fabrication. De plus, le laminé de carbone est tellement mince qu'une faible erreur lors de la fabrication entraîne des modifications importantes du comportement en vol d'une pale.

### 3.1 Banc d'essai

Il est donc impensable que le système de contrôle ait le temps de détecter qu'une pale a une poussée aérodynamique moins grande que les autres et qu'il augmente l'angle d'attaque de cette pale afin d'obtenir la même poussée que les autres pales, sans jamais dépasser l'erreur admissible autour des axes de roulis ou de tangage. Il faut, dès le départ de l'hélicoptère, que les pales aient la même poussée aérodynamique et le même couple. La solution retenue est inspirée d'un système qui appartient au domaine de l'automobile, l'injection électronique. Le système d'injection électronique d'un moteur à essence possède plusieurs capteurs pour le système de contre-réaction, par contre le moteur est calibré sur un banc d'essai où il est mis dans une situation proche de celle rencontrée lorsqu'il est installé dans l'automobile. À la fin de la période de calibrage, on implante dans l'injection électronique un tableau possédant tous les résultats de la période de test. Les paramètres d'entrée sont la vitesse de rotation du moteur, la pression à l'entrée d'air et la température à l'entrée d'air. Celui qui fait l'ajustement du moteur doit trouver la bonne quantité d'essence à injecter pour des paramètres d'entrée donnés. Pour avoir terminé la période de calibrage, il doit avoir fait toutes les combinaisons possibles de paramètres d'entrée. Lors de l'utilisation dans l'automobile, le système lit les trois capteurs principaux et retrouve la valeur associée à ces paramètres dans les tables. Les autres capteurs du système d'injection électronique permettent seulement de modifier légèrement la valeur lue dans le tableau.

Dans le système de contrôle de l'hélicoptère, il sera possible d'utiliser le même principe. Un banc d'essai, dont les paramètres d'entrée sont la vitesse de rotation du rotor et la hauteur du rotor, permet de tester les pales sans l'hélicoptère. Lors du calibrage des pales, une des pales sera la pale maître alors que l'autre sera l'esclave. Le test débutera en fixant la vitesse de rotation, la hauteur du rotor et l'angle de la pale maître. Par la suite, on fait varier l'angle de la pale esclave jusqu'à ce que le vecteur poussée globale ( poussée de la pale maître + celui de la pale esclave) soit parfaitement verticale. À ce point, on est assuré que les deux pales ont une poussée aérodynamique identique, on conserve donc dans un tableau l'angle de la pale esclave en fonction de la vitesse de rotation, de la hauteur et de l'angle de la pale maître. On sauvegarde aussi le couple total et la poussée aérodynamique générés par le rotor. La figure 3.1 montre la procédure à suivre lors de la période de calibrage.

Dans la figure 3.1, PM signifie pale maître et PE, pale esclave. Tous les chiffres de l'algorithme peuvent être modifiés, si le test le nécessite. Il est aussi important qu'en tout temps la poussée d'une pale ne dépasse pas 120% de la poussée définie lors de la conception, sinon il y aura bris de celle-ci. Il est évident qu'à la fin de la période de calibrage, les deux pales forment un couple et qu'on ne peut pas changer une pale. Dans le cas où une pale fait défaut, on doit la remplacer et retourner sur le banc d'essai pour calibrer le nouveau couple.

### 3.2 Algorithme de contrôle en vol (boucle ouverte)

Puisque les tableaux qui ont été enregistrés lors du calibrage sont en fonction de la vitesse de rotation et de la hauteur du rotor, il faut que l'ordinateur de bord ait les capteurs qui lui donnent ces informations. En vol, l'ordinateur lit constamment les deux capteurs et trouve les angles d'attaques des pales à l'aide des tableaux. Par contre, il ne faut pas oublier qu'une des règles du concours est que la nacelle ne fasse pas plus d'un tour lors du vol, il faut donc que les pales du bas génère le même couple que les pales du

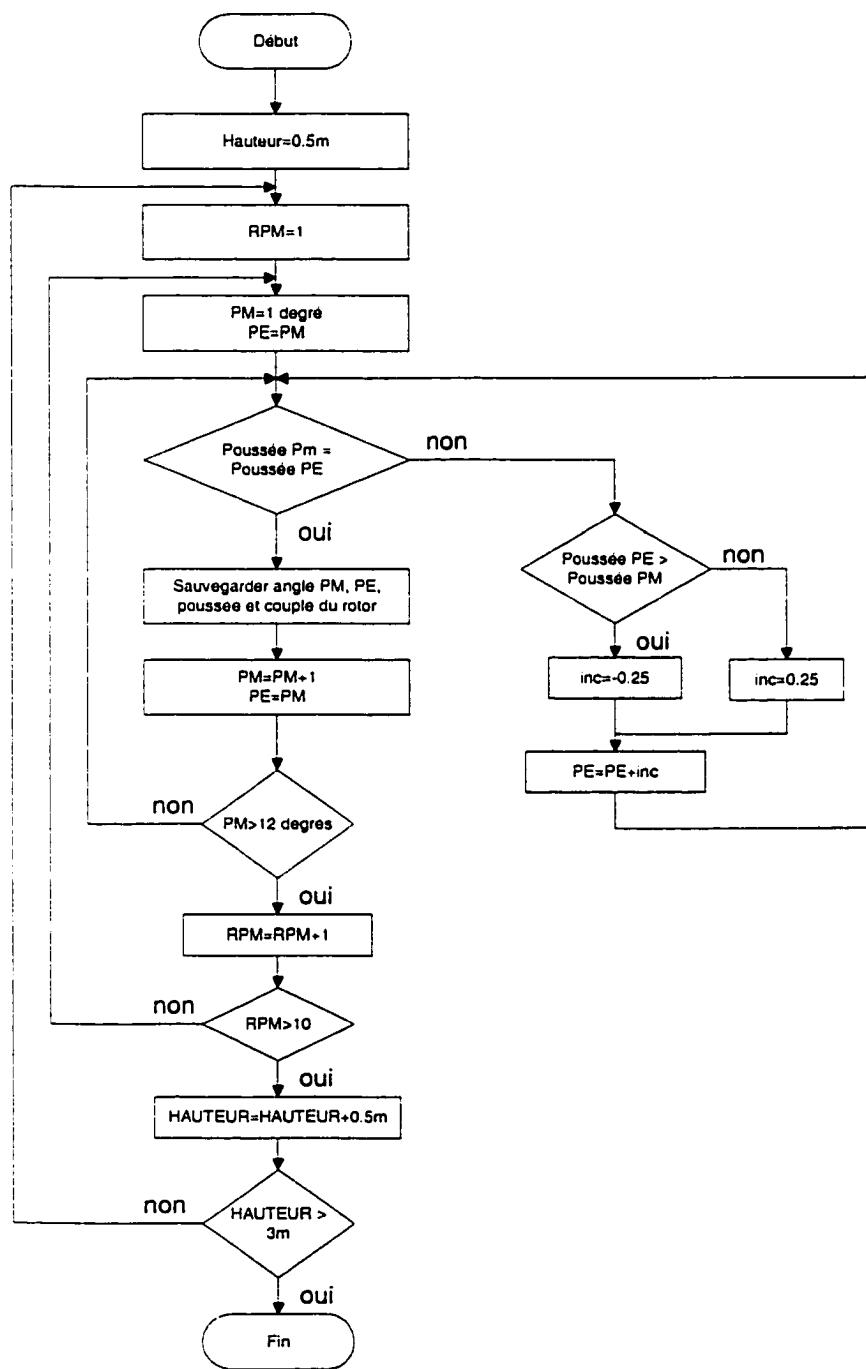


Figure 3.1 Procédure de calibrage d'un rotor

haut. Durant le calibrage des pales, on a conservé le couple généré par celles-ci, il est donc possible de trouver un point où les deux rotors ont le même couple, mais un des deux rotors n'aura plus la poussée aérodynamique optimale pour la vitesse de rotation et la hauteur actuels. Puisque les pales du bas ont toujours une poussée aérodynamique plus importante que les pales du haut, il est plus intéressant de modifier l'angle des pales du haut pour affecter le moins possible la poussée aérodynamique totale. Le rotor du bas sera considéré comme le rotor maître alors que celui du haut sera le rotor esclave. La figure 3.2 présente l'algorithme de contrôle en boucle ouverte utilisé dans l'ordinateur.

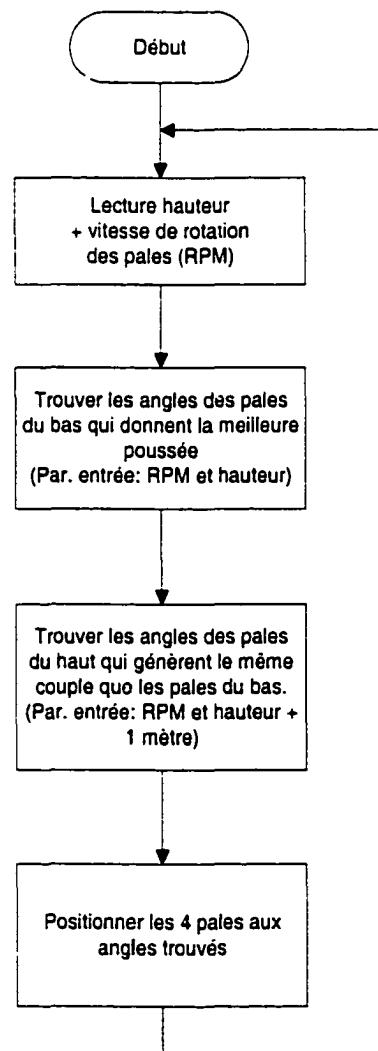


Figure 3.2 Algorithme de contrôle en boucle ouverte

### 3.3 Algorithme de contrôle en vol ( boucle fermée )

Même si les rotors sont balancés, il faut, comme dans le cas de l'injection, faire une boucle de contre-réaction qui éliminera les différences qu'il y a entre les tests sur le banc d'essai et le vol réel. Avant d'implanter une boucle de contre-réaction, il faut s'assurer de pouvoir lire les paramètres d'entrées (angle de roulis, de tangage et de lacet) et de pouvoir agir sur le système avec des actuateurs. Dans le cas des capteurs, la solution la plus simple est d'utiliser un gyroscope à trois axes, par contre c'est une solution coûteuse et la plupart des gyroscopes sont faits pour l'aviation, ils sont donc bien protégés et à cause de cela, ils sont lourds. Ce n'est donc pas une solution envisageable. Il existe maintenant des accéléromètres statiques qui nous permettent de trouver l'angle par rapport à la gravité et ils ont la forme d'un circuit intégré, ils sont donc légers. Ces accéléromètres nous permettent de trouver l'angle de roulis et de tangage mais pas celui de lacet, il faut donc utiliser quand même un gyroscope. Il existe sur le marché des gyroscopes, principalement utilisés dans les manettes de jeux vidéos et dans les souris, qui sont compacts et qui sont assez précis pour notre application. Même s'ils sont vendus comme étant un gyroscope, c'est en réalité un capteur de vitesse en rotation autour d'un axe. Leur défaut est que la position évolue dans le temps à cause de la température dans le module, mais puisque le vol dure seulement une minute, ce défaut nous affecte peu. Maintenant qu'il est possible de lire la rotation autour des trois axes, il faut déterminer les erreurs possibles et comment les corriger.

La première erreur possible est que, malgré les résultats des tableaux, les couples générés par les deux rotors sont différents, la nacelle tourne donc autour de l'axe de lacet. Dès qu'une lecture du gyroscope montre que la nacelle tourne, il faut modifier le couple généré par le rotor esclave, on trouve donc dans le tableau une nouvelle paire d'angle qui va augmenter ou diminuer le couple produit par le rotor esclave. La figure 3.3 montre le modèle dynamique modifié pour tenir compte de cette contre-réaction. La

vitesse de rotation de la nacelle est intégré et multiplié par une constante, ce qui donne le nouvel angle des pales du haut.

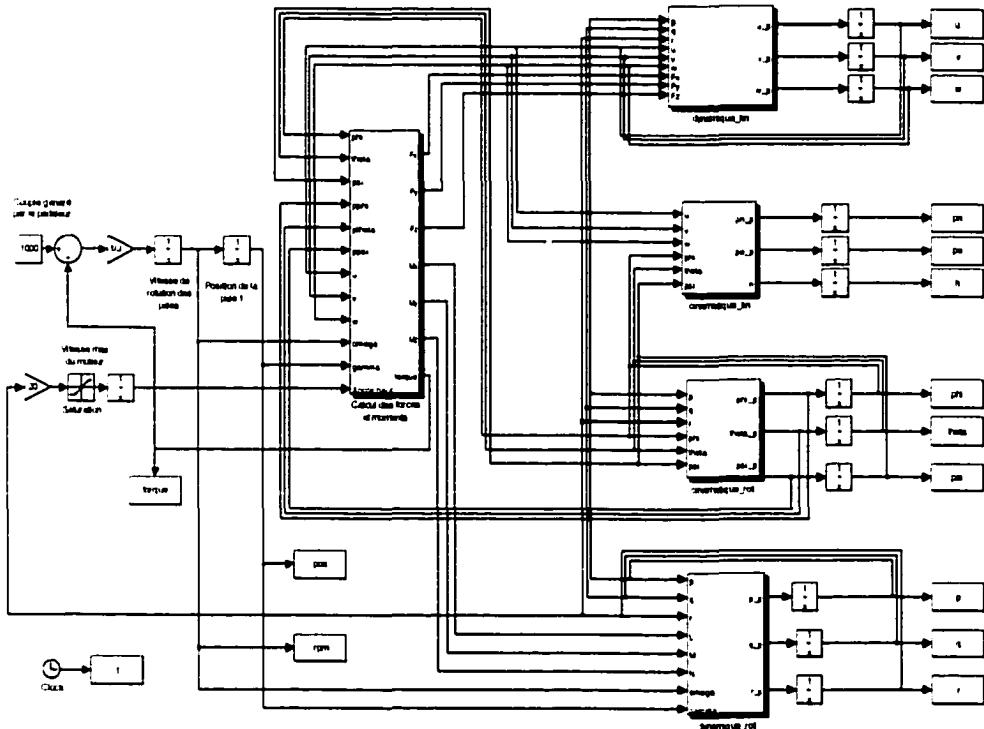


Figure 3.3 Modèle dynamique avec contre-réaction sur la vitesse de rotation de la nacelle

Dans la prochaine simulation, on veut savoir jusqu'à quel angle la nacelle va pivoter en utilisant une boucle de contre-réaction sur la vitesse de rotation de la nacelle. Avec cette boucle, on ne cherche pas à garder la nacelle à la même position du début jusqu'à la fin du vol, mais plutôt de réduire sa vitesse de rotation et ainsi rester réglementaire quant à la rotation maximale d'un tour de la nacelle durant le vol. Les figures 3.4 et 3.5 montrent les résultats de la simulation.

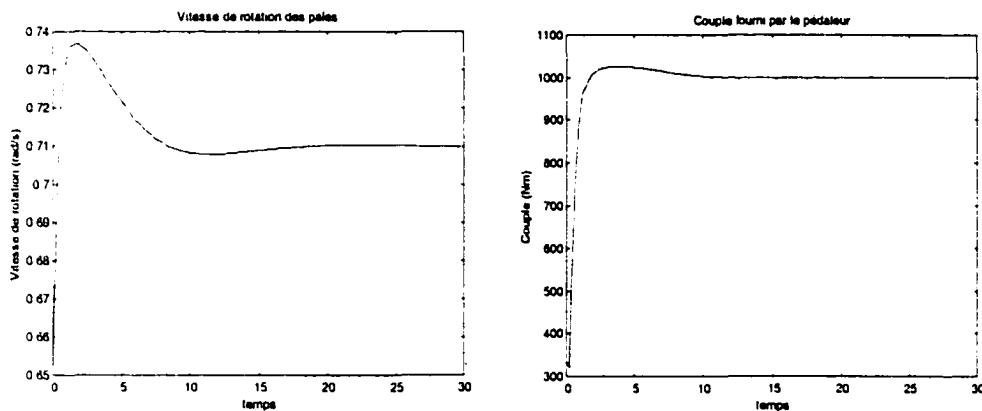


Figure 3.4 Vitesse de rotation des pales et couple fourni par le pédaleur

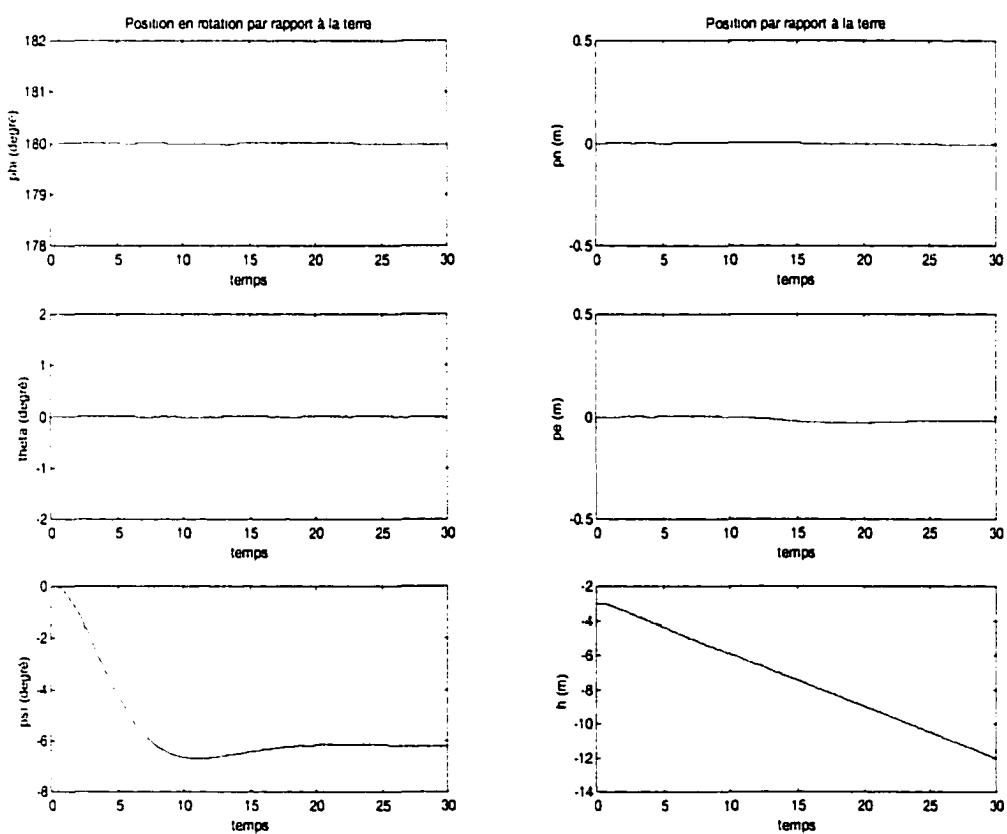


Figure 3.5 Position angulaire et linéaire de l'hélicoptère par rapport au sol

Le graphique le plus important est la rotation de l'hélicoptère autour de z ( $\psi$ ). On peut voir maintenant que la rotation est seulement de  $6^\circ$ , ce qui est une grande amélioration par rapport à la deuxième simulation ( $120^\circ$ ). Il est certain qu'avec ce système, on ne fera pas une rotation de plus de  $360^\circ$  au cours de la minute de vol. Si le vol était beaucoup plus long, il faudrait ajouter une boucle de contre-réaction sur la position pour s'assurer de rester dans les limites permises.

La deuxième erreur est qu'une pale ait une poussée aérodynamique plus grande que les autres. Il est assez difficile de détecter cette erreur puisque le vecteur poussée résultant, qui n'est plus parfaitement vertical, suit la rotation des hélices. Cela va créer un débalancement qui va obliger la nacelle à osciller autour de ses axes de roulis et de tangage sans arrêt, mais l'erreur finale devrait être nulle si la différence de poussée aérodynamique est faible, comme on a pu remarquer dans les résultats de la première simulation. Il serait possible de détecter une erreur de poussée en faisant le lien entre la position de la pale et les erreurs autour des deux axes, et augmenter ou diminuer l'angle d'attaque de la pale fautive.

La troisième erreur possible est que l'hélicoptère soit en train de dériver et de s'approcher de la sortie du carré de  $10m \times 10m$  qui entraîne une disqualification automatique. Pour régler cette erreur, il est nécessaire de trouver un capteur qui permet de mesurer la position de l'hélicoptère par rapport au sol et un actuateur qui permet de replacer celle-ci au centre du carré. Le choix d'un capteur n'est pas facile, il doit permettre d'avoir une précision de moins de 50 centimètres et être léger. Les GPS différentielles permettent d'avoir seulement une précision de un mètre, ce qui est trop imprécis pour l'application. Une autre possibilité est d'utiliser un système de vision artificielle au sol qui calcule la position de l'hélicoptère et qui renvoie l'information à l'hélicoptère.

Pour repositionner l'hélicoptère au centre du carré, il existe deux méthodes : soit utiliser le système de pas variable et augmenter l'angle d'une pale au bon moment pour modifier la poussée de l'hélicoptère vers la direction voulue, soit déplacer le centre de gravité pour que la nacelle pivote légèrement autour des ses axes de roulis et de tangage et ainsi faire pointer la poussée vers la direction voulue. Pour pouvoir modéliser cette dernière possibilité, il faudrait modifier le modèle pour avoir deux systèmes d'axe, un pour le centre géométrique de l'hélicoptère et un pour le centre de gravité. L'avantage de cette dernière méthode est que la poussée dans la direction voulue est continue alors qu'en utilisant les pales il faut attendre que les pales soient bien positionnées.

## CHAPITRE 4

### **SYSTÈME DE CONTRÔLE EN POSITION DES PALES**

Pour pouvoir calibrer les pales et pour que l'ordinateur puisse changer l'angle d'attaque des pales lors du vol, il faut que chaque pale soit contrôlée en position par un moteur électrique. Le moteur utilisé doit être le plus léger possible et développer assez de puissance pour pouvoir faire tourner les pales. Le choix s'est arrêté sur un moteur de la compagnie Maxon de 4.5 Watts. Pour pouvoir contrôler le moteur en position, il faut un système de contre-réaction ainsi qu'un amplificateur de puissance. La plupart des contrôleurs sur le marché sont faits pour contrôler des moteurs plus gros que celui utilisé et fonctionne donc sur le réseau électrique et sont assez lourd. Il est donc nécessaire de concevoir un contrôleur qui répond exactement aux critères suivants :

- léger,
- peu de pertes électriques dans le contrôleur,
- fonctionnant seulement avec une alimentation de 12 Volts,
- contrôle en position,
- limite le courant fourni au moteur.

Pour que le contrôleur soit léger, il ne faut pas installer de dissipateur de chaleur sur l'amplificateur de puissance et que la quantité de circuit intégré soit réduit au maximum. Pour que le contrôleur soit le plus efficace possible, on doit éviter de faire fonctionner les transistors de puissance dans leurs plages linéaires et les utiliser plutôt en saturation. C'est aussi en saturation qu'ils dissiperont le moins de chaleur, donc il sera possible d'éliminer les dissipateurs de chaleur.

Pour pouvoir faire une boucle de contre-réaction, il est nécessaire d'avoir un capteur qui donne la position de la pale en tout temps, les deux choix possibles sont les encodeurs incrémentaux ou les potentiomètres. Le problème des encodeurs incrémentaux est qu'ils donnent seulement la différence entre une position initiale et la position actuelle, il faut donc à chaque fois que le contrôleur démarre, trouver la position que l'on considère comme étant le  $0^\circ$ , par contre ils sont plus précis que les potentiomètres. Les potentiomètres donnent la position absolue, il n'est donc pas nécessaire de retrouver la position  $0^\circ$ . Étant donné que la précision nécessaire est de l'ordre du  $0.1^\circ$  et que la position  $0^\circ$  est importante, le potentiomètre est le meilleur choix.

Le deuxième choix qui doit être fait est de décider si le contrôleur sera numérique, donc par microprocesseur, ou s'il sera purement analogique. Lorsqu'un circuit électronique est produit à plusieurs milliers d'exemplaires, le choix des circuits numériques est plus intéressant car ils n'évoluent pas avec le temps ou la température. Par contre, la conception est beaucoup plus rapide dans le cas d'un circuit seulement analogique, et puisque le contrôleur de moteur est seulement un prototype, il est plus avantageux de le faire de cette façon. La seule difficulté est que l'alimentation est seulement positive, alors que d'habitude on possède une alimentation positive et négative, ce qui permet de faire les opérations mathématiques facilement car le zéro mathématique est à 0 Volts. Pour contourner ce problème, le zéro a été positionné à 5 Volts et tous les résultats mathématiques sont exprimés par rapport à cette référence.

L'amplificateur de puissance, comme on l'a mentionné plus tôt, devra utiliser tous les transistors en saturation. De plus, il devra être capable de faire tourner le moteur dans les deux sens de rotation. Le meilleur choix d'amplificateur sera un pont en H contrôlé par une modulation de la largeur d'impulsion (PWM Pulse Width Modulator). La plupart des compagnies de circuits intégrés possèdent ce type de circuit

pour des applications faible courant (< 5A), un des plus connus est le LMD18200 de National.

Pour transformer la commande donnée par le contrôleur PI en modulation de largeur d'impulsion, on utilise un modulateur de largeur d'impulsion (PWM modulator). Un des plus utilisé dans l'industrie depuis maintenant plusieurs années est le UC3525. Une tension de 0.9 Volts à son entrée génère une onde avec 0% de largeur d'impulsion alors que 3.3V donne 100% de largeur d'impulsion. De plus, il existe une entrée, qui lorsqu'elle est mis à 5V, met les sorties à 0 immédiatement, elle sera utilisée pour implanter le limiteur de courant.

La figure 4.1 permet de bien comprendre le fonctionnement du contrôleur de moteur.

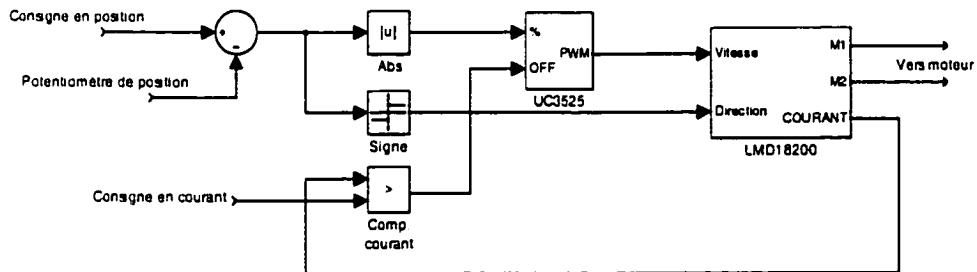


Figure 4.1 Schéma-bloc du contrôleur de moteur

Le contrôleur calcule l'erreur entre la consigne et la position actuelle et injecte l'erreur dans un contrôleur Proportionnel-Integral (PI). Le signal de sortie du PI donne la vitesse (valeur absolue du signal), qui contrôle le pourcentage de largeur d'impulsion à la sortie du UC3525, et la direction (signe du signal) qui contrôle la direction de rotation du moteur dans le LMD18200. Le signal de courant du LMD18200 donne 1.772 Volts par Ampère fournit au moteur. Si le signal du courant est plus élevé que la consigne de courant, la sortie du comparateur devient à 1 et les sorties du UC3525

arrêtent automatiquement. Les schémas électriques complet du contrôleur de moteur sont en annexe.

Le fonctionnement du limiteur de courant est très simple. Dès que le courant dépasse la consigne, les sorties sont mises à 0, ce qui arrête le moteur. Dès l'arrêt du moteur, le courant diminue, ce qui oblige le comparateur à repartir le moteur. Lors du redémarrage du moteur, le courant va encore dépasser la consigne, ce qui va arrêter le moteur et ainsi de suite. Il y a limite de courant lorsque le moteur demande un couple trop élevé, lorsqu'il est bloqué mécaniquement ou lorsqu'il y a un court-circuit électrique. Lorsque le moteur est en limite de courant, la forme d'onde du courant envoyée au moteur est une onde triangulaire qui oscille autour de la consigne en courant. Dès que ce qui cause la limite de courant disparaît, le limiteur de courant devient hors fonction.

## **CONCLUSION**

La première étape dans la conception d'Hélios a été de décider quelle était la forme générale de l'hélicoptère, ce qui a été fait en analysant les différents prototypes qui avaient déjà été construits et d'autres concepts qui semblaient prometteurs. Simplement en faisant des analyses logiques, une des six configurations possédait toutes les caractéristiques nécessaires.

Par la suite, il était important de confirmer que le prototype choisi était stable, un modèle dynamique a donc été fait. Ce modèle, même s'il ne tenait pas compte de l'effet de sol et de la déflexion de la pale, a permis de confirmer que l'hélicoptère était stable dans des situations possibles de vol.

Les règles du concours spécifient que l'hélicoptère doit rester dans un carré de 10 m \* 10 m et que la nacelle doit faire seulement un tour sur elle-même. On a pu voir dans les simulations qu'une faible erreur autour des axes de roulis ou de tangage déplace rapidement l'hélicoptère en-dehors de cette zone, il est donc important que le centre de gravité soit situé exactement au centre de rotation des pales. La vitesse de rotation de la nacelle peut être ralentie en faisant une boucle de contre-réaction qui augmente ou diminue l'angle de calage des pales du haut.

Puisque le modèle considère que les pales ont un comportement aérodynamique identique en vol, il faut donc, dans la pratique, s'approcher le plus de la théorie. Pour ce faire, les pales sont testées sur un banc d'essai où on calibre un couple de pale pour obtenir la même poussée aérodynamique pour les deux pales. Lorsque la calibration est

faite, l'ordinateur reproduit durant le vol les mêmes angles de calage, selon les paramètres d'entrées.

## **RECOMMANDATIONS**

Les principales recommandations concernent le modèle dynamique de l'hélicoptère. Pour que les résultats du modèle soient le plus près de la réalité il serait important que le modèle tienne compte de l'effet de sol et surtout, de la déflexion dans les pales. L'effet de sol diminue la traînée aérodynamique et augmente la poussée aérodynamique puisqu'il diminue la vitesse d'entrée de l'air verticale dans le rotor, ce qui influence directement le comportement du modèle dynamique. La déflexion de la pale est beaucoup plus difficile à modéliser puisque le modèle doit être fonction de la structure mécanique de la pale et des forces aérodynamiques qu'elle subie. Une des solutions plus simple serait d'utiliser les résultats de déflexion lors des tests statiques, cela permettrait d'avoir une position approximative du vecteur poussée lors de la déflexion.

Avec un modèle plus complet, il serait intéressant de concevoir un système de contrôle en boucle fermée, qui permettrait d'améliorer la stabilité. Comme expliqué dans le chapitre sur le système de contrôle, l'approche devrait ressembler à celle de l'injection électronique. Les tables de calibration devrait rester dans le système et le système de contrôle en boucle fermée ferait de légers ajustements sur les valeurs des tables.

## BIBLIOGRAPHIE

- Etkin, B (1972). *Dynamics of atmospheric flight*. éditions J. Wiley & Sons
- Beer, F., Johnston, R. (1977). *Vector mechanics for engineers : Dynamics* (3<sup>e</sup> éd.). éditions McGraw-Hill.
- Meriam, J.L. Kraige, L.G. (1998). *Dynamics* 4<sup>e</sup> éd. éditions J. Wiley & Sons
- Lamarche, L. (1999). *MEC-741 Robotique : Notes de cours*. École de technologie supérieure.
- McCormick, B. (1979). *Aerodynamics, aeronautics, and flight mechanics*. éditions J. Wiley & Sons.
- DeLafontaine, J. (1998) *SYS-851 Analyse de la dynamique d'un aéronef*. École de technologie supérieure.
- Craig, J. (1989). *Introduction to robotics*. éditions Addison-Wesley

## **ANNEXE A**

### **Fabrication des gabarits des moules**

Les gabarits en aluminium permettent de faire le modèle sur lequel sera fait le moule. Ils sont positionnés à moins de 30 cm les uns des autres, ce qui donne la forme du modèle, par la suite du plâtre est coulé entre chaque gabarit. Il reste seulement à sabler l'excédent de plâtre pour revenir exactement à la surface en aluminium des gabarits. La technique complète de fabrication du modèle en plâtre est expliquée dans le mémoire de Simon Joncas « Conception et fabrication d'une aile rotative basse vitesse pour un hélicoptère à propulsion humaine».

Les résultats du programme de conception aérodynamique sont trois vecteurs : les rayons où sont placés les profils, les longueurs de corde de ces profils et leurs angles de rotation par rapport à l'horizontale. Lors de la conception de la pale, on considère que le fluide parcourt toute la longueur du profil, mais puisque l'hélice tourne sur son axe, le fluide suit un parcours circulaire. Il faut donc enruler le profil autour du rayon de rotation.

Puisque les gabarits en aluminium sont découpés dans une feuille d'aluminium, il n'est pas question d'enrouler physiquement les profils, il faut donc enruler mathématiquement les profils et trouver des gabarits plats qui représentent bien la surface enroulée. Il faut donc, dans un premier temps, trouver tous les points représentant la surface enroulée et, dans un deuxième temps, faire des coupes où on veut avoir des gabarits.

Le programme débute en chargeant le profil unitaire. Ce profil a une longueur en  $x$  de 1 et le bord d'attaque est à la coordonnées 0,0. La figure A.1 montre le profil lors du chargement.

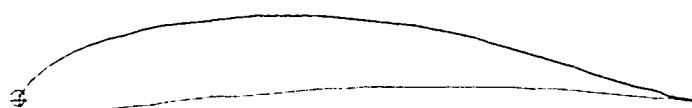


Figure A.1 Profil unitaire DAEDALUS 31

L'axe de rotation du profil est ensuite positionné à l'endroit voulu. Mathématiquement, l'axe de rotation correspond à la coordonnée 0,0 dans les calculs suivants. La figure A.2 montre le profil unitaire avec l'axe de rotation positionné à 33% de la longueur de corde, ce qui est la position de la pale d'Hélios.

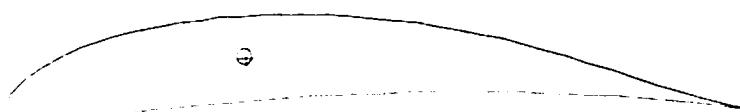


Figure A.2 Profil unitaire DAEDALUS 31 avec axe de rotation à 33%

On multiplie le profil unitaire avec la grandeur contenue dans le vecteur des cordes. Le profil, qui maintenant est à la bonne grandeur, est tourné autour de son axe de rotation de la valeur contenue dans le vecteur des angles de rotation par rapport à l'horizontale. La figure A.3 montre le résultat final de toutes ces opérations.

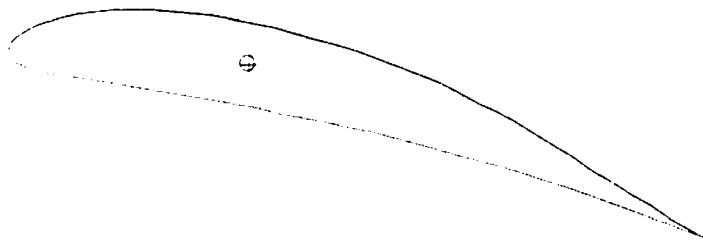


Figure A.3 Profil DAEDALUS positionné à son angle de calage

Ce profil qui est dessiné à plat doit être en réalité courbé selon le rayon où il est positionné. Si on considère un point de la pale ayant les coordonnées  $x_1$ ,  $y_1$  et  $z_1$ , où  $x_1$  correspond à la position du profil sur la pale et  $y_1$ ,  $z_1$  un point du profil, il peut être vu en coordonnées cartésiennes mais pour simplifier les opérations il peut être calculé en coordonnées cylindriques. En coordonnées cylindriques, on a trois variables, le rayon  $r$ , l'angle  $\theta$  et la hauteur  $z$ . Par rapport à nos coordonnées cartésiennes  $x_1$ ,  $y_1$  et  $z_1$ , le rayon

vaut  $x_1$  qui est la position du profil et  $z$  vaut  $z_1$  qui est la hauteur du point, il reste donc à trouver  $\theta$ . Dans la figure A.4, on illustre la relation entre  $\theta$ ,  $x_1$  et  $y_1$ .

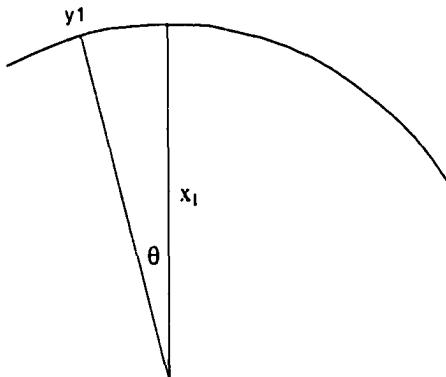


Figure A.4 Vue de dessus d'un profil

La variable  $y_1$  correspond à la longueur de l'arc de cercle et  $x_1$  représente le rayon où est placé le profil. L'équation suivante relie les trois variables.

$$\theta = \frac{y_1}{x_1} \quad (\text{A.1})$$

On connaît maintenant les trois variables en coordonnées cylindriques, il reste seulement à les transformer en coordonnées cartésiennes, ce qui se fait simplement en utilisant la fonction `pol2cart` de Matlab. Puisque les gabarits sont plats, donc à  $x$  constant, il est plus simple de faire les calculs en coordonnées cartésiennes. La figure A.5 montre une vue de dessus de trois profils et d'un gabarit. Puisque chaque profil est composé du même nombre de points, il est possible de faire deux interpolations linéaires pour trouver les points correspondants aux gabarits.

On peut prendre le premier point de chaque profil comme exemple. Les coordonnées de ce point sur le profil 1 s'appelleront  $x_1$ ,  $y_1$  et  $z_1$ , sur le profil 2,  $x_2$ ,  $y_2$  et  $z_2$  et le gabarit est positionné à  $x_g$ . On peut utiliser directement la fonction de Matlab `Interpl` pour trouver  $y_g$  et  $z_g$ .

$$y_g = \text{interp1}([x_1 \ x_2], [y_1 \ y_2], x_g);$$

$$z_g = \text{interp1}([y_1 \ y_2], [z_1 \ z_2], x_g);$$

On fait cette opération pour tous les points d'un gabarit. Lorsque c'est terminé, les points sont transmis à une fonction qui écrit un fichier de format DXF, qui est le format de Autocad en caractère ASCII.

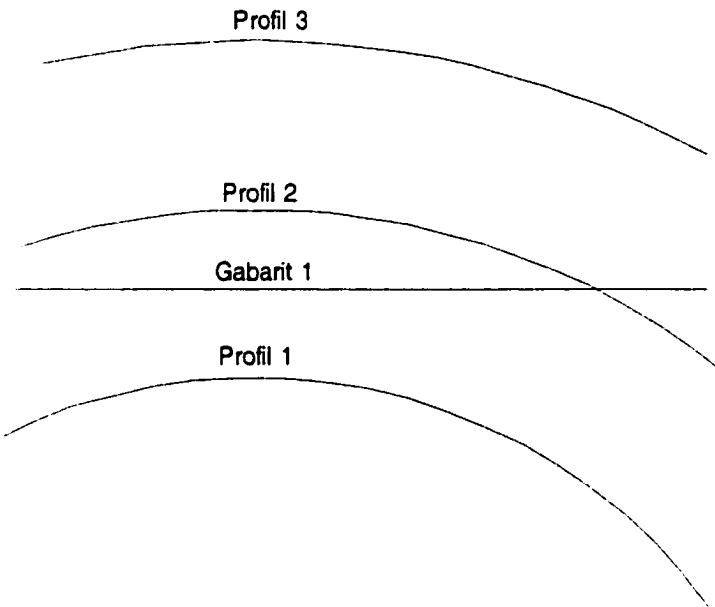


Figure A.5 Vue de dessus de trois profils et d'un gabarit

Lorsque tous les gabarits sont écrits dans le fichier DXF, il reste à utiliser Autocad pour positionner correctement les gabarits sur les feuilles d'aluminium avant d'envoyer les fichiers à la machine de découpe au plasma.

## TEMPLATE.M

```

function template(p)

rayon=p.tr;
corde=p.tc;
twist=p.tphi_alpha+p.tpitch;

w=waitbar(0,'Generation des templates');

xup=[];
yup=[];
zup=[];
xdown=[];
ydown=[];
zdown=[];

% Creation des fichiers de sortie
file_init;

%Creation de la pale 3D en utilisant le rayon, la corde et le twist

for compteur=1:length(rayon)

    [xup1,yup1,zup1,xdown1,ydown1,zdown1]=profil(rayon(compteur),corde(compteur),p.axe.twist(
compteur));

    xup=[xup xup1];
    yup=[yup yup1];
    zup=[zup zup1];

    xdown=[xdown xdown1];
    ydown=[ydown ydown1];
    zdown=[zdown zdown1];
end

template1=linspace(0.611,2.9);
template2=linspace(2.3,14.40);
template3=linspace(14.2,16.10);
template4=linspace(16.2,17.499,9);
template=[template1 template2 template3 template4];



%Interpolation lineaire entre les profils courbes pour générer
% les templates
for c_template=1:length(template)
    %Interpolation de la partie haute
    for compteur=1:size(xup,1)
        %Position en x du template
        xp_up=compteur=template(c_template);
        yp_up=compteur=interp1(xup(compteur,:),yup(compteur,:),xp_up(compteur));
    end
end

```

```
pos_x=1;
while(xup(compteur,pos_x)<template(c_template))
    pos_x=pos_x+1;
end
zp_up(compteur)=interp1(yup(compteur,pos_x-1:pos_x),zup(compteur,pos_x-
l:pos_x),yp_up(compteur));
    end
%interpolation de la partie basse
for compteur=l:size(xdown,1)
    xp_down(compteur)=template(c_template);
    yp_down(compteur)=interp1(xdown(compteur,:),ydown(compteur,:),xp_down(compteur));
    pos_x=1;
    while(xdown(compteur,pos_x)<template(c_template))
        pos_x=pos_x+1;
    end
    zp_down(compteur)=interp1(ydown(compteur,pos_x-1:pos_x),zdown(compteur,pos_x-
l:pos_x),yp_down(compteur));
    end
%Rotation des templates de 14 degrés pour diminuer l'espace occupé
[yp_up,zp_up]=rotation(yp_up,zp_up,14);
[yp_down,zp_down]=rotation(yp_down,zp_down,14);

generate_template(xp_up,yp_up,zp_up,xp_down,yp_down,zp_down,c_template);

waitbar(c_template/length(template));
end

file_end;
close(w);
```

## PROFIL.M

```
function [xup,yup,zup,xdown,ydown,zdown]=profil(rayon,corde,axe,twist)

%Chargement du profil unitaire
[yup,zup,ydown,zdown]=load_prof(axe);

%Dimensionnement du profil
yup=yup*corde;
zup=zup*corde;
ydown=ydown*corde;
zdown=zdown*corde;

%Rotation autour de l'arbre du profil
[yup,zup]=rotation(yup,zup,-twist);
[ydown,zdown]=rotation(ydown,zdown,-twist);

%Calcul de l'angle d'un point par rapport à l'arbre
teta=yup/rayon;
teta2=ydown/rayon;

%Transformation de cylindrique à cartésien
[xup,yup,zup]=pol2cart(teta,rayon,zup);
[xdown,ydown,zdown]=pol2cart(teta2,rayon,zdown);
```

## LOAD\_PROF.M

```
function [xup,yup,xdown,ydown]=load_prof(position_arbre)

% Chargement du fichier de profil Daedalus partie haute
t=load('top31.txt');
xup=t(:,1);
yup=t(:,2);

% partie basse
t=load('bottom31.txt');
xdown=t(:,1);
ydown=t(:,2);

% Translation en x (position de l'arbre) et en y (centre du profil)
x_translate=position_arbre;
y_translate=(spline(xup,yup,x_translate)+spline(xdown,ydown,x_translate))/2;

xup=xup-x_translate;
xdown=xdown-x_translate;
yup=yup-y_translate;
ydown=ydown-y_translate;
```

## **ANNEXE B**

### **Programmes Matlab pour le modèle dynamique**

## Programme de calcul symbolique

### modele1.m

```

syms pn pe h theta phi psi;
syms ptheta pphi ppsi;

% Calcul de la matrice de transformation entre le
% référentiel hélicoptère et le référentiel Terre

trans=[1 0 0 pn;0 1 0 pe;0 0 1 h;0 0 0 1];
rot_z=[cos(psi) -sin(psi) 0 0;sin(psi) cos(psi) 0 0;0 0 1 0;0 0 0 1];
rot_y=[cos(theta) 0 sin(theta) 0;0 1 0 0;-sin(theta) 0 cos(theta) 0;0 0 0 1];
rot_x=[1 0 0 0;0 cos(phi) -sin(phi) 0;0 sin(phi) cos(phi) 0;0 0 0 1];

T_H-vers_T=simple(trans*rot_z*rot_y*rot_x);

% Calcul de la matrice de rotation entre le référentiel Terre
% et le référentiel hélicoptère

R_T-vers_H=simple(inv(T_H-vers_T(1:3,1:3)));

% Vitesse angulaire entre la Terre et l'hélicoptère

w_H-T=simple([0;0;ppsi]+rot_z(1:3,1:3)*[0;ptheta;0]+rot_z(1:3,1:3)*rot_y(1:3,1:3)*[pphi;0;0]);

% Vitesse angulaire entre la Terre et l'hélicoptère par
% rapport à l'hélicoptère

w_TH_H=simple(R_T-vers_H*w_H-T);

% Réécriture de la matrice en mettant pphi,ppsi et ptheta
% en évidence

w_TH_H2=[1 0 -sin(theta);0 cos(phi) sin(phi)*cos(theta);0 -sin(phi) cos(phi)*cos(theta)];

% On trouve pphi,ppsi et ptheta en fonction de p,q et r

w_TH_T=simple(inv(w_TH_H2));

```

**modele2.m**

```
syms ix iy iz ixy ixz iyz;
syms ixp iyp izp ixyp ixzp iyzp;
syms L M N p q r;

I=[ix -ixy -ixz;-ixy iy -iyz;-ixz -iyz iz];
Ip=[ixp -ixyp -ixzp;-ixyp iyp -iyzp;-ixzp -iyzp -izp];
M=[L;M;N];
w=[p;q;r];

% Inverse de I. On enlève le déterminant pour
% simplifier l'équation finale
It=simple(inv(I));
It=simple(It.*det(I));

c l=It*M-It*Ip*w-It*(cross(w,I*w))
```

**modele3.m**

```
modele1
```

```

syms g gp a xp yp zp;
syms u v w pphi ptheta ppsi;
syms vwx vwy;

rot_z=[cos(g) -sin(g) 0 0;sin(g) cos(g) 0 0;0 0 1 0;0 0 0 1];
rot_x=[1 0 0 0;0 cos(a) -sin(a) 0;0 sin(a) cos(a) 0;0 0 0 1];
trans=[1 0 0 xp;0 1 0 yp;0 0 1 zp;0 0 0 1];

T_1_2=simple(rot_z*rot_x*trans);
R_1_2=T_1_2(1:3,1:3);
R_2_1=simple(inv(R_1_2));

R_0_1=T_H_vers_T(1:3,1:3);
R_1_0=simple(inv(R_0_1));

v_0_2=R_0_1*[u;v;w]+cross(w_H_T.R_0_1*T_1_2(1:3,4))+R_0_1*cross([0;0;gp],T_1_2(1:3,4));
v_0_2=simple(v_0_2)+[vwx;vwy;0];
v_2_02=simple(R_2_1*R_1_0*v_0_2);

```

## Programmes utilisés par Simulink

### cinematique\_lin.m

```
function out=cinematique_lin(in)

u=in(1);
v=in(2);
w=in(3);
phi=in(4);
theta=in(5);
psi=in(6);

T_H_vers_T =[cos(psi)*cos(theta) -sin(psi)*cos(phi)+cos(psi)*sin(theta)*sin(phi)
sin(psi)*sin(phi)+cos(psi)*sin(theta)*cos(phi);...
              sin(psi)*cos(theta) cos(psi)*cos(phi)+sin(psi)*sin(theta)*sin(phi) -
cos(psi)*sin(phi)+sin(psi)*sin(theta)*cos(phi);...
              -sin(theta) cos(theta)*sin(phi) cos(theta)*cos(phi)];;

out=T_H_vers_T*[u;v;w];
```

**dynamique\_lin**

```
function out=dynamique_lin(in)
global m;
p=double(in(1));
q=double(in(2));
r=double(in(3));
u=double(in(4));
v=double(in(5));
w=double(in(6));
Fx=double(in(7));
Fy=double(in(8));
Fz=double(in(9));

u_p=Fx/m + r*v - q*w;
v_p=Fy/m - r*u + p*w;
w_p=Fz/m + q*u - p*v;

out=[u_p;v_p;w_p];
```

**cinematique\_rot**

```
function out=cinematique_rot(in)

p=in(1);
q=in(2);
r=in(3);
phi=in(4);
theta=in(5);
psi=in(6);

w_TH_T=[1 tan(theta)*sin(phi) tan(theta)*cos(phi);...
         0 cos(phi) -sin(phi);...
         0 sin(phi)/cos(theta) cos(phi)/cos(theta)];;

out=w_TH_T*[p;q;r];
```

## dynamique\_rot

```

function out=dynamique_rot(in)

global h1;
global h2;
global mp;

p=in(1);
q=in(2);
r=in(3);
L=in(4);
M=in(5);
N=in(6);
omega=in(7);
gamma=in(8);

%Inertie d'une pale
ixp=1.7277;
iyp=1139.3;
izp=1140.8;

%Rotation dans le sens anti-horaire
Ix1=cos(gamma)*ixp-sin(gamma)*iyp+mp*h1^2;
Iy1=sin(gamma)*ixp+cos(gamma)*iyp+mp*h1^2;
Iz1=izp;
Ix_der1=-omega*(sin(gamma)*ixp+cos(gamma)*iyp);
Iy_der1=-omega*(-cos(gamma)*ixp+sin(gamma)*iyp);
Iz_der1=0;

%Rotation dans le sens horaire
Ix2=cos(gamma)*ixp+sin(gamma)*iyp+mp*h2^2;
Iy2=-sin(gamma)*ixp+cos(gamma)*iyp+mp*h2^2;
Iz2=izp;
Ix_der2=-omega*(sin(gamma)*ixp-cos(gamma)*iyp);
Iy_der2=-omega*(cos(gamma)*ixp+sin(gamma)*iyp);
Iz_der2=0;

%Inertie de la nacelle
Ix_nac=2.55;
Iy_nac=9.375;
Iz_nac=8.175;

Ix=2*abs(Ix1)+2*abs(Ix2)+Ix_nac;
Iy=2*abs(Iy1)+2*abs(Iy2)+Iy_nac;
Iz=2*abs(Iz1)+2*abs(Iz2)+Iz_nac;
Ix_der=2*abs(Ix_der1)+2*abs(Ix_der2);
Iy_der=2*abs(Iy_der1)+2*abs(Iy_der2);
Iz_der=2*abs(Iz_der1)+2*abs(Iz_der2);

p_p=(L-Ix_der*p+r*q*(Iy-Iz))/Ix;

```

```
q_p=(M-Iy_der*q+p*r*(Ix-Iz))/Iy;  
r_p=(N-Iz_der*r+p*q*(Ix-Iy))/Iz;
```

```
out=[p_p q_p r_p];
```

### **force\_aero.m**

```

function out=force_aero(in)

% Variables globales
global p_bas;
global sys_bas;
global p_haut;
global sys_haut;
global xp;
global yp;
global h1;
global h2;
global m;

% Paramètres du programme Simulink
phi=in(1);
theta=in(2);
psi=in(3);
pphi=in(4);
ptheta=in(5);
ppsi=in(6);
u=in(7);
v=in(8);
w=in(9);
omega=in(10);
gamma=in(11);
angle=in(12);

g=9.81;

% Calcul des poussées et des trainées aérodynamiques
p1=offdesign3(p_haut,sys_haut,angle,gamma,omega,u,v,w,phi,theta,psi,pphi,ptheta,ppsi,1,1);
p2=offdesign3(p_haut,sys_haut,angle,gamma+pi,omega,u,v,w,phi,theta,psi,pphi,ptheta,ppsi,2,1);
p3=offdesign3(p_bas,sys_bas,5,-gamma,-omega,u,v,w,phi,theta,psi,pphi,ptheta,ppsi,3,2);
p4=offdesign3(p_bas,sys_bas,5,-gamma-pi,-omega,u,v,w,phi,theta,psi,pphi,ptheta,ppsi,4,2);

% Séparation de la poussée et de la trainée
f1z=p1.T/2;
f1y=-(p1.Q/xp)/2;
f2z=p2.T/2;
f2y=-(p2.Q/xp)/2;
f3z=p3.T/2;
f3y=(p3.Q/xp)/2;
f4z=p4.T/2;
f4y=(p4.Q/xp)/2;

% Calcul du couple pour la contre-réaction dans Simulink
Torque=p1.Q/2+p2.Q/2+p3.Q/2+p4.Q/2;

% Les 4 forces aérodynamiques en vecteur

```

```

f1=[0;f1y;f1z];
f2=[0;f2y;f2z];
f3=[0;f3y;f3z];
f4=[0;f4y;f4z];

% Position à l'instant 0 des 4 forces aérodynamiques
pos1=[xp;yp;h1];
pos2=[xp;yp;h1];
pos3=[xp;yp;h2];
pos4=[xp;yp;h2];

% Calcul des matrices de rotation selon la position de la pale 1
T1=[cos(gamma) -sin(gamma) 0;sin(gamma) cos(gamma) 0;0 0 1];
T2=[cos(gamma+pi) -sin(gamma+pi) 0;sin(gamma+pi) cos(gamma+pi) 0;0 0 1];
T3=[cos(-gamma) -sin(-gamma) 0;sin(-gamma) cos(-gamma) 0;0 0 1];
T4=[cos(-gamma-pi) -sin(-gamma-pi) 0;sin(-gamma-pi) cos(-gamma-pi) 0;0 0 1];

% Position actuelle des 4 pales
pos1=T1*pos1;
pos2=T2*pos2;
pos3=T3*pos3;
pos4=T4*pos4;

% Position actuelle des 4 forces
f1=T1*f1;
f2=T2*f2;
f3=T3*f3;
f4=T4*f4;

% Calcul des moments générés autour de x,y et z
M1=cross(pos1,f1);
M2=cross(pos2,f2);
M3=cross(pos3,f3);
M4=cross(pos4,f4);

% Addition des 4 forces aérodynamiques et de la gravité
F=f1+f2+f3+f4;
F=F+[-sin(theta)*m*g;cos(theta)*sin(phi)*m*g;cos(theta)*cos(phi)*m*g];

% Addition des moments
M=M1+M2+M3+M4;

% Paramètre de sortie
out=[F;M;Torque];

```

## offdesign3.m

```

function p=offdesign3(p,sys,calage,g,gp,u,v,w,phi_mod,theta,psi,pphi,ptheta,ppsi,pale,rotor)

% Angle de décrochage du profil
alphad = 8;
% Largeur d'un élément de pale
dr=(p.D/2-p.R1)/(sys.n-1);

for en=1:sys.n-1

    % Angle de la pale par rapport à l'horizontale
    beta=calage+p.tphi_alpha(en);

    % Calcul de l'angle que fait le vent par rapport à la pale
    [Vy,Vz,gamma]=angle_attaque(g, gp, beta*pi/180, u, v, w, phi_mod, theta, psi, pphi, ptheta, ppsi, pale, rotor, p.tr(en));

    % Calcul des vitesses d'entrée horizontale et verticale du vent par
    % rapport au plan de rotation
    phi=(beta-gamma)*pi/180;
    Vr=sqrt(Vy^2+Vz^2);
    Vver=sin(phi)*Vr;
    Vhor=cos(phi)*Vr;

    % Calcul de l'angle d'attaque induit
    [wa,wt,Ve]=calcul_wa_wt(p,en,Vhor,Vver,beta);

    % Calcul de l'angle d'attaque (alpha)
    alphai=atan2(sqrt(wa^2+wt^2),Ve);
    phi_alpha=en+phi;
    alpha=beta-phi_alpha*pi/180;

    % Calcul du Cl et Cd utilisés dans calcul_wa_wt
    p.tCl(en)=0.1125*alpha+0.7+(1+sign(alpha-alphad))/2*(-0.132*(alpha-alphad));
    p.tCd(en)=((alpha-4.75).^2/1600+0.015).*(1-sign(alpha-alphad))/2+(1+sign(alpha-alphad))/2.*(.1437e-2.*alpha+0.0216);
    p.tCl(en)=max(p.tCl(en),0);
    p.tCd(en)=min(p.tCd(en),1.2);
    p.tCd(en)=max(p.tCd(en),0.02);

    % Calcul de la poussée et de la trainée locale par rapport à l'angle d'attaque
    p.dL(en)=0.5*sys.rho*Ve.^2*p.tC(en)*p.tCl(en)*dr;
    p.dD(en)=0.5*sys.rho*Ve.^2*p.tC(en)*p.tCd(en)*dr;

    % Calcul de la poussée et de la trainée par rapport au plan de rotation
    if(phi_alpha>0)
        p.tT(en)=p.dL(en)*cos(phi_alpha) - p.dD(en)*sin(phi_alpha);
        p.tQ(en)=p.tr(en)*(p.dL(en)*sin(phi_alpha) + p.dD(en)*cos(phi_alpha));
    else

```

```
p.tT(en)=p.dL(en)*cos(phi_alpha1) + p.dD(en)*sin(phi_alpha1);
p.tQ(en)=p.tr(en)*(-p.dL(en)*sin(phi_alpha1) + p.dD(en)*cos(phi_alpha1));
end

p.tM(en)=0.5*p.tCm(en)*sys.rho*Ve^2*p.tc(en)^2*dr-((0.25-p.axe)*p.tc(en)*p.dL(en));
end

%Sommation de la poussée et de la trainée
p.T=(2*sum(p.tT)-p.tT(1)-p.tT(sys.n))*p.B/2;
if(p.T<0)
    p.T=0;
end

p.Q=(2*sum(p.tQ)-p.tQ(1)-p.tQ(sys.n))*p.B/2;
p.M=(2*sum(p.tM)-p.tM(1)-p.tM(sys.n))*p.B/2;
```

### angle\_attaque.m

```

function [vy,vz,angle]=angle_attaque(g,gp,a,u,v,w,phi,theta,psi,pphi,ptheta,ppsi,pale,rotor,x)

global h1;
global h2;
global yp;

%Vent terrestre
vwx=0;
vwy=0;

%position du centre de pression
switch pale
case 1, zp=h1;
case 2, zp=h1;
case 3, zp=h2;
case 4, zp=h2;
end

switch rotor
case 1, %Pale du haut
    v_2_02=[[0];...
              [-cos(phi)*ptheta*sin(g)*zp-
sin(g)*cos(a)*u+cos(g)*cos(a)*sin(psi)*sin(theta)*sin(phi)*vwy-cos(g)*cos(a)*sin(psi)*cos(phi)*vwx-
sin(g)*cos(a)*cos(psi)*cos(theta)*vwx+cos(g)*cos(a)*cos(psi)*sin(theta)*sin(phi)*vwx+sin(a)*cos(psi)*sin(theta)*cos(phi)*vwx-sin(a)*cos(psi)*sin(phi)*vwy+cos(a)*gp*x+cos(a)*cos(theta)*ppsi*cos(phi)*x-
cos(a)*ptheta*sin(phi)*x-sin(a)*cos(phi)*ptheta*cos(g)*x+cos(g)*cos(a)*v-
sin(g)*cos(a)*sin(psi)*cos(theta)*vwy+cos(g)*cos(a)*cos(psi)*cos(phi)*vwy+sin(a)*sin(psi)*sin(theta)*cos(phi)*vwy+sin(a)*sin(psi)*sin(phi)*vwx-sin(a)*ppsi*sin(theta)*sin(g)*x+ppsi*sin(theta)*cos(g)*zp-
sin(a)*sin(phi)*ppsi*cos(theta)*cos(g)*x+sin(a)*pphi*sin(g)*x-pphi*cos(g)*zp-
sin(phi)*ppsi*cos(theta)*sin(g)*zp+sin(a)*w];
              [ cos(g)*pphi*yp-cos(g)*sin(a)*v-cos(a)*sin(phi)*ppsi*cos(theta)*cos(g)*x-
cos(g)*sin(theta)*ppsi*yp+sin(g)*cos(theta)*ppsi*sin(phi)*yp+cos(a)*w-
cos(a)*cos(phi)*ptheta*cos(g)*x+sin(g)*sin(a)*sin(psi)*cos(theta)*vwy-
cos(g)*sin(a)*cos(psi)*cos(phi)*vwy-
cos(g)*sin(a)*sin(psi)*sin(theta)*sin(phi)*vwy+cos(a)*sin(psi)*sin(theta)*cos(phi)*vwy-
cos(g)*sin(a)*cos(psi)*sin(theta)*sin(phi)*vwx+cos(g)*sin(a)*sin(psi)*cos(phi)*vwx+cos(a)*cos(psi)*sin(theta)*cos(phi)*vwx+sin(g)*sin(a)*cos(psi)*cos(theta)*vwx+cos(a)*sin(psi)*sin(phi)*vwx-
cos(a)*cos(psi)*sin(phi)*vwy+cos(a)*pphi*sin(g)*x-sin(a)*gp*x+sin(g)*sin(a)*u-
sin(a)*cos(theta)*ppsi*cos(phi)*x+sin(a)*ptheta*sin(phi)*x+sin(g)*ptheta*cos(phi)*yp-
cos(a)*ppsi*sin(theta)*sin(g)*x]];
    angle=(atan2(-v_2_02(3),v_2_02(2)))*180/pi;
    vy=v_2_02(2);
    vz=v_2_02(3);
case 2, %Pale du bas
    v_2_02=[[0];...
              [cos(a)*gp*x+cos(g)*cos(a)*v-pphi*cos(g)*zp+cos(a)*cos(theta)*ppsi*cos(phi)*x-
sin(phi)*ppsi*cos(theta)*sin(g)*zp-
sin(a)*w+ppsi*sin(theta)*cos(g)*zp+sin(a)*sin(phi)*ppsi*cos(theta)*cos(g)*x+sin(a)*ppsi*sin(theta)*sin(g)*x+sin(a)*cos(phi)*ptheta*cos(g)*x-cos(phi)*ptheta*sin(g)*zp-
sin(phi)*x+sin(a)*cos(phi)*ptheta*cos(g)*x-cos(phi)*ptheta*sin(g)*zp];

```

```

sin(a)*pphi*sin(g)*x+cos(a)*ptheta*sin(phi)*x+sin(a)*cos(psi)*sin(phi)*vwy-
sin(a)*sin(psi)*sin(phi)*vwx-sin(g)*cos(a)*u+cos(g)*cos(a)*cos(psi)*cos(phi)*vwy-
sin(g)*cos(a)*sin(psi)*cos(theta)*vwy+cos(g)*cos(a)*sin(psi)*sin(theta)*sin(phi)*vwy-
sin(a)*sin(psi)*sin(theta)*cos(phi)*vwy-
cos(g)*cos(a)*sin(psi)*cos(phi)*vwx+cos(g)*cos(a)*cos(psi)*sin(theta)*sin(phi)*vwx-
sin(a)*cos(psi)*sin(theta)*cos(phi)*vwx-sin(g)*cos(a)*cos(psi)*cos(theta)*vwx];...
[-
sin(g)*sin(a)*u+cos(g)*sin(a)*v+cos(a)*pphi*sin(g)*x+sin(a)*cos(theta)*ppsi*cos(phi)*x-
cos(a)*cos(psi)*sin(phi)*vwy+cos(g)*sin(a)*cos(psi)*sin(theta)*sin(phi)*vwx+cos(a)*cos(psi)*sin(theta)*
*cos(phi)*vwx-cos(g)*sin(a)*sin(psi)*cos(phi)*vwx+cos(a)*sin(psi)*sin(phi)*vwx-
sin(g)*sin(a)*cos(psi)*cos(theta)*vwx+cos(g)*sin(a)*cos(psi)*cos(phi)*vwy+cos(a)*sin(psi)*sin(theta)*c
os(phi)*vwy+cos(g)*sin(a)*sin(psi)*sin(theta)*sin(phi)*vwy-
sin(g)*sin(a)*sin(psi)*cos(theta)*vwy+cos(a)*w-
cos(a)*cos(phi)*ptheta*cos(g)*x+sin(g)*ptheta*cos(phi)*yp+sin(a)*gp*x-
cos(a)*sin(phi)*ppsi*cos(theta)*cos(g)*x-cos(a)*ppsi*sin(theta)*sin(g)*x-cos(g)*sin(theta)*ppsi*yp-
sin(a)*ptheta*sin(phi)*x+cos(g)*pphi*yp+sin(g)*cos(theta)*ppsi*sin(phi)*yp]];
angle=(atan2(-v_2_02(3),-v_2_02(2)))*180/pi;
vy=v_2_02(2);
vz=v_2_02(3);
end

```

### **calcul\_wa\_wt.m**

```

function [wa,wt,Ve]=calcul_wa_wt(p,en,Vhor,Vver,beta)

% Valeurs de départ du calcul numérique
wa2=1;
wt2=1;
wa=0;
wt=0;
alphad = 8;

tol=0.05;
diff=0.01;

while( abs(wa2-wa)>tol | abs(wt2-wt)>tol )
    wa=wa2;
    wt=wt2;
    Ve=sqrt((Vver+wa)^2 + (Vhor-wt)^2);

    % Calcul de l'angle d'attaque actuelle
    alpha=beta-atan2((Vver+p.twa(en)),(Vhor-p.twt(en)))*180/pi;

    % Calcul du Cl et Cd actuelle
    Cl=0.1125*alpha+0.7+(1+sign(alpha-alphad))/2*(-0.132*(alpha-alphad));
    Cd=((alpha-4.75).^2/1600+0.015).*(1-sign(alpha-alphad))/2+(1+sign(alpha-alphad))/2.*(1.437e-2.*alpha-alphad)+0.0216;
    Cl=max(Cl,0);
    Cd=min(Cd,1.2);
    Cd=max(Cd,0.02);

    % Wa
    F1=foncl(wa,wt,Ve,p,en,Vhor,Vver,Cl,Cd);
    F1_moins=foncl(wa-diff,wt,Ve,p,en,Vhor,Vver,Cl,Cd);
    F1_plus=foncl(wa+diff,wt,Ve,p,en,Vhor,Vver,Cl,Cd);
    wa2=wa-(F1/((F1_plus-F1_moins)/(2*diff)));
    % Wt
    F2=fonc2(wa,wt,Ve,p,en,Vhor,Vver,Cl,Cd);
    F2_moins=fonc2(wa,wt-diff,Ve,p,en,Vhor,Vver,Cl,Cd);
    F2_plus=fonc2(wa,wt+diff,Ve,p,en,Vhor,Vver,Cl,Cd);
    wt2=wt-(F2/((F2_plus-F2_moins)/(2*diff)));
end
wa=wa2;
wt=wt2;
Ve=sqrt((Vver+wa)^2 + (Vhor-wt)^2);

%*****
function y=foncl(wa,wt,Ve,p,en,Vhor,Cl,Cd)
y=8*pi*p.tr(en)*wa*(Vver+wa)-p.B*p.tc(en)*Ve^2*(Cl*((Vhor-wt)/Ve)-Cd*((Vver+wa)/Ve));

```

*C<sub>c</sub>*\*\*\*\*\*

function y=fonc2(wa,wt,Ve,p,en,Vver,Vhor,Cl,Cd)

y=8\*pi\*p.tr(en)\*p.tF(en)\*wt-p.B\*p.tc(en)\*Cl\*Ve;

**vari.m**

```
clear;
global p_bas;
global sys_bas;
global p_haut;
global sys_haut;
global h1;
global h2;
global xp;
global yp;
global mp;
global m;

p_bas=load('p_bas_v');
p_bas=p_bas.p_bas;
sys_bas=load('sys_bas_v');
sys_bas=sys_bas.sys;
p_haut=load('p_haut_v');
p_haut=p_haut.p_haut;
sys_haut=load('sys_haut_v');
sys_haut=sys_haut.sys;
J=4600;
h1=0.6;
h2=-0.4;
yp=0;
xp=11.5;
mp=18.1;
m=150;
```

## **ANNEXE C**

### **Schéma électrique du contrôleur en position des moteurs**

