



HAL
open science

Stratégies de guidage visuel bio-inspirées : application à la stabilisation visuelle d'un micro-drone et à la poursuite de cibles

Augustin Manecy

► To cite this version:

Augustin Manecy. Stratégies de guidage visuel bio-inspirées : application à la stabilisation visuelle d'un micro-drone et à la poursuite de cibles. Automatique / Robotique. Université Grenoble Alpes, 2015. Français. NNT : 2015GREAT050 . tel-01218676v2

HAL Id: tel-01218676

<https://theses.hal.science/tel-01218676v2>

Submitted on 21 Oct 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Automatique - Productique**

Arrêté ministériel : 7 août 2006

Présentée par

Augustin MANECY

Thèse dirigée par **Nicolas MARCHAND** et
codirigée par **Stéphane VIOLLET**

préparée au sein du
laboratoire GIPSA-Lab de Grenoble et l'ISM de Marseille
dans l'école doctorale **EEATS**

Stratégies de guidage visuel bio-inspirées : application à la stabilisation d'un micro-drone et à la poursuite de cibles.

Thèse soutenue publiquement le **22/07/2015**,
devant le jury composé de :

Jean-Paul LAUMOND, *Rapporteur*
Directeur de Recherche, LAAS-CNRS, Toulouse

François CHAUMETTE, *Rapporteur*
Directeur de Recherche, IRISA-INRIA, Rennes

Isabelle FANTONI, *Examinateur*
Directrice de Recherche, Heudiasyc-CNRS, Compiègne

Pascal VASSEUR, *Examinateur*
Professeur, Université de Rouen LITIS-CNRS, Rouen

Nicolas MARCHAND, *Directeur de thèse*
Directeur de Recherche, GIPSA-Lab-CNRS, Grenoble

Stéphane VIOLLET, *Directeur de thèse*
Directeur de Recherche, ISM-CNRS, Marseille



UNIVERSITÉ DE GRENOBLE
ÉCOLE DOCTORALE EEATS
Électronique, Électrotechnique, Automatique
et Traitement du Signal

T H È S E

pour obtenir le titre de

docteur en sciences

de l'Université de Grenoble

Mention : Automatique et robotique

Présentée et soutenue par

Augustin MANECY

**Stratégies de guidage visuel bio-inspirées : application à la
stabilisation d'un micro-drone et à la poursuite de cibles.**

Thèse dirigée par Nicolas MARCHAND et co-dirigée par Stéphane VIOLLET

préparée au GIPSA-Lab de Grenoble et à l'ISM de Marseille

soutenue le 22/07/2015

Jury :

<i>Rapporteurs :</i>	Jean Paul LAUMOND	-	Directeur de Recherche	LAAS/CNRS
	François CHAUMETTE	-	Directeur de Recherche	IRISA/INRIA
<i>Examineurs :</i>	Isabelle FANTONI	-	Directeur de Recherche	Heudiasyc/CNRS
	Pascal VASSEUR	-	Professeur, Université de Rouen	LITIS/CNRS
<i>Directeurs :</i>	Nicolas MARCHAND	-	Directeur de Recherche	GIPSA-Lab/CNRS
	Stéphane VIOLLET	-	Directeur de Recherche	ISM/CNRS

Remerciements

Je tiens tout d'abord à remercier mes deux directeur de thèse, Stéphane et Nicolas, pour m'avoir permis de prendre part à cette aventure. Je les remercie pour leur disponibilité, et leur suis reconnaissant de tout ce que j'ai pu apprendre durant ces années de thèse. Je les remercie également pour la confiance qu'il m'ont accordée et l'autonomie dont j'ai disposée durant mes travaux de recherche.

Je tiens également à remercier les équipes pédagogiques qui m'ont permis de participer à leurs activités d'enseignements. Je remercie donc Hayate Khennouf et Ahmad Hably pour les TP d'automatique linéaire à l'ENSE³. Je remercie Zineb Simeu-Abazi les TP et TD d'automatique de M1, et Philip Navarro pour les TP d'automatique de M2P ISTRé à l'UJF de Grenoble. Je remercie également Leticia Gimeno pour m'avoir fait confiance sur les TD et TP en électronique de L1 à l'UJF. Je remercie Julien Serres pour m'avoir permis d'enseigner les rudiments de MATLAB à l'école des Mines de Saint Étienne, à Gardanne.

Je tiens également à exprimer ma gratitude aux différents membres du jury qui ont accepté de relire et évaluer mes travaux.

Je remercie aussi de tout cœur l'ensemble des doctorants et post-doctorants que j'ai eu la chance de côtoyer au Gipsa-Lab, pour leur bonne humeur, les discussions et débats animés, ainsi que l'ambiance fraternelle, conviviale et taquine qui règne grâce à eux. Je remercie donc particulièrement, Soheib, Ignacio, Antoine, Sébastien, Tahar, Rogelio, Valentina, Felipe, Maria, Humberto et tout les autres. Je remercie également mes co-bureaux de Grenoble, pour leur compagnie, leur bonne humeur, leur inventivité et leurs idées farfelues (Rogelio, paix à tes moustiques-mouches). Une mention particulière à Tahar, pour ses interrogations toujours surprenantes, voir même déroutantes. Je regretterai également ces luttes, territoriales, mais toujours amicales, pour le partage de notre espace de travail. Une petite pensée particulière pour Gabriel (et sa voix d'instructeur de TOEIC), pour m'avoir suivi, à plusieurs reprises, dans mes aventures sportives aux quatre coins de la France.

Je remercie également l'ensemble de l'équipe de Marseille : Stéphane, Franck, Thibaut, Julien, Michaël et Nicolas, sans oublier bien sûr Bernadette et Nathalie. Je remercie également l'ensemble des doctorants de Marseille ; je salue donc Frédéric, Raphaël, le gentil Fabien et le méchant Fabien, Stefano notre italien national, et bien sûr Roman, notre biologiste. Salutation à Erik, même si nous n'avons fait que nous croiser. Merci à Fabien C. pour toutes les discussions fructueuses et avisées sur le robot et son œil. Merci à Stefano pour son accent chantant et sa patience face à notre humour insistant, et merci à Roman pour nos débats philosophiques. Je félicite également Bruno pour sa montée en grade, et le remercie pour son aide concernant les tests et la duplication du quadrirotor, souvent très tard dans la nuit. Bien sûr, un merci particulier à Fabien Expert pour son amitié et pour m'avoir aidé à maintenir un minimum d'activité sportive en fin de thèse.

Je remercie également tout particulièrement les ingénieurs et techniciens, qui ont pris part à la réalisation du quadrirotor et de son œil. Je remercie donc Julien Minet et Jonathan Dumon du Gipsa-

Lab. Je remercie également Marc Boyron, l'électronicien de l'ISM, pour sa disponibilité, sa réactivité et son travail d'une immense qualité. Merci aussi pour tout tes conseils, et petites ruses de sioux, ainsi que ta bonne humeur et ta gentillesse. Je remercie également Lucas, qui marche dans les traces de Marc. Merci à notre phénoménal Julien Dipéri, le micro mécanicien, pour son enthousiasme, son sourire communicatif et sa joie de vivre inébranlable. Merci pour ton immense travail et ton aide inestimable à la conception de l'œil et du quadrirotor. Merci aussi pour tous tes petits réglages "à l'œil", qui font qu'une CAO peut s'animer pour de vrai. Merci Julien et Marc, rien n'aurait été possible sans vous !

Je tiens également à remercier ceux qui m'ont aidé à mettre un pied dans le monde du Linux embarqué, que ce soit par leurs conseils, recommandations de lecture ou assistance intensive. Je remercie donc Julien Minet du Gipsa-Lab, Mathieu Herbb du LAAS, François Touchard de l'ISM et bien sûr Guillaume Sanahuja de l'UTC pour son aide et son soutien.

Je tiens également à exprimer toute ma gratitude à ma famille, pour leur soutien continue durant ces années. Je remercie également particulièrement mes parents pour m'avoir poussé dans les études, et sans qui je ne serai, aujourd'hui, pas aussi épanoui dans ce que je fais.

Enfin, je remercie de tout mon cœur ma femme Dounia, pour son soutien sans faille, sa compréhension et pour avoir supporté et accepté tout ce que j'ai pu lui faire endurer et lui imposer ; car une thèse déborde bien souvent sur la vie privée. Je pense que Dounia a su donner, durant ces années, toute sa dimension au mot *patience* et je ne la remercierai jamais assez.

Table des matières

	Remerciements	i
I	Introduction	1
1	Introduction	1
2	Structure de la thèse	1
3	Publications	3
3.1	Brevet	3
3.2	Chapitres de livres	3
3.3	Articles de journaux	4
3.4	Articles de conférences internationales	4
3.5	Autres communications	5
II	Vers une stabilisation bio-inspirée sans accéléromètre	7
1	Introduction	8
2	Présentation du système	10
2.1	Environnement de simulation Processor-In-the-Loop (PIL)	10
2.2	La biologie comme source d’inspiration	12
2.3	Modèle non linéaire du robot	15
3	Vol stationnaire par le regard	16
3.1	L’observateur non-linéaire	17
3.2	Estimation du biais du gyromètre	18
3.3	Le contrôleur de l’œil	19
3.4	Pré-bouclage en vitesse de giration	20
3.5	Contrôleur d’attitude et de position	21
4	Résultats de simulation	21
4.1	Fonctionnement et rôle des réflexes	21
4.2	Robustesse face aux perturbations	22
5	Nouvelles perspectives découlant du contrôle du regard	22
5.1	Vol en saccades	25
6	Les avantages d’un œil découplé	28

7	Conclusion	31
7.1	Application	32
III	Quad-rotor open-source : X4-MaG	35
1	Introduction	37
1.1	Objectifs et contexte	37
1.2	Choix de la plateforme	38
2	Design et dimensionnement	38
2.1	Structure et composants	38
2.2	Modules de calcul	39
2.3	Architecture logicielle et matérielle	40
2.4	Arène de vol et conditions expérimentales	45
2.5	Groupe de propulsion et asservissement local	46
3	Formalisme et angles d'Euler	56
4	Modélisation complète du quadrirotor	58
4.1	Bilan des forces	59
4.2	Bilan des moments	63
4.3	Modèle complet	67
4.4	Identification des paramètres	69
5	Protocoles de calibrations	75
5.1	Calibrations des Accéléromètres	75
5.2	Calibrations des Gyromètres	78
5.3	Estimations en lignes des biais	78
6	Développement d'un support logiciel open-source	79
6.1	Motivations	79
6.2	Quelles applications?	80
6.3	Choix Structurels et considérations techniques	80
6.4	Performances	82
6.5	Perspectives pour RT-MaG	83
7	Stabilisation et asservissement de position dans l'arène de vol	85
7.1	Le contrôleur de vitesse des hélices	86
7.2	Estimation d'attitude : le filtre complémentaire	86
7.3	Contrôleurs d'attitude	87
7.4	Contrôle de position en 3D	90
8	Résultats dans l'arène de vol	93
8.1	Comparaison des deux contrôleurs d'attitude	93

8.2	Précision de positionnement	96
9	Conclusion	98
9.1	La plateforme X4-MaG	98
9.2	Perspectives	100
IV	Système occulo-moteur bio-inspiré	101
1	Introduction à la vision de l'insecte	103
1.1	Contexte et objectifs	103
1.2	L'œil de l'insecte	104
1.3	Hyperacuité et vibration	107
1.4	L'œil du robot	108
2	Description de l'œil	109
2.1	Principes de fonctionnement	109
2.2	Caractérisation et réglages optiques	110
2.3	Design du mécanisme de vibration	113
2.4	Conception du mécanisme de découplage	121
2.5	Conception de la carte de traitement des signaux visuels	125
3	Traitement des signaux visuels	129
3.1	Traitement d'une paire de photo-récepteurs	129
3.2	Fusion multi-paires d'un œil	139
3.3	Fusion des 4 yeux	146
3.4	Fusion généralisée pour n yeux	149
4	Simulateur du système occulo-moteur	152
4.1	Modélisation générique du capteur	152
4.2	Modélisation complète du système occulo-moteur	155
4.3	Asservissement visuel	157
4.4	Résultats de simulations du système occulo-moteur	158
5	Résultats expérimentaux	163
5.1	Mesure de position angulaire	164
5.2	Poursuite de cible	166
6	Conclusion du chapitre	169
V	Stabilisations bio-inspirées du quadrirotor X4-MaG	171
1	Introduction	172
1.1	État de l'art	172
1.2	Motivations	174
2	Description du robot complet	175

2.1	Assemblage complet	175
2.2	Modèle pour l'estimation	176
2.3	Contrôle de position et d'attitude	180
3	Stabilisation visuelle et réflexes	180
3.1	La stabilisation visuelle chez l'insecte	182
3.2	Stabilisation visuelle chez X4-MaG	183
3.3	Résultats et discussions	184
4	Stabilisation bio-inspirée	186
4.1	Asservissement en position basé vision	187
4.2	Stabilisation partielle d'attitude sans accéléromètre, décollage et atterrissage automatique	199
4.3	Stabilisation complète d'attitude bio-inspirée et asservissement en position	205
5	Application à la poursuite d'un Rover	217
5.1	Poursuite du Rover basée vision	217
5.2	Atterrissage sur cible mouvante	219
6	Discussions finales et perspectives de travail	221
VI	Conclusion	225
1	Travail accompli	226
2	Discussions	228
3	Travaux futurs possibles	229
4	Perspectives	229
A	Precision of the VICON localization system	231
1	Introduction	231
2	Procedure and conditions	232
3	The metrics	232
4	Results	233
4.1	Discussion	236
B	Influences du système Vicon et des Néons sur l'œil	239

Table des figures

II.1	Structure Processor-In-the-Loop avec la Gumstix Verdex Pro XL6P	11
II.2	Similarité entre le birotor et l'insecte	12
II.3	Schéma bloc simplifié du robot birotor	14
II.4	Schéma bloc complet du birotor	14
II.5	Schéma bloc du contrôleur du birotor	18
II.6	Réponse du birotor à un déplacement latéral d'un mètre	22
II.7	Influence du découplage tête-corps vis à vis du rejet de perturbation	24
II.8	Evolution des principaux signaux lors d'un saut de cible	29
II.9	Zoom sur l'évolution de l'erreur rétinienne lors d'un saut de cible	30
II.10	Mouvement du robot lors d'un saut de cible	31
II.11	HyperRob et son œil	32
II.12	Poursuite du panneau texturé par HyperRob	33
II.13	Poursuite de mains par HyperRob	34
III.1	Interconnexions des différents matériels de X4-MaG	42
III.2	Structure logicielle de X4-MaG	44
III.3	L'arène de vol de Marseille	45
III.4	Interfaces de l'asservissement des rotors	48
III.5	Réponses en boucle ouverte des différents rotors	49
III.6	Modèles dynamique des rotors en fonction du point de fonctionnement	50
III.7	Boucle fermée de contrôle des rotors	52
III.8	Résultats de l'asservissement de vitesse des rotors	54
III.9	Réponses en vitesses en boucle fermée des différents moteurs	55
III.10	Passage du repère inertiel \mathcal{I} au repère fixe du robot \mathcal{B}	56
III.11	Expression des vitesses de rotations liées au angles d'Euler et le repère fixe.	57
III.12	Convention de rotations des hélices de X4-MaG	59
III.13	Orientation du contre couple d'inertie et du flapping	64
III.14	Banc d'identification des coefficients aérodynamique des hélices	71
III.15	Estimations des coefficients de frottements	74

III.16	Protocole de calibration des accéléromètres	76
III.17	Estimation en ligne des biais gyromètres	78
III.18	Schéma de communication d'une application RT-MaG	81
III.19	Diagramme de génération de RT-MaG	82
III.20	Librairie RT-MaG pour Simulink	82
III.21	Répartition temporelle des périodes d'échantillonnage d'un application RT-MaG	83
III.22	Modèle Simulink de l'autopilote de X4-MaG	88
III.23	Implémentation du générateur de trajectoire	92
III.24	Récupération automatique par le <i>mode de secours</i>	93
III.25	Perturbation artificielle d'attitude	94
III.26	Rejet de perturbations d'attitude	95
III.27	Erreur de poursuite du contrôle en attitude de X4-MaG	97
III.28	Erreur de positionnement lors de vol stationnaire	98
III.29	Poursuite de trajectoire de position par X4-MaG	99
IV.1	Les cibles reconnues par l'œil du robot	104
IV.2	L'œil composé de l'insecte	105
IV.3	Mécanisme de l'hyperacuité chez l'insecte et le robot	108
IV.4	Œil à 6 pixels	110
IV.5	Principe de localisation de cible	111
IV.6	L'œil et ses degrés de liberté	112
IV.7	Banc de caractérisation des paramètres optiques	112
IV.8	Caractérisation et réglages des paramètres optiques	115
IV.9	Système de vibration de l'œil	116
IV.10	Géométrie de la vibration de l'œil	117
IV.11	La carte driver de la vibration de l'œil	119
IV.12	Dispositif de vérification de la vibration de l'œil	120
IV.13	Caractéristiques de la vibration de l'œil	120
IV.14	Vue éclatée de l'œil	123
IV.15	Œil final assemblé	124
IV.16	Identification des micro servomoteurs de l'œil	125
IV.17	Carte de traitement des photodiodes	126
IV.18	Filtrage analogique des photodiodes	127
IV.19	Diagramme de bode du filtrage analogique	128
IV.20	Traitement capteur VODKA	130
IV.21	Sorties caractéristiques du capteur VODKA	131
IV.22	Traitement EyeRing d'une paire de photo-récepteurs	135

IV.23	Démodulation des signaux photodiodes	136
IV.24	Réponse du capteur à différentes fréquences de défilement	137
IV.25	Dispositif de calibration des paires de photodiodes	140
IV.26	Calibration en amplitude des photodiodes	141
IV.27	Filtrage de la caractéristique statique	143
IV.28	Caractéristique des LPU de l'œil 1	144
IV.29	Critères de détection de du ZSL	144
IV.30	Diagramme de la fusion multi-paire d'un œil	146
IV.31	Mesure fusionnée de chaque œil	147
IV.32	Fusion multi-œil généralisée	150
IV.33	Œil de 6 pixels face à une série de contrastes	153
IV.34	Schéma bloc du simulateur de photodiode	154
IV.35	Passage du repère fixe du robot \mathcal{B} aux repères de chaque œil \mathcal{O}_i	156
IV.36	Boucle d'asservissement visuel	158
IV.37	Design du correcteur pour le VFR	159
IV.38	Influence de la vitesse de défilement de la cible sur la mesure de position	160
IV.39	Influence de la luminosité sur la mesure de position de l'œil	162
IV.40	Précision de localisation d'une croix immobile	164
IV.41	Répartition des échantillons de la mesure de position pour chaque œil	165
IV.42	Position mesurée pour des mouvements découplés	167
IV.43	Poursuite de cible vidéo 9584	168
IV.44	Poursuite de cible vidéo 9586	169
IV.45	Asservissement de l'erreur rétinienne	170
V.1	Robot X4-MaG et son œil	175
V.2	Localisation de la cible et repères pour l'estimation	178
V.3	Séquence photographique mettant évidence le VOR	184
V.4	Stabilisation visuelle sur le robot HyperRob	185
V.5	Diagramme en bloc de l'EKF 1	187
V.6	Estimations de position avec l'EKF 1	195
V.7	Estimations de vitesse avec l'EKF 1	196
V.8	Orientation du regard et attitude lors de déplacements latéraux	197
V.9	Diagramme en bloc de l'EKF 2	200
V.10	Estimation des angles de roulis et tangage basée vision	203
V.11	Estimation des positions et vitesses basée vision (EKF2)	204
V.12	Mesure de l'angle de lacet et imperfections de l'œil	206
V.13	Estimation de l'angle de lacet avec l'œil	207

V.14	Recherche de la pondération idéale de la mesure de lacet de l'œil	209
V.15	Diagramme en bloc de l'EKF 3	210
V.16	Estimation de position, vitesse et attitude complète avec l'EKF complet	214
V.17	Poursuite d'un Rover muni d'une cible	216
V.18	Poursuite d'un Rover non coopératif	218
V.19	Atterrissage sur cible mouvante non coopérative	220
V.20	Estimation de distance par louvoisement	224
A.1	Experimental setup used to assess the precision of the Vicon system	233
A.2	Screenshot of the Tracker software used in the flying arena	234
A.3	Diagram of the precision depending on the position of the object in the flying arena	234
A.4	Recording of the 36 tested positions tested to assess the precision of the flying arena	235
B.1	Signaux photodiodes et mesure de position en fonction de l'environnement	240
B.2	Bruit photodiode lié au Vicon	241

Liste des tableaux

II.1	Caractéristiques des capteurs et actionneurs	16
II.2	Paramètres de simulation.	28
III.1	Paramètres des modèles dynamiques des rotors pour différents points de fonctionnement	51
III.2	Paramètres moyens et plage de variations des modèles des rotors	51
III.3	Timing d'une application RT-MaG constituée de 4 tâches	84
III.4	Comparaison des latences mesurées pour PREEMPT-RT seul et XENOMAI	84
III.5	Précisions de positionnement du quadrirotor X4-MaG dans différents modes de vol	99
IV.1	Caractéristiques optique de chaque œil	114
IV.2	Masses des différents éléments de l'œil	122
IV.3	Paramètres des micro servomoteurs de l'œil	126
IV.4	Nouveau filtrage paires photo-récepteurs	134
IV.5	Résumé des précisions des différents yeux	148
V.1	Performances de la poursuite visuelle et des estimation de l'EKF 1.	198
V.2	Statistiques sur les erreurs d'estimations de l'EKF 2	205
V.3	Précision des estimations lors d'une trajectoire circulaire	215
V.4	Précision des estimations lors de la poursuite d'un Rover	219
V.5	Précision des estimations lors de l'atterrissage	221
A.1	The standard deviation and maximum deviation measured at 6 of the 36 positions tested	235
A.2	Statistics about the Vicon precision in the Flying Arena	236

Chapitre I

Introduction

Sommaire

1	Introduction	1
2	Structure de la thèse	1
3	Publications	3
3.1	Brevet	3
3.2	Chapitres de livres	3
3.3	Articles de journaux	4
3.4	Articles de conférences internationales	4
3.5	Autres communications	5

1 Introduction

La problématique de cette thèse s’inscrit dans une démarche visant à mieux comprendre les mécanismes sensorimoteurs associés au vol des insectes et notamment la mouche. Quelles informations les insectes utilisent et lesquelles sont indispensables pour mettre un vol stationnaire de grande précision ? Quels sont les capteurs dont ils disposent ? Autant de questions auxquelles il est possible d’apporter des éléments de réponse en créant des robots présentant des similitudes sensorielles avec la mouche afin de proposer et/ou de valider des hypothèses biologiques. Cette thèse a également été l’occasion de développer une plateforme robotique aérienne open-source, fiable, et générique, permettant d’embarquer divers capteurs bio-inspirés et de valider plus rapidement des principes et hypothèses de contrôle.

2 Structure de la thèse

Ce manuscrit est constitué de 5 chapitres dont voici un rapide descriptif :

Chapitre II : Ce chapitre introduit la problématique de la thèse qui est de proposer des stratégies de stabilisation d'un micro drone en se limitant aux capteurs que l'on sait *a priori* présents chez l'insecte. On propose dans un premier temps de se restreindre au cas d'un birotor attaché au bout d'un bras, afin d'introduire les bases de la stratégie bio-inspirée de stabilisation sans accéléromètre. Ce chapitre prend soin de décrire en détail le découplage entre la tête (centre du système visuel) et le corps du robot qui est un point clé de la stratégie proposée. On démontre alors qu'il est possible de se passer des accéléromètres classiquement utilisés en avionique mais inexistant chez l'insecte. Puis, une discussion permettra de décrire les avantages et les perspectives offertes par ce découplage et la stratégie de stabilisation qui en découle. Enfin on présente un autre type d'application de ce découplage tête-corps lié cette fois ci à une application d'odométrie basée sur la vision et notamment sur la capacité d'un œil composé artificiel à localiser des objets avec une hyperacuité.

Chapitre III Ce chapitre se consacre à la description du quadrirotor X4-MaG open-source, développé durant la thèse. Le développement d'une nouvelle plateforme quadrirotor entièrement ouverte se justifie par la nécessité de maîtriser entièrement l'ensemble des boucles de contrôles nécessaires à la mise en œuvre et à la validation des stratégies bio-inspirées décrites dans la suite de la thèse. Dans un premier temps, le dimensionnement, le choix des différents composants ainsi que l'architecture matérielle et logicielle sont décrits. Puis, un asservissement des différents rotors est proposé afin d'obtenir un asservissement en poussée, indispensable aux algorithmes d'estimations. Un modèle complet de quadri-rotor pour la simulation puis un modèle simplifié pour le contrôle est ensuite présenté. On décrit également les méthodes de calibration des capteurs et d'identification des différents paramètres du modèle qui ont été mises en œuvre sur X4-MaG. La principale originalité de cette plateforme est que son autopilote est entièrement réalisé sous Matlab/Simulink grâce à une nouvelle toolbox développée durant ma thèse ; cette toolbox permettant de générer automatiquement une application temps réel exécutée dans un environnement Linux embarqué. Enfin, différentes stratégies de contrôle d'attitude sont comparées et évaluées dans une arène de vol, puis les différents modes de sécurités instaurés pour fiabiliser l'essai de nouvelles stratégies de contrôle dans l'arène sont décrits.

Chapitre IV : Les stratégies de stabilisation proposées dans cette thèse reposent sur un œil capable de poursuivre efficacement une cible mobile en tirant avantage de son découplage par rapport au reste du robot. Ce chapitre se focalise sur le design d'un système occulo-moteur bio-inspiré complet composé de 4 yeux élémentaires reproduisant les principales caractéristiques de la vision de l'insecte, à savoir un faible nombre de pixel et une sensibilité angulaire gaussienne pour chaque pixel. Ce nouvel œil est capable de localiser l'orientation d'une cible selon 3 axes et capable de la poursuivre en se réorientant grâce à un découplage mécanique présent sur 2 axes (tangage et roulis). Le chapitre commence donc par décrire les similitudes entre l'œil artificiel et celui

de l'insecte puis introduit le concept d'hyperacuité. L'électronique et la mécanique de l'œil sont ensuite présentées ainsi que la caractérisation et le réglage des différents paramètres optiques. Le traitement des signaux de chaque œil est décrit précisément ainsi qu'un algorithme de fusion permettant de remonter à la position angulaire d'une cible et à son orientation. On présente également le simulateur du système occulo-moteur ayant servi à mettre en place les algorithmes de calibration et les asservissements visuels. Enfin les résultats expérimentaux concernant la poursuite de cible par l'œil sont présentés afin de mettre en évidence sa robustesse vis à vis de différents paramètres.

Chapitre V : Le dernier chapitre décrit en détail l'extension au vol libre de la stratégie de stabilisation bio-inspirée sans accéléromètres présentée en chapitre II. On présente alors le système robotique complet, obtenu en combinant le quadrirotor présenté dans le chapitre III et l'œil bio-inspiré du chapitre IV. On commence par montrer que le système oculomoteur peut être utilisé efficacement pour estimer à l'aide d'un EKF la position du robot par rapport à une cible. Cette première application reposant encore sur des techniques d'estimation d'attitude classique (utilisant les accéléromètres), permet également de poursuivre cette cible lorsqu'elle est en mouvement. Puis on utilise ensuite un second EKF, d'abord d'ordre réduit, pour estimer l'ensemble des états du robot excepté les états liés à l'axe de lacet. Puis un troisième EKF est décrit pour estimer cette fois l'attitude complète du robot, sa position et ses vitesses de translation sur la seule base des gyromètres et de son système de vision. On présente alors des expériences démontrant la capacité du robot à poursuivre une cible mouvante et à atterrir sur cette cible en mouvement.

3 Publications

Ces travaux de thèse ont donné lieu à plusieurs publications et communications scientifiques :

3.1 Brevet

- B1 S. Viollet, **A. Manecy**, F. Ruffier, J. Dipéri, *EyeRing, Système de détection d'une cible contrastée*. Brevet FR.15/54051, 2015.

3.2 Chapitres de livres

- CH1 F. Colonnier, **A. Manecy**, R. Juston, S. Viollet, *Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye*. Biomimetic and Biohybrid Systems, Volume 9222, pp 153-163, Papers of the 4th international conference on biomimetics and biohybrid systems (Living Machines 2015), Springer, 2015. [[Colonnier et al., 2015b](#)]

- CH2 **A. Manecy**, R. Juston, N. Marchand, S. Viollet, *Decoupling the Eye : a Key toward a Robust Hovering for Sighted Aerial Robots*. Advances in Aerospace Guidance, Navigation and Control, pp 317-336, Selected Papers of the 2nd CEAS Specialist Conference on Guidance, Navigation & Control. Springer, 2013. [[Manecy et al., 2013a](#)]

3.3 Articles de journaux

- J1 **A. Manecy**, N. Marchand, F. Ruffier, S. Viollet, *X4-MaG : A Low-Cost Open-Source Micro-Quadrotor and its Linux-based Controller*, International Journal of Micro Air Vehicles, 2015. [[Manecy et al., 2015](#)]
- J2 F. Colonnier, **A. Manecy**, R. Juston, H. Mallot, R. Leitel, D. Floreano, S. Viollet, *A tiny hyperacute compound eye featuring active eye tremor : application to visual stabilization, target tracking and short range odometry*, Journal of Bioinspiration & Biomimetics, 2015. [[Colonnier et al., 2015a](#)]
- J3 **A. Manecy**, N. Marchand, S. Viollet, *Hovering by Gazing : a Novel Strategy for Implementing Saccadic Flight-Based Navigation in GPS-Denied Environments*, International Journal of Advanced Robotic System, 2014. [[Manecy et al., 2014a](#)]

3.4 Articles de conférences internationales

- C1 F. Colonnier, **A. Manecy**, R. Juston, S. Viollet, *Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye*. 4th international conference on biomimetics and biohybrid systems : Living Machines 2015. La Pedrera Barcelona, Spain, July 28-31, 2015. *Prix Valentino Braitenberg* [[Colonnier et al., 2015c](#)]
- C2 **A. Manecy**, J. Diperi, M. Boyron, N. Marchand, S. Viollet, *A New Bio-Inspired Eye to Achieve Accurate Hovering, Positioning and Fast Tracking*, ODAS 2015, 15th ONERA-DLR Aerospace Symposium, Toulouse, France, May 27-29, 2015.
- C3 **A. Manecy**, N. Marchand, S. Viollet, *RT-MaG : an open-source SIMULINK Toolbox for Linux-Based Real-Time Robotic Applications*, ROBIO 2014 IEEE, International Conference on Robotics and Biomimetics, Bali, Indonesia, December 5-10, 2014. [[Manecy et al., 2014b](#)]
- C4 **A. Manecy**, N. Marchand, F. Ruffier, S. Viollet, *Implementation of an open hardware-software quadrotor and its 3D control in a flying arena*, 2nd AETOS International Conference on Research challenges for future RPAS/UAV systems, Bordeaux, France, September 9-10, 2014.
- C5 **A. Manecy**, R. Juston, N. Marchand, S. Viollet, *Decoupling the Eye : a Key toward a Robust Hovering for Sighted Aerial Robots*. EuroGNC 2013, 2nd CEAS Specialist Conference on Guidance, Navigation & Control, Delft, Netherland, April 10-12, 2013. [[Manecy et al., 2013b](#)]

- C6 **A. Manecy**, N. Marchand, S. Viollet, *Bio-Inspired Hovering Control for an Aerial Robot Equipped with a Decoupled Eye and a Rate Gyro*. IROS 2012 IEEE/RSJ, International Conference on Intelligent Robots and Systems, Vilamoura Algarve, Portugal, October 7-12, 2012. [[Manecy et al., 2012](#)]
- C7 R. Munguia, **A. Manecy**, *State Estimation for a Bio-Inspired Hovering Robot Equipped with an Angular Sensor*. CEE 2012, 9th International Conference on Electrical Engineering, Computing Science and Automatic Control, Mexico city, Mexico, September 26-28, 2012. [[Munguia et Manecy, 2012](#)]

3.5 Autres communications

- A1 **A. Manecy**, Présentation invitée : *La robotique bio-inspirée*, 4ème rencontre du Pôle Grenoble Cognition, GIPSA-Lab, Grenoble (38), France, 4 Juin 2013.
- A2 **A. Manecy**, Présentation : *Conception et mise en œuvre de drones*, TechDays 2013, 1ère journées technologiques Robotex, LAAS, Toulouse (31), 3-4 Juillet 2013.
- A3 **A. Manecy**, F. Colonnier, Présentation : *Présentation de la plateforme ROBOTEX : Arène de vol*, TechDays 2013, 1ère journées technologiques Robotex, LAAS, Toulouse (31), 3-4 Juillet 2013.
- A4 **A. Manecy**, G. Sanahuja, J. Dumon, F. Elisei, F. Colonnier, Session thématique : *Systèmes de capture du mouvement*, TechDays 2013, 1ère journées technologiques Robotex, LAAS, Toulouse (31), 3-4 Juillet 2013.
- A5 **A. Manecy**, Présentation : *X4-MaG et RT-MaG : Plateforme hardware et software de robotique volante*, TechDays 2014, 2ème journées technologiques Robotex, GIPSA-Lab, Grenoble (38), 3-4 Avril 2013.
- A6 **A. Manecy**, S. Viollet, N. Marchand, Poster : *X4-MaG : open hardware and software Quadrotor and the RT-MaG toolBox*, Stand GDR Robotique InnoRobo, Cité internationale, Lyon (38), France, 18-20 Mars 2014.
- A7 F. Colonnier, **A. Manecy**, S. Viollet, F. Ruffier, Démonstrateur : *Présentation d'un démonstrateur de Curvace (Curved Artificial Compound Eye)*, Stand GDR Robotique InnoRobo, Cité internationale, Lyon (38), France, 18-20 Mars 2014.

Chapitre II

Vers une stabilisation bio-inspirée sans accéléromètre

Sommaire

1	Introduction	8
2	Présentation du système	10
2.1	Environnement de simulation Processor-In-the-Loop (PIL)	10
2.2	La biologie comme source d'inspiration	12
2.3	Modèle non linéaire du robot	15
3	Vol stationnaire par le regard	16
3.1	L'observateur non-linéaire	17
3.2	Estimation du biais du gyromètre	18
3.3	Le contrôleur de l'œil	19
3.4	Pré-bouclage en vitesse de giration	20
3.5	Contrôleur d'attitude et de position	21
4	Résultats de simulation	21
4.1	Fonctionnement et rôle des réflexes	21
4.2	Robustesse face aux perturbations	22
5	Nouvelles perspectives découlant du contrôle du regard	22
5.1	Vol en saccades	25
6	Les avantages d'un œil découplé	28
7	Conclusion	31
7.1	Application	32

1 Introduction

L'autonomie des aéronefs ou drones, semble être un des points clés de la recherche en robotique aérienne pour les prochaines années. Une des premières étapes, pour atteindre cette autonomie, est la stabilisation en attitude des véhicules aériens. Aujourd'hui, cet objectif est classiquement atteint grâce à l'utilisation des centrales inertielles 9 axes (gyromètres 3 axes, accéléromètres 3 axes et magnétomètres 3 axes), comme par exemple, dans [Mahony *et al.*, 2012, Lim *et al.*, 2012, Barton, 2012]. Mais, ces centrales inertielles doivent souvent être combinées avec d'autres capteurs, afin d'estimer la position de l'engin volant sans dérive. Dans des applications extérieures, le GPS est très largement utilisé, comme dans [Wendel *et al.*, 2006, Abdelkrim *et al.*, 2008, Nemra et Aouf, 2010]. Mais de nombreuses techniques consistent également à fusionner les informations inertielles avec des informations venant de la vision, lorsque les signaux GPS ne sont pas disponibles (typiquement pour de la navigation en intérieur). Par exemple, [Gurdan *et al.*, 2007], ont choisi d'utiliser un système de trajectométrie externe capable de déterminer avec précision la position et l'attitude d'un robot équipé de marqueurs. De la même manière, [Kim *et al.*, 2010] utilisent une simple caméra CCD pour mesurer la position de deux LEDs embarquées à bord d'un quadricoptère. Mais, bien que ce genre de systèmes fournissent de très bons résultats, ils ne sont pas embarqués à bord du robot lui-même, et nécessitent bien souvent d'importantes ressources calculatoires fournies, en général, par une station sol. Dans d'autres études, le capteur visuel est embarqué à bord du robot. [Mori *et al.*, 2007a] utilisent deux caméras embarquées pointant sur deux marqueurs elliptiques, l'un placé en face du robot et l'autre en dessous, de façon à déterminer la position et l'attitude du robot par rapport à ces marqueurs. Dans la même idée, cinq marqueurs de formes différentes ont été utilisés par [Zhang *et al.*, 2009a] pour estimer le lacet et la position d'un robot volant, alors qu'une centrale inertielle fournissait le reste de l'attitude et les vitesses de rotations de l'engin. [Wenzel *et al.*, 2009] ont préféré combiner une centrale inertielle avec une solution bas coût basée sur le capteur de la télécommande Wii (Nintendo), pour stabiliser, en vol stationnaire, un multicoptère. [Yang *et al.*, 2013a] ont proposé de fusionner l'estimation de 5 degrés de libertés, obtenue avec une caméra monoculaire, avec le vecteur de gravité mesuré par une IMU pour permettre à un quadricoptère de décoller, faire du vol stationnaire et atterrir. D'autres techniques permettent de stabiliser un quadricoptère en éliminant les vitesses linéaires perçues par des capteurs de flux optiques ([Herissé *et al.*, 2012], [Kendoul *et al.*, 2009]). Le même flux optique a également été utilisé pour estimer l'altitude et la vitesse de translation d'un véhicule aérien dans [Rondon *et al.*, 2010]. [Bošnjak *et al.*, 2012] ont proposé une solution basée sur une caméra également capable de recalibrer les dérives de la centrale inertielle, tout en étant robuste aux forts retards des algorithmes de traitements d'images. Une caméra est également utilisée dans [Ahrens *et al.*, 2009] pour accomplir à la fois la tâche de vol stationnaire sans dérive et d'évitement d'obstacles à bord d'un quadricoptère.

Dans toutes ces études, l'attitude des robots est estimée en utilisant les accéléromètres de l'IMU, afin d'estimer et/ou de compenser les biais des gyromètres. Mais certains auteurs ont également proposé des techniques basées sur une caméra panoramique embarquée afin d'estimer l'attitude du robot (par exemple, [Gurdan *et al.*, 2007]). De même, certains auteurs proposent des techniques d'asservissement visuel direct (visual servoing) pour stabiliser en vol stationnaire un quadrirotor équipé d'une caméra orientée vers le bas ([Kendoul *et al.*, 2010]). [Ozawa et Chaumette, 2011] proposent par exemple, toujours dans des techniques de visual servoing, d'améliorer la stabilité en ajoutant un ressort virtuel au système de contrôle.

Dans notre cadre de travail qui se focalise sur les algorithmes bio-inspirés minimalistes, nous proposons ici une nouvelle stratégie de pilotage, ne reposant que sur trois capteurs :

- un capteur optique ayant un champ de vision très restreint et capable de localiser une cible avec une grande précision ([Kerhuel *et al.*, 2012], [Juston *et al.*, 2013])
- un gyromètre bas coût,
- un capteur proprioceptif permettant de mesurer l'orientation de l'œil (capteur visuel) par rapport au corps du robot (œil découplé).

Ici, le choix d'utiliser un œil découplé, s'explique par la recherche de réduire au minimum la complexité de calcul liée à la vision. En effet, avec un champ de vision très fin, seuls quelques pixels sont nécessaires pour localiser une cible. Le champ de vision est ensuite "augmenté", grâce à un découplage mécanique, qui permet de poursuivre la cible dans un large espace angulaire.

Dans de précédentes applications robotiques, un tel découplage mécanique était utilisé uniquement à des fins de stabilisation de caméra et de poursuite d'une cible. Une telle poursuite a été réalisée avec succès par [Mondragón *et al.*, 2010b], avec un hélicoptère autonome, pour lequel la position et l'attitude étaient obtenues par hybridation des signaux GPS et de la centrale inertielle.

Dans cette thèse, je propose de ne plus utiliser le découplage uniquement à des fins de poursuite, mais d'utiliser également l'information d'orientation du regard, pour remonter simultanément aux informations de position et d'attitude.

Pour cela, on choisit d'utiliser un œil ayant un champ de vision restreint à quelques degrés (une sorte de *fovea*¹), pour les raisons suivantes :

- un petit champ de vision est synonyme d'un petit nombre de pixels, et donc d'une petite région d'intérêt,
- se cantonner à une petite région d'intérêt permet d'atteindre des taux de rafraichissement très élevé,
- une *fovea* permet de réduire fortement la complexité calculatoire liée à la vision, comme décrit dans [Ballard, 1991]. Cela réduit donc la puissance de calcul nécessaire.

1. Une *fovea* est une partie de l'œil où la vision des détails est la plus précise. Elle est située en zone centrale, où le besoin de détails est d'une grande importance.

L'œil du robot est assimilé à un capteur de position angulaire, capable de localiser un contraste (un front ou une barre) avec une grande précision, du moment que ce contraste se trouve dans un champ de vision (supposé ici de l'ordre de $FOV = \pm 5^\circ$, par opposition au champ de vision classiquement rencontré de l'ordre de $\pm 25^\circ$). Ce capteur est donc doté d'hyperacuité ([Westheimer, 1981] et voir chapitre IV pour plus de détails), c'est-à-dire qu'il a la capacité à localiser une cible avec une précision meilleure que la résolution dictée par l'angle séparant les pixels. Des exemples d'implémentation de tels capteurs, dotés d'hyperacuité, sont décrits dans [Viollet et Franceschini, 2005, Kerhuel *et al.*, 2010, Kerhuel *et al.*, 2012, Juston et Viollet, 2012, Juston *et al.*, 2013]. Nous décrirons également, en détail, une évolution de ce genre de capteur dans le chapitre IV, afin d'étendre le concept de localisation 1D au cas 2D.

Je commencerai par décrire le robot et son œil découplé dans la section 2, ainsi que le modèle non linéaire correspondant. L'ensemble des résultats présentés dans ce chapitre, ont été obtenus dans le cadre de simulation Processor-In-the-Loop (PIL), afin de vérifier les capacités temps réel des algorithmes développés. Cette section décrira donc également le système utilisé. Dans la section 3, nous présenterons un observateur non linéaire, permettant d'estimer la vitesse de translation, la position, l'attitude (se résumant ici à un angle de roulis), et le biais du gyromètre. Ensuite, nous introduirons le contrôleur de regard, qui se base sur ces estimations. Puis, nous terminerons par la section 5, dans laquelle je présenterai une nouvelle technique de navigation, basée sur un vol guidé par des saccades oculaires. Enfin, nous discuterons des avantages que peuvent apporter le découplage du regard, dans le cadre de la robotique aérienne.

2 Présentation du système

Dans cette partie, je décrirai d'abord, succinctement, le système de validation PIL utilisé pour les simulations de ce chapitre. Puis, nous expliquerons comment la biologie a inspiré notre approche et finirons par une présentation succincte du birotor et de son modèle de simulation.

2.1 Environnement de simulation Processor-In-the-Loop (PIL)

L'ensemble des résultats qui sont présentés ici, ont été réalisés en Processor-In-the-Loop. C'est-à-dire que le modèle du robot était simulé sur un ordinateur en temps réel, mais que les algorithmes de contrôle étaient exécutés en temps réel, sur le calculateur final. La figure II.1 montre donc l'environnement de simulation et les liens existants entre le calculateur embarqué et le PC hôte. L'ordinateur hôte simule le comportement dynamique du robot en temps réel, mais permet aussi de superviser et de visualiser les calculs faits à bord du calculateur embarqué.

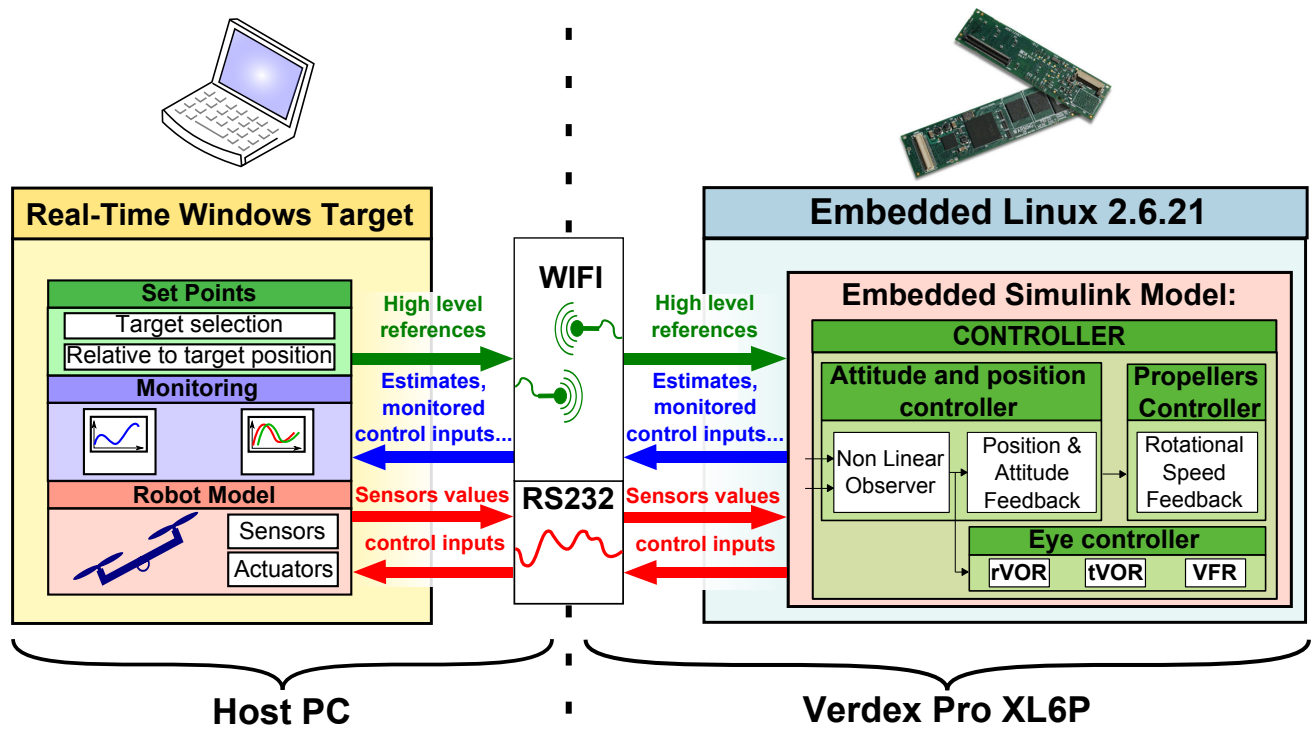


FIGURE II.1 – Description de la structure PIL. Le PC hôte et le ordinateur embarqué sont reliés par un lien Wifi, permettant de superviser les calculs du ordinateur embarqué (la Gumstix VERDEX Pro XL6P), d’émettre les commandes Linux et d’envoyer les consignes haut niveau. Un lien série (UART), permet d’émuler la connexion future entre le ordinateur embarqué et les capteurs du robot.

2.1.1 La carte pilote automatique : Gumstix Verdex Pro XL6P

Au début de ma thèse, nous avons prévu d’équiper le futur quadrirotor avec la carte de contrôle Gumstix Verdex Pro XL6P, qui est une carte de type Computer-On-Module (COM) fonctionnant sous Linux. Pour des raisons de disponibilité et de facteur de forme, nous nous sommes par la suite tournés vers la Gumstix Overo (voir chapitre III) mais cela ne change rien, car cette carte fonctionne également sous Linux. La Gumstix Verdex Pro XL6P est équipée d’un processeur XScale-PXA270 cadencé à 600 MHz, doté de 128MB de DRAM et de 32MB de mémoire Flash. Le choix de ce genre de matériel permet d’assurer, au futur robot, une autonomie complète en terme de ressources calculatoires. De plus, le fait d’avoir un OS embarqué sur la carte de contrôle principale, permet de grandement faciliter son utilisation (connexion via un simple terminal par liaison Wifi).

2.1.2 Outil de prototypage rapide : de Simulink vers la Gumstix

Lors de ces travaux, les boucles de contrôle (contrôleur et observateurs) sont réalisées directement sous Matlab/Simulink. Le modèle Simulink est ensuite converti automatiquement en une application Linux, afin d’être exécuté en temps réel sur la Gumstix. Cela est réalisé à l’aide d’une toolbox que j’ai

écrite durant ma thèse et qui sera décrite plus en détail dans la section 6 du chapitre III. Cet outil de génération de code automatique permet de réduire grandement le temps nécessaire pour développer et tester de nouveaux contrôleurs car :

- il génère et compile directement un code "sans bug",
- il donne accès aux entrées/sorties du système (UART, Wifi, etc.) à travers de simples blocs Simulink,

2.2 La biologie comme source d'inspiration

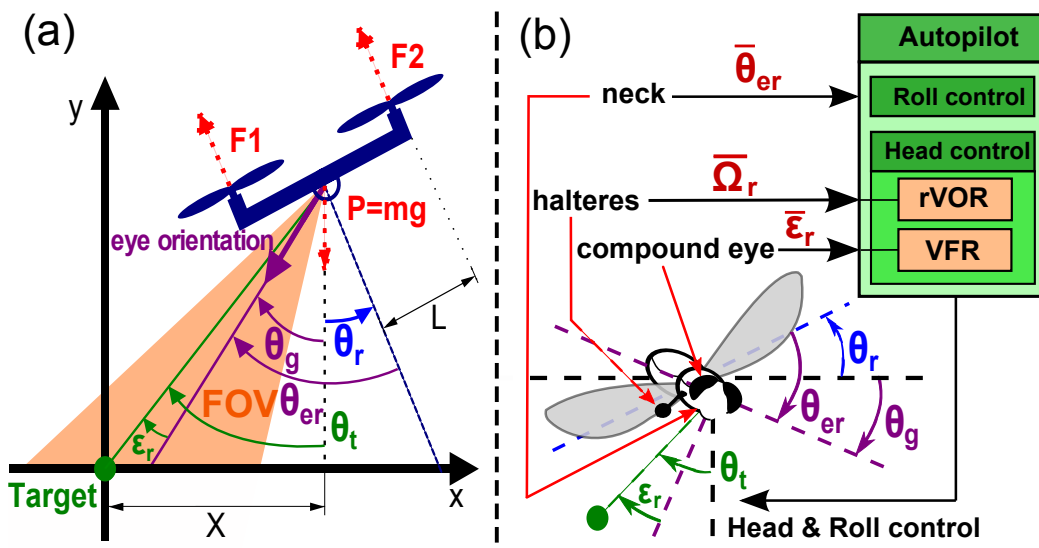


FIGURE II.2 – Similarités entre le birotor et son œil découplé (a) et la mouche (b). Ces deux systèmes sous actionnés, sont capables de mesurer leur vitesse de rotation Ω_r , grâce à un gyromètre (dans le cas du robot) et par de petits balanciers (pour la mouche). Ils sont capables de localiser précisément une cible θ_t placée dans leur petit champ de vision (*fovea*). Alors que la mouche utilise pas moins de 23 paires de muscles pour stabiliser son regard θ_g (en stabilisant sa tête), le robot contrôle l'orientation de son œil θ_{er} grâce à un micro servomoteur. Dans cette figure, on représente le robot et la mouche au-dessus d'une cible placée au sol.

Comme le montre la figure II.2, le robot est capable de se déplacer selon 2 degrés de libertés et on suppose qu'il évolue à une hauteur constante H . Le robot est placé au bout d'un bras et, en jouant sur le différentiel de poussée de ses hélices, il peut ajuster son angle de roulis θ_r , afin de réaliser des translations horizontales le long de l'axe X . Un troisième degré de liberté lui permet de réorienter l'angle de son œil par rapport à son corps (angle θ_{er}), grâce à un découplage mécanique. Le contrôle de ce découplage est réalisé à l'aide d'un micro servomoteur présentant une dynamique rapide.

Comme l'ont par exemple expliqué [Arkin, 1998] et [Webb, 2000], la conception de ce robot et le choix de ses capteurs résulte d'une approche bio-inspirée, où l'on se sert de la biologie comme d'une source

d’inspiration pour les différents choix structurels et le choix des stratégies de contrôles. En ce sens, le robot, avec son gyromètre, sa proprioception et son capteur visuel, regroupe les mécanismes clés observés chez l’insecte en termes de capacités sensorielles (voir [Taylor et Krapp, 2007]), de stabilisation du regard (voir [Hateren et Schilstra, 1999, Boeddeker et Hemmi, 2009, Viollet et Zeil, 2013]), de comportement en vol ([Collett et Land, 1975, Geurten *et al.*, 2010, Niven, 2006, Wijngaard, 2010]) et de localisation de cible ([Nordström *et al.*, 2006]). On résume donc les principales similarités décrites par la figure II.2 ci-dessous :

- **Un gyromètre** : La mouche dispose de petits balanciers gyroscopiques à la base de ses ailes ([Hengstenberg, 1988, Nalbach, 1993, Nalbach et Hengstenberg, 1994]), lui permettant de mesurer ses vitesses de rotations autour de 3 axes (roulis, tangage et lacet). Le robot, quant à lui, est équipé d’un gyromètre de type MEMS, lui permettant de mesurer sa vitesse de rotation autour de son axe de roulis.
- **Un capteur optique de mesure de position angulaire** : Les yeux de la mouche sont capables de localiser avec précision une cible contrastée, placée dans la zone frontale de leur champ visuel ([Collett et Land, 1975] et [Boeddeker *et al.*, 2003], [Wiederman *et al.*, 2008]). L’œil du robot, lui, est doté d’hyperacuité ([Kerhuel *et al.*, 2012] et [Juston *et al.*, 2013]), et peut également localiser une cible contrastée présente dans son champ de vision.
- **Un cou** : les mouvements du cou sont essentiels chez la mouche, qui dispose de pas moins de 23 paires de muscles pour contrôler l’orientation de sa tête ([Strausfeld *et al.*, 1987]). Le robot est équipé d’un mécanisme plus simple, qui permet néanmoins d’orienter son regard selon l’axe de roulis, grâce à un servomoteur rapide (60° en seulement 70 ms) et précis (0.1°).
- **Un capteur proprioceptif dans le cou** : une paire de cils mécano-sensitifs, située à la base du cou, permet à la mouche de mesurer avec précision l’orientation de sa tête par rapport à son corps ([Preuss et Hengstenberg, 1992]). De la même manière, le robot mesure l’orientation de son œil grâce à un capteur de position sans contact (capteur à effet hall).

Dans la première partie de ce travail, on supposera que le robot est en vol stationnaire au-dessus d’une cible unique. Comme le montrent les figures II.3 et II.4, le roulis du robot (θ_r) est contrôlé grâce au différentiel de poussée généré par les deux hélices. [Rondon *et al.*, 2010], [Wenzel *et al.*, 2009] et [Kim *et al.*, 2010] ont établi que, pour les systèmes appartenant à la classe des VTOL (Vertical Taking Off and Landing, c’est-à-dire les véhicules effectuant un décollage et un atterrissage vertical), la dynamique liée à l’altitude peut être complètement découplée. Devant ce fait, on supposera donc ici que ce découplage nous permet de maintenir une altitude H constante tout au long du vol, et l’asservissement en altitude ne sera pas traité. Le contrôle de la position du robot est obtenu, quant à lui, en ajustant l’angle de roulis, sachant que son point d’équilibre est atteint pour un angle de roulis nul : $\theta_r = 0^\circ$.

La figure II.3 montre une vue d’ensemble de l’architecture de contrôle du robot. Cette architecture

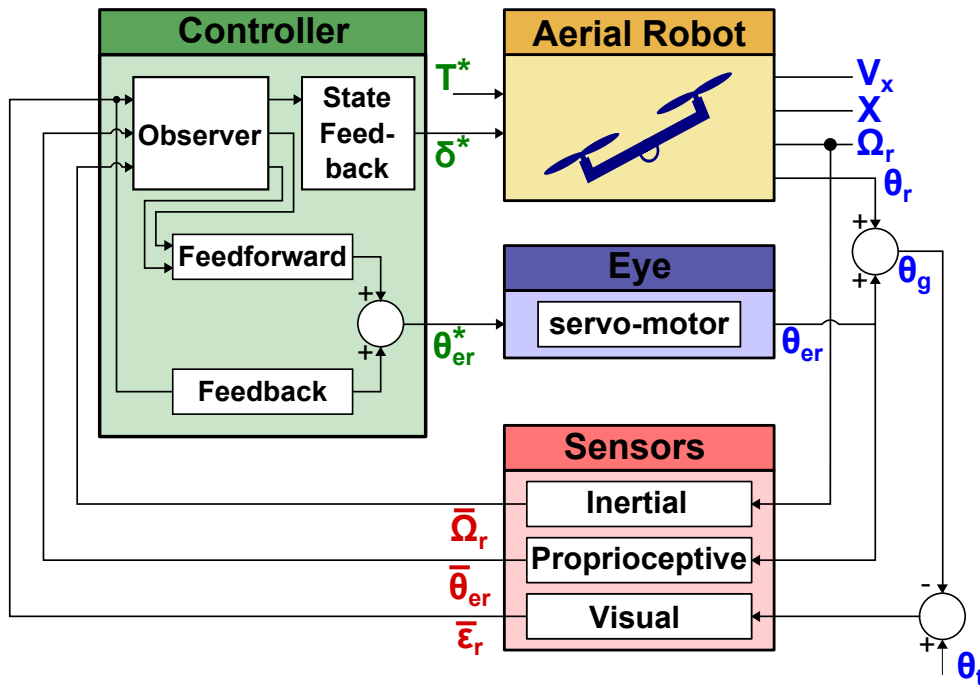


FIGURE II.3 – Schéma bloc simplifié du robot birotor. Du fait du découplage mécanique entre le robot et l’œil, les deux systèmes peuvent être vus comme deux systèmes indépendants.

repose sur un observateur non linéaire estimant les états, et l’attitude et la position sont contrôlées par un contrôle par retour d’état. La figure II.3 présente le robot et son œil comme deux systèmes indépendants recevant chacun ses propres consignes (T^* , δ^* et θ_{er}^*).

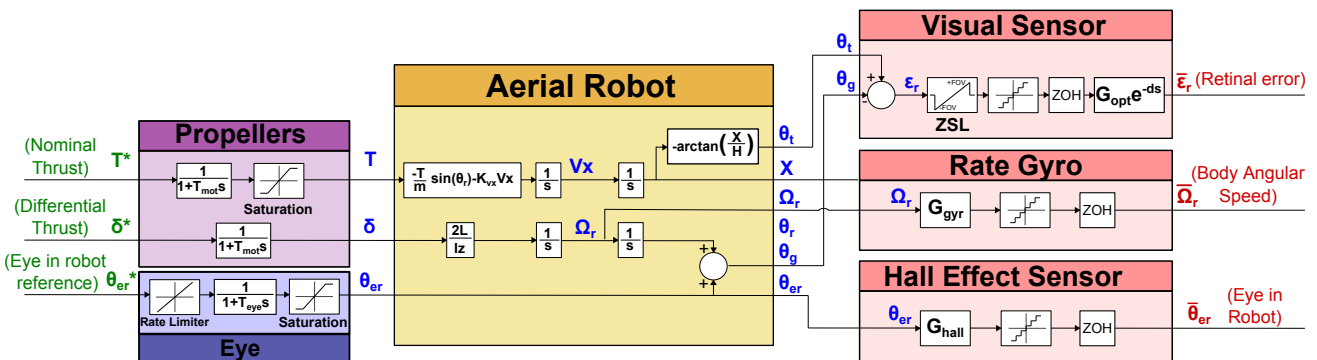


FIGURE II.4 – Schéma bloc complet du birotor. Le robot est équipé d’un gyromètre, mesurant sa vitesse de rotation, et d’un capteur à effet hall, mesurant l’orientation de son œil décollé. Ce dernier mesure la position angulaire d’une cible distante. Les variables écrites en vert, correspondent aux entrées de commandes, les bleues correspondent aux variables physiques d’intérêt, et les rouges, aux signaux mesurés.

Dans la suite, nous utiliserons les notations suivantes :

- θ_t : position angulaire de la cible par rapport au robot dans le référentiel inertiel.

- θ_r : angle de roulis du robot.
- θ_{er} : orientation du regard dans le référentiel du robot. Cet angle est mécaniquement contraint comme suit : $|\theta_{er}| < \theta_{er_MAX}$.
- θ_g : orientation du regard dans le référentiel inertiel, $\theta_g = \theta_{er} + \theta_r$.
- ϵ_r : erreur rétinienne, définie par $\epsilon_r = \theta_t - \theta_g$.
- X : position du robot sur l'axe horizontal du référentiel inertiel.
- V_x : vitesse horizontale du robot dans le référentiel inertiel.
- Y : hauteur de vol stationnaire du robot. Y est ici constante ($Y(t) = H \quad \forall t$).
- Ω_r : vitesse angulaire du robot selon l'axe de roulis.

Dans la suite, on notera avec un chapeau les estimations (par exemple, $\hat{\theta}$), avec une étoile les références ou consignes (par exemple, θ^*) et avec une barre les grandeurs mesurées (par exemple, $\bar{\theta}$).

2.3 Modèle non linéaire du robot

Le modèle dynamique non linéaire du robot résulte simplement du principe fondamental de la dynamique :

$$\begin{cases} \dot{V}_x = \frac{-(F_1 + F_2) \sin(\theta_r)}{m} - K_{vx} V_x \\ \dot{X} = V_x \\ \dot{\Omega}_r = \frac{L(F_2 - F_1)}{I_z} \\ \dot{\theta}_r = \Omega_r \end{cases} \quad (\text{II.1})$$

Où L est la distance séparant les hélices du centre de gravité du robot, I_z est le moment d'inertie autour de l'axe de roulis, F_1 et F_2 sont les poussées respectives des hélices 1 et 2, et K_{vx} est le coefficient de frottements visqueux, supposé ici constant.

Le modèle dynamique de l'œil est assimilé à un système du premier ordre, auquel on rajoute un "rate limiter" (limiteur de pente) correspondant à la vitesse de rotation maximale. Les butées de fin de courses mécaniques sont simplement représentées par des saturations (voir le bloc "Eye" de la figure II.4).

La stabilisation du robot et celle de son œil sont abordées comme deux problèmes indépendants. La stabilisation de l'attitude et de la position du robot est réalisée par un contrôleur haut niveau (voir section 3.5), délivrant des consignes, sous forme de vitesse de giration (Ω_r^*). La poursuite de cette consigne est réalisée par un contrôleur de rotors bas-niveau qui est décrit dans la section 3.4.

On choisit de décomposer l'écriture de la poussée de chaque hélice en la somme d'une poussée nominale notée T et une poussée différentielle, notée δ . La consigne de poussée nominale (T^*) permet

de contrer l'effet de la gravité, alors que la poussée différentielle (δ^*) permet de générer le couple nécessaire pour ajuster l'angle de roulis. On définit ainsi la consigne de poussée de chaque hélice :

$$\begin{cases} F_1^*(p) = T^*(p) - \delta^*(p) \\ F_2^*(p) = T^*(p) + \delta^*(p) \end{cases} \quad (\text{II.2})$$

Ici, la dynamique des hélices est approchée par un système du premier ordre ayant une constante de temps τ_{mot} (voir tableau II.1). La fonction de transfert $F_{1,2}(p)$ est alors donnée par $F_{1,2}(p) = \frac{F_{1,2}^*(p)}{1 + \tau_{mot}p}$

	Capteurs		
	Capteur visuel	Capteur angulaire	Gyromètre
Fonction de transfert	$G_{opt}(p) = G_{opt}e^{-dp}$	$G_{hall}(p) = \frac{1}{1 + \tau_{hall}p}$	$G_{gyr}(p) = \frac{1}{1 + \tau_{gyr}p}$
Résolution	0.1 [°]	0.1 [°]	1 [° · s ⁻¹]
Fréquence d'échantillonnage	40 [Hz]	1 [kHz]	1 [kHz]
Amplitude de bruit	± 0.1 [°]	± 1 [°]	±5[° · s ⁻¹]
	Actionneurs		
	Moteurs des hélices	Servomoteur	
Fonction de transfert	$G_{mot}(p) = \frac{1}{1 + \tau_{mot}p}$	$G_{eye}(p) = \frac{1}{1 + \tau_{eye}p}$	
Limiteur de pente	-	1000 [° · s ⁻¹]	

TABLE II.1 – Caractéristiques des capteurs et actionneurs

3 Vol stationnaire par le regard

Ce travail s'inscrit dans la suite de travaux précédents qui place le regard comme élément central, afin d'atteindre un vol stationnaire robuste et précis. Par exemple, [Kerhuel *et al.*, 2010] avaient développé une première stratégie de " guidage par le regard " afin de maintenir un biorotor orienté face à une cible : le robot poursuit du regard une cible et tente de maintenir son corps aligné avec son regard. Dans de nombreuses études précédentes, ([Mahony *et al.*, 2008], [Salaün *et Martin*, 2010]), l'attitude du robot était résolue à l'aide d'une centrale inertielle (IMU) intégrant des gyromètres, des accéléromètres et/ou des magnétomètres, et la stabilisation en position était atteinte grâce à une caméra ([Meier *et al.*, 2011, Ghadiok *et al.*, 2012, Weiss *et al.*, 2011]). Dans ce travail de thèse, je propose une approche alternative permettant d'apporter quelques éléments de réponses à la question : comment un robot aérien peut-il être stabilisé, uniquement au moyen d'un gyromètre et d'un capteur visuel ?

De récentes études sur le syrphé ([Wijngaard, 2010]) suggèrent que des solutions efficaces existent pour ce problème, et que le système de contrôle du regard est certainement un des éléments clés. Dans cette idée, l'attitude et la position du robot, ainsi que le biais du gyromètre, ne sont déterminés qu'à partir d'une mesure de la vitesse angulaire $\bar{\Omega}_r$, de l'orientation du regard $\bar{\theta}_{er}$ et de l'erreur rétinienne $\bar{\epsilon}_r$. Une stratégie de contrôle, basée sur des réflexes oculomoteurs (voir la section 3.3), permet au robot de maintenir son erreur rétinienne au minimum, et ce, même lors de mouvements brusques.

Dans cette partie, on se propose donc de décrire les différents contrôleurs ainsi que leurs interactions.

3.1 L'observateur non-linéaire

L'observateur non linéaire est une des clés de la stratégie présentée ici, comme le montre la figure II.5. En combinant la mesure d'orientation du regard ($\bar{\theta}_{er}$) avec celle de l'erreur rétinienne ($\bar{\epsilon}_r$), on peut déterminer la position angulaire de la cible ($\bar{\theta}_{tr}$) par rapport à celle du robot :

$$\bar{\theta}_{tr} = \bar{\theta}_{er} + \bar{\epsilon}_r \quad (\text{II.3})$$

Étant donné que la vitesse de rotation du robot ($\bar{\Omega}_r$) est connue (mesurée par le gyromètre), on peut construire un observateur capable d'estimer la vitesse de translation horizontale \hat{V}_x , la position \hat{X} , le roulis $\hat{\theta}_r$ ainsi que le biais du gyromètre $\hat{\Delta}_g$:

$$\left\{ \begin{array}{l} \begin{pmatrix} \dot{\hat{V}}_x \\ \dot{\hat{X}} \\ \dot{\hat{\theta}}_r \\ \dot{\hat{\Delta}}_g \end{pmatrix} = \begin{pmatrix} -\frac{T}{m} \sin(\hat{\theta}_r) - K_{vx} \hat{V}_x + L_1 \tilde{\theta}_{tr} \\ \hat{V}_x + L_2 \tilde{\theta}_{tr} \\ \bar{\Omega}_r - \hat{\Delta}_g + L_3 \tilde{\theta}_{tr} \\ L_4 \tilde{\theta}_{tr} \end{pmatrix} \\ \hat{\theta}_{tr} = \underbrace{-\arctan\left(\frac{\hat{X}}{H}\right)}_{\hat{\theta}_g} - \hat{\theta}_r \end{array} \right. \quad (\text{II.4})$$

$$\text{Où } \tilde{\theta}_{tr} = \underbrace{(\bar{\theta}_{er} + \bar{\epsilon}_r)}_{\tilde{\theta}_{tr}} - \hat{\theta}_{tr}$$

Le choix d'utiliser un filtre non linéaire résulte à la fois de la non linéarité importante reliant les accélérations latérales \dot{V}_x à l'angle de roulis, décrit par l'équation (II.4), et de la non linéarité de la sortie θ_{tr} (qui est une fonction de X , θ_r et H). La non linéarité présente dans \dot{V}_x donne une meilleure estimée de la vitesse durant les phases de translations, et la non linéarité sur $\hat{\theta}_{tr}$ permet d'estimer la position du robot, sans erreurs statiques liées à la linéarisation, et ce, même si le robot n'est pas à l'aplomb de la cible. Le gain de l'observateur $L = (L_1 \ L_2 \ L_3 \ L_4)^T$ a été calculé en utilisant simplement la méthode LQG pour le système linéarisé autour du point d'équilibre.

Remarque II.1: En supposant que la précision angulaire de localisation du capteur visuel est de

l'ordre de 0.1° , le robot est en mesure d'estimer sa position avec une précision de 0.5 cm s'il survole la cible à une hauteur de 2 m. On peut noter que cette précision dépasse fortement celle d'un capteur GPS classique (quelques mètres de précision).

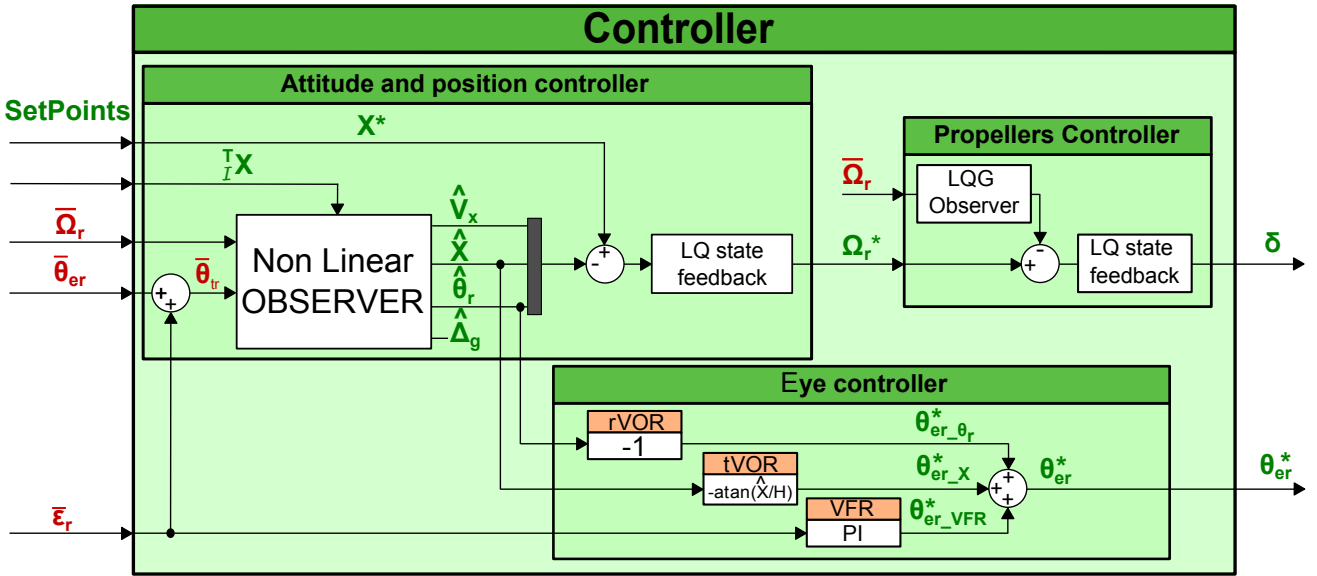


FIGURE II.5 – Schéma bloc du contrôleur du birotor. L’observateur non linéaire détermine l’ensemble des états du robot sur la base de la mesure de vitesse de rotation, de l’orientation de l’œil et de l’erreur rétinienne. Le contrôleur de regard repose sur trois réflexes oculomoteurs, permettant de maintenir l’œil verrouillé sur la cible durant des mouvements du robot ou des perturbations. Les signaux mesurés sont présentés en rouge, et les grandeurs de contrôle en vert. On remarque la présence d’une entrée de consigne $\frac{T}{Z}X$, qui est expliquée dans la section 5.1, et qui permet de déclencher une saccade de l’œil afin de changer de cible d’intérêt.

3.2 Estimation du biais du gyromètre

La plupart des gyromètres présente un biais, qui, s’il n’est pas traité, entraîne des dérives dans l’estimation d’attitude. Les méthodes classiques ([Mahony *et al.*, 2008, Salaün et Martin, 2010, Mahony *et al.*, 2011, Castellanos *et al.*, 2005a, Barton, 2012]) utilisent les accéléromètres pour mesurer le vecteur de gravité, ainsi que des magnétomètres pour mesurer le champ gravitationnel, afin d’estimer et de compenser les biais des gyromètres. La méthode présentée ici se sert, comme le font les insectes, de la vision et de la mesure proprioceptive de l’orientation du regard ($\bar{\theta}_{er}$) pour dé-biaisier les gyromètres. Si le système de regard est verrouillé sur une cible au sol, le biais du gyromètre peut être déduit de l’erreur d’estimation de l’orientation de la cible $\tilde{\theta}_{tr}$ définie par l’équation (II.4). Dans notre cas, les informations basses fréquences d’inclinaison, obtenues classiquement des accéléromètres, sont donc données par la mesure proprioceptive d’orientation du regard $\bar{\theta}_{er}$.

Le biais du gyromètres peut s'écrire de la façon suivante :

$$\bar{\Omega}_r = \Omega_r + \Delta_g + \mu \quad (\text{II.5})$$

$$\dot{\Delta}_g = 0 \quad (\text{II.6})$$

Où Ω_r est la véritable vitesse angulaire, Δ_g est le biais du gyromètre (supposé ici constant ou très lentement variant), et μ est un bruit inconnu supposé centré.

Remarque II.2: La valeur de $\hat{\theta}_r$ dans l'équation (II.4) est obtenue à partir de l'équation (II.5). $\bar{\Omega}_r - \hat{\Delta}_g$ correspond simplement au terme de modèle alors que $L_3\tilde{\theta}_{tr}$ correspond au terme d'innovation. Comme précisé plus haut, le biais du gyromètre est estimé uniquement à partir de l'erreur d'estimation de la position angulaire de la cible ($\hat{\Delta}_g = L_4\tilde{\theta}_{tr}$).

3.3 Le contrôleur de l'œil

Le contrôleur de l'œil permet de maintenir le regard verrouillé sur une cible placé au sol. Étant donné que le champ de vision de l'œil est très réduit, il est nécessaire que la boucle de contrôle soit rapide, précise et efficace. Ces pré-requis sont atteints grâce à une approche bio-inspirée, qui combine trois réflexes complémentaires :

- Un réflexe vestibulo-oculaire de *rotation*, noté rVOR, qui permet de contrer tous les mouvements de rotation réalisés par le robot. Ce réflexe fournit une consigne de position angulaire $\theta_{er_r}^*$, qui n'est autre que l'opposé de l'angle de roulis $\hat{\theta}_r$.
- Un réflexe vestibulo-oculaire de *translation* noté tVOR permettant de compenser les effets de translations du robot sur la rétine. Ce réflexe fournit une consigne $\theta_{er_X}^*$, se basant sur la position estimée du robot \hat{X} et son altitude de vol H , contribuant à maintenir la cible au centre du champ de vision. Il s'agit également du réflexe clé, permettant de déclencher les saccades décrites dans la section 5.1.
- Un réflexe de *fixation visuelle* noté VFR, qui consiste en une boucle de contrôle sur l'erreur rétinienne ϵ_r . Cela permet d'annuler les erreurs rétinienne résiduelles et à poursuivre du regard la cible, si celle-ci se déplace. Ce réflexe permet également d'ajuster l'orientation de l'œil θ_{er} via l'entrée de consigne $\theta_{er_VFR}^*$ (voir la figure II.5).

Le réflexe VFR joue ici un rôle déterminant. C'est ce réflexe qui permet de poursuivre la cible dans toutes les conditions, et grâce à lui, le robot est capable :

- de survoler de façon "robuste" une cible,
- de rejeter efficacement des perturbations du type rafale de vents, ce qui est plus difficilement atteignable sans le découplage et ce réflexe de fixation, du fait de l'inertie du robot.

Le VFR consiste en un simple contrôleur de type Proportionnel-Intégral (PI), permettant d'asservir l'erreur rétinienne à 0° (boucle de feedback). Ce contrôleur contribue à l'orientation de l'œil, grâce à sa sortie $\theta_{er_VFR}^*$, comme le montre l'équation (II.7). Ce réflexe est donc responsable de la poursuite de la cible par le regard dans toutes les situations, ce qui est la première étape pour atteindre un vol stationnaire robuste et une estimation précise d'attitude et de position.

Les deux réflexes VOR aident à maintenir la cible au centre du champ de vision et agissent plus comme des termes d'anticipations (termes de feedforward). Ils sont inspirés de ce qui est fait chez l'insecte, et permettent de contrer deux types de mouvements du robot :

- **le roulis** : le rVOR utilise directement l'estimation de roulis $\hat{\theta}_r$ fournie par l'observateur non linéaire, afin de compenser les mouvements de rotations du corps du robot ($\theta_{er_r}^* = -\hat{\theta}_r$),
- **les translations latérales** : le tVOR, qui se base sur l'estimation de position, minimise les effets des déplacements latéraux sur l'erreur rétinienne ϵ_r , ($\theta_{er_X}^* = -\arctan\left(\frac{\dot{X}}{H}\right)$).

L'angle de référence θ_{er}^* (voir la figure II.5), résulte donc de la contribution de ces trois réflexes (rVOR, tVOR and VFR), et s'exprime comme suit :

$$\theta_{er}^* = \underbrace{\theta_{er_r}^*}_{rVOR} + \underbrace{\theta_{er_X}^*}_{tVOR} + \underbrace{\theta_{er_VFR}^*}_{VFR} \quad (\text{II.7})$$

Pour conclure sur ces réflexes, les réflexes VOR sont vus comme des termes de feedforward, qui permettent d'annuler, *a priori*, les effets perçus par le robot directement (grâce au gyromètre et à la mesure de l'orientation du regard), alors que le VFR, agit comme un terme de feedback permettant d'annuler toutes les erreurs de modélisation, etc. présentes dans les termes précédents. Des résultats de simulations supplémentaires, ainsi que des discussions concernant ces réflexes peuvent être trouvés dans [Manecy *et al.*, 2012] et [Manecy *et al.*, 2013b].

3.4 Pré-bouclage en vitesse de giration

Une première boucle d'asservissement permet de contrôler la vitesse de giration de l'angle de roulis. Cette vitesse de rotation est utilisée comme entrée de commande par le contrôleur d'attitude et de position (figure II.4). La boucle de contrôle interne en vitesse de rotation permet donc de faire en sorte de poursuivre les consignes (Ω_r^*) imposées par le contrôleur d'attitude et de position (voir la section 3.5). On suppose ici que les hélices sont contrôlées directement en poussée, qui est directement liée à la vitesse de rotation de l'hélice elle-même selon l'équation (II.8). La vitesse de rotation du robot (giration de l'angle de roulis) est donc contrôlée en appliquant la commande de poussée différentielle δ^* .

$$F_i = c_T \cdot \omega_i^2 \quad (\text{II.8})$$

Où F_i est la poussée générée par l'hélice i , ω_i est la pulsation de l'hélice i ($2\pi f_i$ avec f_i la fréquence de rotation de l'hélice i), et c_T est un coefficient de poussée qui peut être déterminé par des tests de poussée statique (voir le chapitre III).

Afin de prendre en compte la dynamique du gyromètre et d'asservir la vitesse de rotation directement, on utilise une structure observateur-contrôleur LQG. La représentation d'état du système est alors donnée par le système suivant :

$$\begin{pmatrix} \dot{\Omega}_r \\ \dot{\bar{\Omega}}_r \end{pmatrix} = \begin{pmatrix} \frac{-2L}{I_z \tau_{mot}} & 0 \\ \frac{1}{\tau_{gyr}} & \frac{-1}{\tau_{gyr}} \end{pmatrix} \begin{pmatrix} \Omega_r \\ \bar{\Omega}_r \end{pmatrix} + \begin{pmatrix} \frac{2L}{I_z \tau_{mot}} \\ 0 \end{pmatrix} \delta^* \quad (\text{II.9})$$

Où Ω_r est la véritable vitesse de rotation et $\bar{\Omega}_r$ est la vitesse de rotation mesurée par le gyromètre. Le temps de réponse en boucle fermée sur la grandeur Ω_r , a été réglé de façon à être de l'ordre de 20ms et on a fait en sorte que le bruit de commande ne dépasse pas 2%.

3.5 Contrôleur d'attitude et de position

Le contrôleur de position et d'attitude est un contrôleur par retour d'état, dont le gain a été déterminé par la méthode LQR et dont les états estimés (\hat{V}_x , \hat{X} and $\hat{\theta}_r$) proviennent de l'observateur non linéaire. Lors du design de ce contrôleur, on a supposé que la boucle sur la vitesse de rotation décrite précédemment (voir section 3.4), est suffisamment rapide pour être négligée.

Le contrôleur par retour d'état a été calculé sur le système (II.1) linéarisé autour de l'origine avec la commande à l'équilibre $\Omega_{req} = 0$. Afin d'annuler l'erreur statique en position, un effet intégral a été ajouté.

4 Résultats de simulation

4.1 Fonctionnement et rôle des réflexes

La figure II.6 montre la contribution des trois réflexes, durant un déplacement volontaire d'un mètre par rapport à la cible (obtenu en changeant la référence X^*), ainsi que lors d'une perturbation latérale de 20 cm, assimilable à une rafale de vent. On remarque que lors d'une translation volontaire, le rVOR réagit en premier (courbe verte) afin de compenser l'angle de roulis pris par le robot, suivi par le tVOR (courbe rouge) qui corrige l'orientation de l'œil, au fur et à mesure que l'on s'éloigne de la cible. Le VFR (courbe noire) intervient seulement ensuite, afin de corriger les erreurs résiduelles. Dans le cas d'une perturbation latérale, on remarque que c'est le VFR qui réagit en premier, afin de compenser le mouvement relatif involontaire entre le robot et la cible. Cette réorientation de l'œil, découlant de la fixation rétinienne, permet d'estimer la nouvelle position relative du robot et d'engager une manœuvre pour corriger l'erreur de positionnement. Ce n'est alors que dans un deuxième temps que les réflexes

rVOR et tVOR interviennent, afin de contrer l'angle de roulis adopté par le robot et la translation résultante.

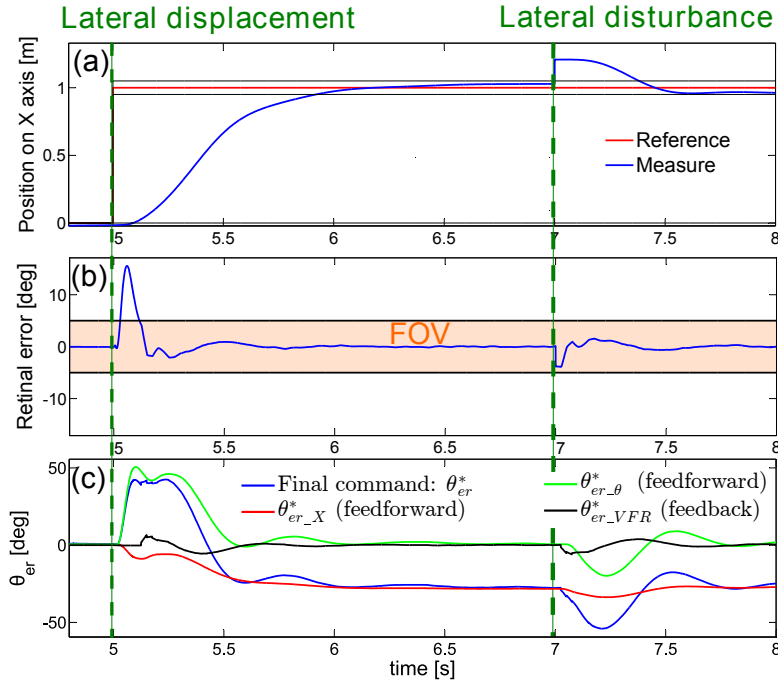


FIGURE II.6 – Réponse du robot à un déplacement latéral d'un mètre (a) et à une perturbation latérale de 20 cm (rejetée à 90% en 0.9 s). Durant le déplacement volontaire, les réflexes VOR réagissent en premiers, et compensent les mouvements du robot. Le VFR permet de compenser les erreurs du modèle d'observation ainsi que les dynamiques des actionneurs, afin de maintenir, malgré tout, la cible au centre du champ de vision. Lors d'une perturbation latérale, le VFR répond rapidement en générant une rotation de l'œil. Puis quelques milli-secondes après, les VOR apparaissent (lorsque le robot commence à se déplacer afin de contrer la perturbation).

4.2 Robustesse face aux perturbations

Dans certains de mes travaux, non détaillées ici ([Manecy *et al.*, 2012] et [Manecy *et al.*, 2013b]), j'avais démontré qu'un robot équipé d'un œil découplé (noté D-EYE) était capable de rejeter plus efficacement des perturbations latérales ou des perturbations de roulis, qu'un même robot (noté F-EYE) dépourvu de découplage, mais équipé d'un champ de vision plus grand.

5 Nouvelles perspectives découlant du contrôle du regard

L'utilisation d'un œil découplé à bord de véhicule aérien, ouvre donc un certain nombre de perspectives :

- cela permet au robot d'ajuster sa position relative à la cible, sans avoir besoin d'informations provenant d'une caméra, d'un système de capture du mouvement ou d'un GPS,
- cela permet également de se placer volontairement de façon décalée par rapport à la cible. Ce point présente un attrait particulier en ce qui concerne les capacités de rejet de perturbations. De façon plus générale, l'œil n'étant pas contraint dans une direction particulière, le robot peut évoluer librement autour de la cible, sans pour autant prendre le risque de perdre la cible de vue. (voir section 5.0.1),
- il est également possible au robot de "sauter" d'une cible à une autre, en générant une saccade avec son œil, ce qui lui permet ainsi de sélectionner une nouvelle cible (voir la section 5.1). Cela ouvre la possibilité de suivre une trajectoire spécifique en se déplaçant d'une cible à une autre.

5.0.1 Vol stationnaire décalé

Dans cette section, nous allons démontrer que l'œil découplé, en permettant de se placer de façon décalé par rapport à la cible, donne au robot la capacité d'être plus robuste aux perturbations latérales. Pour cela, nous introduisons la notion de *perturbation latérale maximale* (MLD, pour Maximum Lateral Disturbance) :

Définition II.1: Une perturbation latérale maximale (notée MLD) correspond au déplacement horizontal maximal qui peut se produire, sans que la cible soit perdue de vue (c'est-à-dire, sans que la cible ne quitte le champ de vision de l'œil). En d'autres termes, la MLD correspond à la translation maximale pouvant apparaître entre deux mises à jour consécutives de la boucle de vision (c'est-à-dire durant la période d'échantillonnage de la boucle visuelle). Si une perturbation est supérieure à la MLD, la boucle de vision sera trop lente pour détecter ce mouvement, et le robot sera dans l'incapacité de rejeter cette perturbation. La MLD est exprimée en mètres, et on note MLD_R et MLD_L les perturbations maximales à droite et à gauche respectivement.

La figure II.7 montre les capacités de rejet de perturbations latérales d'un robot F-EYE ayant un champ de vision de $\pm 25^\circ$ ainsi que celles d'un robot D-EYE ayant un champ de vision 5 fois inférieur ($\pm 5^\circ$). De cette figure, on détermine la MLD correspondant à chacun des robots (F-EYE et D-EYE) :

Le robot F-EYE : les positions atteignables par le robot F-EYE sont contraintes par la relation suivante $|X| \leq H \tan\left(\frac{FOV}{2}\right)$, en effet :

$$\begin{cases} MLD_R(X) = H \tan\left(\frac{FOV}{2}\right) - X \\ MLD_L(X) = H \tan\left(\frac{FOV}{2}\right) + X \end{cases} \quad (\text{II.10})$$

le robot D-EYE : théoriquement, le robot D-EYE peut adopter toutes les positions le long de

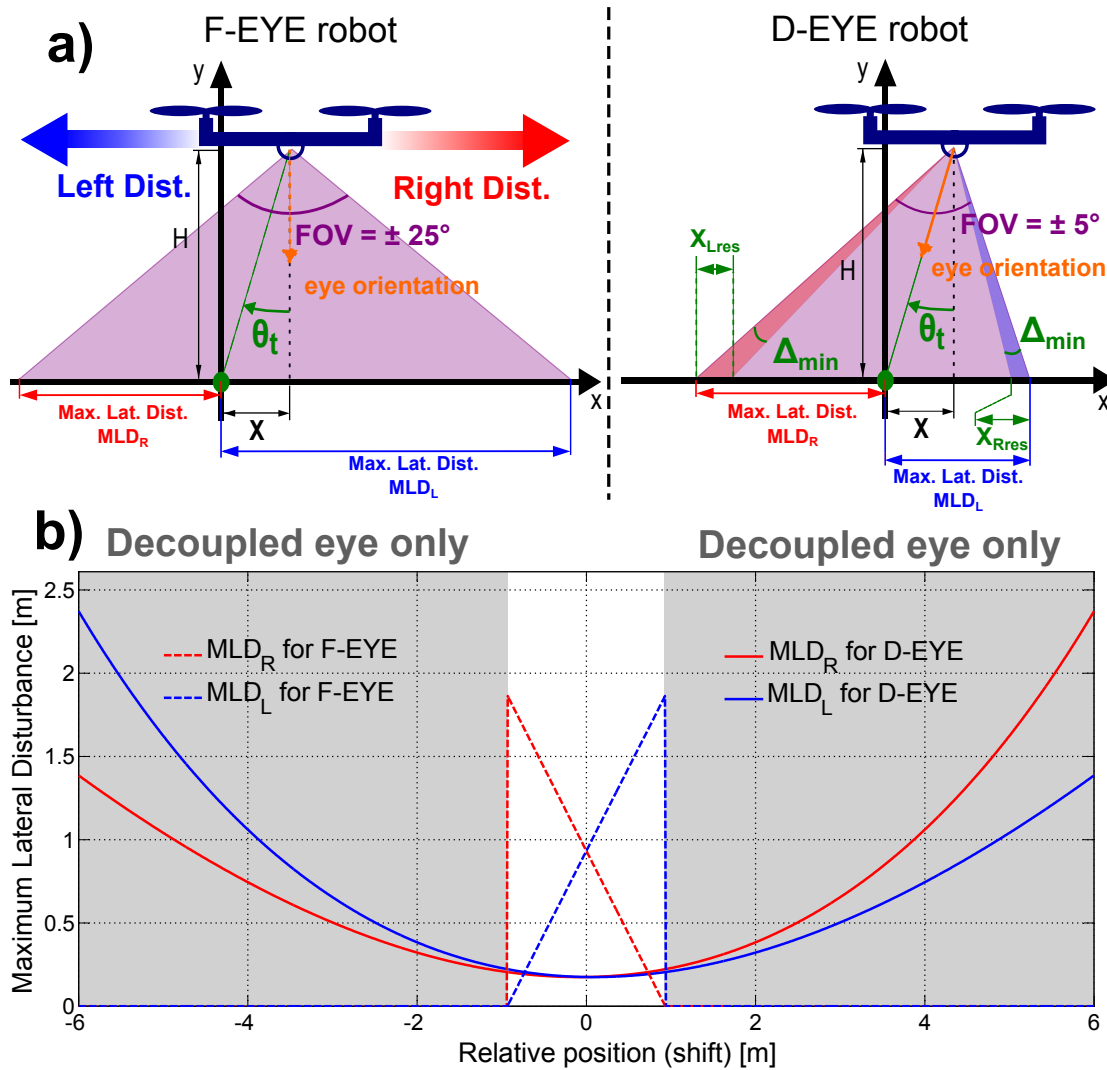


FIGURE II.7 – a) Résolution au niveau du sol et perturbation latérale maximale (MLD). Grâce au découplage du regard du robot D-EYE, plus on se place de façon éloignée de la cible ($|X|$ grandit), plus la MLD augmente. b) La perturbation latérale maximale (MLD) dépend donc de la position latérale du robot. La zone grise correspond à la zone non atteignable par le robot ayant un œil fixe (le robot F-EYE), c'est-à-dire la zone où il perd la cible de vue. Pour le robot F-EYE, l'augmentation de la MLD d'un côté entraîne forcément une diminution équivalente de l'autre côté, alors que pour le robot D-EYE, en s'éloignant de la cible, la MLD augmente des deux côtés (gauche et droite). La figure est réalisée avec un robot F-EYE, robot ayant un champ de vision de $\pm 25^\circ$ et un robot D-EYE ayant un champ de vision de $\pm 5^\circ$ avec une course maximale de $\pm 80^\circ$ pour l'œil ($\theta_{er_{MAX}} = 80^\circ$).

l'axe horizontal.

$$\begin{cases} \text{MLD}_R(X) = H \tan\left(\frac{FOV}{2} + \arctan\left(\frac{X}{H}\right)\right) - X \\ \text{MLD}_L(X) = X - H \tan\left(\frac{FOV}{2} + \arctan\left(\frac{X}{H}\right)\right) \end{cases} \quad (\text{II.11})$$

Lorsqu'il vole à l'aplomb de la cible, le robot F-EYE bénéficie de tout son champ de vision et peut rejeter des perturbations allant jusqu'à 93 cm, alors que le robot D-EYE ne peut rejeter qu'une MLD de 17.5 cm du fait de son champ de vision restreint. Cependant, en éloignant le robot de cette position centrale, il est possible d'améliorer grandement les capacités de rejet de MLD pour le robot D-EYE. En effet, elles deviennent même supérieures à celle du robot F-EYE dès lors que l'on se décale de plus de 2 m sur le côté. Cette méthode simple, permet de compenser facilement le désavantage initial, lié à l'utilisation d'un champ de vision restreint. Même avec son champ de vision 5 fois plus restreint, le robot D-EYE est capable, en se déplaçant latéralement, de rejeter des MLD plus importantes que le robot F-EYE. De plus, contrairement au robot F-EYE, l'œil découplé du robot D-EYE permet de rejeter des perturbations d'amplitude quasi-égale des deux côtés (gauche et droite).

L'avantage principal de cette nouvelle stratégie, consistant à se décaler par rapport à la cible, est donc d'améliorer la résistance du robot en terme de perturbation latérale maximale (MLD, assimilable à des rafales de vent par exemple). On peut déduire de la figure II.7, qu'un robot sans œil découplé, ne peut même pas se décaler de 2 m par rapport à la cible, sans que celle-ci ne sorte de son champ de vision de $\pm 25^\circ$. Alors qu'à cette distance, un robot équipé d'un œil découplé de seulement $\pm 5^\circ$ de champ de vision, peut rejeter des perturbations de plus de 38 cm, tout en conservant une précision de positionnement meilleure que 1 cm.

5.1 Vol en saccades

Une autre propriété intéressante, découlant de l'œil découplé, et la possibilité de changer très rapidement de cible. Par exemple, lorsque le robot survole une première cible (et que son œil est verrouillé dessus), on peut générer une saccade très brève, assimilable à une saccade de rattrapage (catch-up saccade), afin de faire "sauter" le robot sur une cible dont la position est connue. Une saccade est un des quatre type de mouvements oculaire (saccadique, poursuite continue, réflexe vestibulo-oculaire et vergence) utiles pour poursuivre une trajectoire. Typiquement, une saccade est utilisée chez l'œil humain lorsque la poursuite continue ne parvient plus à maintenir la cible au centre de champ visuel ([Carpenter, 1988]).

Ici, nous utilisons ce type de saccades, afin de permettre au robot de changer de cible très rapidement. Ensuite, découlant de la stratégie de guidage par le regard décrite précédemment, le robot se déplace là où son regard le porte. C'est-à-dire, qu'il cherche à asservir sa position relative par rapport à la cible, à partir de l'angle que forme son œil avec son corps.

5.1.1 Notations :

Ici, nous allons introduire certaines notations nécessaires pour bien définir le saut de cible, et les différents référentiels que nous allons utiliser dans cette partie :

${}^A_B U$ décrit donc la variable U de l'objet A (A est l'objet "propriétaire" de la variable U) et exprimée dans le référentiel \mathcal{B} (dans lequel la mesure est faite.)

Par exemple, ${}^R_{\mathcal{I}} X$ donne la position du robot R dans le référentiel inertiel \mathcal{I} et ${}^R_{\mathcal{T}} X^*$ représente la référence de position du robot R dans le référentiel de la cible \mathcal{T} .

Pour les besoins de cette étude, nous allons introduire en tout, trois référentiels différents :

le référentiel du robot, noté \mathcal{R} ,

le référentiel inertiel, noté \mathcal{I} ,

le référentiel de la cible, noté \mathcal{T} .

Remarque II.3: Notez que dans le début de ce chapitre, le référentiel de la cible \mathcal{T} était le même que le référentiel inertiel \mathcal{I} étant donné que les origines de ces deux référentiels était confondues. C'est-à-dire que précédemment : ${}^T_{\mathcal{I}} X = X_T$ et ${}^T_{\mathcal{T}} X = 0$.

Dorénavant, la relation entre la position estimée du robot dans le référentiel inertiel, la position estimée du robot dans le référentiel de la cible et la position de la cible dans le référentiel inertiel, peut s'écrire :

$${}^R_{\mathcal{I}} \hat{X} = {}^R_{\mathcal{T}} \hat{X} + {}^T_{\mathcal{I}} X \tag{II.12}$$

Où ${}^T_{\mathcal{I}} X$ est la position de la cible courante dans le référentiel inertiel. On peut donc réécrire le système d'équations(II.4) correspondant à l'observateur non-linéaire comme suit :

$$\left\{ \begin{array}{l} \begin{pmatrix} {}^R_{\mathcal{T}} \dot{\hat{V}}_x \\ {}^R_{\mathcal{T}} \dot{\hat{X}} \\ {}^R_{\mathcal{T}} \dot{\hat{\theta}}_r \\ {}^R_{\mathcal{T}} \dot{\hat{\Delta}}_g \end{pmatrix} = \begin{pmatrix} -\frac{T}{m} \sin({}^R_{\mathcal{T}} \hat{\theta}_r) - K_{vx} {}^R_{\mathcal{T}} \hat{V}_x + L_1 {}^R_{\mathcal{T}} \tilde{\theta}_{tr} \\ {}^R_{\mathcal{T}} \hat{V}_x + L_2 {}^R_{\mathcal{T}} \tilde{\theta}_{tr} \\ {}^R_{\mathcal{T}} \tilde{\Omega}_r - {}^R_{\mathcal{T}} \hat{\Delta}_g + L_3 {}^R_{\mathcal{T}} \tilde{\theta}_{tr} \\ L_4 {}^R_{\mathcal{T}} \tilde{\theta}_{tr} \end{pmatrix} \\ {}^R_{\mathcal{T}} \hat{X} = {}^R_{\mathcal{I}} \hat{X} - {}^T_{\mathcal{I}} X \\ {}^R_{\mathcal{T}} \hat{\theta}_{tr} = -\arctan\left(\frac{{}^R_{\mathcal{T}} \hat{X}}{{}^R_{\mathcal{T}} H}\right) - {}^R_{\mathcal{T}} \hat{\theta}_r \end{array} \right. \tag{II.13}$$

où ${}^R_{\mathcal{T}} \tilde{\theta}_{tr} = {}^R_{\mathcal{I}} \tilde{\theta}_{tr} - {}^R_{\mathcal{T}} \hat{\theta}_{tr}$

5.1.2 Procédure de "saut de cible" :

Grâce à l'écriture précédente, il suffit d'un changement de variable pour faire passer le robot d'une cible à une autre. En effet, il suffit de mettre à jour la position de la cible, c'est-à-dire de remplacer la

valeur de l'ancienne cible $\frac{T}{\mathcal{I}}X_k$ par celle de la nouvelle cible $\frac{T}{\mathcal{I}}X_{k+1}$ dans la position de cible courante $\frac{T}{\mathcal{I}}X(t)$:

$$\frac{T}{\mathcal{I}}X(t) = \frac{T}{\mathcal{I}}X_k \quad \text{avant le saut.} \quad (\text{II.14})$$

$$\frac{T}{\mathcal{I}}X(t + \Delta t) = \frac{T}{\mathcal{I}}X_{k+1} \quad \text{après le saut.} \quad (\text{II.15})$$

Où Δt correspond à la période d'échantillonnage de l'observateur non linéaire et $\frac{T}{\mathcal{I}}X_k$ avec les indices k et $k + 1$ sont les identifiants des cibles. $\frac{T}{\mathcal{I}}X_k$ et $\frac{T}{\mathcal{I}}X_{k+1}$ sont donc les positions (dans le référentiel inertiel \mathcal{I}) de la cible k avant le "saut de cible" et de la cible $k + 1$ après le "saut de cible", respectivement.

L'ordre de saccade ou de "saut" est donc passé en mettant à jour la position de la cible dans le terme de feedforward $\theta_{er_X}^*$, lié au tVOR, et qui devient avec les nouvelles notations : $\theta_{er_X}^* = -\arctan\left(\frac{\frac{R}{\mathcal{I}}\dot{X}}{\frac{R}{\mathcal{I}}H}\right)$. Lorsque l'opérateur désire changer de cible, il lui suffit de mettre à jour la position de la nouvelle cible $\frac{T}{\mathcal{I}}X_k$, afin de déclencher la saccade. Ce changement peut également être automatique (au bout d'un certain temps, d'un certain critère, etc.)

Une fois que les nouvelles coordonnées de la cible ont été renseignées, le tVOR déclenche une saccade, réorientant l'œil sur la nouvelle cible. De la nouvelle orientation de l'œil, le robot déduit sa nouvelle position relative à la cible (grâce à l'observateur non linéaire), et se repositionne immédiatement afin de respecter la consigne de position relative $\frac{R}{\mathcal{I}}X^*$. Une simulation d'une séquence de saut de cible est décrite par les figures II.8, II.9 et II.10.

Remarque II.4: Les simulations présentées ici, ont été réalisées en Processor-In-the-Loop (PIL), comme décrit dans la section 2.1, et avec le jeu de paramètres donné dans le tableau II.2. L'ensemble de la stratégie présentée ici, a été développée sous Matlab/Simulink et les boucles de contrôle ont été exécutées à une fréquence de 400 Hz (excepté pour le VFR qui a été implémenté à seulement 40 Hz du fait de la version du capteur que l'on suppose utiliser ici). La charge totale du programme représente moins de 20% du CPU disponible. Durant les simulations, l'ensemble des résultats des calculs et des entrées de commandes étaient visualisées et enregistrées en temps réel à 400 Hz également, grâce à une connexion WIFI, sur le modèle Simulink de la station sol (qui servait ici aussi, à simuler le comportement dynamique du robot et à générer les sorties des différents capteurs).

Les figures II.8 et II.10 correspondent au scénario suivant :

- étape 1 : le robot se déplace latéralement afin de se placer à 1 m de la cible,
- étape 2 : après 5 s, l'œil du robot fait une saccade afin de se focaliser sur la cible 2 et le robot se replace, afin de respecter sa consigne de position relative par rapport à la cible (la figure II.9 montre uniquement ce saut de cible à l'instant $t = 5.0s$),
- étape 3 : après 5s, le robot "saute" sur la cible 3,

Nous avons donc : $\frac{T}{\mathcal{I}}X_1 = 0m$, $\frac{T}{\mathcal{I}}X_2 = -1m$ et $\frac{T}{\mathcal{I}}X_3 = -2m$.

Paramètres	Description	Valeur	Unité
H	Altitude de vol	2.0	m
L	Demi envergure du robot	0.15	m
m	Masse du robot	0.1	kg
I_z	Moment d'inertie	2.0×10^{-5}	$\text{kg} \cdot \text{m}^2$
K_{vx}	Coefficient de frottement	0.5	$\text{N} \cdot \text{s} \cdot \text{m}^{-1}$
FOV	Champ de vision	± 5	deg
d	Retard de traitement de vision	10	ms
G_{opt}	Gain du capteur optique	1	-
τ_{hall}	Constante de temps du capteur à effet hall	1	ms
τ_{gyr}	Constante de temps du gyromètre	4.3	ms
τ_{mot}	Constante de temps des hélices	20	ms
τ_{eye}	Constante de temps de l'œil	10	ms

TABLE II.2 – Paramètres de simulation.

En conclusion, grâce à une dynamique rapide de l'œil (liée à sa faible inertie), un saut de l'œil est réalisé très rapidement car en moins de 0.13s la nouvelle cible entre dans le champ de vision de l'œil et n'en sort plus, comme le montre la figure II.9. Une fois que cette nouvelle cible est verrouillée, l'observateur non linéaire met à jour l'estimation de position relative par rapport à la cible, et le robot rejoint, en moins de 1.5 s, sa position de référence ${}^R X^*$, comme le montrent les figures II.8 et II.10.

6 Les avantages d'un œil découplé

Ici, nous résumons les principaux avantages que peut apporter l'utilisation d'un œil dans des applications de robotique aérienne :

- **Moins de distorsion** : Dans le cas d'un œil fixe, il est nécessaire d'avoir un large champ de vision impliquant par exemple l'utilisation d'un objectif fish-eye ainsi que de nombreux pixels afin de conserver une résolution spatiale suffisante. Les objectifs fish-eyes, ou grands angles en général, sont connus pour apporter des distorsions qui doivent être compensées. De même, un plus grand nombre de pixels doit être utilisé. Ces deux phénomènes, augmentent d'autant plus la complexité de calcul et par conséquent les ressources calculatoires nécessaires. Un œil découplé, quant à lui, permet de réduire le champ de vision nécessaire, sans pour autant prendre le risque de perdre la cible de vue. Cela permet ainsi de réduire les distorsions liées à l'objectif et de maintenir, à tout moment, la cible au centre du champ de vision, là où l'acuité visuelle est la meilleure.
- **Réduction de la complexité de calcul** : comme montré dans la section 4.2.2 of [Manecy *et al.*, 2013b], un robot D-EYE rejette des perturbations plus efficacement qu'un robot F-EYE, présentant un champ visuel large de 1000 pixels (une ligne d'une caméra avec 1000 pixels a la même résolution angulaire de 0.1° que le robot D-EYE équipé de son capteur doté d'hyperacuité).

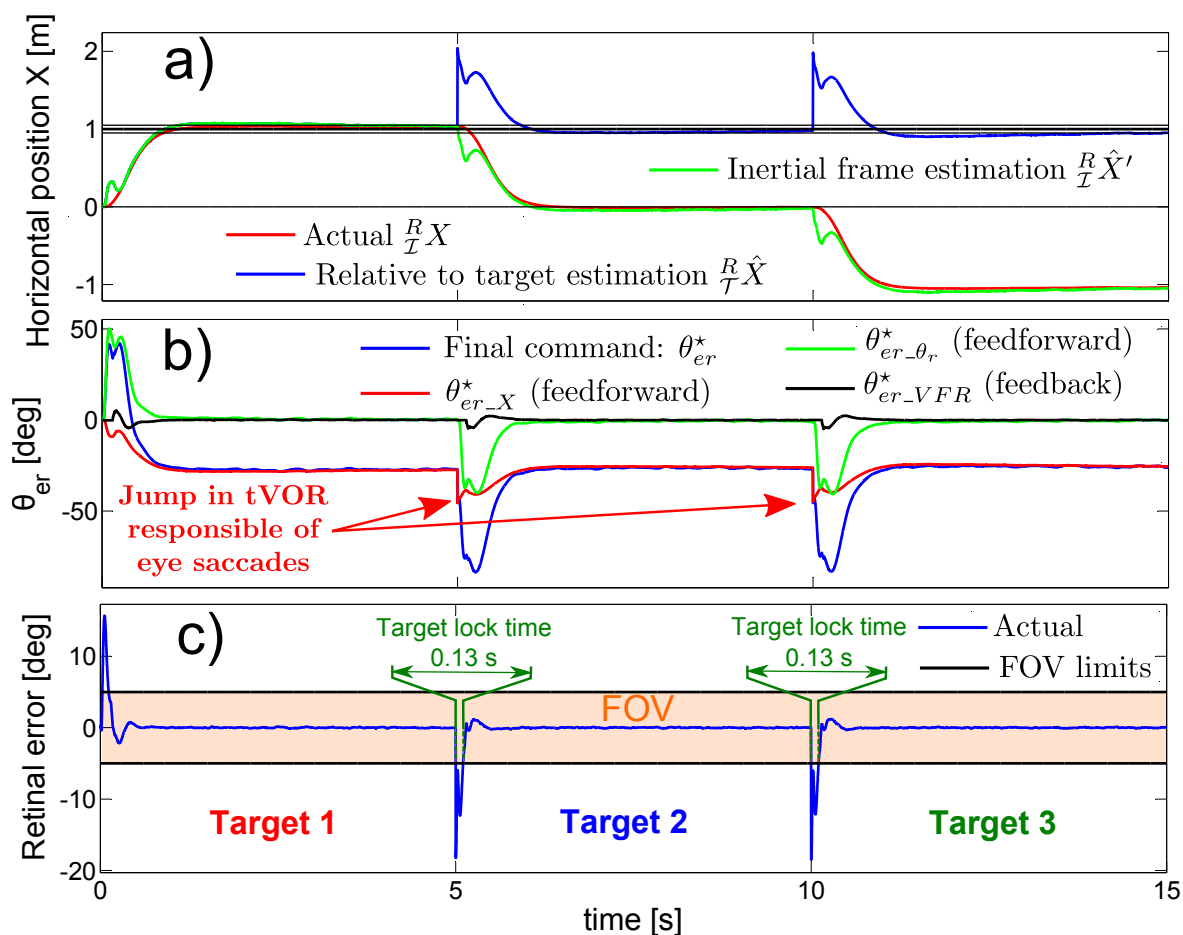


FIGURE II.8 – a) Estimation de la position et de la poursuite de la cible, exprimée dans les différents référentiels. b) Evolution de l'angle de l'œil. La mise à jour des coordonnées de la cible, décrite par l'équation (II.15), implique un saut dans le terme de feedforward θ_{er-X}^* (courbe rouge), déclenchant ainsi le changement de cible. c) L'erreur rétinienne durant les phases de saut de cible. Il faut seulement 0.13 s pour changer de cible, c'est-à-dire perdre de vue la précédente et voir apparaître la nouvelle. La figure II.9 montre plus précisément l'évolution de l'erreur rétinienne durant une phase de saut de cible.

Cette résolution est atteinte pour l'œil doté d'hyperactuité qu'avec 6 pixels seulement. Attention cependant, l'œil de 6 pixels ne peut pas traiter n'importe quelle cible, contrairement à une caméra classique.

- **Rejet de perturbation plus rapide** : avec le robot F-EYE robot (équipé d'un champ visuel large), une forte rafale de vent ne peut pas être compensée rapidement. En effet, pour compenser rapidement une perturbation latérale, le robot doit adopter des angles de roulis élevés, ce qui implique une perte de vue de la cible (ceci est d'autant plus vrai que la perturbation déplace la cible proche de la limite du champ de vision). Le robot D-EYE quant à lui, ne connaît pas de limitation forte sur son attitude, car son œil s'adapte afin de maintenir la cible au centre de champ de vision quelle que soit l'attitude (voir la section IV de [Manecy *et al.*, 2012]).

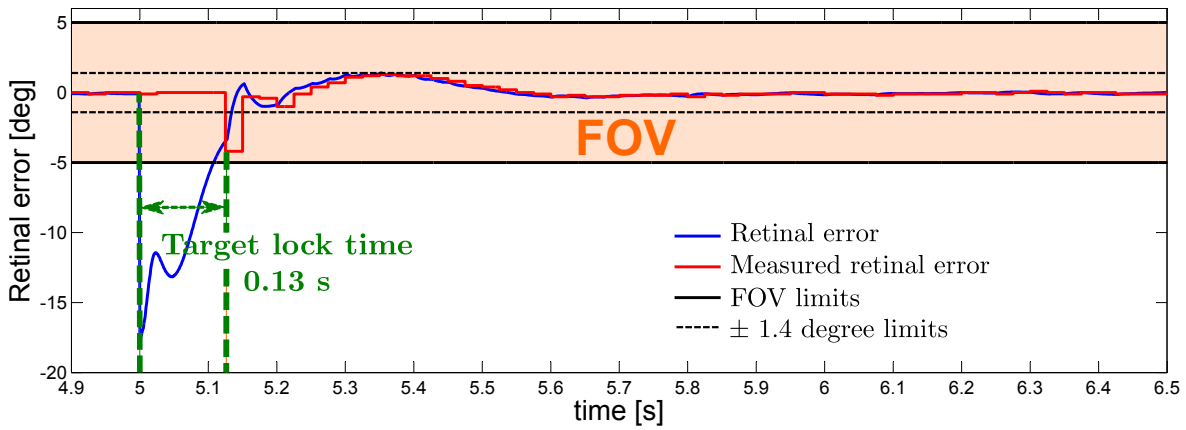


FIGURE II.9 – Evolution de l’erreur rétinienne durant une phase de saut de cible. La courbe bleue, décrit l’erreur rétinienne réelle (erreur théorique que l’œil devrait mesurer), et la courbe rouge montre la valeur échantillonnée correspondant à la sortie du capteur (incluant la dynamique du capteur, les délais de traitement, le champ de vision, etc. selon le tableau II.1). Ici, en 0.13 s la cible entre dans le champ de vision du capteur et est verrouillée. En moins de 0.15 s, l’erreur rétinienne atteint moins de 1.4° ($|\epsilon_r| < 1.4^\circ$) et en 0.7 s, elle est maintenue en dessous de 0.2°. Il faut donc moins de 0.7 s pour que le saut de cible soit parfaitement réalisé.

- **Robustesse accrue vis-à-vis du vent latéral :** De la même manière, le robot D-EYE n’est pas contraint de rester au-dessus de la cible pour la maintenir au centre de son champ de vision. Plus particulièrement, il peut se placer volontairement de façon décalée, afin d’augmenter la zone couverte par son champ de vision, et par conséquent, augmenter la perturbation latérale maximale (MLD) qui peut être rejetée. De même, cela peut également permettre au robot de s’éloigner de la cible sans pour autant prendre le risque de la perdre de vue (ce qui n’est pas possible avec un robot ayant un œil fixe, comme le montre la figure II.7).
- **Trajectoire point par point :** Finalement, l’utilisation d’un œil découplé ouvre la possibilité de naviguer de cible en cible, comme expliqué dans la section 5.1. Cela peut permettre à un robot de naviguer de cible en cible, sans pour autant nécessiter d’une localisation absolue à tout instant (GPS).
- **GPS et guidage par le regard :** Dans le cas d’une utilisation en extérieur, avec couverture GPS, on pourrait imaginer utiliser le regard pour poursuivre une cible (le robot "subit " les mouvements de la cible), et utiliser le positionnement GPS du robot pour estimer la position de la cible. En effet, la stratégie de guidage par le regard permet de poursuivre de façon efficace une cible (voir par exemple le chapitre V. La position de la cible serait alors simplement la somme de la position du GPS du robot (éventuellement filtrée et fusionnée avec le modèle de l’observateur non linéaire) et de la position relative du robot par rapport à la cible :

$${}^T_{\mathcal{I}}X = {}^R_{\mathcal{I}}X_{f_{GPS}} + {}^T_{\mathcal{R}}X_{gaze} = {}^R_{\mathcal{I}}X_{f_{GPS}} - {}^R_{\mathcal{T}}X_{gaze} \tag{II.16}$$

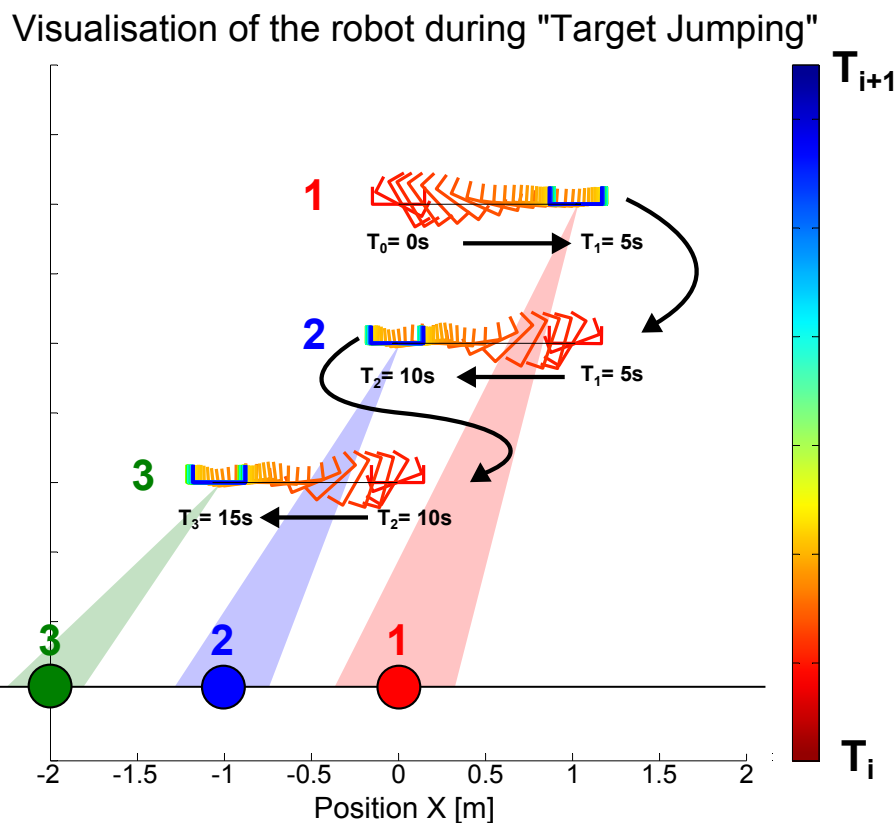


FIGURE II.10 – Attitude du robot durant une phase de "saut de cible". La position et l'attitude du robot sont représentées durant la trajectoire à une fréquence de 20 Hz. Pour une raison de clarté, les trois séquences ont été décalées sur l'axe vertical, mais l'altitude H est en réalité bien la même pour les trois séquences. A $t = 0s$, va se placer à 1 m sur le côté de la cible. A $t = 5s$, le robot change de cible de référence et va se placer à 1 mètre sur le côté de la cible 2. Finalement, à $t = 10s$, le robot se déplace encore afin d'aller se placer à 1 m de la cible 3, la nouvelle cible de référence.

Où ${}^R_I X_{f_{GPS}}$ est la position GPS filtrée et ${}^R_I X_{gaze}$ et la position estimée du robot par rapport à la cible.

7 Conclusion

Dans ce chapitre, nous avons donc présenté les résultats de simulation concernant la stabilisation d'un birotor équipé d'un œil découplé. La stratégie de contrôle et de stabilisation repose en grande partie sur la stabilisation du regard, qui est obtenue grâce à une combinaison de réflexes. Nous avons montré que l'utilisation d'un tel œil découplé présentait l'avantage de rendre le robot plus robuste aux perturbations. Nous avons également montré, que contrairement à ce qui est fait classiquement, on peut utiliser efficacement l'orientation du regard pour estimer le biais du gyromètre. L'ensemble de la stratégie a été validé et testé par simulations de type Porcessor-In-the-Loop, durant lesquelles les

algorithmes de contrôle ont été exécutés sur une Gumstix Verdex Pro XL6P.

Nous avons également introduit une nouvelle technique de navigation point par point, reposant sur la génération de saccades oculaires. De plus, nous pouvons dire que l'utilisation d'un œil découplé peut aller bien plus loin que la simple stabilisation d'un robot volant. Cela ouvre des possibilités prometteuses, en ce qui concerne le contrôle de la position 3D d'un robot se fixant selon les besoins sur différents amers visuels. Dans la suite de ce manuscrit, on décrira donc la généralisation de cette stratégie de guidage par le regard, à l'utilisation dans le cas 3D. Et on démontrera sa faisabilité lors d'expérimentation sur un micro-drone de type quadrirotor.

7.1 Application

Dans le cadre de la poursuite des travaux de thèse d'un précédent doctorant et de la réalisation d'un démonstrateur robotique, j'ai pu mettre en œuvre une partie des stratégies présentées ici. L'idée de cette application était de démontrer l'intérêt d'utiliser, dans le cadre de la robotique aérienne, des nouvelles technologies de capteurs visuels reproduisant fidèlement l'œil composé de l'insecte [Floreano *et al.*, 2013].

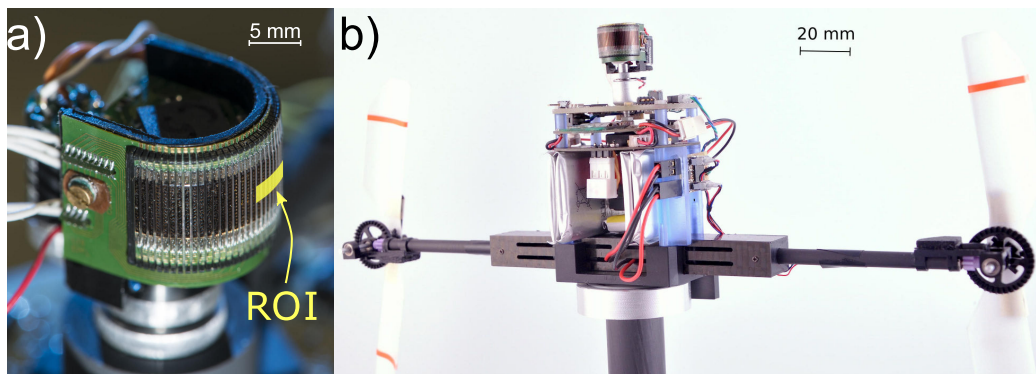


FIGURE II.11 – a) l'œil composé artificiel développé lors d'un précédent projet européen CurvACE, et sa région d'intérêt composée de quelques pixels. b) Le robot HyperRob équipé de son œil. *Adapté de [Colonnier et al., 2015a].*

L'objectif était alors de permettre à un birotor de se stabiliser vis à vis de son environnement visuel. Le birotor, appelé HyperRob, pèse environ 145 g et est équipé d'hélices en polystyrène. Il a été entièrement pensé et conçu au sein du laboratoire biorobotique de Marseille, et est équipé de deux micro-contrôleurs. Le premier micro-contrôleur permet de traiter les informations visuelles, et le second réalise les différents asservissements. De plus, un contrôleur de moteur à courant continu a été développé spécifiquement au sein du laboratoire, afin d'asservir finement la vitesse de rotation des hélices.

Le principe était de réutiliser la stabilisation du regard dans un cadre légèrement différent. Le regard est, cette fois-ci, maintenu aligné avec la verticale. C'est à dire que l'on estime, à l'aide de l'IMU embarqué, l'attitude du robot, et que l'on utilise cette information pour maintenir la direction

du regard fixe, quelques soient les mouvements du robot. En annulant ainsi, les mouvements de rotation liés aux changements d'attitude du robot, on considère que l'œil composé du robot capte uniquement les mouvements de translations du robot. Il est alors possible, de réaliser des tâches de rejet de perturbations (type rafales de vent), d'odométrie visuelle, poursuite d'objets (par exemple une main) au dessus de fond texturé, etc.

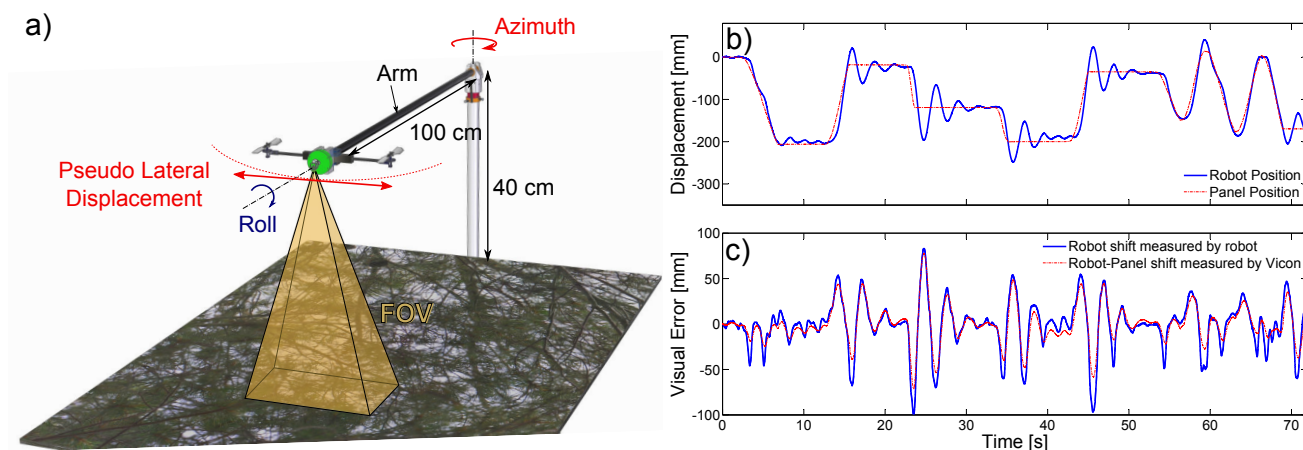


FIGURE II.12 – a) Principe de l'expérimentation. Le robot est placé au bout d'un bras rotatif, au dessus d'un panneau texturé. Le regard est découplé du corps du robot à l'aide d'un servomoteur, afin de maintenir l'orientation du regard à la verticale. Ainsi, l'œil mesure directement les pseudo-translations réalisées par le robot. b) En pointillés rouges, la position du panneau texturé, mesuré par un système de capture du mouvement externe (système Vicon). En bleu, la position du robot. c) En rouge l'écart réel entre le robot et le panneau texturé. En bleu, l'écart mesuré par le robot. *Adapté de [Colonnier et al., 2015a].*

La figure II.12b montre donc le principe de l'expérience. Le robot est placé au dessus d'un panneau texturé, et on utilise seulement quelques pixels de la zone frontale de l'œil composé (figure II.12a) pour la tâche de vision. Le robot est monté au bout d'un bras rotatif, lui permettant de réaliser des mouvements de pseudo-translations latérales. Comme le montrent les figures II.12b-c, lorsqu'on déplace manuellement le panneau texturé situé en dessous du robot, celui-ci mesure la translation grâce à son œil et cherche à annuler cette translation. Cela entraîne donc naturellement le robot, à suivre les mouvements du panneau texturé.

La figure II.13 montre une autre application de cet œil avec le même robot, où cette fois-ci, le robot poursuit les mains d'un opérateur, placées au dessus d'un fond texturé.

Ces résultats sont le fruit de la collaboration avec un autre doctorant, Fabien Colonnier, et ont repris aussi les travaux de fin de thèse de Raphaël Juston. J'ai alors mis au point les différents algorithmes d'estimation d'états, de stabilisation du regard et les différentes boucles d'asservissements. Ce travail a ensuite donné lieu à un article de journal dans la revue *Bioinspiration & Biomimetics*, et le détail de ces expériences est donné dans [Colonnier et al., 2015a].

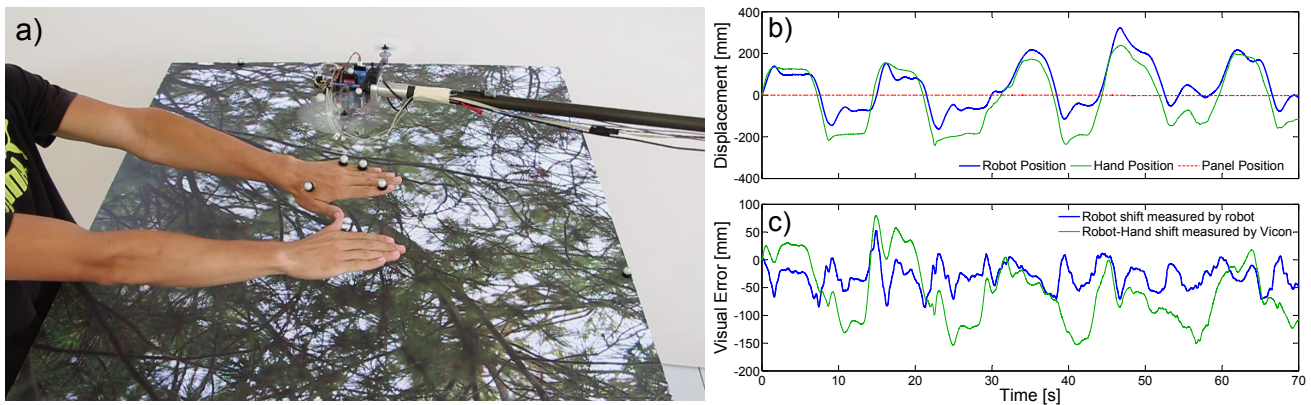


FIGURE II.13 – a) Principe de l'expérimentation. Un opérateur déplace ses mains au dessus d'un panneau texturé, le robot suit les déplacements des mains. b) En pointillés rouges, la position du panneau texturé, mesuré par un système de capture du mouvement externe (système Vicon). En bleu, la position du robot, et en vert, la position de la main. c) En vert l'écart réel entre le robot et la main. En bleu, l'écart mesuré par le robot. *Adapté de [Colonnier et al., 2015a].*

Chapitre III

Design d'un quad-rotor open-source : X4-MaG

Sommaire

1	Introduction	37
1.1	Objectifs et contexte	37
1.2	Choix de la plateforme	38
2	Design et dimensionnement	38
2.1	Structure et composants	38
2.2	Modules de calcul	39
2.3	Architecture logicielle et matérielle	40
2.4	Arène de vol et conditions expérimentales	45
2.5	Groupe de propulsion et asservissement local	46
3	Formalisme et angles d'Euler	56
4	Modélisation complète du quadrirotor	58
4.1	Bilan des forces	59
4.2	Bilan des moments	63
4.3	Modèle complet	67
4.4	Identification des paramètres	69
5	Protocoles de calibrations	75
5.1	Calibrations des Accéléromètres	75
5.2	Calibrations des Gyromètres	78
5.3	Estimations en lignes des biais	78
6	Développement d'un support logiciel open-source	79
6.1	Motivations	79
6.2	Quelles applications ?	80

6.3	Choix Structurels et considérations techniques	80
6.4	Performances	82
6.5	Perspectives pour RT-MaG	83
7	Stabilisation et asservissement de position dans l'arène de vol	85
7.1	Le contrôleur de vitesse des hélices	86
7.2	Estimation d'attitude : le filtre complémentaire	86
7.3	Contrôleurs d'attitude	87
7.4	Contrôle de position en 3D	90
8	Résultats dans l'arène de vol	93
8.1	Comparaison des deux contrôleurs d'attitude	93
8.2	Précision de positionnement	96
9	Conclusion	98
9.1	La plateforme X4-MaG	98
9.2	Perspectives	100

1 Introduction

Dans ce chapitre, nous allons décrire en détail le quadrirotor ainsi que la toolbox Matlab/Simulink qui ont été développés durant cette thèse. Le quad-rotor et sa suite logicielle ont été développés pour simplifier la mise en œuvre et le test de nouvelles lois de commandes.

1.1 Objectifs et contexte

Dans cette thèse, nous souhaitons développer de nouvelles lois de commandes bio-inspirées pour la stabilisation d'un robot aérien en vol libre. Bien que l'élaboration de ces lois puisse être testée et validée en simulation, j'ai choisi de tester la faisabilité pratique et d'évaluer leurs performances sur un micro-aéronef. Il a alors été nécessaire de choisir une plateforme permettant de répondre à ces différents critères ; la plateforme doit :

- **être "petite"** : son envergure ne doit pas dépasser les 30 *cm* de façon à maintenir des dynamiques rapides,
- **permettre d'embarquer entre 50g et 100g de charge utile**. La plateforme doit être équipée d'un système occulo-moteur complet, pesant en tout une cinquantaine de grammes. Il est donc nécessaire que l'aéronef présente, malgré sa petite taille, une capacité d'emport supérieure au poids du système visuel bio-inspiré (voir chapitre [IV](#)).
- **être bon marché**, le but étant de pouvoir réutiliser cette plateforme dans d'autres projets de recherches afin de faciliter la capitalisation,
- **fournir un accès à l'ensemble des capteurs embarqués à son bord** : Les nouvelles stratégies nécessitent le développement de nouveaux algorithmes de fusion de capteurs. Il est donc indispensable de pouvoir accéder à l'ensemble des capteurs présents sur le robot de façon individuelle. Par exemple, certaines IMU du commerce fournissent directement les estimations d'attitudes, voire même de positions, sans donner accès aux mesures "bas niveaux" des différents capteurs (gyromètres, accéléromètres, etc.)
- **être robuste aux crash** pour des raisons évidentes, étant donné que de nouveaux algorithmes vont être testés,
- **être entièrement ouverte** de façon à faciliter sa programmation, ainsi que la modification des différents contrôleurs à tous les niveaux,
- **Idéalement, offrir une interface de programmation avec Matlab/Simulink**. J'ai choisi de développer l'ensemble des lois de commandes, ainsi que le simulateur de l'aéronef sous Matlab/Simulink, car il offre une grande flexibilité, une grande versatilité et propose beaucoup d'outils, permettant de faciliter le design et l'analyse de lois de commandes.

Étant donné que le projet de thèse se fait en collaboration entre deux entités de recherche, le Gipsa-Lab et l'Institut des Sciences du Mouvement, il a été décidé de développer une solution complète en interne. Bien que de nombreux aéronefs soient présents dans le commerce, développer notre propre solution, permet de mieux répondre à chacun des points énoncés précédemment. Cela permet également de proposer une plateforme fortement réutilisable de part son ouverture totale, ce qui permettra de l'utiliser dans d'autres projets et de la rendre plus fiable et polyvalente.

1.2 Choix de la plateforme

Nous avons choisi une plateforme de type quadrirotor. En effet, ce type de plateforme fait preuve de beaucoup d'agilité en vol, ainsi que d'une grande robustesse de par sa structure simple. Ce type de plateforme a été largement étudiée dans la littérature, offrant des modèles et des lois de contrôle éprouvés. De plus, ce type de plateforme fait partie des aéronefs que l'on classe comme VTOL (Vertical Take Off and Landing), permettant de décoller et d'atterrir de façon verticale, ce qui limite l'espace nécessaire pour ces manœuvres. Pour finir, ce genre de plateforme présente de bonnes performances en ce qui concerne l'emport de charge utile.

2 Design et dimensionnement

2.1 Structure et composants

Le quadrirotor est composé essentiellement de composants bas coûts disponibles dans le commerce.

- **La structure :** La structure du quadrirotor, appelée également "frame", est entièrement faite en PCB d'épaisseur $2mm$ de $20cm \times 20cm$ et pèse environ $62g$. L'utilisation d'une frame en PCB permet de limiter le nombre de fils présents sur la plateforme, mais surtout de profiter d'une structure, qui assure à la fois un rôle mécanique et électronique. Bien que plus lourde que d'autre frame présente dans le commerce (*micro-quad* de chez Hobbyking par exemple), cette frame (*NanoQuad*) de chez Flyduino présente l'avantage d'être très solide, car elle est faite d'une seule pièce. De plus, les raccords reliant l'extrémité de chaque bras, permettent de limiter les torsions en bout de bras, lors de fortes accélérations angulaires des moteurs. Ceci posait problème dans une première version du robot, utilisant la frame *micro-quad* de chez Hobbyking. Cette frame souffrait de fortes vibrations en bout de bras, créant de forts couplages entre les différents axes : roulis, tangage et lacet. Ces raccords engendrent cependant une diminution de l'efficacité de la poussée (voir sous section 4.4.2) à cause de l'occlusion qui en découle.
- **Le groupe de propulsion :** Le groupe de propulsion est l'élément clé d'un quadrirotor et conditionne directement les performances dynamiques, ainsi que l'endurance de la plateforme. On appellera ici propulsion, l'ensemble ESC (contrôleur brushless) + moteur brushless + hélice ;

j'ai testé durant ma thèse deux types de propulsion :

- *Les moteurs Turnigy* : Ces moteurs de 3800KV ont l'avantage d'être très bon marché, et présentent de bonnes performances dynamiques. Chaque moteur est piloté par un ESC TURNIGY PLUSH 6A, équipé d'une hélice 5030 en plastique. Cependant, le couple moteur-ESC est peu efficace d'un point de vue énergétique (environ 7-9 minutes d'autonomie pour une plateforme de 307g).
- *Les moteurs Tiger* : Ces moteurs, bien que plus chers, présentent des performances dynamiques un peu moins bonnes que les TURNIGY, mais avec un rendement bien meilleur et une meilleure fiabilité. Ils sont pilotés par des ESC Afro 12A de chez Hobbyking, équipé d'une hélice plastique 5030 hautes performances. Ce triplet ESC-moteur-hélice présente une grande efficacité sur le plan énergétique et permet d'atteindre plus de 16 minutes de vol pour une plateforme de 315 grammes.
- **Les tachymètres** : Contrairement à ce que laisse penser le nom "ESC" (Electronic Speed Controller), les contrôleurs brushless de base ne réalisent qu'un pilotage en boucle ouverte des moteurs et n'asservissent donc pas la vitesse de rotation. Pour des besoins évidents (voir section 2.5), j'ai mis en œuvre un asservissement local de la vitesse des moteurs, grâce à des tachymètres (également conçus durant ma thèse) permettant de mesurer la vitesse de rotation de chaque moteur. En fonction des moteurs (Turnigy ou Tiger), les tachymètres diffèrent légèrement. Pour plus de détails se référer à la partie 2.5.
- **Les supports de cartes** : Dans un premier temps, nous avons choisi de monter les cartes sur "silentbloc" afin de filtrer passivement les vibrations des moteurs et améliorer les mesures de la centrale inertielle. Cependant, nous avons remarqué que ces derniers n'apportaient pas grand chose et qu'il fallait de toute façon utiliser le filtre passe-bas analogique de la centrale inertielle, pour limiter les bruits de mesures. Les différentes cartes électroniques sont donc montées directement sur la structure, grâce à des colonnettes en plastique et une pièce de prototypage rapide.
- **Les pieds** : Il a été ajouté des pieds en carbone de 9 grammes chacun (pieds Hobbyking) afin de protéger l'œil lors des atterrissages et décollages. Chaque pied a ensuite été raccourci, afin de gagner du poids et passer de 9 à moins de 6 grammes par pied.

2.2 Modules de calcul

X4-MaG est composé en tout de trois modules de calcul distincts jouant chacun un rôle différent :

- Le micro-contrôleur DsPic33FJ128GP206 de chez Microchip permet de réaliser à 500Hz l'asservissement en vitesse de chacun des rotors.
- La Nanowii est une carte Arduino, équipée d'un micro-contrôleur ATmega32u4 et d'une centrale inertielle 6 axes (accéléromètres 3 axes et gyromètres 3 axes), permet de réaliser l'asser-

vissement en attitude du quadrirotor en mode manuel, grâce à des commandes émises par une radio-commande de modélisme.

- La Gumstix Overo AirSTORM, équipée d'un processeur de 1GHz, de 128MB de RAM et fonctionnant sous Linux, permet de réaliser l'asservissement en position du X4, lorsque celui-ci est connecté à une station sol et au système de capture de mouvement Vicon de l'arène de vol. Cette carte est entièrement programmable depuis MATLAB/Simulink grâce à la toolbox que j'ai développée durant ma thèse (voir la section 6).

L'utilisation de ces trois contrôleurs permet de mieux répartir la charge de calcul, mais aussi de séparer les boucles de contrôle de haut niveau et de bas niveau. Cela permet également de mettre en place une certaine hiérarchie, dans les contrôleurs, permettant à la fois d'apporter de la redondance en cas de défaillance d'un des contrôleurs et d'assurer la reprise en contrôle manuel à tout instant.

Ces différents points sont abordés dans la partie suivante.

2.3 Architecture logicielle et matérielle

Dans cette partie, je présenterai les différents choix de conception qui ont été réalisés pour le quadrirotor X4-MaG. Durant ma thèse, j'ai mis au point plusieurs configurations qui se présentent sous forme de modules :

1. **Basic configuration** : cette configuration de base ne fournit qu'un pilote manuel via une radio-commande de modéliste, et pilote les moteurs en boucle ouverte,
2. **Deluxe configuration** : cette configuration ajoute à la première une boucle de contrôle de la vitesse des rotors, permettant d'améliorer la réactivité et la stabilité,
3. **Accademic configuration** : cette configuration ajoute un contrôleur haut niveau fonctionnant sous Linux et entièrement programmable via Matlab/Simulink,
4. **Fly configuration** : cette configuration ajoute un système de vision découplé du corps du robot (voir chapitre IV pour la conception du système oculo-moteur et chapitre V pour les résultats obtenus en vol autonome),

Chacune de ces versions peut être équipée de deux systèmes de propulsion différents : le premier étant basé sur des moteurs TURNIGY bas coût, et le second sur des moteurs TIGER hautes performances. Les configurations 1 à 3 sont décrites en détails dans la rubrique "X4-MaG documentation" du site internet lié au projet (<http://www.gipsa-lab.fr/projet/RT-MaG>) et ne seront pas décrites ici. Par la suite, nous décrirons uniquement la dernière version qui regroupe l'ensemble des modules et qui utilise le système de propulsion basé sur les moteurs Tiger (notés T-Motor).

2.3.1 Architecture matérielle

La figure III.1 montre les connexions qui existent entre les différents composants de X4-MaG. La carte contrôle moteur, a été placée comme un élément central et permet de gérer l'asservissement en vitesses des moteurs, de distribuer les alimentations aux différents composants et de réaliser les adaptations de niveaux de tension sur les bus de communications. En effet, la Gumstix fonctionne à une logique de 1.8V, alors que la Nanowii et la carte contrôle-moteurs, dénommée RCB dans la suite (Rotor Control Board), fonctionne à 3.3V. La carte contrôle-moteurs distribue également l'alimentation fournie par les BEC des contrôleurs brushless (Battery Eliminator Circuit) aux différents composants, à savoir la Gumstix, la Nanowii, les servomoteurs (utilisés pour actionner l'œil présenté dans le chapitre IV).

Le choix de cette configuration matérielle s'explique par la modularité qu'elle offre. En effet, j'ai choisi de l'organiser de façon hiérarchique, afin de faciliter le "debug" et de pouvoir moduler la complexité de la plateforme. Il est alors très facilement possible de disposer de trois configurations différentes :

- **Nanowii seule** : La Nanowii fournit des consignes de vitesses pour les moteurs sous forme de PWM. Ainsi, si l'on choisit de ne pas utiliser la carte contrôle moteur, il suffit de relier directement la Nanowii aux différents ESC pour avoir accès au mode manuel. Les moteurs sont alors pilotés en boucle ouverte.
- **Nanowii et RCB** : La Nanowii n'est plus connectée directement aux moteurs, mais envoie ses consignes à la RCB (toujours sous forme de PWM, ce qui permet de ne rien changer au software de la Nanowii). La RCB quant à elle, grâce aux retours tachymétriques, assure l'asservissement en boucle fermée de la vitesse des quatre rotors. Cette solution permet d'obtenir un comportement plus dynamique et plus fiable (voir section 2.5).
- **Nanowii, RCB et Gumstix** : Dans cette dernière configuration, la Nanowii fournit toujours un mode manuel (qui peut être activé à tout moment par la télécommande) mais peut également fonctionner en tant qu'esclave de la Gumstix. Dans ce dernier cas, elle ne fait que relayer les consignes moteurs générées par la Gumstix. Cette configuration permet de tester des algorithmes beaucoup plus complexes, grâce à la forte capacité de calcul de la Gumstix. De plus, la Gumstix étant directement programmable depuis Simulink, cette configuration offre une plateforme de développement idéale pour tester rapidement de nouveaux algorithmes. La RCB assure alors, en plus de l'asservissement en vitesses des moteurs, les différentes adaptations de tensions pour les bus de communications partagés par la Nanowii et la Gumstix.
- **Nanowii, RCB, Gumstix + Vision** : Cette configuration est identique à la précédente, mais fournit en plus un système visuel (voir le chapitre IV). Le système de vision, doté d'un micro-contrôleur DsPIC33EP512GP806, est alors relié à la Gumstix via une liaison SPI cadencée à 12 MHz.

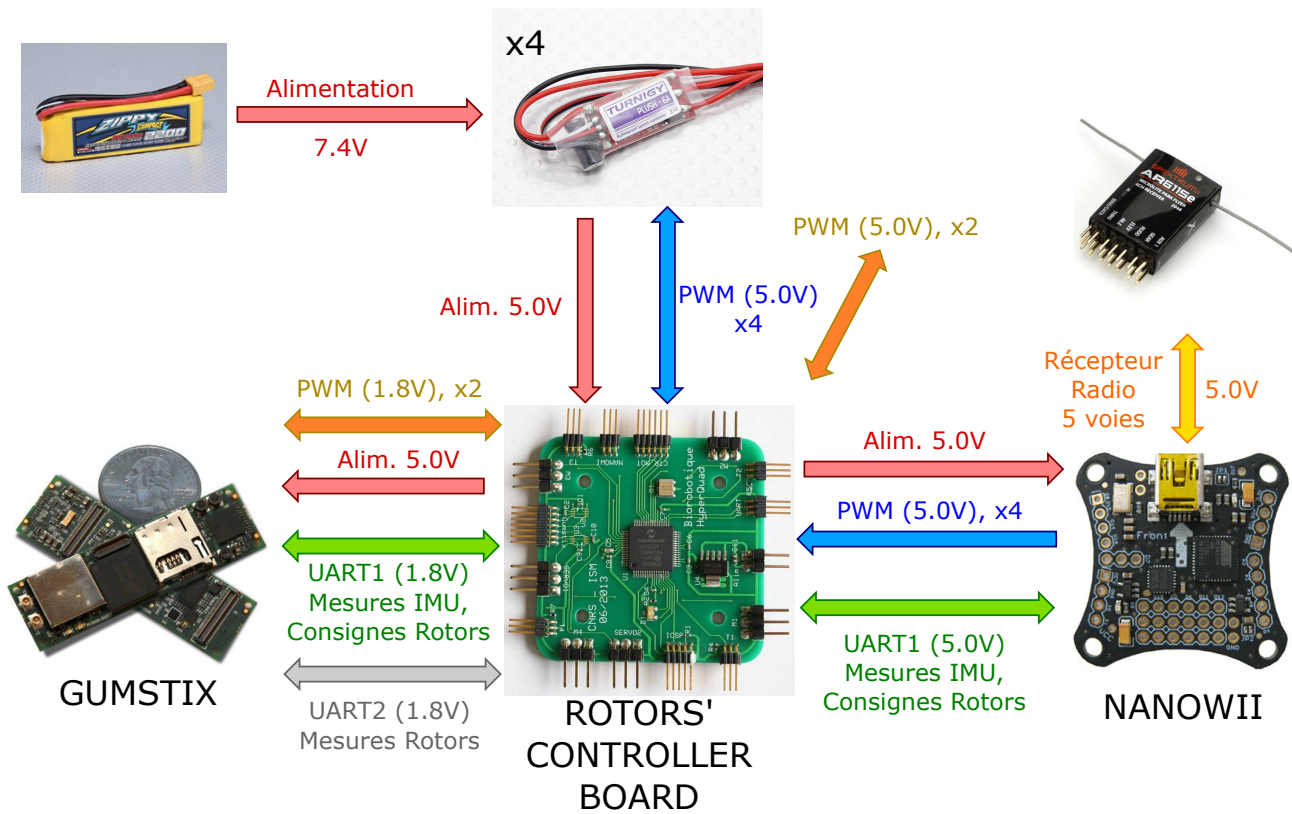


FIGURE III.1 – Les interconnexions des différents composants de X4-MaG. Les quatre ESC fournissent une alimentation stabilisée à 5V à partir de la batterie. La Gumstix est reliée par une liaison UART à 115200 bauds avec la Nanowii afin de transmettre les consignes de vitesses de rotations des rotors, et de récupérer les données inertielles de la Nanowii ; l'adaptation de tension de 1.8V à 3.3V est assurée par des "level translators" situés sur la carte contrôle moteur. La Nanowii envoie des consignes moteurs grâce à 4 PWM à la carte contrôle-moteur (RCB) qui asservit la vitesse des rotors via les ESC. La mesure de vitesse de rotation des moteurs arrive de la tachy de chaque moteur. De plus la Nanowii récupère à 50 Hz les commandes manuelles provenant d'une télécommande grâce à un récepteur radio. Enfin la Gumstix peut piloter 2 servomoteurs grâce à 2 sorties PWM. L'alimentation des servomoteurs, ainsi que l'adaptation des signaux de commandes provenant de la Gumstix, sont encore une fois gérées par la carte contrôle moteur. Enfin, une supervision de la vitesse des moteurs est réalisée grâce à une liaison UART à 230400 bauds entre la RCB et la Gumstix.

Dans ce mémoire, nous ne nous intéresserons qu'aux deux dernières configurations. Des informations à propos des deux premières peuvent être obtenues sur le site internet du projet à l'adresse suivante : <http://gipsa-lab.fr/projets/RT-MaG/>.

2.3.2 Architecture logicielle

La partie logicielle a été pensée et construite de façon à être compatible avec les différentes configurations matérielles citées précédemment. Ainsi, il est possible de passer d'une configuration à une autre, sans avoir à apporter de modifications à la structure du logiciel.

Le contrôleur bas niveau (la Nanowii) se trouve donc à une position centrale, d'un point de vue logiciel, car c'est elle qui recueille les mesures de l'IMU, et transmet les consignes des rotors (soit aux rotors directement, soit à la RCB). En effet, quelque soit le mode de fonctionnement (pilotage manuel par la Nanowii ou le pilotage automatisé par la Gumstix), c'est la Nanowii qui génère systématiquement les commandes PWM pour les rotors. Ceci présente l'avantage de passer d'un mode à l'autre sans rien avoir à changer d'un point de vue hardware. Cela permet également de reprendre le contrôle manuel à n'importe quel moment, via la radio-commande. Enfin, une sécurité additionnelle a été prévue, afin de permettre à la Nanowii de reprendre le contrôle de façon automatique, en cas de défaillance du contrôleur haut niveau (Gumstix). Les différents modes de fonctionnements sont décrits dans le paragraphe suivant.

Architecture Logicielle et sécurités : La figure III.2, met en évidence la structure logicielle qui a été mise en place pour X4-MaG. Le quadrirotor X4-MaG a été pensé pour fournir une plateforme, permettant de tester en toute sécurité, des algorithmes de navigation dans l'arène de vol de Marseille (voir la section 2.4 pour une description de cette arène et de son fonctionnement). Pour cela, il présente deux calculateurs différents : un calculateur haut niveau (la Gumstix) et un calculateur bas niveau (la Nanowii). Chacun de ces calculateurs présente plusieurs types de contrôleurs utilisés dans différents modes :

- Le calculateur haut-niveau (Gumstix) présente :
 - Un *mode expérimental* : c'est un mode qui repose sur le contrôleur à tester/valider et qui utilise les capteurs embarqués. Ce mode est enclenché à la demande de l'utilisateur (Flag), mais ne s'enclenche que sur validation du bloc de *supervision Gumstix*.
 - Un *mode de secours* : il repose sur les mesures du système Vicon et sur une architecture de contrôle, qui a été validée et qui est considérée fiable. Ce contrôleur reprend la main (sur décision d'un bloc de supervision) lorsque le *mode expérimental* s'écarte trop des objectifs de mission fixés.
- Le calculateur bas niveau (Nanowii) présente :
 - Un *mode automatique* : ce mode repose sur le calculateur haut niveau. Il lui transmet les informations de l'IMU et applique les consignes déterminées par le calculateur haut niveau aux rotors.
 - Un *mode fail-safe* : ce mode a pour but de maintenir le robot en vol stationnaire à la position actuelle, en cas de défaillance du contrôleur haut niveau, lorsque le mode automatique était enclenché. Il s'enclenche automatiquement dès lors que le calculateur haut niveau ne répond plus pendant plus de 50 ms. Étant donné que dans sa configuration actuelle, la Nanowii ne dispose pas de mesure de l'altitude (baromètre, télémètre, etc.), le contrôleur "maintient" au mieux l'altitude, en appliquant une poussée constante compensant la pesanteur. Le but de

ce mode est donc juste de stabiliser au mieux le robot, afin de laisser le temps à l'opérateur de reprendre le contrôle manuel.

- Un *mode manuel* : ce mode permet un contrôle manuel du robot via une radio-commande de modéliste. Ce mode peut être activé à n'importe quel moment via un interrupteur dédié sur la radio-commande.

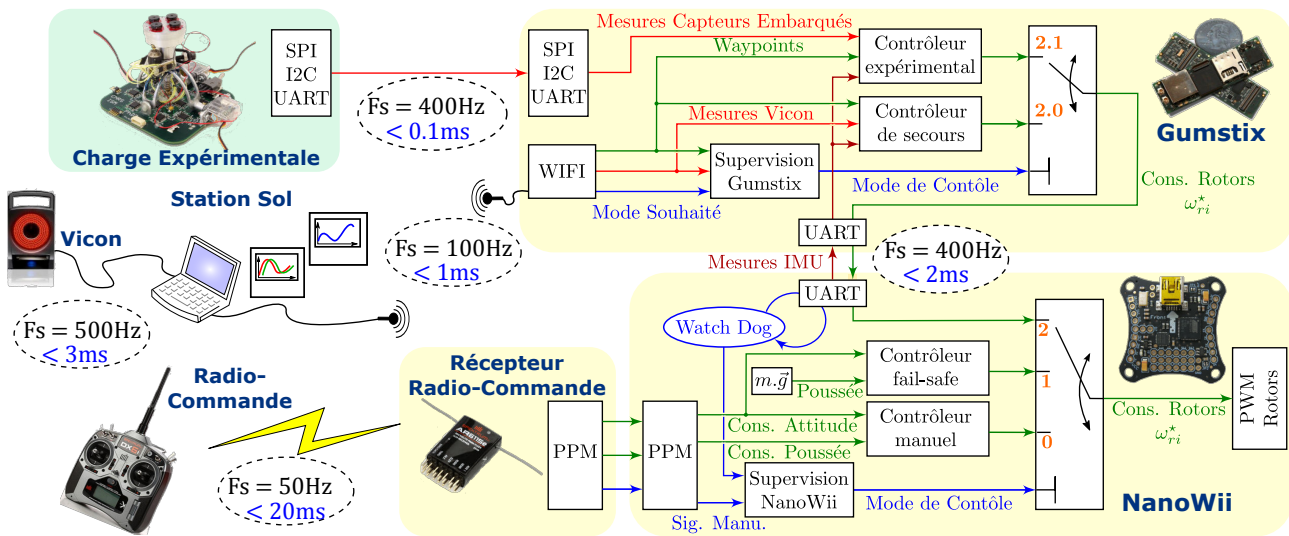


FIGURE III.2 – La structure logicielle du quadrirotor X4-MaG. Différents modes de fonctionnement sont disponibles afin de sécuriser la plateforme. Le *mode expérimental* de la Gumstix permet de tester de nouveaux algorithmes utilisant une charge expérimentale (capteurs embarqués). Dans le *mode expérimental ou de secours*, la Nanowii se contente de transmettre les consignes aux rotors calculées par la Gumstix. Un bloc de *supervision Gumstix* permet de basculer automatiquement sur le *contrôleur de secours*, si le robot s'écarte trop de son objectif. De même, en cas de défaillance de la Gumstix, le *superviseur Nanowii* enclenche le *mode fail-safe* pour stabiliser au mieux le robot, en attendant une reprise manuelle. Enfin, le *mode manuel* de la Nanowii permet d'asservir la poussée et l'attitude sur les consignes envoyées par la radio-commande. Ce *mode manuel* peut être enclenché à tout moment par l'utilisateur car il dispose de la priorité la plus élevée.

La figure III.2 présente donc la structure logicielle avec les niveaux de priorité des différents modes, qui apparaissent en orange dans les "switches" de la Gumstix et la Nanowii ; le niveau de priorité maximum étant 0. Chaque contrôleur peut préempter (c'est à dire prendre la main) tous les contrôleurs de niveau inférieur et être préempté par un contrôleur de priorité supérieure (sur décision des blocs de supervision de chaque calculateur). Le contrôleur de secours de la Gumstix peut préempter le contrôleur *expérimental*, si celui ci ne remplit pas ses objectifs sur décision du bloc *supervision*. Actuellement, le bloc *supervision* active le contrôleur de *secours*, si la position réelle du robot est éloignée de plus de 75cm de la consigne fixée (la procédure de transition est décrite plus longuement dans la section 7.4.2). Le contrôleur *fail-safe* de la Nanowii reprend la main si la Gumstix n'a pas envoyé de donnée sur la liaison UART pendant plus de 50 ms, ce qui est symptomatique d'un dysfonctionnement du

calculateur haut niveau. Enfin, le contrôleur *manuel* de la Nanowii peut reprendre la main à n'importe quel moment sur l'ordre de la radio-commande de l'utilisateur.

Remarque III.1: La reprise automatique du quadrirotor par le *contrôleur de secours* est illustrée dans la section 7.4 par la figure III.24.

2.4 Arène de vol et conditions expérimentales

L'ensemble des expérimentations qui sont présentées dans ma thèse, ont été réalisées dans la nouvelle arène de vol de Marseille (voir figure III.3 et le site internet www.marseilles-flying-arena.eu). Cette dernière fournit un espace de vol reconfigurable d'une taille maximale de $6\text{m} \times 6\text{m} \times 8\text{m}$ et couvert par 17 caméras Vicon T-40s. Une vue CAO de cette salle est disponible sur la figure III.3a. Le volume de vol peut être ajusté en faisant monter ou descendre la structure motorisée supportant les caméras. Une photographie de l'arène de vol durant le protocole de calibration est donné dans la figure III.3b. Dans cette arène, un ou plusieurs robots, peuvent être localisés simultanément avec une précision sub-millimétrique. En effet, une fois la calibration réalisée, la précision de localisation (écart type des mesures) est inférieure à 0.2mm (voir l'annexe A).

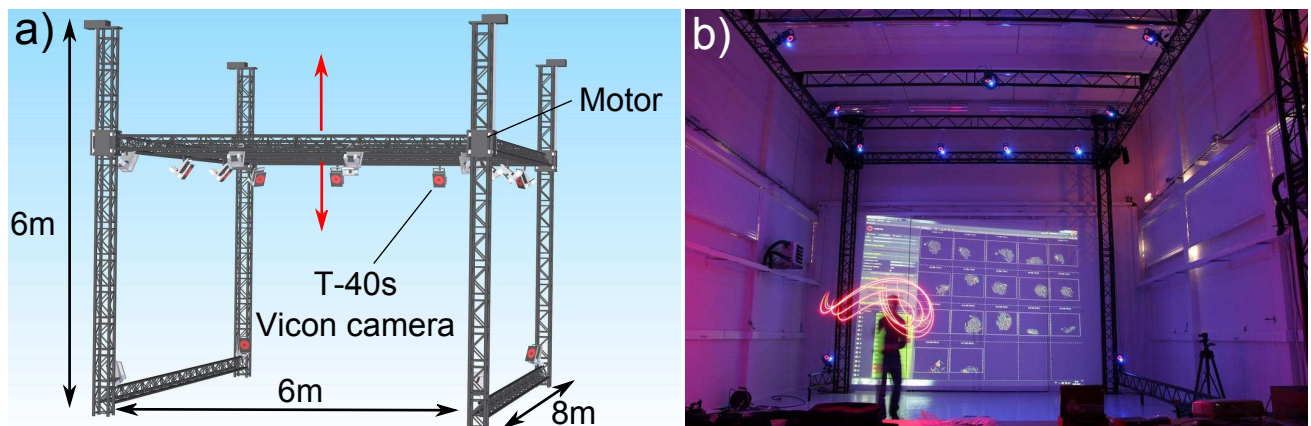


FIGURE III.3 – a) CAO (Conception Assistée par Ordinateur) de la nouvelle arène de vol de Marseille (Institut des Sciences du Mouvement, à Marseille, www.flying-arena.eu) composée d'une structure tubulaire motorisée en aluminium équipée de 17 caméras Vicon T-40s. b) Photographie de l'arène de vol de $6 \times 8 \times 6\text{m}^3$ lors d'une phase de calibration.

Durant ma thèse, j'ai fortement contribué au déploiement de cette salle. J'ai notamment réalisé un bloc Simulink permettant de recueillir en temps réel jusqu'à 500Hz l'ensemble des données Vicon (positions, quaternion matrice de rotation, etc.) via une interface utilisateur GUI (Graphical User Interface) donnant accès à l'ensemble des objets, marqueurs ou segments visibles par le système Vicon. Le bloc consiste en une s-fonction C++ exploitant les SDK C++ (Software Development Kit) fournis par Vicon. Ce bloc Simulink est disponible en open-source sur le site internet de la toolbox RT-MaG et

est d'ores et déjà utilisé par d'autres universités (West Virginia University par exemple). J'ai également mis en place un protocole (décrit dans le détail dans l'annexe A) afin de déterminer la précision de la localisation de ce système en fonction de la position 3D. De même, j'ai évalué les latences du système d'acquisition complet (motion capture + réseau). Les latences du système de capture sont évaluées et données directement par le système Vicon, et j'ai évalué les latences liées aux réseaux, grâce à des outils de capture réseaux (le "sniffeur" réseaux WireShark, voir [Wireshark Foundation, 2006]) sur de longues périodes de fonctionnement (plus de 30 minutes).

Fonctionnement dans l'arène de vol : Durant l'ensemble des expériences présentées dans cette thèse, trois systèmes sont systématiquement présents et interagissent entre eux :

- **Le système de capture du mouvement Vicon**, qui permet de localiser avec une grande précision (< 0.2 mm) les marqueurs équipant les robots. La localisation est réalisée à une fréquence de 500 Hz et présente des latences très faibles (< 3 ms, qui regroupe les latences dues au système Vicon lui même et aux latences liées au réseau Ethernet).
- **Une station sol**, qui envoie les positions du robot mesurées par le Vicon au calculateur haut niveau de X4-MaG. Cette station sol permet d'envoyer les consignes de position (waypoints), mais également d'enregistrer et de superviser l'ensemble des données calculées et envoyées via une connexion WIFI.
- **Le quadrirotor X4-MaG** est équipé d'une Gumstix Overo entièrement programmée grâce à la toolbox RT-MaG, développée pendant ma thèse. Le pilote automatique calcule de façon complètement autonome les trajectoires permettant de relier les waypoints envoyés par la station sol.

L'ensemble des liens de communications et des programmes du calculateur haut niveau (Gumstix) a été conçu, réalisé et testé sur Matlab/Simulink, qui permet également de réaliser la tâche de supervision temps réel via une station sol. Durant les expérimentations, les données du robot sont supervisées via le WIFI à une fréquence de 100Hz. L'exécution temps réel du modèle Simulink de la station sol est réalisée grâce au logiciel QUARC® 2.0 ([Quanser, 2009]). Le robot exécute également un modèle Simulink en temps réel (un exemple est donné par la figure III.22) sur le Linux embarqué de la Gumstix grâce à la toolbox RT-MaG (voir [Manecy, 2014]). Le quadrirotor est entièrement autonome d'un point de vue ressources calculatoires, étant donné que l'ensemble des estimateurs et contrôleurs sont exécutés sur la Gumstix (le contrôle de position est exécuté à 100Hz, l'estimation et le contrôle d'attitude sont opérés à 400Hz).

2.5 Groupe de propulsion et asservissement local

Comme expliqué précédemment, il est possible d'utiliser pour X4-MaG, deux systèmes de propulsions différents. Ces deux systèmes de propulsions sont basés sur le même schéma de commandes : Les

moteurs sont pilotés en tensions par des ESC qui reçoivent des consignes de vitesses sous forme de PWM. Ces ESC réalisent un pilotage en boucle ouverte des rotors, ne permettant donc pas de contrôler finement la vitesse de rotations des hélices.

Nous ne présenterons ici que le système de propulsion utilisant les moteurs *Tiger* (voir la section 2.1). La mise en œuvre pour l'autre système de propulsion est identique, et seuls les paramètres intrinsèques des moteurs changent (valeur de seuil, vitesses max, temps de réponse, etc.) Nous présentons donc ici la mise en œuvre d'un asservissement en boucle fermée de la vitesse des moteurs afin d'améliorer les performances dynamiques des rotors et de linéariser leur comportement.

2.5.1 Type de propulsion et Interfaces

Depuis maintenant plusieurs années, les moteurs de type "Brushless", se sont démocratisés dans le domaine des multi-rotors. En effet, ce type de moteur présente de nombreux avantages par rapport à leurs prédécesseurs (les moteurs à courant continu). Par exemple :

- Ils présentent de meilleurs rendements, du fait de l'absence de balais (réductions des frottements),
- L'absence de balais augmente grandement leur durée de vie, évitant ainsi une usure prématurée et par conséquent une variabilité des paramètres durant la durée de vie du moteur,
- Ils permettent également un montage des hélices directement sur l'axe du moteur, sans passer par un étage de réduction, améliorant ainsi grandement le rendement et la fiabilité du groupe de propulsion.

Cependant, le pilotage de ce type de moteur est plus complexe étant donné qu'il s'agit de moteurs triphasés asynchrones. L'utilisation de circuit électronique de commandes spécialisées que l'on appelle ESC, ou contrôleur Brushless, est donc indispensable. Ces éléments permettent de générer les signaux de puissance pour le moteur, et fournissent la plupart du temps une alimentation 5V stabilisée, permettant d'isoler l'alimentation du circuit de commande (BEC, Battery Electronic Cancellor). Les ESC reçoivent des consignes sous forme de PPM et réalise la plupart du temps un pilotage en boucle ouverte des moteurs, ce qui rend leur fonctionnement sensible aux perturbations aérodynamique, et aux variations de l'alimentation électrique. Les consignes sont alors plus exactement assimilables à des consignes de puissances fournies au moteur, qu'à de véritables consignes de vitesses.

Dans le cadre de ces travaux, nous souhaitons disposer d'une boucle de contrôle efficace sur la vitesse de rotation des rotors, car cela nous permettra par la suite, de contrôler les couples appliqués au quadrirotor (sous certaines hypothèses, voir section 2.5.3 et 4.4.2). Nous décrivons donc ici la mise en place de cet asservissement, ainsi que sa mise en œuvre sur la carte RCB.

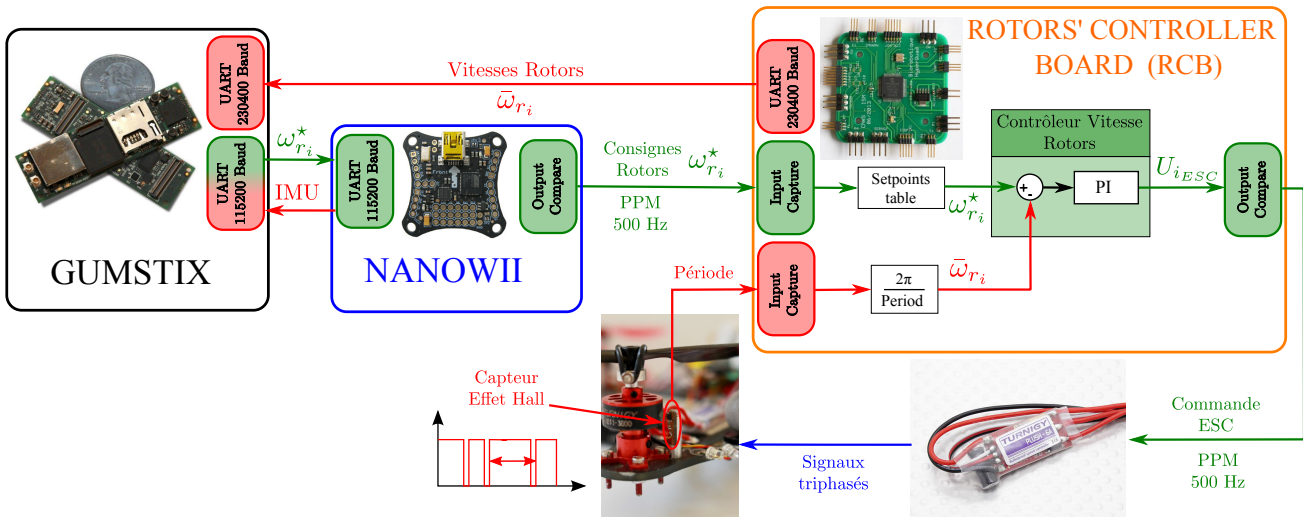


FIGURE III.4 – Boucle d’asservissement des rotors. Les consignes de position sont envoyées à la RCB par la Nanowii sous forme de signaux PPM. La vitesse de rotation des rotors est obtenue à partir des impulsions générées par un capteur à effet hall (placé très près de la cage tournante du moteur brushless). Un correcteur PI avec anti-windup et feedforward permet de générer les commandes à envoyer aux ESC, qui pilotent les moteurs en puissance. Les consignes de vitesses des rotors proviennent directement de la Nanowii, en cas de pilotage manuel. En mode automatique, les consignes sont générées par la Gumstix qui monitorise également, via une liaison UART, la vitesse de rotation des rotors.

2.5.2 Identification en boucle ouverte de la dynamique des rotors

La première étape consiste donc à identifier le modèle dynamique du couple ESC+moteur ainsi que les différents paramètres tels que les seuils de démarrage, les temps de synchronisation, les vitesses maximales et minimales atteignables, etc. La figure III.4 résume donc les interfaces du système de propulsion dans son ensemble, incluant les tachymètres permettant la mesure de vitesse de rotation. L’entrée de commande U_{i_ESC} du rotor i doit être un signal de type PPM (signal en créneau), ayant une fréquence comprise entre 50 Hz et 500 Hz. La consigne est codée sur la durée à l’état haut du signal, qui doit être comprise entre 1000 μ s et 2000 μ s, et correspondant respectivement à une consigne de vitesse nulle et maximale. En réalité, toutes les valeurs de commande inférieures à un certain seuil, sont considérées comme étant une consigne de vitesse nulle. Les mesures de vitesses sont obtenues grâce à des tachymètres, réalisés spécialement et qui sont disposés au plus près du rotor (voir figure III.4). Les tachymètres ne sont autres que des capteurs à effet hall, générant une impulsion à chaque passage d’un pôle du moteur. Connaissant le nombre de pôles d’un moteur, il est alors aisé de remonter à sa fréquence de rotation, en observant le temps écoulé entre chaque impulsion. Si un trop grand nombre d’impulsions est généré par tour (du fait d’un trop grand nombre de pôles), il est possible de diviser la fréquence en utilisant une simple bascule logique D (ou encore le diviseur de fréquence des entrées Input Capture du micro-contrôleur).

L’identification du modèle dynamique du système de propulsion peut être réalisé directement à par-

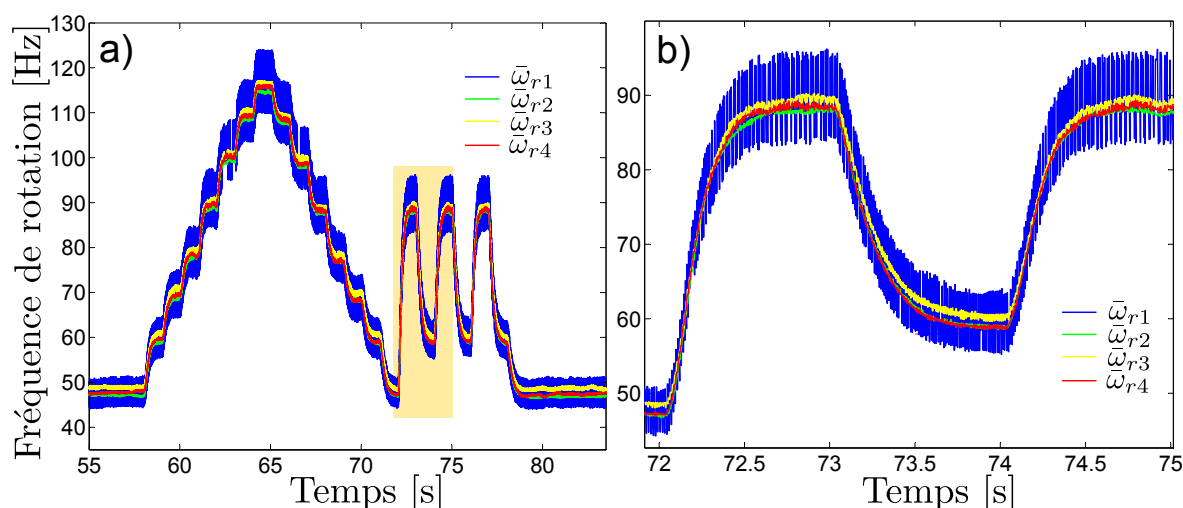


FIGURE III.5 – a) La réponse en boucle ouverte des différents rotors pour des échelons de commande de 100. En bleu la réponse en vitesse du robot 1, en vert du rotor 2, en jaune du rotor 3 et en rouge du rotor 4. On remarque que pour des mêmes échelons de commande, plus on demande des vitesses de rotation élevées, plus l'échelon de vitesse obtenu est faible (on obtient échelon de 12 Hz à 50 Hz et de seulement 6.5 Hz à 110 Hz). b) Zoom correspondant à l'encadré jaune de la figure a). La réponse de chacun des rotors diffère en amplitude et temps de réponse. On observe également, vers la fin ($t = 74.7s$), une variation de toutes les vitesses, liée à une variation de l'alimentation électrique. On note que le rotor 1 présente un bruit de mesure très important, probablement dû à du bruit électronique sur la voie du tachymètre 1.

tir du robot, sans avoir à ajouter de système de mesure ou de contrôle extérieur. En effet, l'émission des commandes et la lecture des mesures se font entièrement par le bus de communication UART, existant entre la RCB (contrôleur bas niveau des moteurs) et la Gumstix (contrôleur haut niveau). La carte RCB dispose d'un mode open-loop qui, une fois activé, permet d'appliquer directement les commandes venant du contrôleur haut niveau (c'est à dire la Gumstix), sans effectuer aucun asservissement. Ainsi, l'identification du modèle dynamique des rotors peut être effectuée à n'importe quel moment sans avoir besoin d'un banc extérieur. Le robot est simplement lesté à l'aide de poids, de façon à ne pas pouvoir décoller lors de la phase d'identification en open-loop. Cette méthode présente également l'avantage d'identifier le modèle dans sa globalité c'est à dire avec les retards introduits par le bus de communication. L'identification se fait simplement, en appliquant une série d'échelons de commande depuis le contrôleur haut niveau. Les commandes appliquées ainsi que les vitesses mesurées par les tachymètres sont enregistrées afin d'en déduire un modèle dynamique du système sous forme de fonction de transfert.

La réponse en boucle ouverte des moteurs *Tiger* est donnée en figure III.5. On remarque que le temps de réponse des moteurs ainsi que le gain, sont assez différents selon le point de fonctionnement (la vitesse de rotation du moteur). De même, le temps de réponse varie fortement, selon que l'on demande au moteur d'accélérer ou de ralentir (jusqu'à 2 fois plus long). Il a donc été décidé d'identifier un modèle, pour plusieurs point de fonctionnement, afin de décrire ces variations. Pour cela, on applique

une succession d'échelons en montée puis en descente (accélération et décélération) et on enregistre les commandes appliquées ainsi que la réponse en vitesses des moteurs. Ensuite, les données sont filtrées à l'aide d'un filtrage parallèle, afin de rendre l'identification moins sensible au bruit de mesure (on applique simultanément un filtre du premier ordre, ayant une fréquence de coupure à 7 Hz sur la commande et la mesure). Ensuite, une fonction de transfert de type premier ordre est identifiée grâce à un modèle ARMAX, pour la montée et la descente, à chacun des points de fonctionnement.

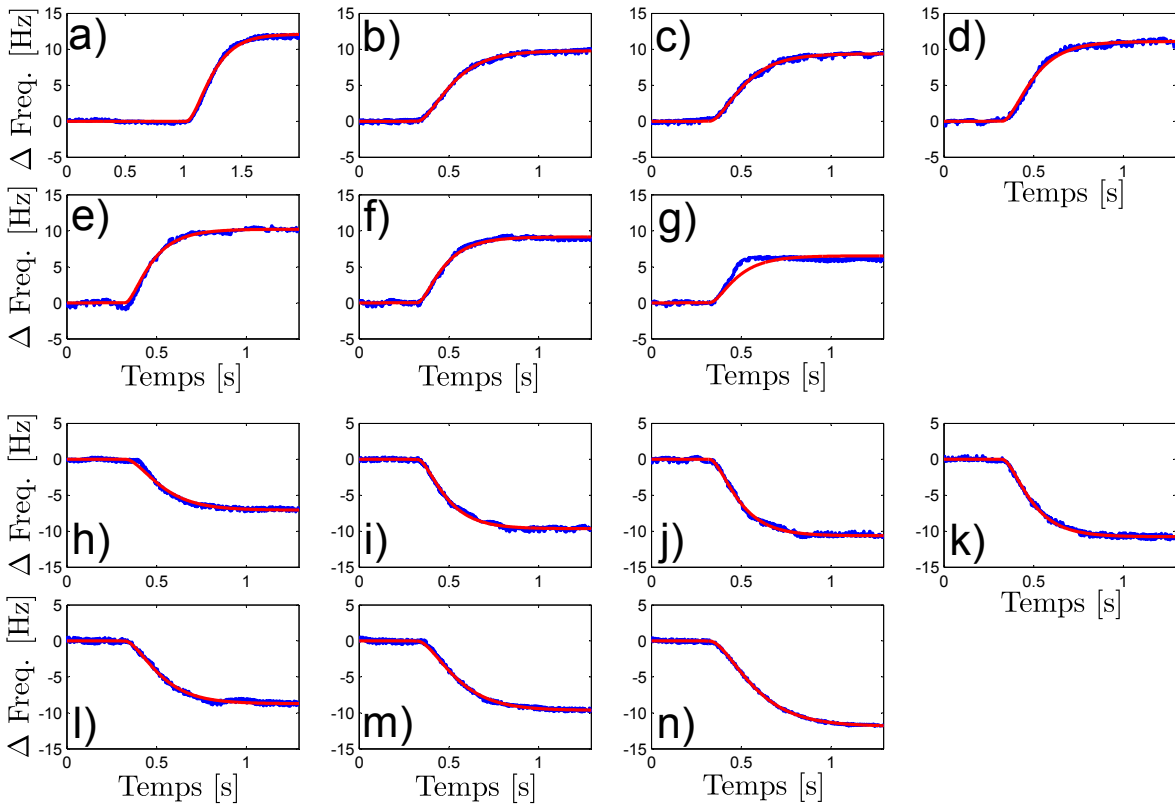


FIGURE III.6 – En bleu la vitesse mesurée du moteur, et en rouge le modèle identifié. Ici pour faciliter la comparaison, l'axe des ordonnées représente un écart de fréquence de rotation en Hertz. a) Échelon de 1200 à 1300. b) Échelon de 1300 à 1400. c) Échelon de 1400 à 1500. d) Échelon de 1500 à 1600. e) Échelon de 1600 à 1700. f) Échelon de 1700 à 1800. g) Échelon de 1900 à 1800. h) Échelon de 1900 à 1800. i) Échelon de 1800 à 1700. j) Échelon de 1700 à 1600. k) Échelon de 1600 à 1500. l) Échelon de 1500 à 1400. m) Échelon de 1400 à 1300. n) Échelon de 1300 à 1200.

Les figures III.6a-n comparent la réponse du modèle identifié, à la réponse réelle du système pour quelques points de fonctionnement, pour des accélérations (III.6a-g) et des décélérations (III.6h-n). Le tableau III.1 recense les paramètres des fonctions de transferts identifiées pour l'ensemble des points de fonctionnement. On remarque que plus la fréquence de rotation est élevée, moins l'échelon de commande a d'effet; alors que pour un échelon de commande allant de 1200 à 1300, la fréquence de rotation augmente de 12 Hz, elle n'augmente que de 6.5 Hz pour une échelon de 1800 à 1900. Cela s'explique par les forces de frottement aérodynamique qui sont de plus en plus importantes. L'amplitude de fréquence

obtenue étant plus faible, le régime permanent est atteint plus rapidement. C'est à dire que plus la fréquence de rotation augmente, plus la constante de temps et donc le temps de réponse diminue.

Modèles à la montée							
	a	b	c	d	e	f	g
G	0.1211	0.0978	0.0937	0.1108	0.1021	0.0913	0.0651
τ	0.1249	0.0953	0.0911	0.0711	0.0442	0.0448	0.0509
Modèles à la descente							
	h	i	j	k	l	m	n
G	0.0707	0.0963	0.1061	0.1074	0.0870	0.0963	0.1184
τ	0.1009	0.0559	0.0616	0.0605	0.0943	0.1184	0.1507

TABLE III.1 – Les paramètres des modèles dynamiques qui ont été identifiés pour différents points de fonctionnement. Les modèles sont identifiés par une lettre qui correspond au résultats présentés dans la figure III.6

On déduit donc de cette étude une plage de variation pour chacun des paramètres G et τ , ainsi qu'un modèle moyen, donné par le tableau III.2. On remarque que la constante de temps τ varie sur une plage de plus de 80% et le gain G sur une plage de plus de 30%. Cela prouve bien que les couples générés pour contrôler l'attitude du robot, sont fortement dépendants du point de fonctionnement des moteurs. Cela justifie donc bien la nécessité de réaliser un asservissement local des moteurs, afin de rendre les performances des moteurs identiques quelque soit le point de fonctionnement, ce qui permettra d'obtenir un asservissement d'attitude plus précis. De plus, en réalisant cet asservissement, on peut facilement connaître la poussée totale générée, ce qui s'avérera particulièrement utile pour la suite, lorsqu'on aura à estimer les différents états du robot comme les vitesses latérales, positions, etc. (voir le chapitre V).

	Min	Max	Mean	Std	Std [%]	MaxVar [%]
G	0.065	0.121	0.097	0.016	±16.3	±33.1
τ	0.044	0.151	0.053	0.033	±39.6	±81.2

TABLE III.2 – Paramètres moyens et plage de variations des modèles des propulseurs du quadrirotor X4-MaG avec moteur Tiger et les Afro ESC 12A de chez HobbyKing.

2.5.3 Asservissement en boucle fermée

Enjeux et rappel théorique Le but de l'asservissement en vitesse des rotors est de contrôler la poussée générée par chaque rotor, qui est directement liée au couple appliqué au corps du quadrirotor. Dans le cas d'un quadrirotor en configuration X_4 (en opposition a $+4$), la relation existant entre les

couples et la poussée de chaque rotor est décrite par :

$$\begin{pmatrix} \Gamma_\phi \\ \Gamma_\theta \\ \Gamma_\psi \end{pmatrix} = \underbrace{\sin\left(\frac{\pi}{4}\right) \begin{pmatrix} -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}}_M \begin{pmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{pmatrix} \quad (\text{III.1})$$

Où Γ_ϕ , Γ_θ et Γ_ψ représentent les couples de roulis, tangage et lacet (c'est à dire autour des axes \vec{x} , \vec{y} et \vec{z} respectivement) ; T_i représente la poussée générée par le rotor i .

Les couples résultent donc d'une combinaison linéaire des poussées de chacun des rotors, et le passage de l'un à l'autre, s'opère grâce à la matrice M (qui est une matrice constante). Il suffit donc, pour contrôler les couples, d'asservir la poussée des rotors. Or, celle-ci est facilement déduite de la vitesse de rotation des rotors grâce à la relation suivante :

$$T_i = c_T \cdot \omega_{r_i}^2 \quad (\text{III.2})$$

Où T_i est la poussée du rotor i , et ω_{r_i} , sa vitesse de rotation. Le paramètre c_T est un paramètre aérodynamique constant qui dépend de la géométrie et la "souplesse" de l'hélice, et qui peut être facilement identifié par des tests de poussée statique (voir section 4.4).

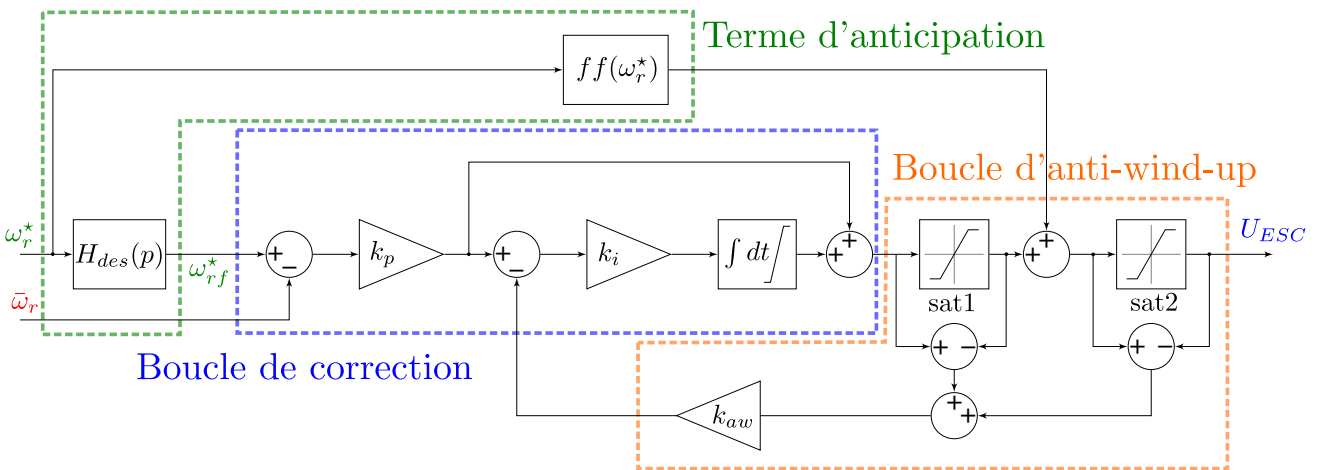


FIGURE III.7 – La boucle de contrôle de la vitesse des rotors est constituée d'un pré-filtre de consigne, d'un correcteur PI classique, associé à un double anti-wind-up et à un terme d'anticipation. Le correcteur PI est ajusté de façon à obtenir une dynamique bien plus rapide que celle du pré-filtre, ce qui assure une bonne poursuite de la consigne filtrée. Le terme d'anticipation permet de générer, à priori, la réponse souhaitée afin de limiter encore plus l'écart de poursuite. Enfin une double boucle d'anti-windup permet d'éviter les "overshoots" et "undershoots", liés à l'intégrateur.

Cahier des charges et implication Pour obtenir une réponse linéaire sur le couple, il est donc nécessaire d'obtenir une réponse linéaire et parfaitement identique sur chacun des rotors. En effet, si l'on note Γ_x^* les références de couples souhaitées, et $H_{des}(p)$ la dynamique en boucle fermée souhaitée, on a alors $\Gamma_x(p) = H_{des}(p) \cdot \Gamma_x^*(p)$. Ce qui implique que l'équation III.1 devient :

$$\begin{pmatrix} \Gamma_\phi \\ \Gamma_\theta \\ \Gamma_\psi \end{pmatrix} = H_{des}(p) \begin{pmatrix} \Gamma_\phi^* \\ \Gamma_\theta^* \\ \Gamma_\psi^* \end{pmatrix} = M \cdot H_{des}(p) \cdot \begin{pmatrix} T_1^* \\ T_2^* \\ T_3^* \\ T_4^* \end{pmatrix} = M \cdot \begin{pmatrix} H_{des}(p) \cdot T_1^* \\ H_{des}(p) \cdot T_2^* \\ H_{des}(p) \cdot T_3^* \\ H_{des}(p) \cdot T_4^* \end{pmatrix} \quad (\text{III.3})$$

On note donc qu'il est important que $T_i(p) = H_{des}(p) \cdot T_i^*(p)$, avec $H_{des}(p)$ toujours la même fonction.

Pour obtenir cela nous allons utiliser la structure de contrôleur donnée par la figure III.7 et décrit par l'équation suivante :

$$\left\{ \begin{array}{l} U_i = k_p(\omega_{r_{f_i}}^* - \bar{\omega}_{r_i}) + k_i \int_0^t (k_p(\omega_{r_{f_i}}^* - \bar{\omega}_{r_i}) - k_{aw} \cdot \Delta sat) dt \\ \Delta sat = \underbrace{U_i - \sigma_{sat1_{min}}^{sat1_{max}}(U_i)}_{\Delta sat1} + \underbrace{\sigma_{sat1_{min}}^{sat1_{max}}(U_i) + ff(\omega_{r_i}^*) - \sigma_{sat2_{min}}^{sat2_{max}}(\sigma_{sat1_{min}}^{sat1_{max}}(U_i) + ff(\omega_{r_i}^*))}_{\Delta sat2} \\ U_{ESC_i} = \sigma_{sat2_{min}}^{sat2_{max}}(\sigma_{sat1_{min}}^{sat1_{max}}(U_i) + ff(\omega_{r_i}^*)) \end{array} \right. \quad (\text{III.4})$$

Où i représente le numéro du rotor, σ_a^b représente la fonction saturation avec comme saturation basse a et saturation haute b . $\omega_{r_{f_i}}^*$ et $\bar{\omega}_{r_i}$ sont respectivement la consigne de vitesse filtrée et la mesure de vitesse du rotor i . Les saturations sont choisies comme suit :

$$\left\{ \begin{array}{l} sat1_{min} = T_{up_{min}} - T_{up_{max}} \\ sat1_{max} = T_{up_{max}} - T_{up_{min}} \\ sat2_{min} = T_{up_{min}} \\ sat2_{max} = T_{up_{max}} \end{array} \right. \quad (\text{III.5})$$

Où $T_{up_{min}} = 1120$ et $T_{up_{max}} = 2000$ correspondent respectivement à la commande minimale et maximale pour U_{ESC_i} . Leur valeur représente un temps à l'état haut de la commande PPM, exprimé en microsecondes.

Mise en œuvre Sur la base des résultats précédents, on réalise donc un asservissement en boucle fermée de la vitesse. Étant donné qu'il n'est pas possible de faire freiner le moteur (les ESC utilisés ne le permettent pas), la dynamique du moteur ne peut presque pas être modifiée sur les décélérations. Or, si l'on veut obtenir un comportement linéaire sur le couple comme décrit par l'équation III.3, il est nécessaire de choisir une dynamique en boucle fermée qui soit accessible, aussi bien pour les

accélération que pour les décélérations. Pour obtenir cela on utilise la structure de contrôleur illustrée par la figure III.7. La dynamique du préfiltre de consigne $H_{des}(p) = \frac{1}{\left(\frac{1}{2\pi f_c}p + 1\right)^2}$ avec $f_c = 7\text{Hz}$ a été choisie pour correspondre à la dynamique de décélération la plus rapide atteignable par les rotors.

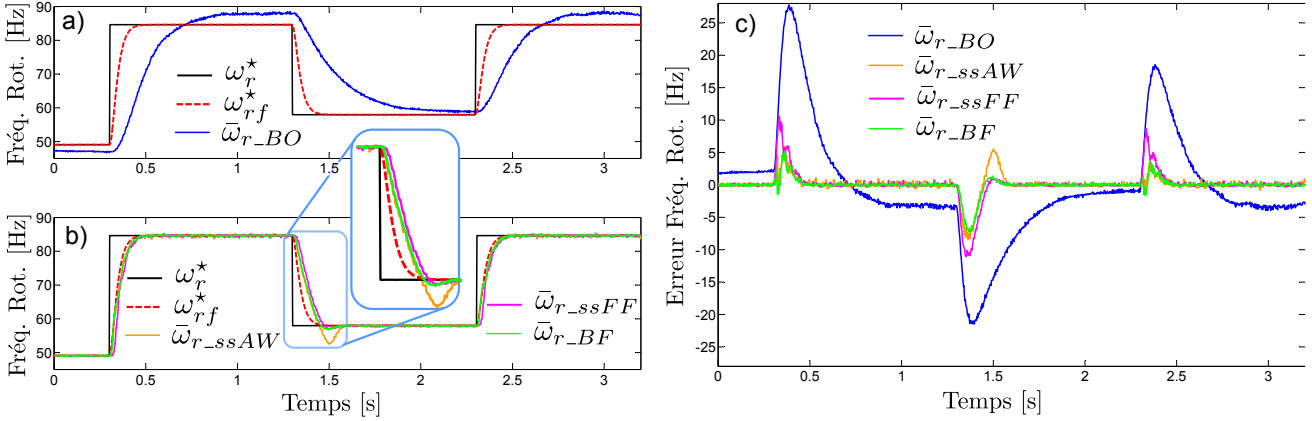


FIGURE III.8 – a) Réponse indicielle d'un rotor en boucle ouverte en bleu continu et en pointillés rouge la réponse souhaitée en boucle fermée. b) Réponse d'un rotor en fonction des différentes versions du contrôleur de rotors. En noir, la consigne de vitesse à atteindre, en pointillés rouge la trajectoire désirée pour atteindre cette consigne. En orange continu, la réponse en boucle fermée, lorsque l'anti-windup est désactivé. En mauve, la réponse en boucle fermée avec anti-windup mais sans le terme d'anticipation. En vert la réponse en boucle fermée avec l'anti-windup, et le terme d'anticipation actifs. c) Évolution de l'erreur de poursuite (par rapport à la trajectoire souhaitée ω_{rf}^*) pour les différentes variantes du contrôleur de rotors. Le contrôleur avec anti-windup et terme d'anticipation présente les meilleures performances.

L'asservissement en vitesses des rotors est donc réalisé à l'aide d'un correcteur PI classique et d'un anti-windup, associé à un terme d'anticipation. Les consignes des rotors sont également filtrées, afin d'harmoniser la dynamique d'accélération et de décélération et d'assurer une réponse identique pour chacun des rotors. La dynamique du correcteur PI est choisi de façon à obtenir une dynamique de boucle bien plus rapide que celle du préfiltre, ce qui assure une bonne poursuite. Le terme d'anticipation $ff(\omega_r^*)$ consiste en une table associant à chaque consigne de vitesse une valeur de commande nominale. La commande nominale correspondant à une vitesse a été construite grâce à des essais en boucle ouverte, réalisés avec une batterie pleine. En effet, une commande u_0 génère, avec une batterie pleine ayant une tension V_0 , une vitesse de rotation ω_0 . Pour une tension $V_1 < V_0$ on obtient pour cette même commande u_0 plus qu'une vitesse de rotation ω_1 , avec $\omega_1 < \omega_0$. Il est important de construire la table sur la base d'une batterie pleine, afin de ne pas surestimer la commande nominale nécessaire. En effet, surestimer cette commande aurait pour effet de rendre la dynamique de la boucle fermée plus lente lorsque l'on demande au moteur de ralentir. La table est donc constituée de 10 couples "consigne/commande"; les valeurs manquantes sont calculées en ligne par interpolation linéaire.

L'anti-windup permet d'annuler les overshoots et undershoots liés à l'effet intégrale de la commande,

comme l'illustrent les figures III.8b-c. On observe alors que l'anti-windup agit surtout sur les phases de décélérations. En effet, sans anti-windup, on obtient un dépassement de plus de 20% (courbe orange) alors qu'avec l'anti-windup, le dépassement est inférieur à 3.8% (courbe mauve et verte). Comme le montre le zoom de la figure III.8b que l'anti-windup est donc un élément clé pour obtenir une réponse quasi-identique, lors des phases d'accélération et de décélération. Enfin, on voit que le terme d'anticipation permet de répondre un peu plus rapidement et ainsi de réduire l'erreur de poursuite. La contribution de ce terme $ff(\omega r^*)$ est nettement visible sur la figure III.8 en observant l'erreur de poursuite sans ce terme d'anticipation (courbe mauve) et avec ce terme (courbe verte). Celui-ci contribue à éliminer près de 35% de l'erreur de trainée et donc d'améliorer encore un peu la poursuite de la référence.

Finalement, la figure III.9 montre la réponse en boucle fermée, des quatre rotors à une même consigne. On observe que la boucle fermée permet (en plus d'éliminer les erreurs statiques) de améliorer fortement le temps de réponse, qui passe de 159 ms en moyenne en boucle ouverte (453 ms dans le pire cas) à seulement 68 ms en boucle fermée. L'utilisation du même pré-filtre de consignes permet, de surcroît, d'atteindre des réponses quasi identiques sur chacun des rotors. Ainsi, on obtient une réponse linéaire sur les couples permettant de contrôler l'attitude du robot.

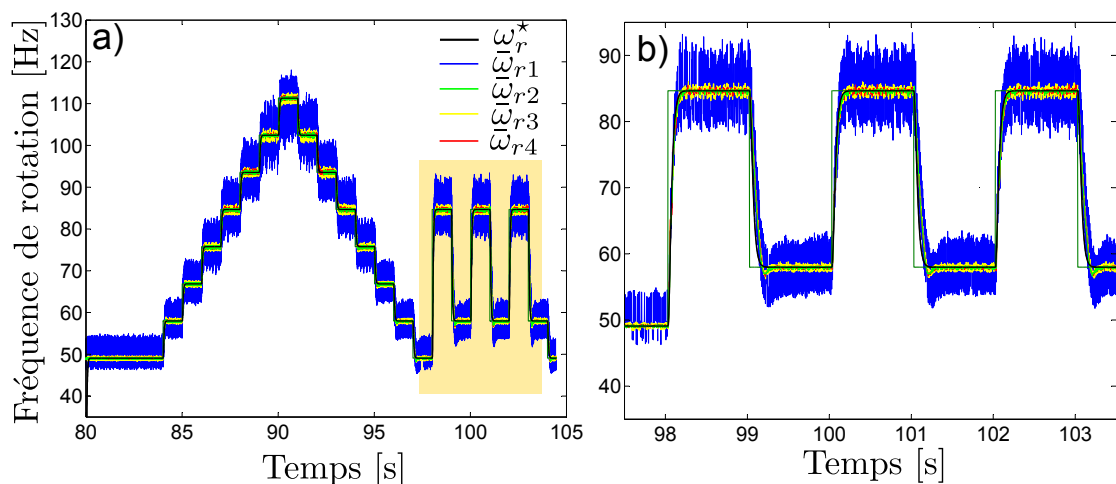


FIGURE III.9 – a) Réponses en vitesse à une même consigne des quatre rotors. b) Zoom sur la réponse des moteurs. On observe qu'au bruit près, le réponse des quatre rotors est strictement identique, et que la réponse à un échelon montant ou descendant, est également presque identique. On note que la présence de bruit sur la mesure du rotor 1, ne dégrade pas les performances de l'asservissement de vitesse de ce rotor.

3 Formalisme et angles d'Euler

A partir de ce chapitre et pour la suite du document, nous définirons deux repères pour lesquels nous notons $\{\vec{x}, \vec{y}, \vec{z}\}$ les vecteurs unitaires de chacun des axes. Le premier repère, noté $\{E\}$ est le repère orthonormé inertiel (ou référentiel terrestre) ayant pour vecteur unitaire $\{\vec{e}_1, \vec{e}_2, \vec{e}_3\}$ (c'est à dire que l'on a $\vec{e}_1 = \vec{x}$, $\vec{e}_2 = \vec{y}$ et $\vec{e}_3 = \vec{z}$ dans $\{E\}$). Le second repère est noté $\{B\}$ et correspond au repère fixe du référentiel du robot dont les vecteurs unitaires sont notés $\{\vec{b}_1, \vec{b}_2, \vec{b}_3\}$. On définit alors la matrice de rotation $R \in SO(3)$ permettant de définir l'orientation du repère fixe du robot $\{B\}$ dans le repère inertiel $\{E\}$. C'est à dire que l'on peut écrire $\vec{b}_1 = R\vec{e}_1 = R\vec{x}$, $\vec{b}_2 = R\vec{e}_2 = R\vec{y}$ and $\vec{b}_3 = R\vec{e}_3 = R\vec{z}$. Ces transformations de repères sont décrites par la figure III.10.

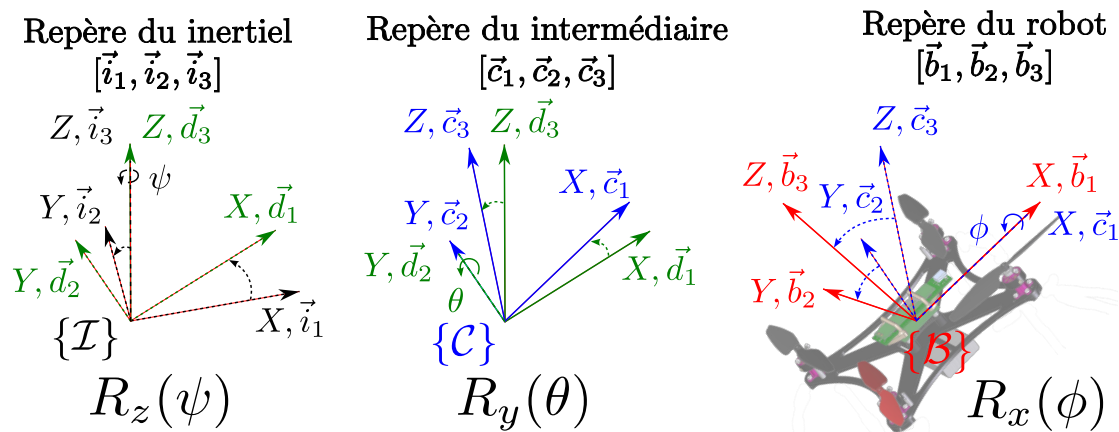


FIGURE III.10 – Ensemble des rotations permettant de passer du repère inertiel \mathcal{I} au repère fixe du robot \mathcal{B} . Le passage du repère inertiel au repère fixe du robot est obtenu en réalisant une combinaison de trois rotations successives notées R_z , R_y et R_x d'angles ψ , θ et ϕ respectivement. Ces trois angles sont appelés angles d'Euler et permettent de décrire l'attitude du robot, c'est à dire son orientation dans l'espace.

La figure III.10 décrit le passage du repère inertiel au repère fixe du robot qui permet de définir les trois angles d'Euler. Le repère inertiel est noté \mathcal{E} et est décrit par une base orthonormée $(\vec{e}_1, \vec{e}_2, \vec{e}_3)$. On effectue alors une première rotation d'angle ψ autour de l'axe Z du repère inertiel, ce qui donne lieu à un nouveau repère \mathcal{D} de base $(\vec{d}_1, \vec{d}_2, \vec{d}_3)$. Ensuite, une seconde rotation d'angle θ autour de l'axe Y de ce nouveau repère \mathcal{D} permet de passer au repère \mathcal{C} de base $(\vec{c}_1, \vec{c}_2, \vec{c}_3)$. Enfin une dernière rotation d'angle ϕ s'effectue autour de l'axe X du repère \mathcal{C} et donne lieu au repère fixe noté \mathcal{B} et de base orthonormée $(\vec{b}_1, \vec{b}_2, \vec{b}_3)$.

Cette matrice de rotation R correspond donc au formalisme d'Euler X-Y-Z intrinsèque (rotation

autour de X, puis de Y puis Z) où Z-Y-X extrinsèque et sa matrice de rotation s'écrit :

$$\begin{aligned}
 R &= R_z R_y R_x \\
 &= \begin{pmatrix} c(\psi) & -s(\psi) & 0 \\ s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} c(\theta) & 0 & s(\theta) \\ 0 & 1 & 0 \\ -s(\theta) & 0 & c(\theta) \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi) & c(\phi) \end{pmatrix} \\
 &= \begin{pmatrix} c(\theta)c(\psi) & s(\phi)s(\theta)c(\psi) - c(\phi)s(\psi) & s(\phi)s(\psi) + c(\phi)s(\theta)c(\psi) \\ c(\theta)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\theta)s(\psi) & c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi) \\ -s(\theta) & s(\phi)c(\theta) & c(\phi)c(\theta) \end{pmatrix}
 \end{aligned}$$

Où $c(\alpha)$ et $s(\alpha)$ représentent respectivement les cosinus et sinus de l'angle α .

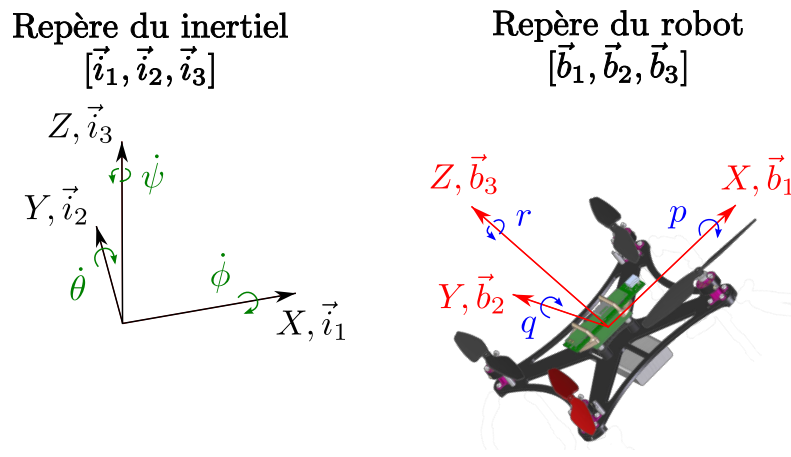


FIGURE III.11 – Expression des vitesses de rotations liées aux angles d'Euler et le repère fixe.

Des transformations précédentes, on peut déduire la relation existant entre les vitesses de rotations du repère du robot \mathcal{B} notées, $(p \ q \ r)^T$ et les vitesses de rotations liées aux angles d'Euler que l'on note $(\dot{\phi} \ \dot{\theta} \ \dot{\psi})^T$. Ces deux vecteurs vitesses de rotations sont décrits par la figure III.11. On obtient donc la relation suivante :

$$\begin{aligned}
 \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi) & c(\phi) \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi) & c(\phi) \end{bmatrix} \begin{bmatrix} c(\theta) & 0 & s(\theta) \\ 0 & 1 & 0 \\ -s(\theta) & 0 & c(\theta) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\
 \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= J^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}
 \end{aligned} \tag{III.6}$$

L'inversion de la matrice J^{-1} donne alors la relation permettant de déduire à partir des vitesses angulaires du repère fixe, les taux de rotations des angles d'Euler :

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = J \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix}}_W \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (\text{III.7})$$

La matrice de passage W est appelée matrice Wronksienne, et on notera qu'elle n'est pas définie dans le cas où l'angle de tangage proche de 90° $\theta = 90^\circ$ [180°]. Cependant, dans le cas de petites manœuvres, c'est à dire tant que le robot conserve des angles proches de zéro, la matrice W peut être approximée :

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (\text{III.8})$$

On utilisera donc la relation III.7 dans le cas d'un simulateur et on supposera que III.8 est vrai lors de l'élaboration de contrôleurs, afin de simplifier leur calcul et mise en œuvre.

4 Modélisation complète du quadrirotor

Dans cette partie, je présente le modèle complet du quadrirotor ayant servi à développer un simulateur, afin de tester et valider les différents algorithmes de commande et d'estimation. J'essaierai donc ici, de présenter les principaux phénomènes impactant la dynamique d'un quadrirotor et d'apporter la signification physique chacun. Ici, je m'inspire des différents modèles que l'on peut trouver dans la littérature de ces dernières années ([Hamel *et al.*, 2002], [Nice, 2004], [Bouabdallah *et al.*, 2004], [Bouabdallah, 2007], [Pounds *et al.*, 2006], [Huang *et al.*, 2009], [Pounds *et al.*, 2010]).

La modélisation complète du quadri-rotor est obtenue grâce au modèle classique d'un corps rigide en vol libre ([Mahony *et al.*, 2012]) :

$$\dot{\xi} = v \quad (\text{III.9})$$

$$m\dot{v} = -mg\vec{e}_3 + F_\Sigma \quad (\text{III.10})$$

$$\dot{R} = R\Omega_\times \quad (\text{III.11})$$

$$I\dot{\Omega} = \underbrace{-\Omega \times I\Omega}_{\tau_{bge}} + \tau_\Sigma \quad (\text{III.12})$$

Où ξ correspond à la position du centre de gravité G du robot, v correspond à la vitesse de ce centre, R est la matrice d'orientation du robot dans le repère inertiel (comme décrit précédemment) et Ω

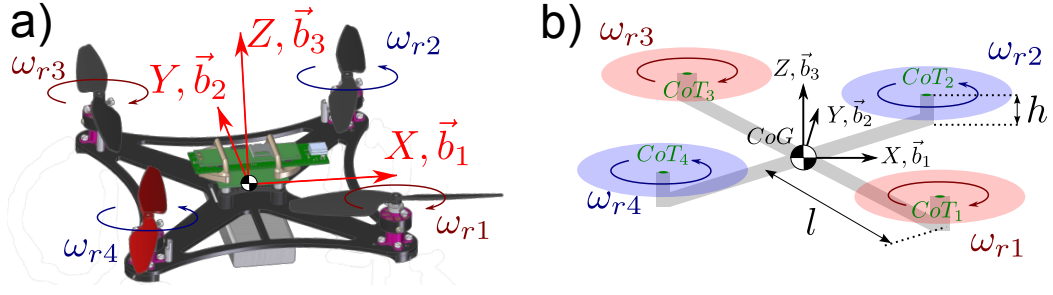


FIGURE III.12 – a) Convention utilisée pour numérotter les hélices de X4-MaG. L'axe X pointe entre les hélices 1 et 2, l'axe Y pointe entre les hélices 2 et 3 et l'axe Z pointe vers le haut (direction principale de la force de poussée des hélices). b) Convention utilisée pour le sens de rotation des hélices. Les hélices 1 et 3 sont dites "clockwise", (tournant dans le sens des aiguilles d'une montre, vue de dessus) et les hélices 2 et 4 "counter-clockwise" (tournant dans le sens inverse des aiguilles d'une montre). L'axe X de vecteur directeur \vec{b}_1 donne la direction du sens "avant". La distance l représente la longueur d'un bras et h l'offset vertical (selon la direction Z de vecteur directeur \vec{b}_3) entre les centres de poussée (CoT_i) et le centre de gravité du quadricoptère (CoG).

correspond au vecteur des vitesses de rotation du corps rigide. On définit Ω_\times comme étant la matrice antisymétrique, associée au vecteur des vitesses de rotations Ω , c'est à dire que pour toute matrice A de dimension 3×3 on a $A \times \Omega = A\Omega_\times$:

$$\Omega_\times = \begin{pmatrix} 0 & p & -q \\ -p & 0 & r \\ q & -r & 0 \end{pmatrix} \quad (\text{III.13})$$

Le terme τ_{bge} correspond à l'effet gyroscopique induit par les vitesses de rotation du corps. On notera par la suite le vecteur de position $\xi = (x \ y \ z)^T$, le vecteur des vitesses de translation $v = (V_x \ V_y \ V_z)^T$, le vecteur des vitesses angulaires $\Omega = (p \ q \ r)^T$.

La modélisation complète est alors obtenue de façon classique, en effectuant le bilan des forces F_Σ et des moments Γ_Σ s'appliquant sur le robot.

4.1 Bilan des forces

Les forces F_Σ qui s'appliquent sur le robot dans l'équation (III.10) peuvent être décomposées en deux catégories :

- les forces aérodynamiques,
- les forces générées par les hélices,

4.1.1 Les forces aérodynamiques

Il s'agit de forces non conservatives directement liées aux frottements visqueux de l'air sur le fuselage du robot. On suppose que ces forces sont dépendantes de la vitesse de translation du robot et d'un facteur de forme que l'on suppose proportionnel à l'aire de la partie du fuselage qui est orthogonale à la direction du mouvement. On peut donc écrire ces forces en fonction des vitesses de translation ${}_{\mathcal{B}}V_i$ dans le référentiel \mathcal{B} du robot, étant donné que, dans ce référentiel, les aires sont constantes :

$$F_v = \begin{pmatrix} F_{vx} \\ F_{vy} \\ F_{vz} \end{pmatrix} = \begin{pmatrix} -\frac{1}{2}\rho C_x A_{fx} {}_{\mathcal{B}}V_x |{}_{\mathcal{B}}V_x| \\ -\frac{1}{2}\rho C_y A_{fy} {}_{\mathcal{B}}V_y |{}_{\mathcal{B}}V_y| \\ -\frac{1}{2}\rho C_z A_{fz} {}_{\mathcal{B}}V_z |{}_{\mathcal{B}}V_z| \end{pmatrix} = \begin{pmatrix} -k_{vx} {}_{\mathcal{B}}V_x |{}_{\mathcal{B}}V_x| \\ -k_{vy} {}_{\mathcal{B}}V_y |{}_{\mathcal{B}}V_y| \\ -k_{vz} {}_{\mathcal{B}}V_z |{}_{\mathcal{B}}V_z| \end{pmatrix} \quad (\text{III.14})$$

Pour se déplacer le robot ajuste son attitude ce qui a pour effet de modifier son facteur de forme, où ρ est la densité de l'air, A_{fi} est l'aire du fuselage (projection sur le plan orthogonal¹ à l'axe i), $C_{x,y,z}$ sont des coefficients de frottements visqueux que l'on suppose constants.

4.1.2 Les forces générées par les hélices

Lorsqu'une hélice est mise en rotation, elle génère une force perpendiculaire à son plan de rotation (la poussée) ainsi qu'un couple (le couple de trainée). Ces deux éléments sont liés à la vitesse de rotation de l'hélice ainsi qu'à des coefficients aérodynamiques :

$$T_i = \underbrace{C_T \cdot \rho \cdot A_{ri} \cdot r_i^2}_{c_T} \cdot \omega_i^2 \quad (\text{III.15})$$

$$Q_i = \underbrace{C_Q \cdot \rho \cdot A_{ri} \cdot r_i^3}_{c_Q} \cdot \omega_i^2 \quad (\text{III.16})$$

Où C_T est un coefficient de poussée (qui dépend de la géométrie de l'hélice) et C_Q est le coefficient de couple, ρ est la densité de l'air, A_{ri} est l'aire balayée par le rotor i , r_i est le rayon de l'hélice et ω_i est la vitesse de rotation de l'hélice i . Dans le cas où la densité de l'air ne change pas (l'altitude reste sensiblement la même durant tout le vol), on peut déduire un unique coefficient de poussée noté c_T et un unique coefficient de couple noté c_Q .

Dans le cas d'un multirotor, la poussée totale générée est donnée par la somme des contributions de chaque hélice (dans notre cas 4 hélices) :

$$T_{\Sigma} = \left(\sum_{i=1}^4 T_i \right) \vec{b}_3 = \left(\sum_{i=1}^4 c_T \omega_i^2 \right) \vec{b}_3 \quad (\text{III.17})$$

1. Par exemple A_{fx} est la projection du fuselage (l'ensemble de ses constituants) sur le plan (yOz) .

L'effet de "flapping" : Dans l'équation (III.17), la poussée est supposée orientée selon l'axe Z . Or en réalité, la direction de la poussée est affectée par un effet connu sous le nom de flapping. Cet effet est bien connu pour les hélicoptères et a été largement décrit et documenté ([Johnson, 1994], [Prouty, 2003]). Plus récemment, [Huang *et al.*, 2009] ont montré que ce flapping avait une réelle influence sur le contrôle d'attitude des quadrirotors, d'où l'intérêt de le représenter dans le modèle. Le flapping consiste en une inclinaison du plan de rotation de l'hélice sous l'effet d'un vent apparent. Lorsqu'un hélicoptère avance, un vent apparait dans le sens opposé du mouvement créant ainsi une portance plus élevée sur un côté selon le sens de rotation de l'hélice. Cette différence de portance entraîne alors une inclinaison selon l'axe longitudinal, qui couplé à l'effet gyroscopique et aux vitesses de rotations de l'hélicoptère, implique une inclinaison selon l'axe latéral. Ainsi, on peut décrire la poussée comme étant inclinée selon deux directions. Cela peut s'écrire en appliquant une matrice de rotation au vecteur de poussée :

$$F_{T_i} = \begin{pmatrix} T_{xi} \\ T_{yi} \\ T_{zi} \end{pmatrix} = R_{flap_i} \cdot T_i \cdot \vec{b}_3 \quad (\text{III.18})$$

Où T_i est la poussée du rotor i orientée selon l'axe idéal $\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$ dans $\{B\}$ sans considérer le flapping comme décrit dans l'équation (III.17). R_{flap_i} est la matrice de rotation associée aux angles de flapping du rotor i . Cette matrice s'écrit sous la forme :

$$\begin{aligned} R_{flap} &= R_{x_{flap}} \cdot R_{y_{flap}} \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & c(b_{1s}) & -s(b_{1s}) \\ 0 & s(b_{1s}) & c(b_{1s}) \end{pmatrix} \cdot \begin{pmatrix} c(a_{1s}) & 0 & s(a_{1s}) \\ 0 & 1 & 0 \\ -s(a_{1s}) & 0 & c(a_{1s}) \end{pmatrix} \\ R_{flap} &= \begin{pmatrix} c(a_{1s}) & 0 & s(a_{1s}) \\ s(a_{1s})s(b_{1s}) & c(b_{1s}) & -c(a_{1s})s(b_{1s}) \\ -s(a_{1s})c(b_{1s}) & s(b_{1s}) & c(a_{1s})c(b_{1s}) \end{pmatrix} \end{aligned} \quad (\text{III.19})$$

Cet angle de flapping est spécifique à chaque rotor, car il dépend du sens et de la vitesse de rotation. Il est principalement produit par les taux de rotations du roulis et du tangage (p et q respectivement) et de la translation du rotor par rapport au vent apparent wind, c'est à dire : \vec{b}_1 ou \vec{b}_2 . D'après [Prouty, 2003], ces deux effets de flapping peuvent être approximés par les équations (III.20) et (III.21) pour une pale d'hélicoptère :

$$a_{1s} = \underbrace{\frac{8}{3}\theta_0\mu + 2\theta_1\mu - c_T}_{a_{1\mu}} + \underbrace{\frac{-16}{\gamma} \frac{q}{\omega_r} + \frac{p}{\omega_r}}_{a_{1\omega}} \quad (\text{III.20})$$

$$b_{1s} = \underbrace{\frac{4}{3} \frac{\mu^2}{1 + \frac{\mu^2}{2}} \left(\frac{2 \frac{c_T}{\sigma} \mu \gamma}{3a} + \mu \right)}_{b_{1\mu}} + \underbrace{\frac{-\frac{16}{\gamma} \frac{p}{\omega_r} - \frac{q}{\omega_r}}{1 + \frac{\mu^2}{2}}}_{b_{1\omega}} \quad (\text{III.21})$$

Où $a_{1\mu}$ et $b_{1\mu}$ sont respectivement les angles de flapping longitudinaux et latéraux dûs aux vitesses horizontales. $a_{1\omega}$ et $b_{1\omega}$ sont les angles de flapping longitudinaux et latéraux dûs aux vitesses de rotations p et q et a_{1s} et b_{1s} sont les angles de flapping finaux.

Le paramètre θ_0 est l'angle d'attaque de l'hélice et θ_1 son angle de twist² (ou angle de torsion supposé ici linéaire). a est une constante aérodynamique (égale ici à 6.0 d'après [Prouty, 2003]), μ est le ratio d'avance (défini comme étant le rapport entre la vitesse horizontale et la vitesse en bout de pale), σ est la solidité du disque rotor (le rapport de l'aire du rotor sur l'aire couverte lors de sa rotation) et γ est le nombre de Lock donné par l'équation (III.25). μ est donc défini par :

$$\mu = \frac{\|V_h\|}{\omega_r \cdot r} = \frac{\sqrt{{}^{\mathcal{B}}V_x^2 + {}^{\mathcal{B}}V_y^2}}{\omega_r \cdot r} \quad (\text{III.22})$$

Où ω_r est la vitesse de rotation du rotor, r est le rayon de l'hélice, V_h est le vecteur vitesse de translation horizontale dans le repère du robot, décrite par ${}^{\mathcal{B}}V_x$ et ${}^{\mathcal{B}}V_y$, qui sont les vitesses d'air "horizontales" apparentes dans le repère \mathcal{B} du robot. C'est à dire, que si on considère un vent nul, il s'agit des vitesses de translation orientées selon \vec{b}_1 et \vec{b}_2 .

Remarque III.2: Le vecteur V_h est la projection sur le plan horizontal du vecteur de vitesse ${}^{\mathcal{B}}v = R^T \cdot v$. C'est à dire que :

$$V_h = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \cdot R^T \cdot v \quad (\text{III.23})$$

Le rapport de solidité du disque rotor est donné par :

$$\sigma = \frac{n_{pale} \cdot c}{\pi \cdot r} = \frac{2 \cdot c}{\pi \cdot r} \quad (\text{III.24})$$

Où c est la corde de l'hélice (sa largeur) et n_{pale} le nombre de pales (ici égal à 2). Et le nombre de Lock est défini comme :

$$\gamma = \frac{\rho \cdot a \cdot c \cdot r^4}{I_b} \quad (\text{III.25})$$

2. Plus on se dirige vers l'extrémité de la pale, et plus la vitesse air est élevée (du fait du rayon de plus en plus élevé). La vitesse air augmentant, il en est de même pour la portance, ce qui a tendance à déformer la pale et à répartir les efforts en bout de pale. Pour éviter cela, les pales sont en général vrillées, de façon à diminuer l'angle d'attaque au fur et à mesure que l'on se rapproche de l'extrémité de la pale. On parle d'un angle de "twist" le plus souvent linéaire (qui est une bonne approximation de l'angle de twist idéal).

Où I_b est l'inertie de l'hélice.

Remarque III.3: Ici, les hélices utilisées n'ont pas exactement le même profil que celui d'une hélice d'hélicoptère. En effet, les hélices utilisées ne présentent pas d'angle de twist, qui est ici remplacé par une corde se réduisant vers l'extrémité de l'hélice. J'ai donc recherché l'angle de twist θ_1 équivalent et pris la valeur moyenne de la corde de l'hélice pour c .

L'équation (III.18) donne donc l'expression de la poussée d'un rotor dans le référentiel du robot. On peut donc en déduire la poussée totale dans le référentiel du robot, en sommant la contribution de chaque rotor, et en déduire une force totale dans le référentiel inertiel :

$$F_T = R.F_T = R. \sum_{i=1}^4 \left(R_{flap_i}.T_i.\vec{b}_3 \right) \quad (\text{III.26})$$

Avec R la matrice de rotation décrivant l'attitude du robot.

Bilan des forces Pour conclure, la force totale s'exerçant sur le robot est la contribution des frottements visqueux et de la poussée des hélices influencée par le flapping. On peut donc réécrire l'équation (III.10) sous la forme :

$$\begin{aligned} m\dot{v} &= -mge_3 + F_T + F_v \\ m\dot{v} &= -mge_3 + R. \left(\sum_{i=1}^4 R_{flap_i}.c_T.\omega_{ri}^2.\vec{b}_3 \right) - K_v v |v| \end{aligned} \quad (\text{III.27})$$

4.2 Bilan des moments

On résume ici les principaux moments (ou couples) s'exerçant sur le quadrirotor et influençant ses rotations.

4.2.1 Effet gyroscopique dû aux hélices

[Hamel *et al.*, 2002] ont proposé d'inclure l'effet gyroscopique des hélices dans le modèle du quadrirotor :

$$\begin{aligned} \Gamma_{pge} &= \sum_{i=1}^4 I_r \left(\Omega \times \vec{b}_3 \right) \omega_{ri} \\ \Gamma_{pge} &= I_r \left(\Omega \times \vec{b}_3 \right) \sum_{i=1}^4 \omega_{ri} \end{aligned} \quad (\text{III.28})$$

Où I_r est la matrice d'inertie d'un rotor. Il est important de noter que $\omega_{r1,3}$ sont négatives car les hélices tournent dans le sens des aiguilles d'une montre, et $\omega_{r2,4}$ sont positives car elles tournent dans le

sens inverse des aiguilles d'une montre. Ce qui veut dire que l'on peut réécrire ce couple sous la forme :

$$\Gamma_{pge} = I_r \left(\Omega \times \vec{b}_3 \right) \sum_{i=1}^4 (-1)^i |\omega_{ri}|$$

4.2.2 Contre couple d'inertie

Le contre couple d'inertie Γ_I est le couple généré par l'accélération résiduelle de rotation de chacune des hélices. Cette accélération est déterminée à partir de la vitesse de rotation résiduelle qui est notée ω_{res} et qui se définit comme :

$$\omega_{res} = \sum_{i=1}^4 \omega_{ri} = \sum_{i=1}^4 (-1)^i |\omega_{ri}| = -\omega_{r1} + \omega_{r2} - \omega_{r3} + \omega_{r4} \quad (\text{III.29})$$

Ensuite on peut réécrire Γ_I :

$$\Gamma_I = I_r \dot{\omega}_{res} \vec{b}_3 \quad (\text{III.30})$$

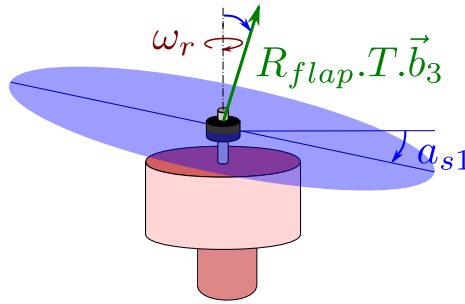


FIGURE III.13 – Axe de rotation du moteur et plan de rotation de l'hélice dus au flapping. L'axe de rotation du moteur reste inchangé, alors que le flapping induit un changement de plan de battement de l'hélice (ici, par simplicité, seul le flapping selon l'axe longitudinal est représenté). Le contre couple d'inertie est donc bien orienté selon l'axe du moteur.

Il est intéressant de noter ici que le couple est dirigé selon \vec{b}_3 et n'est pas influencé par les angles de flapping (comme la poussée l'était précédemment). Cela s'explique car le couple généré ici, est issu de l'inertie des rotors et dépend donc de l'axe de rotation des moteurs. Or les angles de flapping sont dus à une déformation de l'hélice (comme le montre la figure III.13) et non à une inclinaison de l'axe moteur. Le contre couple d'inertie est donc bien dirigé uniquement selon l'axe \vec{b}_3 .

4.2.3 Couple de lacet

L'équation (III.16) décrit le couple résultant de la traînée d'une hélice. Une hélice n'étant pas parfaite, elle génère en plus de la poussée, une traînée. Cette traînée, entraîne par réaction un contre-couple, opposé au sens de rotation de l'hélice. Dans le cas d'un hélicoptère classique, qui est équipé

d'une seule hélice, il est donc nécessaire d'ajouter un rotor de queue (également appelé anti-couple), afin de contrer cet effet et pouvoir contrôler l'angle de lacet. Pour un quadrirotor, ce sont également les couples liés à la trainée, qui vont permettre de générer les mouvements de lacet mais d'une façon différente. En effet, deux hélices tournent dans un sens et deux autres dans l'autre, ce qui permet d'ajuster le couple de lacet résiduel en ajustant la vitesse de rotation des hélices. Ce couple de lacet résiduel est obtenu en sommant les contributions de chacune des hélices :

$$\Gamma_D = \sum_{i=1}^4 (-1)^{i+1} c_Q \omega_i^2 \vec{b}_3 \quad (\text{III.31})$$

Il est important de noter, que pour les mêmes raisons que celles énoncées précédemment (voir section 4.2.2), ce couple n'est pas influencé par les angles de flapping et est donc bien orienté le long de l'axe de rotation des moteurs, c'est à dire \vec{b}_3 .

4.2.4 Couples induits par la poussée des hélices

Les hélices sont disposées en bout d'un bras de levier, par rapport au centre de gravité, ce qui a pour effet de générer des couples. Le couple généré par chacune des hélices dépend donc de la position de l'hélice dans le repère du robot. On peut donc écrire :

$$\Gamma_T = \sum_{i=1}^4 (F_{T_i} \times D_i) = \sum_{i=1}^4 \left[\left(R_{flap_i} \cdot (c_T \cdot \omega_{r_i}^2 \vec{b}_3) \right) \times D_i \right] \quad (\text{III.32})$$

Où D_i correspond à la ligne i de la matrice D représentant la position des hélices dans le référentiel du robot $\{\mathcal{B}\}$. Ici pour un quadrirotor en configuration X_4 , on obtient :

$$P = \begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{pmatrix} = \begin{pmatrix} l' & -l' & h \\ l' & l' & h \\ -l' & l' & h \\ -l' & -l' & h \end{pmatrix} \quad (\text{III.33})$$

Où $l' = \cos(45^\circ) \cdot l = \frac{\sqrt{2} \cdot l}{2}$ avec l la longueur d'un bras et h la hauteur séparant une hélice du centre de gravité selon l'axe \vec{b}_3 (voir III.12).

On remarque donc, dans le cas où il n'y aurait pas de flapping, que les couples ne se répartissent que sur le roulis et le tangage. Si on ajoute la contribution du couple de lacet découlant de la trainée (voir 4.2.3), on retombe sur la matrice M bien connue, dite matrice de mixage ([Mahony *et al.*, 2012]) qui est ici exprimée pour une configuration X_4 :

$$\begin{aligned}
\Gamma_T + \Gamma_D &= \sum_{i=1}^4 \left[\left(c_T \cdot \omega_{ri}^2 \vec{b}_3 \right) \times P_i \right] + \sum_{i=1}^4 (-1)^{i+1} c_Q \cdot \omega_{ri}^2 \vec{b}_3 \\
&= \begin{pmatrix} -c_T \cdot l' & c_T \cdot l' & c_T \cdot l' & -c_T \cdot l' \\ -c_T \cdot l' & -c_T \cdot l' & c_T \cdot l' & c_T \cdot l' \\ c_Q & -c_Q & c_Q & -c_Q \end{pmatrix} \begin{pmatrix} \omega_{r1}^2 \\ \omega_{r2}^2 \\ \omega_{r3}^2 \\ \omega_{r4}^2 \end{pmatrix} \tag{III.34}
\end{aligned}$$

Dans le cas où il n'y a pas de flapping, la dynamique de lacet est donc entièrement découplée des dynamiques de roulis et tangage. De plus, on remarque que la hauteur h n'a aucune influence sur le comportement du robot.

Effet du Flapping : Si l'on considère maintenant les angles de flapping, une conséquence directe de l'équation (III.18) est l'apparition de couples parasites sur l'axe de lacet. En effet, en reprenant (III.18), (III.34) devient :

$$\begin{aligned}
\Gamma_T + \Gamma_D &= \sum_{i=1}^4 \left[\left(R_{flap_i} c_T \cdot \omega_{ri}^2 \vec{b}_3 \right) \times P_i \right] + \sum_{i=1}^4 (-1)^{i+1} c_Q \cdot \omega_{ri}^2 \vec{b}_3 \\
&= \sum_{i=1}^4 \left[\begin{pmatrix} T_{xi} \\ T_{yi} \\ T_{zi} \end{pmatrix} \times D_i \right] + \sum_{i=1}^4 (-1)^{i+1} c_Q \omega_i^2 \vec{b}_3 \\
\Leftrightarrow \Gamma_T + \Gamma_D &= \begin{pmatrix} l' \cdot (T_{z1} - T_{z2} + T_{z3} + T_{z4}) + h \cdot (T_{y1} + T_{y2} + T_{y3} + T_{y4}) \\ l' \cdot (T_{z1} + T_{z2} + T_{z3} - T_{z4}) - h \cdot (T_{x1} + T_{x2} + T_{x3} + T_{x4}) \\ l' \cdot (T_{x2} - T_{x1} - T_{z3} - T_{z4}) - h \cdot (T_{y1} + T_{y2} + T_{y3} - T_{x4}) \end{pmatrix} \\
&\quad + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ c_Q & -c_Q & c_Q & -c_Q \end{pmatrix} \begin{pmatrix} \omega_{r1}^2 \\ \omega_{r2}^2 \\ \omega_{r3}^2 \\ \omega_{r4}^2 \end{pmatrix} \tag{III.35}
\end{aligned}$$

On voit alors que dans ce cas ci, les angles de flapping sont responsables de couplages, étant donné que le couple de lacet ne dépend plus uniquement de la traînée, mais également de la poussée.

Bilan des moments : Pour conclure, les couples totaux sont obtenus en réalisant la somme des contributions des différents couples énoncés précédemment ($\Gamma_\Sigma = \Gamma_T + \Gamma_D + \Gamma_{pge}$), et l'équation (III.12) peut être réécrite comme :

$$\begin{aligned}
I\dot{\Omega} &= \Gamma_{bge} + \Gamma_T + \Gamma_D + \Gamma_I + \Gamma_{pge} \\
I\dot{\Omega} &= \underbrace{-\Omega \times I\Omega}_{\Gamma_{bge}} + \underbrace{\sum_{i=1}^4 \left[\left(R_{flap_i} \cdot (c_T \cdot \omega_{ri}^2 \vec{b}_3) \right) \times P_i \right]}_{\Gamma_T} \\
&\quad + \underbrace{\sum_{i=1}^4 (-1)^{i+1} c_Q \cdot \omega_{ri}^2 \vec{b}_3}_{\Gamma_D} + \underbrace{I_r \dot{\omega}_{res} \vec{b}_3}_{\Gamma_I} \\
&\quad + \underbrace{\sum_{i=1}^4 I_r \left(\Omega \times \vec{b}_3 \right) \omega_{ri}}_{\Gamma_{pge}}
\end{aligned} \tag{III.36}$$

4.3 Modèle complet

4.3.1 Modèle pour la simulation

Il est maintenant possible de réécrire le modèle complet du quadrirotor à partir du modèle de corps rigide libre (III.9)-(III.12) en utilisant le bilan des forces (III.27) et des moments (III.36) que nous avons établi :

$$\left\{ \begin{array}{l}
\dot{\xi} = v \\
m\dot{v} = -mge\vec{3} + R \cdot \left(\sum_{i=1}^4 R_{flap_i} c_T \cdot \omega_{ri}^2 \right) \vec{b}_3 - K_v v |v| \\
\dot{R} = R\Omega_{\times} \\
I\dot{\Omega} = -\Omega \times I\Omega + \sum_{i=1}^4 \left[\left(R_{flap_i} \cdot (c_T \cdot \omega_{ri}^2 \vec{b}_3) \right) \times P_i \right] + \sum_{i=1}^4 (-1)^{i+1} c_Q \cdot \omega_{ri}^2 \vec{b}_3 \\
\quad + I_r \dot{\omega}_{res} \vec{b}_3 + \sum_{i=1}^4 I_r \left(\Omega \times \vec{b}_3 \right) \omega_{ri}
\end{array} \right. \tag{III.37}$$

Ce modèle a été utilisé dans tous mes travaux de simulation qui ont permis de régler les différents contrôleurs, estimateurs, et de valider les différentes stratégies de commandes et algorithmes. Les résultats de simulation ne seront pas présentés ici ; seuls les résultats sur le robot réel seront abordés à la fin de ce chapitre dans la section 7 et dans le chapitre V.

4.3.2 Modèle pour le contrôle

Le modèle précédent est nécessaire dans le cadre de la simulation, afin de représenter au mieux les non linéarités présentes sur le système. Cela permet de soumettre les contrôleurs et estimateurs proposés, à un monde un peu moins parfait, que celui décrit par le modèle simplifié, utilisé pour

concevoir ces contrôleur et estimateurs.

En effet, afin de résoudre la complexité du contrôle, on utilise souvent un modèle simplifié du système, permettant d'appliquer les outils de l'automatique. On simplifie donc ici le modèle précédent, grâce au hypothèses et constats suivants :

- Dans le simulateur, nous considérons des angles de flapping différents pour chaque rotor. Or en étudiant de plus près les équations (III.20) et (III.21), on observe que les termes de flapping sont décomposés en une partie $a_{1\mu}$ et $b_{1\mu}$ ne dépendant que des paramètres intrinsèques de l'hélice et du ratio d'avance μ . L'autre partie $a_{1\omega}$ et $b_{1\omega}$ dépend, quant à elle, également de la vitesse de rotation du rotor. Or, d'après les simulations, les termes $a_{1\omega}$ et $b_{1\omega}$ apparaissent comme étant négligeables dans des conditions de vol normal (vitesses angulaires du robot faibles.)

[Mahony *et al.*, 2012] proposent donc de remplacer la matrice de rotation, et de représenter le flapping comme la somme de la poussée idéale et d'un terme de "trainée" dépendant de la vitesse :

$$F_{T_i} = T_i \cdot (\vec{b}_3 - D_i \cdot V_h) \quad (\text{III.38})$$

Où D_i est une matrice décrivant le flapping et les frottement visqueux liés au déplacement : $D_i = A_{flap_i} + \text{diag}(K_{vx}, K_{vy}, 0)$. La matrice de flapping étant alors définie par :

$$A_{flap_i} = \frac{1}{\omega_{r_i} \cdot r} \begin{pmatrix} A_{1s} & -B_{1s} & 0 \\ B_{1s} & A_{1s} & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (\text{III.39})$$

Avec $A_{1s} = \frac{8\theta_0}{3} + 2\theta_1$ ³ et $B_{1s} = \frac{4}{3} \left(\frac{2 \cdot c_T \cdot \gamma}{3 \cdot a \cdot \sigma} + 1 \right)$ obtenus depuis les équations (III.20) et (III.21).

On peut ensuite n'utiliser qu'une unique matrice de flapping moyenne \bar{A}_{flap} en utilisant la moyenne des vitesses de rotations des rotors $E[\omega_r]$ au lieu de ω_{r_i} , ce qui nous permet d'obtenir :

$$F_T = \sum_{i=1}^4 T_i (\vec{b}_3 - \bar{D} \cdot V_h) \quad (\text{III.40})$$

Avec $\bar{D} = \bar{A}_{flap} + \text{diag}(k_{vx}, k_{vy}, 0)$

- Les termes de frottements visqueux donnés dans (III.14) sont linéarisés autour de 0, et ajoutés aux termes de flapping décrits ci-dessus, de façon à obtenir un terme unique, agissant comme des frottements selon la vitesse v .
- Par souci de simplicité, le couple généré par les angles de flapping sera négligé. Même si dans les faits, il entraîne une perturbation non négligeable sur la commande d'attitude lors de translations latérales (voir [Huang *et al.*, 2009]). On utilisera donc l'approximation décrite par l'équation

3. Ici le coefficient c_T est négligé devant θ_0 et θ_1

(III.34) pour les couples de commande.

- Pour le modèle de contrôle, le contre couple Γ_I dû à l'accélération angulaire des hélices sera également négligé,
- Dans les faits, le couple lié à l'effet gyroscopique des hélices Γ_{pge} peut être négligé aussi.

Suite aux hypothèses et constats précédents, on peut alors décrire à partir de III.37 un modèle simplifié qui sera utilisé pour le contrôle.

$$\begin{cases} \dot{\xi} = v \\ m\dot{v} = -mg\vec{e}_3 + R \cdot \left(\sum_{i=1}^4 c_T \cdot \omega_{r_i}^2 \right) \vec{b}_3 - K_v \cdot v \\ \dot{R} = R \cdot \Omega_\times \\ I\dot{\Omega} = -\Omega \times I\Omega + \sum_{i=1}^4 \left[(c_T \cdot \omega_{r_i}^2 \vec{b}_3) \times P_i \right] + \sum_{i=1}^4 (-1)^{i+1} c_Q \cdot \omega_{r_i}^2 \vec{b}_3 \end{cases} \Leftrightarrow \begin{cases} \dot{\xi} = v \\ m\dot{v} = -mg\vec{e}_3 + R \cdot u_T \cdot \vec{b}_3 - K_v \cdot v \\ \dot{R} = R \cdot \Omega_\times \\ I\dot{\Omega} = -\Omega \times I\Omega + u_\Gamma \end{cases} \quad (\text{III.41})$$

Où $K_v = \bar{A}_{flap} + \text{diag}(k_{vx}, k_{vy}, k_{vz})$. L'entrée de commande est donnée par le vecteur $u = (u_T \quad u_\Gamma)^T$ avec u_T un scalaire et u_Γ un vecteur colonne de dimension 3. Ces deux entrées de commandes sont reliées aux vitesses de rotations des rotors par la relation suivante :

$$\begin{pmatrix} u_T \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} u_T \\ u_{\Gamma_\phi} \\ u_{\Gamma_\theta} \\ u_{\Gamma_\psi} \end{pmatrix} = \begin{pmatrix} c_T & c_T & c_T & c_T \\ -c_T \cdot l' & c_T \cdot l' & c_T \cdot l' & -c_T \cdot l' \\ -c_T \cdot l' & -c_T \cdot l' & c_T \cdot l' & c_T \cdot l' \\ c_Q & -c_Q & c_Q & -c_Q \end{pmatrix} \begin{pmatrix} \omega_{r_1}^2 \\ \omega_{r_2}^2 \\ \omega_{r_3}^2 \\ \omega_{r_4}^2 \end{pmatrix} \quad (\text{III.42})$$

Où c_T est le coefficient de poussée, c_Q le coefficient de couple lié à la trainée et l' le bras de levier de la configuration X_4 .

4.4 Identification des paramètres

4.4.1 Paramètres inertiels

Les paramètres inertiels peuvent être obtenus assez simplement. La masse totale du robot est obtenue par une simple pesée. Les termes de la matrice d'inertie I sont obtenus par modélisation fine du robot à l'aide d'un logiciel de CAO (Conception Assistée par Ordinateur). Pour cela, chaque

composant a été soigneusement mesuré et pesé individuellement (moteur, hélice, ESC, frame, cartes électroniques, pieds, divers supports, etc.) Ensuite ces différents composants ont été créés sous le logiciel de CAO *Solid Edge* et des matériaux fictifs ont été créés, afin de leur attribuer une masse volumique permettant de respecter la masse totale de chaque composant (qui est en règle générale composé de plusieurs matériaux). Cette conception a donc permis de déterminer de façon précise la matrice d'inertie ainsi que la position du centre de gravité. Déterminer la position du centre de gravité était nécessaire notamment pour trouver la distance h décrite par la matrice P de l'équation (III.33).

Cette étape de conception par ordinateur a donc été une phase particulièrement importante, qui a été réalisée grâce à l'aide précieuse de Julien Dipéri, micro-mécanicien de l'équipe biorobotique de l'ISM. En plus d'avoir permis d'identifier des paramètres, elle a également permis de choisir la meilleure disposition possible pour les divers composants, et de définir la forme idéale de certains (choix de la forme de la batterie, des cartes électroniques, etc.) Nous avons notamment cherché, pendant cette phase, à disposer les divers éléments de façon à obtenir un centre de gravité situé en dessous du centre de poussée (obtenir une distance $h > 0$). En effet, [Bristeau *et al.*, 2009] ont montré que dans le cas de mouvements de translation, les couples générés par le bras de levier h et le flapping ont un effet de stabilisation naturelle, dans le cas où le centre de gravité est positionné en dessous du centre de poussée (bien qu'ils aient un effet inverse lors de rejet de perturbations).

4.4.2 Paramètres aérodynamiques

Les coefficients des hélices : Les coefficients c_T et c_Q sont des coefficients liés aux hélices utilisées et peuvent être identifiés à l'aide de test de poussée statique. Ces coefficients c_T et c_Q permettent respectivement de lier la vitesse de rotation d'une hélice à la poussée et le couple de trainée générés. On considère alors que ces coefficients restent valables dans le cas où la vitesse de translation du rotor reste faible par rapport à la vitesse du flux d'air généré par le rotor. On considèrera cette hypothèse vérifiée, car les vitesses de translation verticale du robot sont considérées comme relativement faibles dans l'ensemble des travaux décrits ici. Ce qui permet de supposer que :

$$T = c_T \cdot \omega_r^2 \quad (\text{III.43})$$

$$D = c_Q \cdot \omega_r^2 \quad (\text{III.44})$$

Le principe est assez simple, et on obtient une bonne identification grâce à un banc de poussée, réalisé spécialement à cet effet. On place le rotor en bout d'un bras relié à une liaison pivot et dont un second bras, fixé orthogonalement au premier, vient prendre appui sur une balance. Un capteur à effet hall, placé sur le côté du rotor, permet de réaliser un tachymètre, comme présenté précédemment dans la section 2.5.2 (voir également la figure III.4). La figure III.14 montre donc le banc moteur utilisé, ainsi que le relevé de poussée et de trainée réalisés. La poussée ou la trainée peut être lue directement après

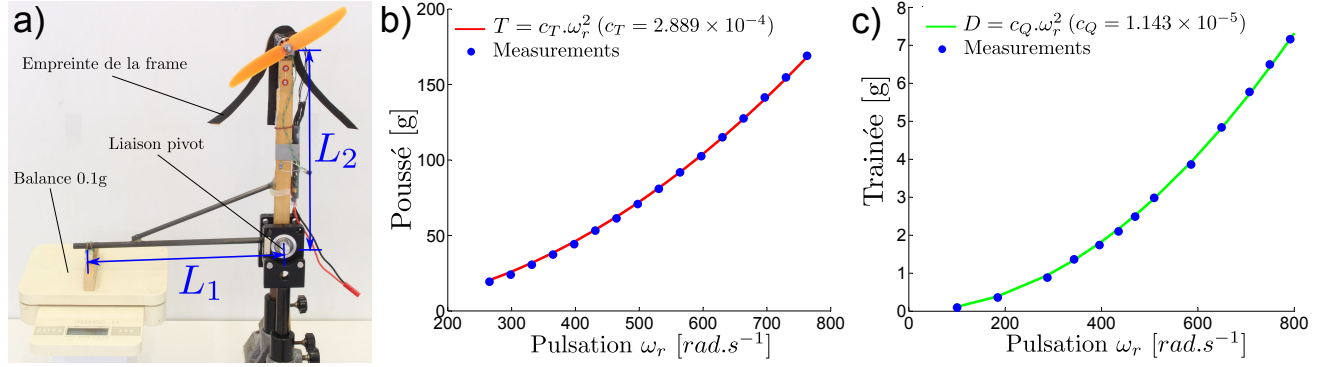


FIGURE III.14 – a) Le banc moteur en position pour identifier le coefficient c_Q (axe de rotation de l'hélice colinéaire à l'axe de la liaison pivot.) Pour identifier le coefficient c_T il suffit de repositionner le moteur en appliquant une rotation de $\pm 90^\circ$. Une "empreinte" de la frame est ajoutée en dessous du rotor de façon à être au plus proche des conditions réelles. b) En bleu les mesures de poussée en fonction de la pulsation du rotor, et en rouge la courbe théorique optimale au sens des moindres carrés approchant la poussée du rotor $T = c_T \cdot \omega_r^2$. c) En bleu les mesures de traînée en fonction de la pulsation du rotor, et en vert la courbe théorique optimale au sens des moindres carrés approchant la traînée du rotor $D = c_Q \cdot \omega_r^2$.

avoir appliqué le rapport des bras de levier : $\bar{T} = L_2/L_1 \cdot T_{lue}$ et $\bar{Q} = L_2/L_1 \cdot Q_{lue}$. Afin de s'approcher au mieux des conditions d'écoulement d'air présent sur le quadrirotor, on ajoute en dessous du rotor une empreinte semblable à la frame du quadrirotor. De même, on place le banc sur le bord d'une table (comprendre, loin de tout obstacle dans la direction d'écoulement de l'air) de façon à perturber le moins possible l'écoulement d'air.

Les coefficients de poussée c_T et de traînée c_Q sont ensuite obtenus en minimisant, au sens des moindres carrés, l'écart entre les échantillons et la loi attendue :

$$\min_{c_T \in \mathbb{R}^+} f_1(c_T) = \min_{c_T \in \mathbb{R}^+} \sum_{k=1}^N (\bar{T}_k - c_T \cdot \bar{\omega}_{rk}^2)^2 \quad (\text{III.45})$$

$$\min_{c_Q \in \mathbb{R}^+} f_2(c_Q) = \min_{c_Q \in \mathbb{R}^+} \sum_{k=1}^N (\bar{D}_k - c_Q \cdot \bar{\omega}_{rk}^2)^2 \quad (\text{III.46})$$

Le minimum au sens des moindres carrés est donc obtenu en résolvant $\frac{\partial f_1}{\partial c_T} = 0$ et $\frac{\partial f_2}{\partial c_Q} = 0$. Les valeurs

de c_T et c_Q minimisant respectivement f_1 et f_2 sont donc :

$$c_{T_{\text{Banc}}} = \frac{\sum_{k=1}^N \bar{\omega}_{rk} \cdot \bar{T}_k}{\sum_{k=1}^N \bar{\omega}_{rk}^4} \quad (\text{III.47})$$

$$c_{Q_{\text{Banc}}} = \frac{\sum_{k=1}^N \bar{\omega}_{rk} \cdot \bar{D}_k}{\sum_{k=1}^N \bar{\omega}_{rk}^4} \quad (\text{III.48})$$

On remarque donc que les lois de poussée et de traînée proportionnelle au carré de la vitesse sont une bonne représentation du phénomène physique, étant donnée que les points mesurés sont très proches des courbes théoriques identifiées (voir figures III.14b-c). Cependant, on observe une différence entre le coefficient de poussée identifié grâce au banc et celui obtenu en vol réel. En effet, lors d'un vol réel, on peut déterminer le coefficient de poussée en effectuant un vol stationnaire et en récupérant la vitesse de rotation des hélices mesurée lors de ce stationnaire. Connaissant la masse du robot, on peut remonter facilement au coefficient de poussée :

$$\sum_{i=1}^4 c_{T_{\text{Statio}}} \cdot E[\omega_{ri}]^2 = m \Leftrightarrow c_T \sum_{i=1}^4 E[\omega_{ri}]^2 = m \Leftrightarrow c_{T_{\text{Statio}}} = \frac{m}{c_{T_{\text{Statio}}} \sum_{i=1}^4 E[\omega_{ri}]^2} \quad (\text{III.49})$$

Où $E[\omega_{ri}]$ est la vitesse moyenne du rotor i sur une fenêtre temporelle de 30 secondes, et m est la masse exacte du robot lors du vol.

Lors de cette seconde identification du paramètre de poussée, on obtient un coefficient 10% plus faible que celui identifié avec le banc : $c_{T_{\text{Statio}}} = 0.9003 \cdot c_{T_{\text{Banc}}}$. Cela est assez surprenant, et peut sans doute s'expliquer par le fait que, sur le quadrirotor réel, des éléments supplémentaires sont présents en dessous du disque balayé par le rotor (des fils, bien qu'ils ne représentent qu'une aire très réduite) et des cartes électroniques sont proches de l'extrémité des rotors. On suppose donc que ces éléments perturbent un peu le flux d'air, réduisant ainsi l'efficacité des rotors. Enfin, une interaction entre les flux des différents rotors peut également expliquer cette différence.

Les termes de frottements et flapping : La matrice K_v , décrite dans 4.3.2, décrit à la fois les termes de flapping et de frottements visqueux. On a donc 5 termes à identifier :

$$K_v = \begin{pmatrix} K_{vx1} & K_{vx2} & 0 \\ K_{vy1} & K_{vy2} & 0 \\ 0 & 0 & K_{vz} \end{pmatrix} \quad (\text{III.50})$$

Pour les identifier, on propose de construire un filtre de Kalman étendu, permettant d'estimer ces paramètres qui seront ajoutés au vecteur d'états. On choisit donc le vecteur d'états :

$x = \left(X \ Y \ Z \ V_x \ V_y \ V_z \ K_{vx1} \ K_{vx2} \ K_{vy1} \ K_{vy2} \ K_{vz} \right)^T$ où l'on considèrera que les états K_{vi} liés aux frottements sont des constantes que l'on cherche à estimer. C'est à dire que l'on supposera (dans le modèle) que leur dérivée par rapport au temps est nulle.

Nous obtenons donc le modèle dynamique suivant pour notre système dont on cherche à estimer les termes de frottement :

$$\left\{ \begin{array}{l} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{V}_x \\ \dot{V}_y \\ \dot{V}_z \\ \dot{K}_{vx1} \\ \dot{K}_{vx2} \\ \dot{K}_{vy1} \\ \dot{K}_{vy2} \\ \dot{K}_{vz} \end{array} \right\} = \left(\begin{array}{c} V_x \\ V_y \\ V_z \\ \frac{1}{m} \left(\hat{T} \cdot c(\bar{\phi}) s(\bar{\theta}) c(\bar{\psi}) - K_{vx1} \cdot V_x - K_{vx2} \cdot V_y \right) \\ \frac{1}{m} \left(\hat{T} \cdot c(\bar{\phi}) s(\bar{\theta}) s(\bar{\psi}) - s(\bar{\phi}) c(\bar{\psi}) - K_{vy1} \cdot V_x - K_{vy2} \cdot V_y \right) \\ \frac{1}{m} \left(-m \cdot g + \hat{T} \cdot c(\bar{\phi}) c(\bar{\theta}) - K_{vz} \cdot V_z \right) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right) \quad (\text{III.51})$$

$$\bar{y} = \left(\bar{X} \ \bar{Y} \ \bar{Z} \ \bar{V} \ \bar{V} \ \bar{V} \right)^T$$

Où $\bar{\phi}$, $\bar{\theta}$ et $\bar{\psi}$ sont les angles d'Euler mesurés par le système Vicon et qui sont considérés ici comme les entrées du système. \hat{T} est la poussée estimée, que l'on peut déterminer à partir des consignes moteurs demandées (issus du contrôle de position et d'attitude, voir section 7) et de dynamique fixée par la boucle fermée sur les rotors (voir le chapitre V pour plus de détail). \bar{y} est le vecteur de mesure que l'on obtient également grâce au système Vicon. Une étude de la matrice d'observabilité du système (qui ne sera pas développée ici), nous montre que le système est bien observable du moment que les vitesses de translations restent différentes de zéro (des termes de la matrice d'observabilité dépendent des vitesses qui font chuter le rang si elles s'annulent).

On réalise donc un premier vol consistant en un décollage, puis à une trajectoire circulaire de rayon 50 cm à une altitude de 1.3 m. Lors de ce premier enregistrement, les gains des matrices de covariances sont réglés de façon à obtenir une convergence assez rapide vers les coefficients de frottements. L'EKF, fournit donc au bout de 150 secondes, une première estimation des coefficients. Puis on rejoue l'expérience, en simulation, en initialisant les états correspondant aux coefficients K_{vi} avec les valeurs qui viennent d'être trouvées. Pour cette simulation, les termes de pondérations liés au modèle des coefficients K_{vi} sont divisés par un facteur 1000, afin d'obtenir un temps de convergence plus lent, et donc une estimation

moins bruitée.

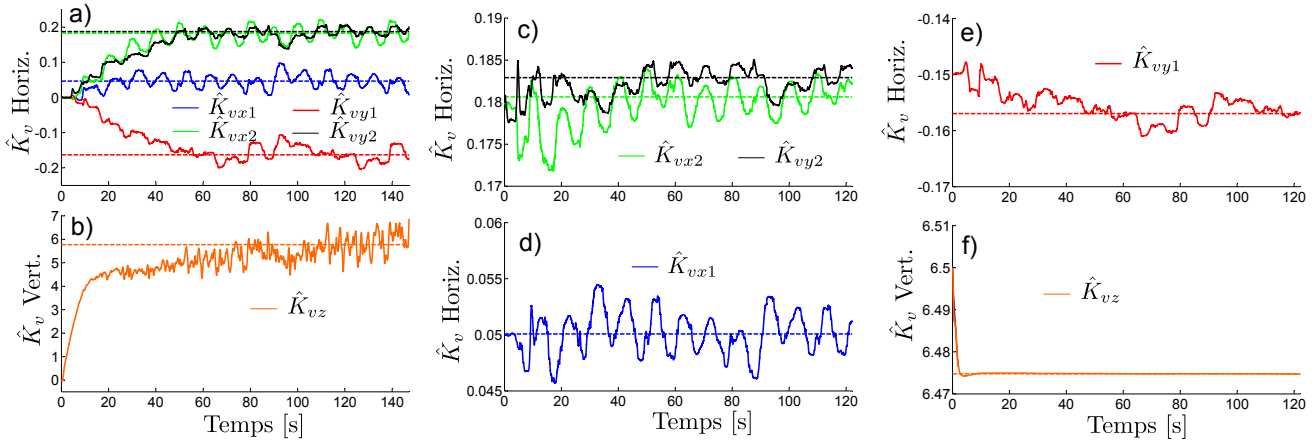


FIGURE III.15 – a) Estimations des coefficients de frottements visqueux selon les directions horizontale et verticale avec des dynamiques d’estimations rapides, afin d’obtenir une valeur d’initialisation. En trait continu, les valeurs estimées à chaque instant et en pointillée la valeur moyenne sur les 50 dernières secondes. b) Le coefficient de frottement estimé selon la direction verticale. c-f) Estimation plus fine des paramètres lors d’un rejeu en simulation avec une boucle d’estimation plus lente de la séquence donnée en a-b). En trait continu les valeurs estimées à chaque instant et en pointillés la valeur moyenne sur les 50 dernières secondes.

On voit donc sur la figure III.15a que les coefficients K_{vx2} et K_{vy2} sont très proches. On remarque également que K_{vy1} est négatif, ce qui est normal au vue du signe - présent devant $s(\phi)$ dans l’équation différentielle de V_y . On observe également sur les figures III.15b et f, que l’estimation du coefficient K_{vz} converge selon deux dynamiques distinctes. En effet, lors des 10 premières secondes, le robot est en phase d’ascension, apportant des informations "riches" à l’estimateur. Une fois l’altitude "de croisière" atteinte, la vitesse verticale est alors quasi nulle, ce qui rend le système quasi inobservable (sans vitesse il n’est pas possible d’estimer des termes de frottements). Enfin, on remarque que le coefficient K_{vz} est nettement supérieur aux autres coefficients (environ 30 fois supérieur). Cela est dû à la forme du robot selon cette direction. En effet, l’aire du robot projetée sur le plan horizontale est plus importante que dans les autres directions. Cela explique donc que les frottements soient plus importants car le coefficient de frottements dépendent du facteur de forme de l’objet.

Les figures III.15c-f montrent ensuite l’estimation des mêmes paramètres avec une dynamique d’estimation plus lente. L’initialisation est faite à partir des paramètres obtenus précédemment et on obtient au bout de 120 secondes, une estimation bien plus précise. Les coefficients finaux retenus sont obtenus en moyennant les dernières 50 secondes de cet essai.

Bilan des paramètres du modèle : On présente donc dans le tableau suivant l’ensemble des paramètres ayant été identifiés. A noté, il s’agit ici des paramètres obtenu pour le robot X4-MaG muni de son œil, tel qu’il est présenté dans le chapitre V.

Description	Paramètre	Valeur	Unité
masse	m	0.395	[kg]
Matrice d'inertie	I	$\begin{pmatrix} 1.027e^{-3} & -7e^{-6} & -2e^{-6} \\ -7e^{-6} & 1.628e^{-3} & 0 \\ -2e^{-6} & 0 & 1.142e^{-3} \end{pmatrix}$	[N · m]
Coefficient de poussée (banc)	$c_{T_{\text{Banc}}}$	2.889×10^{-7}	[kg]
Coefficient de poussée (stationnaire)	$c_{T_{\text{Statio}}}$	2.601×10^{-7}	[kg · rad ⁻² · s ²]
Coefficient de couple de trainée	c_Q	$1.143e^{-8}$	[kg · rad ⁻² · s ²]
Matrice de frottements et flapping	K_v	$\begin{pmatrix} 0.050 & 0.181 & 0 \\ -0.157 & 0.183 & 0 \\ 0 & 0 & 6.475 \end{pmatrix}$	[N · m ⁻² · s ²]

5 Protocoles de calibrations

5.1 Calibrations des Accéléromètres

Les accéléromètres utilisés sont ceux de la MPU6050 et sont de technologie MEMS et présentent souvent deux types de défauts : à savoir un biais et un facteur d'échelle pour chaque axe. Le biais peut être dû à deux phénomènes, un mauvais alignement des 3 accéléromètres qui ne forment pas un trièdre parfait et/ou un positionnement pas parfaitement "à plat" sur le robot.

D'un point de vue pratique, j'ai pu constaté la présence de ces deux biais. En effet, on relève les valeurs maximale et minimale mesurées par les accéléromètres de chaque axe, en alignant uns par uns les axes du robot avec la verticale. Puis on relève également les valeurs des accéléromètres lorsque le robot est placée parfaitement à l'horizontale (positionnée à l'horizontale grâce à un niveau à bulle.) On obtient donc le tableau 5.1 :

	MIN	MAX	Position 0°
X	-9.63	10.15	0.3
Y	-9.81	9.72	-0.02
Z	-9.3	10.7	-9.214

On remarque donc qu'il y a bien un facteur d'échelle ainsi qu'un biais sur chacun des axes. On écrit alors la sortie des accéléromètres, entachées de leurs défauts, de la façon suivante :

$$\ddot{\xi} + \vec{g} = R.T(\bar{a}_{acc} - b_{acc}) \quad (\text{III.52})$$

En supposant des accélérations latérales faibles devant g ($\ddot{\xi} \ll \|\vec{g}\|$), on obtient :

$$\begin{aligned}
 T.\vec{g} &= R.(T\bar{a}_{acc} - b_{acc}) \\
 \Leftrightarrow R^T.\vec{g} &= \begin{pmatrix} a & -1 \end{pmatrix} \begin{pmatrix} T \\ T.b \end{pmatrix} \\
 \Leftrightarrow \begin{pmatrix} 0 & 0 & -g \end{pmatrix} . R &= \begin{pmatrix} a^T & -1 \end{pmatrix} \begin{pmatrix} T^T \\ b^T.T^T \end{pmatrix}
 \end{aligned} \tag{III.53}$$

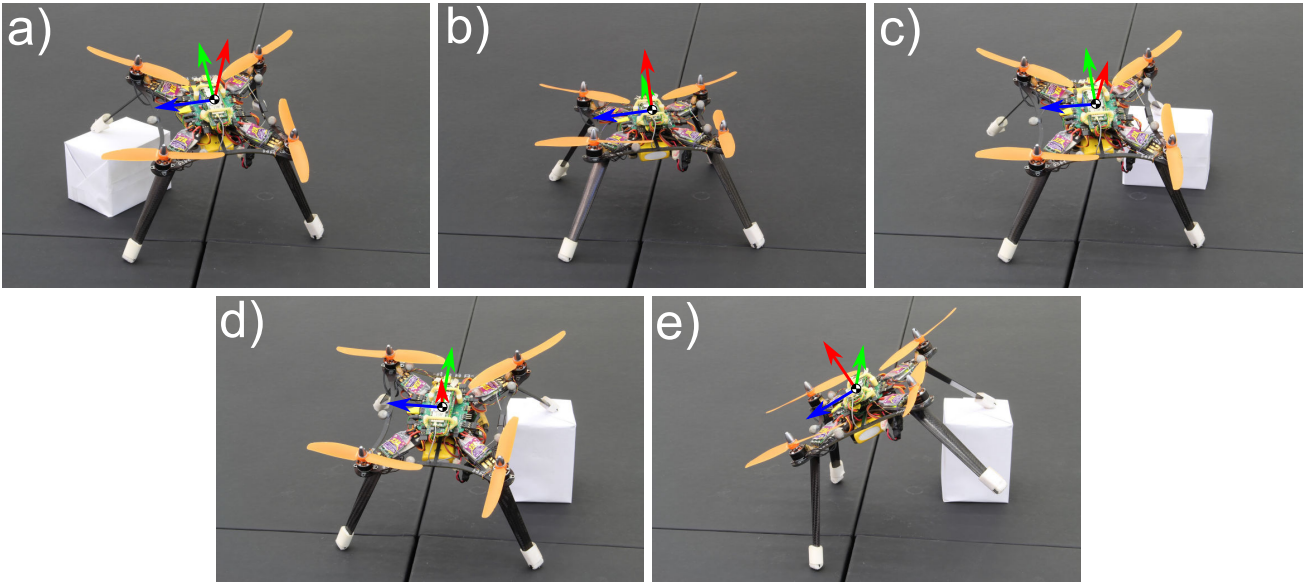


FIGURE III.16 – a-e) Photographie montrant le quadrirotor dans différentes positions angulaires lors de la phase de calibration des accéléromètres. L'orientation réelle du robot est également mesurée par le système Vicon, et n identifie ensuite par moindres carrés un facteur d'échelle s et un biais b pour chaque accéléromètre de la centrale inertielle (chaque axe).

Ces défauts peuvent être identifiés de façon précise, en relevant sur quelques secondes, les valeurs mesurées par les accéléromètres pour différentes orientations (le robot étant parfaitement immobile durant l'enregistrement), comme proposé par exemple dans [Spica *et al.*, 2013]. Ces orientations sont également mesurées par le système Vicon et on se sert de ces mesures, considérées comme vraies, pour calibrer les accéléromètres. On note \bar{R} la mesure de l'attitude (matrice de rotation) réalisée par le Vicon. Afin de garantir que $R = I_{3 \times 3}$ lorsque le robot est parfaitement à l'horizontale, on détermine une matrice de rotation de calibration \bar{R}_0 pour le Vicon. Cette matrice \bar{R}_0 est simplement la mesure d'attitude faite par le Vicon, quand le robot est parfaitement à l'horizontale (ce qui est vérifié à l'aide d'un niveau à bulle de précision). Ainsi, on applique donc cette calibration en redéfinissant pour la suite les mesures Vicon comme étant :

$$\bar{R} \leftarrow \bar{R}_0^T . \bar{R} \tag{III.54}$$

On réalise donc k enregistrements correspondant à k orientations du robot différentes et obtient donc n échantillons par enregistrements. On peut ensuite rassembler l'ensemble de ces données et construire un système matriciel :

$$\underbrace{\begin{pmatrix} g^T \cdot \bar{R}_{1,1} \\ \vdots \\ \bar{g}^T \cdot \bar{R}_{1,n} \\ \bar{g}^T \cdot \bar{R}_{2,1} \\ \vdots \\ \bar{g}^T \cdot \bar{R}_{2,n} \\ \vdots \\ \bar{g}^T \cdot \bar{R}_{k,n} \end{pmatrix}}_{G_r} = \underbrace{\begin{pmatrix} a_{1,1}^T & -1 \\ \vdots & \\ a_{1,n}^T & -1 \\ a_{2,1}^T & -1 \\ \vdots & \\ a_{2,n}^T & -1 \\ \vdots & \\ a_{k,n}^T & -1 \end{pmatrix}}_A \begin{pmatrix} T^T \\ b^T \cdot T^T \end{pmatrix} \quad (\text{III.55})$$

Où les matrices $\bar{R}_{i,j}$ correspondent à la $j^{\text{ième}}$ mesure d'orientation faite par le Vicon pour l'orientation i . On peut ensuite résoudre ce système afin de déterminer la matrice d'échelle T et le vecteur de biais b . La matrice A n'étant pas carrée, on utilise la pseudo-inverse notée A^+ ⁴. On a alors :

$$C = A^+ \cdot G_r \quad (\text{III.56})$$

Ce qui permet d'obtenir une estimation des biais b et facteur d'échelle réels T :

$$\begin{cases} T = C_{1 \rightarrow 3, 1 \rightarrow 3}^{+T} \\ b = C_{4, 1 \rightarrow 4}^+ \cdot T^{-1} \end{cases} \quad (\text{III.57})$$

On note donc \hat{b}_{acc} et \hat{T} ces estimations, que l'on utilise pour corriger systématiquement les mesures des accéléromètres et fournir une mesure corrigée notée \bar{a}'_{acc} :

$$\bar{a}'_{acc} = \hat{T}(\bar{a}_{acc} - \hat{b}_{acc}) \quad (\text{III.58})$$

Par exemple, pour le robot utilisé lors de ma thèse, on obtient :

4. Pour une matrice A de dimension $n \times m$ de rang p , la pseudo inverse est ici obtenue à partir de la décomposition SVD de A (Moore-Penrose). Soit $S_{n \times m}$, $V_{m \times m}$ et $U = n \times n$ la décomposition SVD de C telle que $A = U \cdot S \cdot V^T$. On a alors, $A^+ =$

$V \cdot S^{+T} \cdot U^T$ Où S^+ est elle même la pseudo-inverse de $S = \begin{pmatrix} \Lambda_{p \times p} & 0_{p \times m-p} \\ 0_{n-p \times p} & 0_{n-p \times m-p} \end{pmatrix}$ avec $\Lambda_{p \times p} = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_p \end{pmatrix}$ la

matrice des valeurs singulières de S . On a alors $S^+ = \begin{pmatrix} \Lambda_{p \times p}^{-1} & 0_{p \times m-p} \\ 0_{n-p \times p} & 0_{n-p \times m-p} \end{pmatrix}^T = \begin{pmatrix} \frac{1}{\lambda_i} & 0_{n-p \times p} \\ 0_{p \times m-p} & 0_{m-p \times n-p} \end{pmatrix}$.

$$\hat{T} = \begin{pmatrix} 0.9854 & -0.0293 & -0.0277 \\ 0.0956 & 1.0345 & 0.1221 \\ -0.0344 & -0.0122 & 0.8682 \end{pmatrix} \quad \hat{b} = \begin{pmatrix} 0.5233 \\ -1.2842 \\ 1.9094 \end{pmatrix}$$

5.2 Calibrations des Gyromètres

La calibration des gyromètres dépend fortement des conditions de l'essai et ces biais ne sont pas constants. Il sera donc nécessaire de les estimer en ligne de façon à s'adapter à leurs évolutions.

Cependant une première estimation de ces biais peut être réalisée simplement en moyennant les mesures obtenues sur quelques secondes lorsque le robot est immobile. Cette première valeur permettra simplement d'initialiser le filtre complémentaire décrit dans la suite (voir section 7.2).

5.3 Estimations en lignes des biais

Biais des accéléromètres Les biais et la matrice d'échelle des accéléromètres peut être déterminée de la façon décrite précédemment.

Biais des gyromètres Comme dit précédemment, les biais sont systématiquement estimés en ligne, grâce au filtre complémentaire décrit dans la section 7.2. Cette estimation est possible, en utilisant le vecteur de gravité mesuré par les accéléromètres, dont on suppose que la composante basse fréquence est parfaitement égale à \vec{g} . La figure III.17 donne un exemple de l'estimation en ligne des gyromètres lors d'un vol stationnaire de quelques minutes.

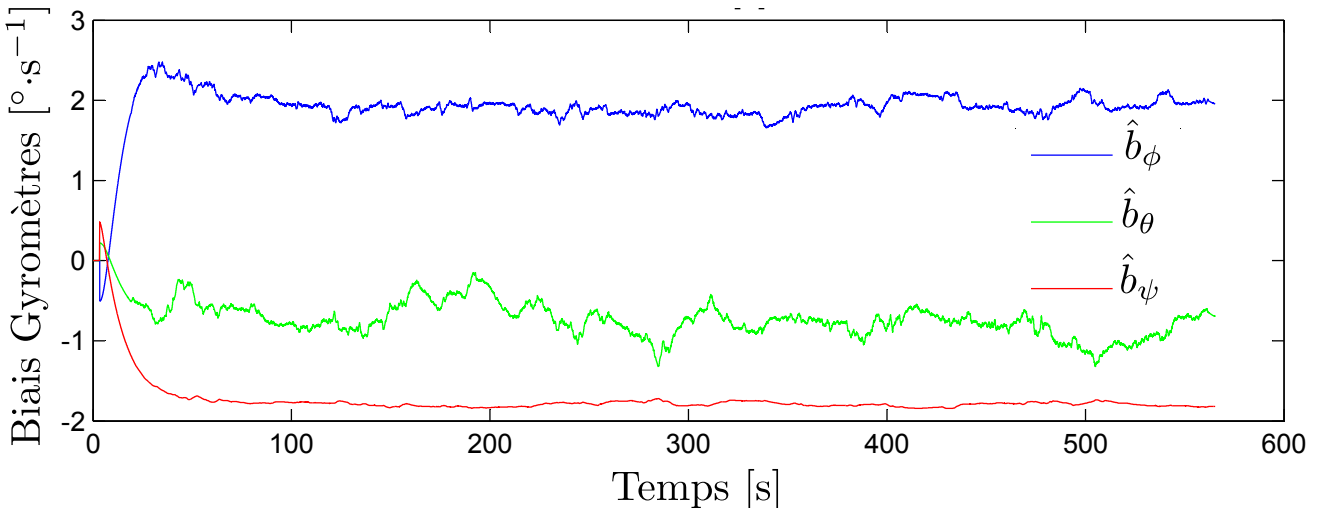


FIGURE III.17 – Exemple de l'estimation en ligne des biais des gyromètres avec le filtre complémentaire.

Dans le chapitre V, les accéléromètres ne seront plus utilisés, et on utilisera la direction du regard. Cette direction de regard jouera alors le même rôle "d'inclinomètre" que les accéléromètres. Et la

construction d'un EKF permettra donc de déterminer les biais des gyromètres à l'aide de cette mesure.

6 Développement d'un support logiciel open-source

6.1 Motivations

J'ai décidé, durant ma thèse, de développer une toolbox Matlab/Simulink permettant d'automatiser entièrement la génération de code pour l'exécution temps réel d'un modèle Simulink sur une cible dotée d'un Linux embarqué. Cette volonté découle de deux constats :

- Durant ces dernières années, de nombreuses cartes de calcul jouant le rôle de micro-ordinateur ont vu le jour. On peut citer par exemple le célèbre Raspberry Pi, la BeagleBone, les Gumstix, les IGEP, etc. Ces cartes ont l'intérêt de présenter des puissances de calculs conséquentes pour des facteurs de forme assez minimalistes, et des consommations relativement faibles (quelques Watt). Ce type de carte est donc tout indiqué pour des applications robotiques et leur utilisation dans ce domaine est de plus en plus commune.
- Ces architectures sont parfaitement compatibles avec Linux, qui est un système d'exploitation fortement configurable et dont on peut affiner les caractéristiques afin d'adapter son contenu à nos besoins (configuration du noyau.) Cela fait donc de Linux un candidat idéal pour les applications robotiques.

Cependant, développer des applications temps réel sous Linux reste une tâche difficile, nécessitant de solides connaissances des mécanismes temps réel, ainsi que des méthodes permettant de lier efficacement le hardware à l'application logicielle. Cela nécessite donc un choix minutieux des drivers à utiliser (permettant de gérer les entrées/sorties du système), du noyau Linux (qui déterminera les mécanismes temps réels qui pourront être utilisés), de l'ordonnanceur (qui permettra de gérer les priorités des différentes tâches), des méthodes de cadencement des tâches, etc.

Le but de la Toolbox RT-Mag est donc de fournir un cadre de développement qui rend transparent ces différents choix et garantit l'exécution temps réel d'un modèle Simulink sur un système Linux embarqué. Elle automatise l'ensemble des étapes de développement d'une application robotique classique (lecture/écriture des entrées/sorties temps réel, cadencement des tâches, etc.) et fournit une interface intuitive, afin de configurer l'accès aux entrées/sorties avec un haut niveau d'abstraction. Elle donne également accès à de nombreux modes et outils de débogage permettant d'avoir un retour sur les performances temps réel, et fournissant des "métriques" (temps d'exécution de portions de code, charges CPU, etc.) permettant d'optimiser les algorithmes de contrôle. Cet outil permet également de mettre en place facilement une station sol afin de superviser et enregistrer l'ensemble des données de l'application embarquée. Cela permet non seulement de limiter considérablement le temps de développement, mais également d'augmenter la versatilité des développements (car le développement des algorithmes de simulation et de l'application embarquée se font sous le même environnement qu'est Simulink).

6.2 Quelles applications ?

La toolbox RT-MaG peut donc être utilisée pour des applications robotiques reposant sur un calculateur de type Computer-On-Module (COM) et doté de Linux ; comme par exemple les cartes Gumstix, les Beaglebone, les Raspberrypi, les module IGEP, etc. Actuellement, RT-MaG existe sous deux versions :

- une version pour applications de type temps réel "mou". Il s'agit de la version qui a été utilisée pour présenter l'ensemble des travaux de cette thèse, et qui permet d'atteindre des fréquences de tâche allant jusqu'à 1 kHz. Cette version est *plug & play* avec la plupart des systèmes Linux et ne nécessite que la présence du patch PREEMPT-RT et des drivers d'entrées/sorties classiques (SPI, I2C, UART, et Réseaux) qui sont nativement présents dans la plupart des distributions Linux. Étant donné que cette version repose sur les drivers Linux natifs ainsi que sur une version de Linux non temps réel (le patch PREEMPT-RT permet d'atteindre des comportements proches du temps réel par des mécanismes de préemption, mais cela ne suffit pas toujours), elle ne fournit qu'un support temps réel "mou". De fortes limitations sont en effet dues aux drivers qui ne sont pas tous temps-réel, et les tâches liées aux entrées/sorties peuvent être "hasardeuse" dans le cas de forte charge CPU.
- Une version pour les applications de type temps-réel "dur". Cette version a été développée et continue de l'être, afin de fournir une maîtrise parfaite du temps réel, ce qui passe par l'utilisation d'un vrai noyau temps réel : XENOMAI (qui n'est en réalité qu'un co-noyau, mais qui permet de faire de Linux un vrai OS temps réel), et de drivers temps réel spécifiques. Ainsi, des tâches peuvent être planifiées à des fréquences jusqu'à quelques dizaines de kilo-hertz et le temps réel est garanti également pour les E/S, et ce même pour des charges CPU très élevées.

6.3 Choix Structurels et considérations techniques

6.3.1 Architecture typique d'application

La toolbox RT-Mag permet de créer facilement une connexion entre l'application distante et la station sol. La figure III.18 résume donc les différentes connexions et interactions du système embarqué et de la station sol. Le système embarqué est contrôlé via un lien Wifi (via un tunnel SSH) permettant de communiquer avec le système Linux embarqué via un terminal. Ce terminal permet de déclencher ou stopper le programme correspondant au modèle Simulink à tout moment et permet également de récupérer des informations sur son exécution (charge CPU, temps d'exécution moyens par tâche, overruns, etc. à une fréquence de 1Hz). Le lien Wifi permet également d'envoyer, si besoin, des consignes de haut niveau ou des paramètres au programme embarqué et de récupérer, en temps réel, l'ensemble des signaux à surveiller. L'envoi et la visualisation des informations se font également via un modèle Simulink sur la station sol. L'application embarquée est liée aux différents capteurs et autres calculateurs

via les connections Série, SPI, I2C, ADC ou GPIO comme renseigné dans le modèle Simulink. Une liaison série peut également être utilisée afin de valider le modèle embarquée en HIL : le système robotique réel est alors émulé par la station sol qui simule le comportement du système et génère les différentes E/S des capteurs.

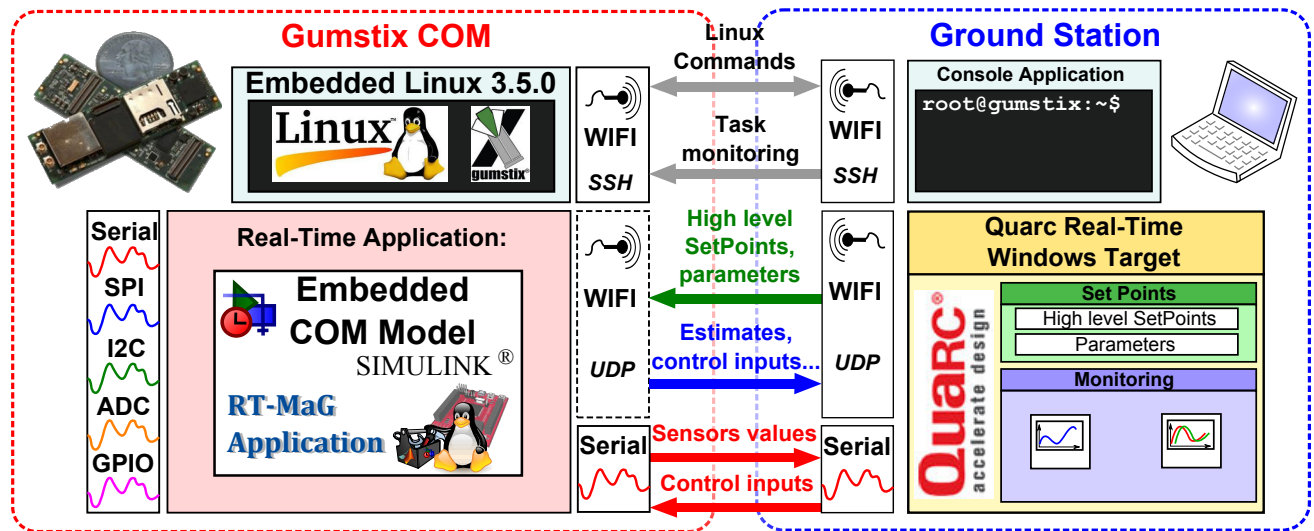


FIGURE III.18 – Schéma de communication classique d'une application RT-MaG. Le modèle Simulink embarqué est exécuté en temps réel sur le module COM (Computer-On-Module) et accède directement aux différentes entrées/sorties. L'application embarquée est reliée via wifi à la station sol afin de changer les paramètres ou échanger des données en temps réel. Un mode Processor-In-the-Loop est également disponible si le calculateur est relié à une station sol émulant le comportement du robot via une connexion série. L'utilisateur peut démarrer ou stopper le programme à tout moment (via le Wifi) et accéder à de nombreuses informations grâce à la console (charge CPU, temps d'exécution des tâches, overruns, etc.)

6.3.2 Génération de code automatique

Le modèle Simulink de la station sol et des contrôleurs embarqués peuvent être réalisés comme n'importe quel modèle Simulink. L'accès aux différentes entrées/sorties du système (SPI, I2C, etc.) est possible grâce à une librairie de blocs Simulink fournie par RT-MaG. Le modèle Simulink est ensuite traduit en langage C grâce à Embedded Coder de Matlab/Simulink. Ce code fournit une fonction *Model_stepi()* correspondant à l'ensemble des opérations relatives à une période d'échantillonnage *i*. Puis un ensemble de routine Matlab (appelé RT-MaG Coder) va générer le code manquant pour cadencer l'appel à ses fonctions et les lier aux entrées/sorties nécessaires. Une archive contenant l'ensemble des sources est alors générée automatiquement et déposée sur un serveur FTP. Les sources sont ensuite téléchargées et compilées sur la cible afin de générer l'exécutable correspondant au modèle.

De plus amples informations concernant la génération de codes, le support, ou encore la structure de l'application temps réel générée peuvent être trouvées dans [Manecy *et al.*, 2014b] ou sur le site

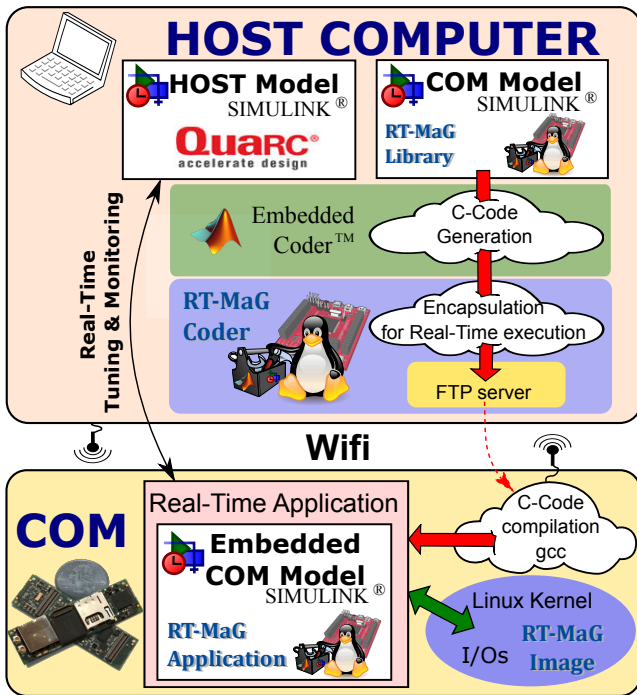


FIGURE III.19 – Diagramme de génération de RT-MaG. Le code C du modèle est généré par Embedded Coder. RT-MaG Coder génère ensuite le code nécessaire pour cadencer l'application et accéder aux entrées/sorties. Puis les sources sont compilées sur la cible et le programme est exécuté en temps réel et supervisé via le modèle Simulink de la station sol (exécuté également en temps réel).

internet du projet [Manecy, 2014].

6.4 Performances

Pour illustrer les performances temporelles de la toolbox RT-MaG, on propose d'étudier de plus près le cadencement obtenu pour une application présentant différentes fréquences d'échantillonnage et ayant été réalisée avec RT-MaG. L'application présente donc 4 fréquences différentes utilisant différentes E/S avec une charge CPU faible (voir III.21) et une charge CPU élevée (voir table III.3). La charge CPU est ici réalisée à l'aide de longue boucle "for" intégré dans des *Embedded Matlab Function*. Les différentes E/S présentes sont donc : le wifi échangeant avec une station sol à plus de 100 *float* à 1 kHz, deux ports série (UART) échangeant chacun 5 *float* à 500 Hz, 2 PWM dont les valeurs sont mises à jour à 333 Hz, un bus SPI échange cinq entiers 16-bit à 200 Hz et un port I2C échange 5 entiers 8-bit avec une carte Arduino (la Nanowii) à 200 Hz.

La figure III.21 montre donc l'évolution de chacune des périodes d'échantillonnage, mesurée dans le

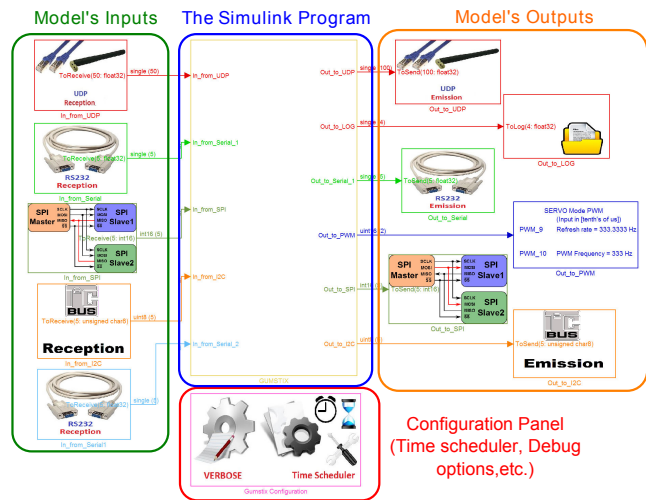


FIGURE III.20 – Exemple de modèle Simulink utilisant les blocs de la bibliothèque RT-MaG. Ici le modèle utilise une connexion Wifi (émission et réception UDP), 2 ports séries, 1 port SPI, 1 sortie vers un fichier de log, 1 port I2C et 2 sorties PWM. L'ensemble des options de configuration est réalisé via un block spécifique (entouré en rouge).

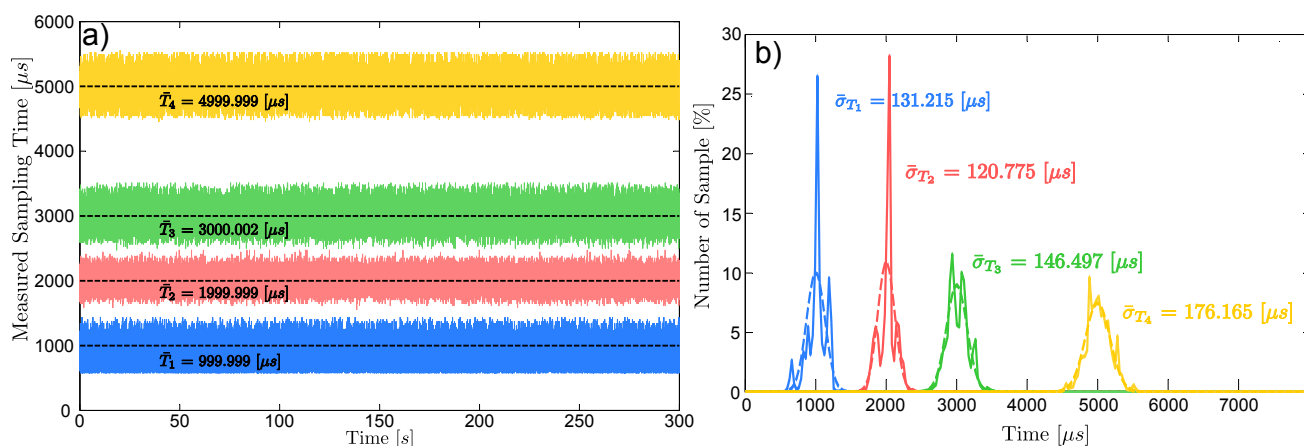


FIGURE III.21 – a) Mesure de la période effective pour 4 tâches cadencées à 1000Hz, 500Hz, 333Hz et 200Hz pour la version utilisant Linux PREEMPT-RT. La période moyenne ne diffère que de 0.1% par rapport à la période. b) Distribution de chacune des périodes obtenues. L'écart type est considéré suffisamment petit pour des applications robotiques allant jusqu'à 1 kHz.

cas d'un système Linux PREEMPT-RT. On observe que la période souhaitée est plutôt bien respectée car elle ne diffère que de 0.01% par rapport à la période cible désirée. Avec une faible charge CPU, aucun overrun n'a été mesuré, et les écarts type qui peuvent paraître assez élevés (de l'ordre de 131 μs pour la tâche la plus rapide) sont des valeurs typiques obtenues par le benchmark *Cyclictest* pour des structures ARM de ce type [Ficheux et Bénard, 2012]. On peut donc considérer qu'il s'agit des meilleures performances atteignables avec un Linux utilisant simplement un patch PREEMPT-RT. La table III.3 montre donc un résumé des performances de la même application pour une charge CPU supérieure à 90%. Dans ce cas, on observe que quelques overruns apparaissent (moins de 0.03% pour la tâche 3 et 0.01% pour la tâche 1). Mais, malgré cela, les mises à jour des E/S sont assez fiables pour être utilisées dans le cadre d'application robotique, comme le prouve la suite de ce chapitre, étant donné que la plateforme X4-MaG est capable de voler avec une très bonne précision dans l'arène de vol (voir section 8 de ce chapitre) ou de poursuivre une cible mouvante grâce à sa vision (voir section 5 du chapitre V) et ce avec des auto-pilotes exclusivement réalisés grâce à la toolbox RT-MaG.

6.5 Perspectives pour RT-MaG

Plus récemment, la toolbox RT-MaG a été modifiée afin de fonctionner également sous XENOMAI qui offre des performances temps réel bien meilleures, comme le montre le tableau III.4. On observe alors que les latences du système XENOMAI pour traiter une tâche planifiée par timer (ici une tâche à 1 kHz) sont bien plus faibles et bien plus constantes que sur un système Linux PREEMPT-RT classique. En effet, alors que pour le système Linux l'écart type des latences est de l'ordre de 130 μs , il est de l'ordre de 1 à 2 μs et ce, même pour un système fortement chargé. De plus on remarque que les latences maximales sont de l'ordre de plusieurs centaines de micro secondes sous PREEMPT-RT alors qu'elles

	Tâche 1	Tâche 2	Tâche 3	Tâche 4
Période cible [μ s]	1000	2000	3000	5000
Période moyenne [μ s]	999.9	2000.4	2999.7	5013.8
Overruns/Nb exécution	0/300000	13/149977	0/99999	14/59832
Temps moyen d'exécution [μ s]	47.9	74.5	489.8	3303.2
Temps min d'exécution [μ s]	10	30	427	2014
Temps max d'exécution [μ s]	458	3631	702	5462
Charge CPU [μ s]	90%			

TABLE III.3 – Timing d’une application RT-MaG constituée de 4 tâches utilisant un lien Wifi, 2 liens UART, 1 lien I2C, 1 lien SPI et générant 2 PWM.

ne dépassent pas les 30 μ s sous XENOMAI. En effet, pour une expérience de plusieurs dizaines de minutes, on observe avec PREEMPT-RT une latence maximale de 435 μ s pour système non chargé et de 536 μ s pour un système chargé. Avec même, dans de très rares occasions (une à deux occurrences sur plusieurs dizaines de minutes), des latences de plusieurs millisecondes (4852 μ s pour un système non chargé et 8972 μ s pour un système fortement chargé). Ce phénomène n’apparaît absolument pas sur le système XENOMAI. On observe même que les latences maximales atteintes par le système XENOMAI sont inférieures d’un facteur 20 et les écarts types inférieurs d’un facteur 100 environ.

	CPU <50%		CPU >90%	
	PREEMPT-RT	XENOMAI	PREEMPT-RT	XENOMAI
Tps. Décl. min. [μ s]	-	4.769	-	8.999
Tps. Décl. moy. [μ s]	-	7.307	-	11.076
Tps. Décl. max. [μ s]	435 (4852)	15.076	526 (8972)	26.307
Écart type [μ s]	134.327	0.978	125.335	1.662

TABLE III.4 – Latences mesurées pour les systèmes Linux PREEMPT-RT et XENOMAI en cas de faibles et fortes charges CPU.

De même, plus récemment, nous avons créé un groupe de travail regroupant plusieurs laboratoires (ISM, Heudiasyc de l’UTC et Gipsa-Lab) afin de mutualiser les développements du noyau et des drivers concernant le système Linux XENOMAI pour les cartes Gumstix et IGEP (architecture reposant sur le même processeur OMAP). Cela permettra de réduire considérablement les latences du système et d’obtenir des drivers temps réel offrant également des capacités asynchrones. Ce qui permettra de considérablement réduire la charges CPU liée aux E/S. En effet, à l’heure actuelle, sur le système PREEMPT-RT seul, le temps réel est obtenu en réalisant des lectures et écritures bloquantes sur les E/S, ce qui représente, pour des fréquences de travail élevées, un gaspillage des ressources processeur substantiel.

Cela permettra donc d’atteindre des performances temps réel encore plus élevées bien qu’actuelle-

ment la toolbox présente des résultats satisfaisants et suffisants pour asservir par exemple un quadrirotor comme le démontre les travaux de cette thèse.

7 Stabilisation et asservissement de position dans l'arène de vol

Maintenant que nous avons décrit la plateforme ainsi que sa structure logicielle, nous allons nous intéresser aux algorithmes de contrôle constituant l'auto-pilote du quadrirotor.

Dans cette partie, nous allons donc nous focaliser sur la stabilisation du robot X4-MaG dans l'arène de vol Vicon. Dans ce cadre, on utilisera les données de positions et de vitesses de translation fournies par le Vicon, afin de réaliser un contrôleur offrant de bonnes performances de poursuite de trajectoire.

On réutilise le modèle précédemment développé dans la section 4, à la différence qu'ici les rotations sont exprimées sous forme de quaternions et que l'on ne conserve que les forces et moments principaux :

$$\begin{cases} \dot{\boldsymbol{\xi}} = \mathbf{v} \\ m\dot{\mathbf{v}} = -m\vec{\mathbf{g}} + \mathbf{F}_{\Sigma} \\ \dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \mathbf{p}(\boldsymbol{\Omega}) \\ \mathbf{I}\dot{\boldsymbol{\Omega}} = -\boldsymbol{\Omega} \times \mathbf{I}\boldsymbol{\Omega} + \boldsymbol{\Gamma}_{\Sigma} \end{cases} \quad (\text{III.59})$$

Où $\boldsymbol{\xi}$ est la position du robot, \mathbf{v} sa vitesse de translation et $\boldsymbol{\Omega}$ ses vitesses de rotations angulaires. Pour la suite de ce chapitre, on introduit les notations suivantes : \otimes représente le produit de quaternions, \times est le produit vectoriel et $\mathbf{p}(\boldsymbol{\Omega})$ représente un quaternion pur $\mathbf{p}(\boldsymbol{\Omega}) = \begin{pmatrix} 0 & \boldsymbol{\Omega} \end{pmatrix}^T$ (avec $\boldsymbol{\Omega}$ le vecteur de dimension 3 correspondant aux vitesses de rotation du robot). \mathbf{F}_{Σ} est la somme des forces agissant sur le robot, et $\boldsymbol{\Gamma}_{\Sigma}$ la somme des couples. Pour le design de contrôleurs, ces deux entités peuvent être simplement décrites par :

$$\mathbf{F}_{\Sigma} = \mathbf{q} \odot \left(T_{\Sigma} \cdot \vec{\mathbf{b}}_3 \right) + \mathbf{F}_{\Delta} = \mathbf{q} \odot \left(\sum_{i=0}^4 c_T \cdot \omega_{r_i}^2 \vec{\mathbf{b}}_3 \right) + \mathbf{F}_{\Delta} \quad (\text{III.60})$$

$$\boldsymbol{\Gamma}_{\Sigma} = \boldsymbol{\Gamma}_{\Theta} + \boldsymbol{\Gamma}_{\Delta} \quad (\text{III.61})$$

où \odot est le produit de quaternion-vecteur défini par $\mathbf{q} \odot \mathbf{v} = \mathbf{q} \cdot \begin{pmatrix} 0 & \mathbf{v} \end{pmatrix}^T \cdot \mathbf{q}^{-1}$ avec $\vec{\mathbf{b}}_3 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$ (c'est à dire la verticale dans le repère du robot), où c_T est le coefficient de poussée (voir 4.4.2) et ω_{r_i} la vitesse de rotation du rotor i . \mathbf{F}_{Δ} et $\boldsymbol{\Gamma}_{\Delta}$ sont des perturbations aérodynamiques et des perturbations en couples que l'on considère ici inconnues. On considère que la poussée totale T_{Σ} et les couples $\boldsymbol{\Gamma}_{\Theta}$

permettant de contrôler le robot sont décrits par (configuration X4) :

$$\begin{pmatrix} T_\Sigma \\ \mathbf{\Gamma}_\Theta \end{pmatrix} = \underbrace{\begin{pmatrix} c_T & c_T & c_T & c_T \\ -l'.c_T & l'.c_T & l'.c_T & -l'.c_T \\ -l'.c_T & -l'.c_T & l'.c_T & l'.c_T \\ c_Q & -c_Q & c_Q & -c_Q \end{pmatrix}}_M \begin{pmatrix} \omega_{r_1}^2 \\ \omega_{r_2}^2 \\ \omega_{r_3}^2 \\ \omega_{r_4}^2 \end{pmatrix} \quad (\text{III.62})$$

où T_Σ est un scalaire et $\mathbf{\Gamma}_\Theta = (\Gamma_\phi \ \Gamma_\theta \ \Gamma_\psi)^T$ est un vecteur de dimension 3. On rappelle qu'ici $l' = \frac{\sqrt{2}}{2}.l$ où l est la longueur d'un bras (demi envergure du robot) et c_Q coefficient de couple lié à la trainée. M est appelé la matrice de mixage.

7.1 Le contrôleur de vitesse des hélices

L'asservissement en vitesse de rotation de chacun des rotors, est réalisé à l'aide du contrôleur décrit en détail dans la section 2.5.3. Il s'agit d'un contrôleur PI, avec anti-windup, auquel on ajoute un terme d'anticipation (feedforward). Ici, on rappelle la boucle de contrôle simplifiée (sans anti-windup) :

$$U_{\text{ESC}_i} = k_p(\omega_{r_{fi}}^* - \bar{\omega}_{r_i}) + k_i \int_0^t (\omega_{r_{fi}}^* - \bar{\omega}_{r_i}) dt + ff(\omega_{r_i}^*) \quad (\text{III.63})$$

Ce contrôleur fournit donc une sortie PWM U_{ESC_i} à 490 Hz. A partir d'ici, les valeurs de consignes seront notées avec une étoile (comme par exemple, $\omega_{r_i}^*$), les valeurs estimées par un chapeau (par exemple, $\hat{\Omega}$), et les valeurs mesurées par une barre (par exemple, $\bar{\omega}_{r_i}$).

7.2 Estimation d'attitude : le filtre complémentaire

Le problème d'estimation d'attitude est résolu ici en mettant en place un filtre complémentaire comme celui proposé par exemple dans [Mahony *et al.*, 2008] ou [Mahony *et al.*, 2012]. Ce filtre est exprimé ici en quaternion afin d'éviter les singularités présentes dans la représentation sous forme de matrice de rotation. Le filtre permet donc d'estimer à la fois l'attitude du robot, mais également les biais des gyromètres, afin de fournir une estimée des vitesses de rotation non biaisées. Il a été montré que ce filtre converge exponentiellement à la fois théoriquement et dans la pratique ([Mahony *et al.*, 2008]). La centrale inertielle (MPU6050) utilisée ici ne fournit pas de magnétomètre. Cela implique donc l'utilisation d'une autre mesure permettant d'estimer le biais du gyromètre selon l'axe de lacet. L'avantage du filtre complémentaire est sa facilité à intégrer différents capteurs (fusion des informations d'un capteur par simple ajout d'un terme dans le terme de correction α). On choisit donc d'utiliser une partie de l'information (quaternion d'attitude) fournie par le système Vicon. Et grâce à un gain adapté k_v , on peut n'utiliser que l'information liée au lacet dans ce quaternion :

$$\begin{cases} \dot{\hat{\mathbf{q}}} = \frac{1}{2} \hat{\mathbf{q}} \otimes \mathbf{p}(\underbrace{\bar{\boldsymbol{\Omega}} - \hat{\mathbf{b}}}_{\hat{\boldsymbol{\Omega}}} - \boldsymbol{\alpha}) \\ \dot{\hat{\mathbf{b}}} = \mathbf{k}_b \cdot \boldsymbol{\alpha} \\ \boldsymbol{\alpha} = \mathbf{k}_a \circ \frac{\bar{\mathbf{g}} \times (\hat{\mathbf{q}} \odot \bar{\mathbf{g}}_{IMU})}{\|\bar{\mathbf{g}}\|^2} + \mathbf{k}_v \circ \tilde{\mathbf{v}} \end{cases} \quad (\text{III.64})$$

où \circ est le produit d'Hadamard de vecteur (multiplication élément par élément), $\tilde{\mathbf{s}}$ et $\tilde{\mathbf{v}}$ sont respectivement la partie scalaire et vectorielle de l'erreur de quaternion entre le quaternion estimé et le quaternion mesuré par le Vicon. Cette erreur de quaternion est définie par $\tilde{\mathbf{q}} = \bar{\mathbf{q}}_{Vicon}^{-1} \otimes \hat{\mathbf{q}} = (\tilde{\mathbf{s}} \quad \tilde{\mathbf{v}})^T$ avec $\bar{\mathbf{q}}_{Vicon}$ le quaternion mesuré par la système Vicon. Ici $\mathbf{k}_a = (k_{a1} \quad k_{a2} \quad 0)$ avec $k_{a1} > 0$ et $k_{a2} > 0$, et $\mathbf{k}_v = (0 \quad 0 \quad k_{v3})$ avec $k_{v3} > 0$. L'utilisation de ces pondérations permet de ne pas utiliser l'information des accéléromètres pour l'estimation du lacet, car cette mesure n'apporte aucune information (pour des angles de roulis et tangage nuls, qui est le cas nominal, le vecteur de mesure des accéléromètres est colinéaire à $\bar{\mathbf{g}}$ et indépendant de l'angle de lacet.) De même, un magnétomètre n'apporte que très peu d'informations sur les angles de roulis et tangage, et on représente cela en extrayant uniquement les informations de lacet du quaternion fournies par le Vicon.

7.3 Contrôleurs d'attitude

Ici nous allons donc tester deux contrôleurs d'attitude différents afin de comparer leurs performances et leurs robustesses à différentes perturbations. Le but étant de sélectionner le meilleur contrôleur afin d'assurer la fonction de contrôleur de *secours* nécessaire pour fiabiliser la plateforme lors d'élaboration de nouveaux algorithmes (comme dans le chapitre V). Nous proposons donc ici de comparer deux contrôleurs : un premier consistant en deux boucles de contrôleurs PI imbriqués permettant d'asservir les vitesses de rotations ainsi que l'attitude. Le deuxième contrôleur est un contrôleur géométrique conçu directement sur l'espace $SO(3)$ comme présenté par [Lee et al., 2010] et mis en œuvre dans ma thèse sous forme de quaternions. Ces contrôleurs d'attitude vont donc recevoir en entrée des consignes d'attitude $\boldsymbol{\Theta}^* = (\phi^* \quad \theta^* \quad \psi^*)^T$ fournies par le contrôleur de position, et délivre des consignes de couples $\boldsymbol{\Gamma}^*$ à générer.

7.3.1 Contrôleur d'attitude sous forme de PI imbriqués

Ici, on propose de stabiliser l'attitude en utilisant simplement deux boucles de PI imbriqués. Le premier contrôleur permet donc d'asservir les vitesses de rotation et délivre des consignes de couples à générer :

$$\boldsymbol{\Gamma}_{\boldsymbol{\Theta}}^* = \underbrace{\mathbf{k}_p(\boldsymbol{\Omega}_f^* - \hat{\boldsymbol{\Omega}}) + \mathbf{k}_i \int_0^t (\boldsymbol{\Omega}_f^* - \hat{\boldsymbol{\Omega}}) dt}_{\tau_{\boldsymbol{\Theta}}^*_{feedback}} + \underbrace{\hat{\boldsymbol{\Omega}} \times \mathbf{I} \hat{\boldsymbol{\Omega}} + \mathbf{I} \frac{d\boldsymbol{\Omega}_f^*}{dt}}_{\tau_{\boldsymbol{\Theta}}^*_{feedforward}} \quad (\text{III.65})$$

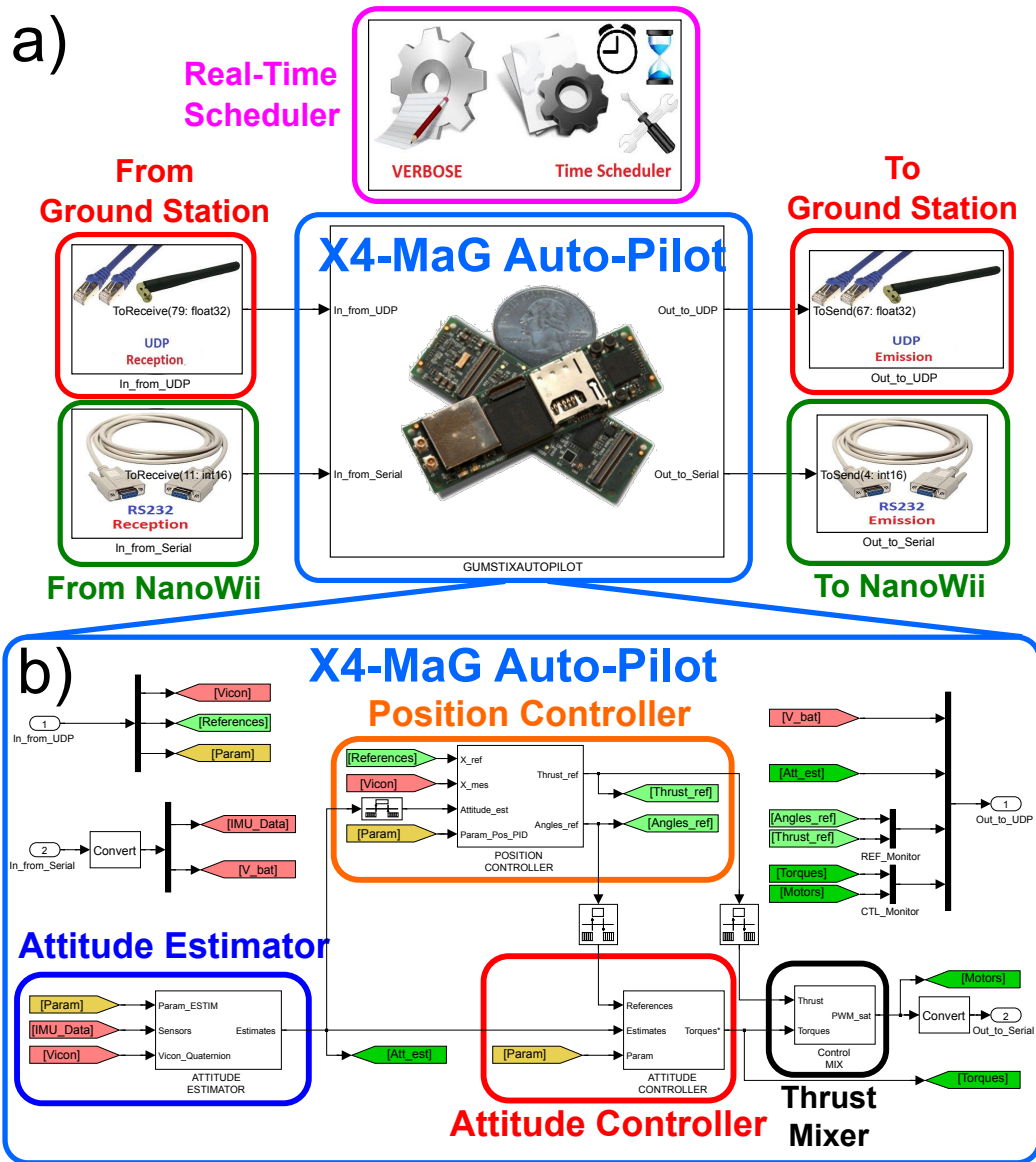


FIGURE III.22 – Le modèle Simulink de l’auto-pilote de X₄-MaG qui réalise à la fois l’estimation d’attitude, l’asservissement d’attitude et l’asservissement de position. a) Les entrées/sorties de la Gumstix sont accessibles directement grâce à la toolbox RT-MaG. Ici, l’auto-pilote reçoit directement via le wifi, les "waypoints", les mesures du Vicon et les paramètres des contrôleurs envoyés par la station sol. L’ensemble des résultats d’estimations et des commandes générées est envoyé à la station sol qui les enregistre et permet de superviser le tout en temps réel. Les données inertielles et les consignes rotors sont échangées via liaison série (UART) avec la Nanowii. b) Une version simple d’auto-pilote, intégrant un contrôleur de position, un contrôleur d’attitude et l’estimateur d’attitude.

où Γ_{Θ}^* est un vecteur de dimension 3 correspondant aux couples, $\hat{\Omega} = \bar{\Omega} - \hat{b}$ est le vecteur de vitesse de rotation obtenu en retirant le biais estimé des gyromètres aux mesures gyrométriques; I est la matrice d’inertie. k_p et k_i sont des matrices diagonales dont les termes sont positifs. Les deux éléments

composant le terme de feedforward $\Gamma_{\Theta_{feedforward}}^*$ permettent de compenser les effets gyroscopiques liés aux vitesses de rotation du robot, et générer *a priori* le couple correspondant à la consigne de vitesse filtrée Ω_f^* . Cette consigne de vitesse est obtenue en filtrant les consignes Ω^* fournies par le contrôleur d'attitude afin d'assurer leur faisabilité et leur différentiabilité :

$$\Omega_f^*(s) = \frac{\Omega^*(s)}{(\tau_{f_\Theta} s + 1)} \quad (\text{III.66})$$

où τ_{f_Θ} est la constante de temps du filtre du premier ordre qui assure que la dérivée première Ω_f^* est continue (nécessaire dans III.65). La consigne de vitesse de rotation Ω^* est fournie par le contrôleur d'attitude qui est décrit par :

$$\Omega^* = k_p(\Theta^* - \hat{\Theta}) + k_i \int_0^t (\Theta^* - \hat{\Theta}) \quad (\text{III.67})$$

où k_p et k_i sont des matrices diagonales positives, et $\hat{\Theta} = (\hat{\phi} \ \hat{\theta} \ \hat{\psi})^T$ est le vecteur décrivant les angles d'Euler correspondant au quaternion d'attitude estimé \hat{q} . $\Theta^* = (\phi^* \ \theta^* \ \psi^*)^T$ où ϕ^* et θ^* sont fournies par le contrôleur de position et ψ^* est un angle de cap spécifié par l'utilisateur.

7.3.2 Contrôleur d'attitude géométrique en quaternions

Le deuxième contrôleur proposé est une version en quaternions du contrôleur décrit par [Lee *et al.*, 2010], auquel un effet intégral a également été ajouté. On notera que la stabilité asymptotique de ce type de contrôleur ([Lee *et al.*, 2010]) ne tient que sous des contraintes assez restrictives (la matrice d'inertie parfaitement connue, la dynamique des moteurs est négligeable, etc.). Ces hypothèses ne peuvent pas vraiment être garanties dans un cadre expérimental, mais ses bonnes performances et sa robustesse (sous la forme de matrices de rotation) ont néanmoins été vérifiées expérimentalement comme par exemple dans [Mellinger et Kumar, 2011]. Dans mes travaux de thèse, je montre que la version sous forme de quaternion avec effet intégral fonctionne également de façon satisfaisante d'un point de vue expérimental (voir la section 8.2).

A partir des consignes d'attitude délivrées par le contrôleur de position $\Theta^*(t)$ (où $\Theta^*(t)$ est la consigne d'attitude décrite par $\Theta^*(t) = (\phi^*(t) \ \theta^*(t) \ \psi^*(t))^T$), on définit une nouvelle consigne $\Theta_f^*(t)$, correspondant à la consigne initiale $\Theta^*(t)$ filtrée à l'ordre 3, afin de garantir que les dérivées temporelles d'ordre 1 et 2 Ω_f^* et $\dot{\Omega}_f^*$ soient continues.

Finalement, la consigne en quaternion q_f^* est extraite à partir du vecteur $\Theta_f^*(t)$ et on définit un vecteur d'erreur e_q de dimension 3×1 entre le quaternion estimé et le quaternion de consigne qui peut être défini comme suit :

$$e_q = \tilde{s}_{e_q} \cdot \tilde{v}_{e_q} \quad (\text{III.68})$$

où \tilde{s}_{e_q} et \tilde{v}_{e_q} sont respectivement la partie scalaire et vectorielle de l'erreur de poursuite en quaternion $\tilde{q}_{e_q} = \hat{q}^{-1} \otimes q^*$.

L'erreur de poursuite en vitesse de rotation peut être décrite par :

$$e_\Omega = [(\hat{q}^{-1} \otimes q_f) \odot \Omega_f^*] - \hat{\Omega} \quad (\text{III.69})$$

Ensuite, le contrôleur géométrique sur $SO(3)$ peut être construit de la façon suivante :

$$\Gamma_\Theta^* = I \left(k_R e_q + k_I \int_0^t e_q dt + k_\Omega e_\Omega \right) + \hat{\Omega} \times I \hat{\Omega} - I \left(\hat{\Omega}_\times (\tilde{q}_{e_q} \odot \Omega_f^*) - \tilde{q}_{e_q} \odot \hat{\Omega}_f^* \right) \quad (\text{III.70})$$

7.4 Contrôle de position en 3D

7.4.1 Boucles de contrôle

Le contrôleur de position proposé ici, est basé sur celui proposé par [Michael *et al.*, 2010], auquel on ajoute les non linéarités \sin^{-1} . Ce contrôleur permet donc, en ajustant les angles de roulis ϕ et de tangage θ , de poursuivre les consignes de position $\xi_1^* = x^*$ et $\xi_2^* = y^*$. Les consignes de roulis et de tangage sont donc obtenues de la façon suivante :

$$\begin{cases} \phi^* = \sin^{-1} \left(\frac{m}{T_\Sigma^*} \left(\ddot{\xi}_{f1}^* \sin \hat{\psi} - \ddot{\xi}_{f2}^* \cos \hat{\psi} \right) \right) \\ \theta^* = \sin^{-1} \left(\frac{m}{T_\Sigma^*} \left(\ddot{\xi}_{f1}^* \cos \hat{\psi} + \ddot{\xi}_{f2}^* \sin \hat{\psi} \right) \right) \end{cases} \quad (\text{III.71})$$

où $\ddot{\xi}_{f1}^*$ et $\ddot{\xi}_{f2}^*$ sont des consignes d'accélération horizontales composées par un terme de feedforward (anticipation) et de feedback (déterminé par un contrôleur PID) et sont obtenues comme suit :

$$\ddot{\xi}_{fi}^* = \underbrace{k_P \tilde{\xi}_i + k_I \int_0^t \tilde{\xi}_i dt + k_D \left(\frac{d\xi_{fi}^*}{dt} - \bar{v}_i \right)}_{\tilde{\xi}_{i,feedback}^*} + \underbrace{\frac{d^2 \xi_{fi}^*}{dt^2}}_{\tilde{\xi}_{i,feedforward}^*} \quad (\text{III.72})$$

où $\tilde{\xi}_i$ est l'erreur de position 3D donnée par $\tilde{\xi}_i = (\xi_{fi}^* - \bar{\xi}_i)$, $\bar{\xi}_i$ et \bar{v}_i sont respectivement les positions et vitesses linéaires mesurées par le Vicon sur le plan horizontal. ξ_{fi}^* correspond à une consigne de position filtrée du waypoint ξ_i^* , qui assure au signal d'être de classe C^3 et permet d'introduire les saturations physiques du système (c'est à dire que le terme de feedforward qui en découlera sera continu et réalisable) :

$$\xi_{fi}^*(s) = \frac{\xi_i^*(s)}{(\tau_{f_\xi} s + 1)^3} \quad (\text{III.73})$$

La poursuite de la consigne de position $z^* = \xi_3^*$ sur l'axe vertical est réalisée en ajustant simplement la poussée totale T_Σ définie par :

$$T_\Sigma^* = \underbrace{\frac{k_P \tilde{\xi}_3 + k_I \int_0^t \tilde{\xi}_3 dt + k_D \left(\frac{d\xi_{f3}^*}{dt} - \bar{v}_z \right)}{\cos \hat{\phi} \cos \hat{\theta}}}_{\tilde{z}_{feedback}} + \underbrace{\frac{m}{\cos \hat{\phi} \cos \hat{\theta}} \cdot \left(g + \frac{d^2 \xi_{f3}^*}{dt^2} \right)}_{\tilde{z}_{feedforward}} \quad (\text{III.74})$$

Le rôle de ce préfiltre de consigne est donc de construire une trajectoire réalisable, pour rallier les waypoints fournis par la mission (déterminés par l'utilisateur par exemple).

7.4.2 Générateur de trajectoires et sécurités

Dans les faits, ce préfiltre de consigne (qui peut être vu comme un générateur de trajectoire permettant de planifier une trajectoire réalisable) est implémenté avec des intégrateurs saturés qui permettent de respecter les saturations du système physique et/ou les limitations fixées par le cahier des charges. La figure III.23 montre donc l'implémentation de ce préfiltre qui permet de calculer les trajectoires ξ_f^* , $\frac{d\xi_f^*}{dt}$ et $\frac{d^2 \xi_f^*}{dt^2}$ nécessaires aux différents termes de feedforward. Les saturations, quant à elles, permettent de garantir que la trajectoire générée sera réalisable par le robot. Les saturations en accélération assurent de ne pas atteindre des angles de roulis et tangage trop importants. De plus, dans le cas où un nouveau waypoint est placé très loin de la position actuelle, les saturations en vitesses permettent de limiter la vitesse maximale à atteindre pour relier deux waypoints ; ce qui n'est pas possible à garantir avec un préfiltre linéaire classique. Enfin, les saturations en positions permettent de fixer un volume de travail (cela peut permettre de séparer l'espace de vol lors de l'utilisation de plusieurs robots, ou de ne pas chercher à atteindre des coordonnées incohérentes).

Ainsi, les saturations des différents intégrateurs sont données par :

$$\left\{ \begin{array}{l} S_\xi^+ = \xi_{max} = \begin{pmatrix} x_{max} \\ y_{max} \\ z_{max} \end{pmatrix}, \quad S_\xi^- = \xi_{min} = \begin{pmatrix} x_{min} \\ y_{min} \\ z_{min} \end{pmatrix} \\ S_v^+ = v_{max} = \begin{pmatrix} v_{x_{max}} \\ v_{y_{max}} \\ v_{z_{max}} \end{pmatrix}, \quad S_v^- = v_{min} = \begin{pmatrix} v_{x_{min}} \\ v_{y_{min}} \\ v_{z_{min}} \end{pmatrix} \\ S_a^+ = a_{max} = \begin{pmatrix} \frac{T_\Sigma}{m} \cdot \sin(\theta_{max}) \\ \frac{T_\Sigma}{m} \cdot \sin(\phi_{max}) \\ \sum_{i=1}^4 c_T \omega_{ri_{max}}^2 \end{pmatrix}, \quad S_a^- = a_{min} = \begin{pmatrix} \frac{T_\Sigma}{m} \cdot \sin(\theta_{min}) \\ \frac{T_\Sigma}{m} \cdot \sin(\phi_{min}) \\ \sum_{i=1}^4 c_T \omega_{ri_{min}}^2 \end{pmatrix} \end{array} \right. \quad (\text{III.75})$$

Où ξ_{max} et ξ_{min} permettent de définir un volume de travail dont on ne veut pas que le robot sorte, v_{max}

et v_{min} définissent les vitesses linéaires maximales autorisées. $\theta_{max/min}$ et $\phi_{max/min}$ sont les angles de roulis et de tangage maximaux autorisés. Ici, l'accélération maximale est déterminée en fonction de la poussée de façon à ce que les angles d'attitude nécessaires ne dépassent pas les angles maximaux fixés par l'utilisateur. Les vitesses maximales et minimales des rotors $\omega_{ri_{max}}$ et $\omega_{ri_{min}}$ sont ici respectivement de $125 \text{ rad} \cdot \text{s}^{-1}$ et $1162 \text{ rad} \cdot \text{s}^{-1}$. Ainsi, lors de l'arrivée d'un nouveau waypoint, quelque soit la position actuelle du robot, on garantit que la trajectoire demandée au robot pour atteindre ce waypoint, respectera l'ensemble des limites fixées par le cahier des charges.

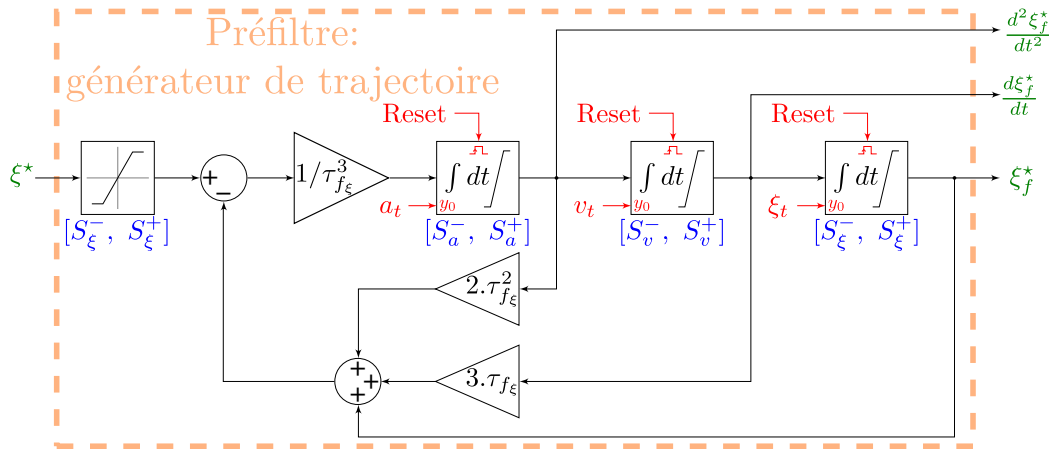


FIGURE III.23 – Implémentation du préfiltre (générateur de trajectoire). Ce préfiltre permet de prendre en compte les limitations du système physique et/ou du cahier des charges grâce aux saturations S_x^+ et S_x^- des états ξ , v et a . De plus, grâce à une entrée de *reset* activée par le *superviseur Gumstix* lors de changement de mode, la trajectoire peut être re-planifier à partir de l'état actuel $(a_t \quad v_t \quad \xi_t)$

Remarque III.4: Ce préfiltre de consigne joue également un rôle primordial lorsqu'on réalise un changement de mode de contrôle (par exemple, lorsque le mode de *secours* reprend la main sur le mode *experimental*). En effet, lors d'un basculement de mode, il suffit de réinitialiser les états du préfiltre avec l'état courant du système, mesuré par le Vicon. Cela permet, lors du basculement, de re-générer une trajectoire réalisable à partir de l'état courant et donc de garantir une commande "souple". La figure III.24 montre, par exemple, plusieurs basculements automatiques du mode *experimental* vers le mode de *secours*, lors du test d'un contrôleur *experimental* ne respectant pas les exigences du *superviseur* (contrôleur volontairement instable). Aux instants $t = 149\text{s}$, $t = 158\text{s}$ et $t = 176\text{s}$, on bascule donc sur le *contrôleur experimental* qui est instable et s'écarte fortement de la trajectoire désirée (courbes vertes). On voit donc que, dès lors que la trajectoire du robot (courbes noires) croise les seuils de sécurité qui ont été fixés, le *contrôleur de secours* reprend la main et re-planifie une trajectoire ξ_f^* pour rejoindre le waypoint actuel ξ^* . On observe bien sur la figure III.24b que lorsque le *mode de secours* s'enclenche, la trajectoire est replanifiée à partir de l'état actuel du robot (positions : voir figure III.24a-b, vitesses : voir III.24c).

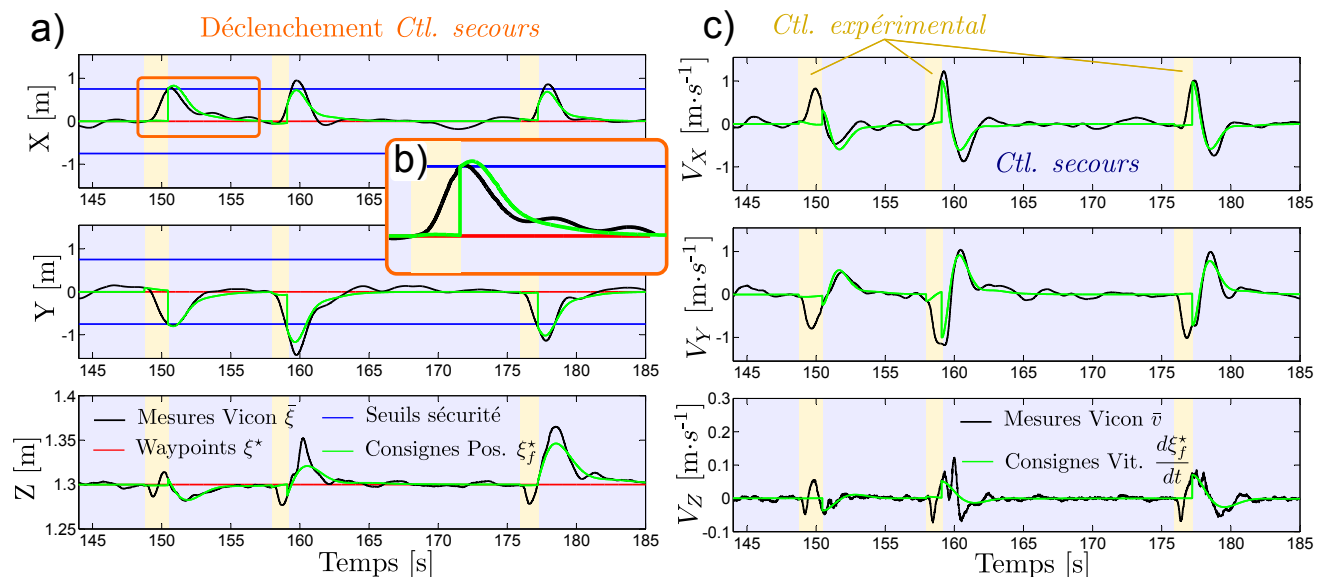


FIGURE III.24 – a) Évolution des positions X , Y et Z lors du test de récupération automatique par le mode de secours. Le *mode expérimental* est lancé par l'utilisateur 3 fois (fond jaune) avec un contrôleur volontairement instable. Dès lors que la position du robot dépasse le seuil de sécurité fixé (lignes bleues), le contrôleur de secours s'enclenche, re-planifie une trajectoire (lignes vertes) et rectifie la trajectoire (lignes noires). b) Zoom sur la première récupération automatique par le *contrôleur de secours*. c) Les consignes de vitesses linéaires (en vert) fournies par le préfiltre générateur de consigne et les vitesses mesurées par le Vicon (en noir).

Finalement, on obtient la consigne de vitesse de rotation pour chaque rotor correspondant au vecteur de commande $(T_{\Sigma}^* \quad \Gamma_{\Sigma}^*)^T$ en inversant la matrice M qui est donnée par l'équation (III.62).

Comme M est une matrice constante (la configuration des rotors est fixe), l'inversion de cette matrice peut être réalisée hors ligne et une seule fois.

8 Résultats dans l'arène de vol

Dans cette partie, nous allons tout d'abord, comparer les résultats obtenus avec les deux contrôleurs d'attitude présentés précédemment et quantifier l'influence de leurs performances sur la précision de positionnement dans le volume de vol. Puis, nous évaluerons les précisions de positionnement obtenues pour le meilleur des deux contrôleurs sur une trajectoire circulaire.

8.1 Comparaison des deux contrôleurs d'attitude

Nous allons donc comparer le rejet de perturbations en attitude qu'offrent les deux contrôleurs d'attitude décrit par les équations (III.65) et (III.70).

8.1.1 Modèle de perturbations d'attitude

Afin de pouvoir comparer les deux contrôleurs pour ces perturbations, il est nécessaire de générer toujours la même perturbation. On choisit de générer artificiellement cette perturbation (plutôt qu'avec un système extérieur générant une rafale de vent), en ajoutant un terme à la commande de couple qui est appliquée au robot (Γ_{Θ}^* devient $\Gamma_{\Theta}^* + \Gamma_p$).

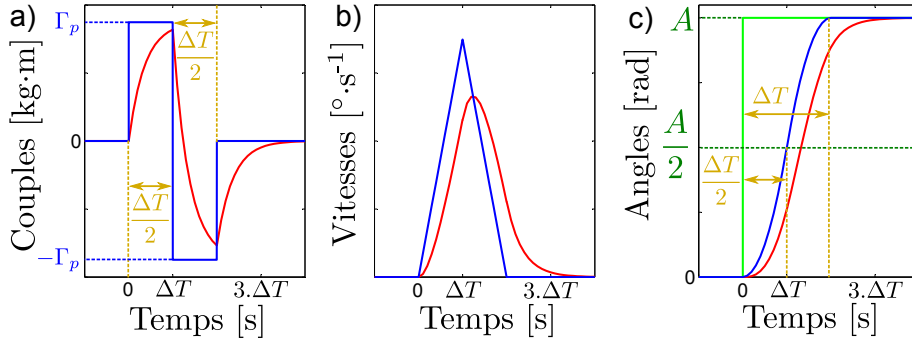


FIGURE III.25 – a) En bleu, le profil de perturbation de commande utilisé pour générer une perturbation d'attitude répétable, et en rouge, le couple obtenu incluant la dynamique des rotors. b) En bleu, le profil de vitesse de rotation correspondant, en négligeant la dynamique des rotors et en rouge, en considérant la dynamique des rotors. c) En vert, la perturbation en attitude souhaitée, en bleu la perturbation sans prendre en compte la dynamique des rotors et en rouge, la perturbation réellement obtenue une fois réalisée par le robot (réponses simulées).

Le profil de perturbation est donnée par la courbe bleue de la figure III.25a. Cette perturbation génère donc un profil de vitesse de rotation correspondant à la figure III.25b (courbe bleue) et une perturbation angulaire donnée par la figure III.25c (courbe bleue). On remarque que l'introduction d'une dynamique d'actionneur (courbes rouges) ne modifie pas l'amplitude de perturbations générée au final. On peut donc déterminer l'amplitude Γ_p des échelons de couples à générer pour obtenir une perturbation d'amplitude A atteinte en une période ΔT . En effet, en supposant que la dynamique de rotation sur un axe Θ_i est donnée par :

$$I_{\Theta_i} \cdot \dot{\Omega}_{\Theta_i} = \Gamma_{p_i} \quad (\text{III.76})$$

La matrice d'inertie est supposée ici diagonale, et I_{Θ_i} correspond au terme de la diagonale correspondant à l'axe Θ_i ($I_{\Theta_1} = I_{xx} = I_{1,1}$, $I_{\Theta_2} = I_{yy} = I_{2,2}$ et $I_{\Theta_3} = I_{zz} = I_{3,3}$). Si on suppose $\Gamma_{p_i}(t) = \Gamma_{p_i}$ constante pour $0 < t < \frac{\Delta T}{2}$ et $\Gamma_{p_i}(t) = -\Gamma_{p_i}$ constante pour $\frac{\Delta T}{2} < t < \Delta T$, on en déduit que :

$$\begin{aligned} & \iint_0^{\frac{\Delta T}{2}} \dot{\Omega}_{\Theta_i} dt^2 = \frac{A}{2} \\ \Leftrightarrow & \iint_0^{\frac{\Delta T}{2}} \frac{\Gamma_{p_i}}{I_{\Theta_i}} dt^2 = \frac{A}{2} \\ \Rightarrow & \Gamma_{p_i} = \frac{4 \cdot A \cdot I_{\Theta_i}}{\Delta T^2} \end{aligned} \quad (\text{III.77})$$

En appliquant cette formule, on détermine donc l'amplitude Γ_{p_i} de la perturbation selon le coefficient d'inertie correspondant à chacun des axes, ce qui permet de générer des perturbations parfaitement identiques pour chacun des contrôleurs à tester.

8.1.2 Rejet de perturbation d'attitude

Chaque contrôleur a donc été testé 5 fois pour chaque angle (roulis, tangage et lacet). Les figures III.26a et III.26b montrent donc, que le rejet de perturbation obtenu est hautement reproductible pour les deux contrôleurs (PI et quaternion). Par exemple, sur l'axe de roulis, 80% de la perturbation de 20° sont rejetés en 0.7 s par le contrôleur géométrique (quaternion), et la même perturbation est rejetée en 0.75 s par le contrôleur PI. Les deux contrôleurs se comportent donc sensiblement pareil en ce qui concerne les perturbations d'attitude.

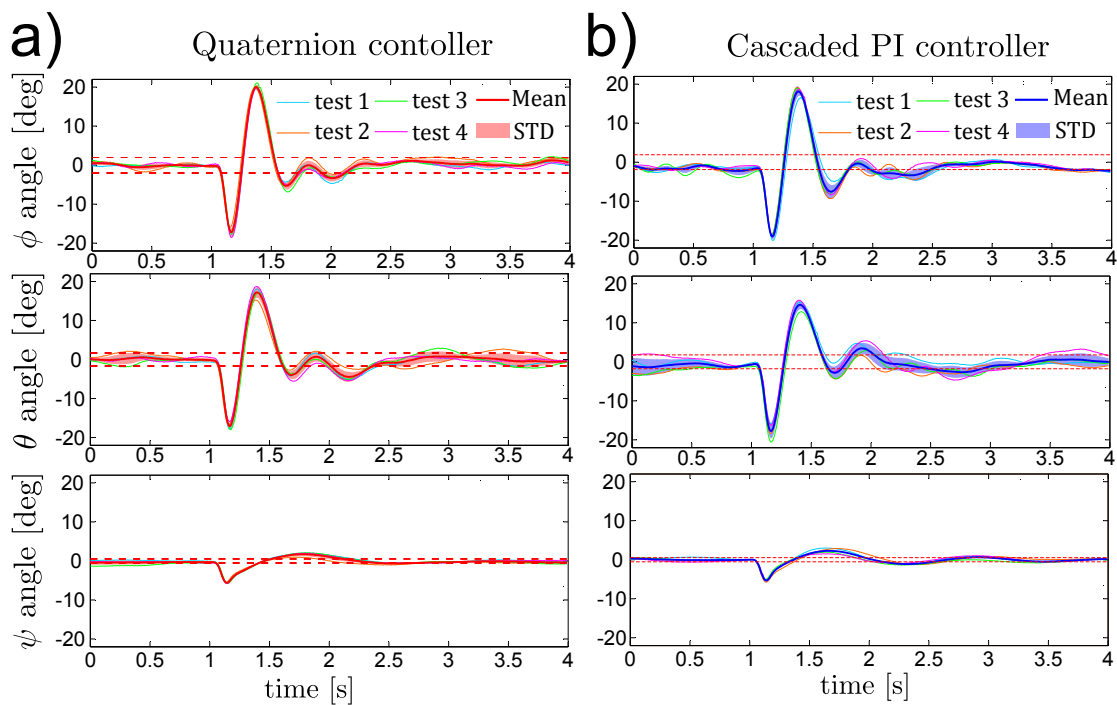


FIGURE III.26 – a) Rejets de perturbations obtenus avec le contrôleur géométrique (contrôleur basé sur les quaternions). La perturbation a été réalisée plusieurs fois et on affiche les résultats de quatre essais dans différentes couleurs. L'aire colorée en rouge transparent correspond à l'écart type des mesures par rapport à la trajectoire moyenne (courbe rouge). b) Rejets de perturbations obtenus avec le contrôleur utilisant des PI imbriqués. La perturbation a été réalisée plusieurs fois et on affiche les résultats de quatre essais dans différentes couleurs. L'aire colorée en bleu transparent correspond à l'écart type des mesures par rapport à la trajectoire moyenne (courbe bleue).

8.1.3 Poursuite des consignes d'attitude

Ensuite, on propose de réaliser plusieurs fois une trajectoire identique et de mesurer l'erreur de poursuite en attitude, obtenue pour les deux contrôleurs. Le robot décolle, puis après 15 secondes de vol stationnaire, on applique successivement des perturbations sur le roulis, le tangage puis le lacet. Ensuite, le robot effectue trois échelons de position de 1 mètre sur l'axe X et Y . Enfin, on réalise un cercle incliné de 45° autour de l'axe Y et d'un mètre de rayon à une vitesse totale de $1.0 \text{ m} \cdot \text{s}^{-1}$, avant d'atterrir automatiquement. Le scénario a été répété 4 fois pour chaque contrôleur, et les résultats sont présentés sur la figure III.27. On observe que le contrôleur géométrique (quaternion) offre de meilleurs résultats que celui basé sur des contrôleurs PI imbriqués. Les écarts type des erreurs de poursuite sont de 2.67° et 2.15° respectivement sur les axes de roulis et tangage pour le contrôleur géométrique, alors qu'elles sont de 3.05° et 4.14° avec les contrôleurs PI imbriqués. De même, l'erreur maximale enregistrée pour les contrôleurs PI imbriqués est de 20° alors qu'elle n'est que de 8° avec le contrôleur géométrique les valeurs sont données pour la trajectoire après. Les perturbations d'attitude, c'est à dire à partir de $t = 25\text{s}$).

8.2 Précision de positionnement

Ici, nous allons donc évaluer les précisions de positionnement que l'on peut atteindre avec X4-MaG dans l'arène de vol, lorsque le robot est localisé par le système Vicon. Les précisions de positionnement sont obtenues en utilisant le contrôleur de position décrit par (III.72) et en choisissant comme contrôleur d'attitude le contrôleur géométrique décrit par (III.70).

Pour évaluer cette précision, une première série d'essai est réalisé : on fait décoller le robot et on lui demande d'effectuer un vol stationnaire pendant 45 secondes. Cela est répété en tout 5 fois afin d'extraire des données statistiques plus fiables. La figure III.28 montre donc que X4-MaG est capable de se positionner très précisément lors de vol stationnaire. En effet, lors d'un vol stationnaire, l'erreur maximale de positionnement par rapport à la position demandée est de 3 cm sur les axes X et Y et l'écart type est de seulement 0.77 cm. L'erreur sur l'axe Z est, quant à elle, encore plus petite étant donné que l'erreur maximale est de quelques millimètres et l'écart type inférieur au millimètre. Ces très bonnes performances sur l'axe Z sont principalement dues à l'asservissement en vitesses des rotors, qui permet de maîtriser parfaitement la poussée totale générée.

Ensuite, on évalue les performances du robot en ce qui concerne la poursuite de trajectoire. La figure III.29a montre donc les réponses du robot pour des déplacements sur les axes X , Y et Z répétés lors de 5 tests différents. L'écart type de l'erreur de poursuite obtenu est de 1.67 cm pour l'axe X , 1.13 cm pour Y et de seulement 0.12 cm sur l'axe Z . On observe que les trajectoires obtenues sont très reproductibles (grâce au préfiltre de consigne qui génère toujours une trajectoire réalisable) et que la précision obtenue est bonne. Ensuite, on évalue les performances du robot lors d'une trajectoire

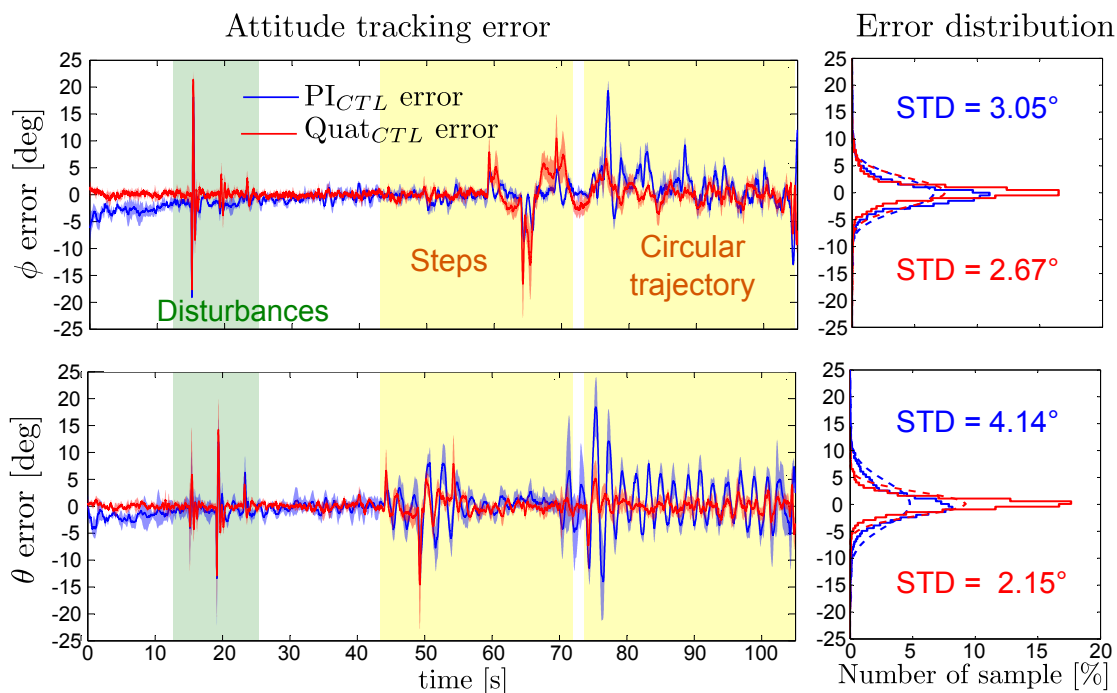


FIGURE III.27 – A gauche, l’erreur de poursuite obtenue pour chaque contrôleur durant une trajectoire complexe (décollage, vol stationnaire, perturbation d’attitude, déplacements latéraux, cercle incliné et atterrissage). La trajectoire a été répétée 4 fois pour chaque contrôleur. Les lignes rouges et bleues représentent l’erreur moyenne pour chaque instant de la trajectoire, pour le contrôleur géométrique et les PI imbriqués respectivement. L’aire transparente représente l’écart type de l’erreur de poursuite par rapport à la valeur moyenne. A droite, on représente par des lignes continues la répartition de l’erreur d’attitude, et les courbes en pointillés sont les gaussiennes correspondantes (bleue pour les PI imbriqués et rouge pour le contrôleur géométrique.)

circulaire, comme le montre la figure III.29b. Dans ce cas, le robot effectue trois cercles d’un mètre de rayon, et inclinés de 45° par rapport à l’horizontal à une vitesse totale de $0.8 \text{ m} \cdot \text{s}^{-1}$. Dans ce cas, le robot a été capable de poursuivre assez fidèlement la trajectoire demandée, étant donné que l’erreur de poursuite est de l’ordre de 2.9 cm (toujours en terme d’écart type).

Des essais supplémentaires ont été réalisés pour une vitesse un peu plus élevée. On résume les précisions obtenues pour l’ensemble des essais dans la table III.5. On remarque notamment que pour des trajectoires circulaires, le robot est capable à des vitesses de $1.2 \text{ m} \cdot \text{s}^{-1}$ d’être précis à 3.2 cm en terme d’écart type, pour une erreur maximale de 8.3 cm (ce qui est inférieur à sa demi envergure). Ces résultats sont similaires à ceux obtenus par exemple par [Michael *et al.*, 2010] et [Mellinger *et al.*, 2012] avec la plateforme Hummingbird (500 grammes environ) vendue par Ascending Technologies ([Ascending Technologies GmbH, 2011]).

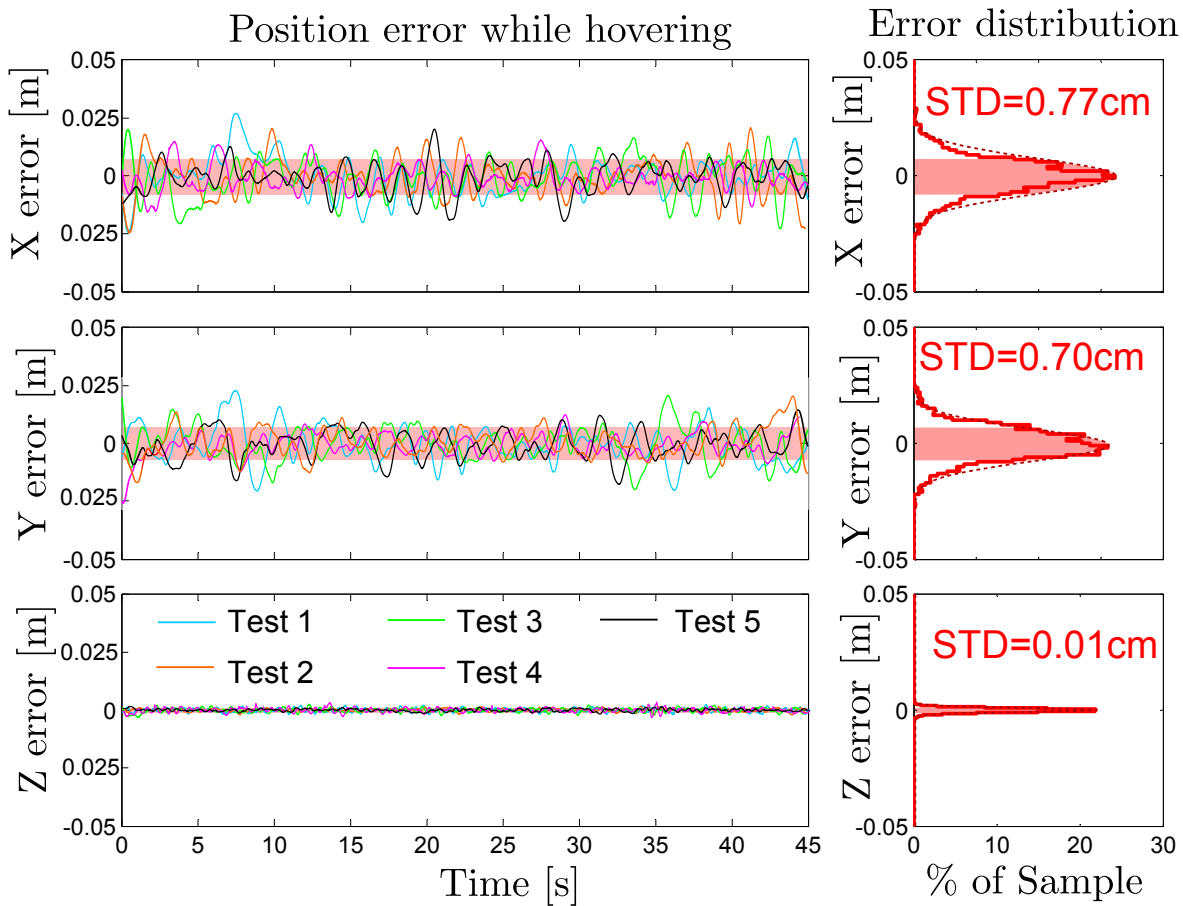


FIGURE III.28 – A gauche, l'évolution des positions X , Y et Z de X4-MaG lors de 5 tests en vol stationnaire. Chaque test est affiché dans une couleur différente, et on présente l'écart type sous la forme d'une zone rouge transparente. L'association du contrôleur de position décrit par (III.72) et (III.74) avec le contrôleur d'attitude géométrique, permet d'atteindre des précisions de positionnement en 3D tout à fait satisfaisantes, étant donné que le robot est capable de se maintenir dans une ellipsoïde de 3.0 cm de rayon sur le plan horizontal et de seulement quelques millimètres sur le plan vertical. D'après [Manecy et al., 2015].

9 Conclusion

9.1 La plateforme X4-MaG

Durant ce chapitre, j'ai donc expliqué, en détail, les différents choix de conception et algorithmes de contrôle du quadrirotor X4-MaG. Cette nouvelle plateforme open-source et open-hardware peut être assemblée pour un coût d'environ 500€. Elle peut être pilotée directement par un système de télécommande de modélisme ou évoluer de façon entièrement autonome entre des coordonnées fixées par une station sol. La programmation du pilote automatique est grandement simplifiée et est entièrement faite sous Matlab/Simulink, grâce à la nouvelle toolbox open source que j'ai développée (RT-MaG). De plus, l'architecture logicielle et matérielle a été conçue de façon à fournir des modes de secours

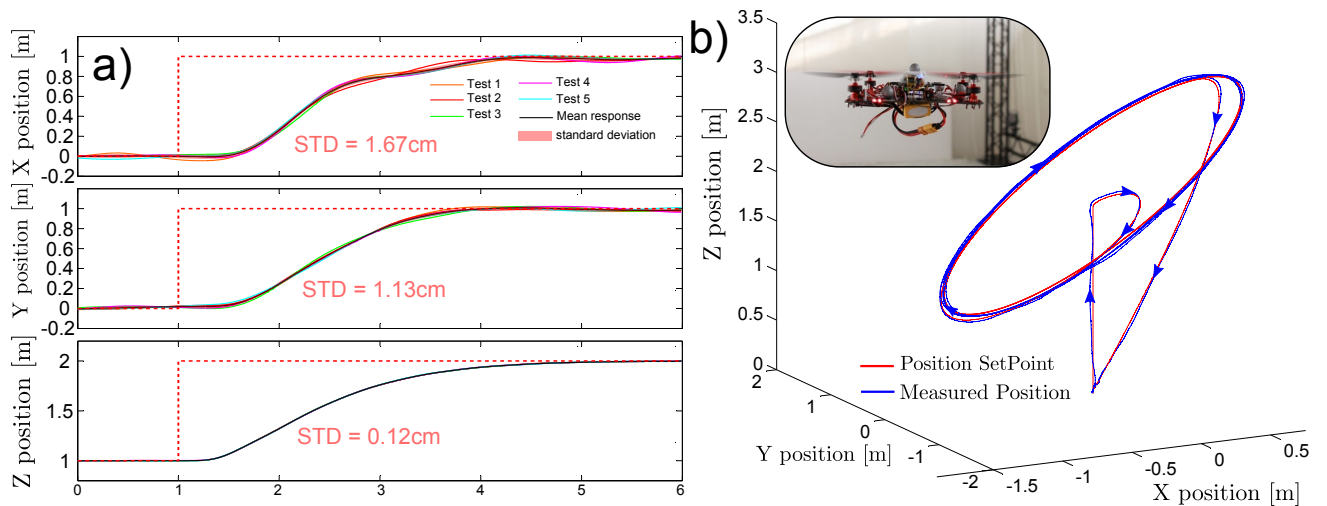


FIGURE III.29 – a) Réponses à des échelons de positions selon les trois axes lors d’essais répétés 5 fois. Chaque test est affiché dans une couleur différente : la ligne noire représente la réponse moyenne obtenue pour les 5 essais et l’aire rouge représente l’écart type par rapport à la trajectoire moyenne. b) Exemple de la réalisation d’une trajectoire circulaire inclinée de 45° par rapport à l’horizontale, pour une vitesse de translation de $0.8 \text{ m} \cdot \text{s}^{-1}$. En rouge la trajectoire demandée et en bleu la trajectoire réalisée par le robot. D’après [Manecy et al., 2015].

Flying Mode		Max Error	Mean Error	STD	
Hover	x	2.8cm	0.03cm	0.7cm	
	y	2.7cm	0.02cm	0.6cm	
	z	0.4cm	0.00cm	0.3cm	
Path Following	$0.8 \text{ m} \cdot \text{s}^{-1}$	x	5.3cm	0.04cm	1.7cm
		y	6.7cm	0.02cm	2.9cm
		z	3cm	0.14cm	1.5cm
Path Following	$1.2 \text{ m} \cdot \text{s}^{-1}$	x	8.3cm	0.3cm	2.3cm
		y	7.0cm	0.7cm	3.2cm
		z	6.6cm	0.5cm	2.0cm

TABLE III.5 – Précisions de positionnement du quadrirotor X4-MaG dans différents modes de vol.

en cas de défaillance à plusieurs niveaux. J’ai alors proposé une architecture simple, pour doter le calculateur haut niveau d’un mode de secours, permettant de faciliter l’essai de nouveaux contrôleurs tout en limitant le risque de crash dans l’arène de vol. Le contrôleur de secours est alors un contrôleur fiable et validé, reposant sur les mesures du système Vicon, qui reprend automatiquement la main, si le contrôleur expérimental ne parvient pas à atteindre les trajectoires à réaliser. De même, une fonction fail-safe, permet de maintenir une stabilisation d’attitude et de position minimale (basée sur

la centrale inertielle) via le ordinateur bas-niveau, en cas de défaillance du ordinateur haut niveau (pilote automatique). J'ai également mis à disposition l'ensemble des développements réalisés sur cette plateforme : la documentation, les tutoriels, les instructions de montages, et l'ensemble de la suite logicielle qui sont disponibles sur un site internet dédié à ce projet : <http://www.gipsa-lab.fr/projet/RT-MaG/>.

En ce qui concerne les algorithmes de contrôle, j'ai proposé une formulation du filtre complémentaire en quaternions, ainsi qu'une évolution d'un contrôleur géométrique d'attitude, basé également sur les quaternions. Les performances de ce contrôleur ont alors été comparées avec celles d'un contrôle plus classique reposant sur des boucles imbriquées de contrôleurs PI.

J'ai ensuite démontré la fiabilité de la plateforme ainsi que quantifié les performances atteignables en terme de positionnement. Le robot est donc capable d'effectuer des vols stationnaires avec une précision de 0.8 cm (écart type) et de poursuivre des trajectoires complexes avec une précision de 3.2 cm (écart type) pour des vitesses de $1.2 \text{ m} \cdot \text{s}^{-1}$.

9.2 Perspectives

J'ai donc proposé et mis en place une première version du quadrirotor X4-MaG ainsi que la suite logicielle permettant de facilement tester et intégrer de nouveaux algorithmes. Mais dans sa version actuelle, le robot présente encore quelques limitations qui sont principalement liées aux capteurs qu'il embarque. Il serait donc intéressant de le faire évoluer, afin d'étendre son champ d'application, en intégrant de nouveaux capteurs. Une prochaine version pourrait, par exemple, intégrer des capteurs supplémentaires tel qu'un magnétomètre 3 axes, un module GPS, et un capteur d'altitude (baromètre ou autre), ce qui permettrait au robot d'évoluer également en extérieur.

Cette plateforme permettra donc de tester plus tard des stratégies de stabilisations et de guidage bio-inspirés comme celles décrites dans [Expert et Ruffier, 2012], [Roubieu *et al.*, 2014] ou encore [Expert et Ruffier, 2015]. Ces stratégies reposent sur des capteurs visuels minimalistes et légers tels que CurvACE qui est le premier œil artificiel fonctionnel reproduisant l'œil composé de l'insecte [Floreano *et al.*, 2013, Viollet *et al.*, 2014]. [Mafrica *et al.*, 2015] propose une rétine avec de fortes capacités d'adaptation à la lumière ce qui permet d'envisager de naviguer dans des environnements avec de fortes variations lumineuses. La validation de telles stratégies peut être un premier pas pour des intégrations futures pour des robots-insectes (comme par exemple [de Croon *et al.*, 2009] ou [Ma *et al.*, 2013]), ne reposant que sur des capteurs embarqués minimalistes et de petites tailles.

Une première application de ce genre de capteurs minimalistes va être décrite dans la suite de ce manuscrit avec la description du capteur dans le chapitre suivant (chapitre IV). Puis, nous montrerons que ce capteur peut être utilisé efficacement, pour estimer finement la position d'une cible, afin d'asservir entièrement en attitude et en position le robot X4-MaG, et ce, en se passant des données accélérométriques (voir chapitre V).

Chapitre IV

Design d'un système occulo-moteur bio-inspiré

Sommaire

1	Introduction à la vision de l'insecte	103
1.1	Contexte et objectifs	103
1.2	L'œil de l'insecte	104
1.3	Hyperacuité et vibration	107
1.4	L'œil du robot	108
2	Description de l'œil	109
2.1	Principes de fonctionnement	109
2.2	Caractérisation et réglages optiques	110
2.3	Design du mécanisme de vibration	113
2.4	Conception du mécanisme de découplage	121
2.5	Conception de la carte de traitement des signaux visuels	125
3	Traitement des signaux visuels	129
3.1	Traitement d'une paire de photo-récepteurs	129
3.2	Fusion multi-paires d'un œil	139
3.3	Fusion des 4 yeux	146
3.4	Fusion généralisée pour n yeux	149
4	Simulateur du système occulo-moteur	152
4.1	Modélisation générique du capteur	152
4.2	Modélisation complète du système occulo-moteur	155
4.3	Asservissement visuel	157
4.4	Résultats de simulations du système occulo-moteur	158
5	Résultats expérimentaux	163

5.1	Mesure de position angulaire	164
5.2	Poursuite de cible	166
6	Conclusion du chapitre	169

1 Introduction à la vision de l'insecte

Ce chapitre présente la mise en œuvre d'un système occulo-moteur bio-inspiré complet qui permet de reproduire le découplage entre la tête et le corps existant chez la plupart des êtres vivants. Il a été montré dans le chapitre II que ce découplage est probablement un des points clés de la stabilisation en vol des insectes. Ce chapitre décrit donc la conception et la validation du système de vision (qui sera appelé "l'œil" tout au long de ce chapitre) et qui vise à reproduire les principes de base de la vision de l'insecte, à savoir :

- Une sensibilité gaussienne de chacun des photo-récepteurs,
- Un décodage de l'information basé sur des paires de photo-récepteurs adjacents,
- La présence d'une vibration rétinienne permettant de mettre en œuvre une hyperacuité lors de la localisation d'objets contrastés.

1.1 Contexte et objectifs

Le but de ce système occulo-moteur n'est pas de reproduire en tout point de vue la vision de l'insecte, mais d'en réutiliser les principes considérés comme clés, afin de les adapter aux besoins de la robotique. Les travaux de cette thèse s'appuient sur une technologie de capteurs développés durant ces dix dernières années. Cette technologie de capteurs, appelé VODKA (pour "Vibrating Optical Device for the Kontrol of Autonomous robots"), est présentée en détail dans [Kerhuel *et al.*, 2012] et [Kerhuel, 2009] et certaines évolutions sont disponibles dans [Juston *et al.*, 2014] et [Juston, 2013]. Ces capteurs réutilisent des principes fondamentaux observés chez l'œil de l'insecte ou de l'humain (voir section 1.2) afin de les doter d'hyperacuité (voir section 1.3) et sont capables de localiser un contraste de type barre ou front (voir figure IV.1) avec seulement une paire de pixels [Violet, 2014]. Le système de vision présenté ici, combine quatre yeux, chacun composé de 6 pixels offrant donc 4×5 paires de pixels.

Le principal intérêt d'utiliser des capteurs "minimalistes" de ce type est qu'ils fournissent une mesure très précise avec un très faible nombre de pixels (24 pixels dans notre cas.) Le faible nombre de pixels à traiter va donc permettre d'atteindre des taux de rafraîchissement élevés (de l'ordre de 1 kHz) pour des charges de calculs faibles. Un haut taux de rafraîchissement est essentiel dans une application robotique, afin de faire fonctionner efficacement les différentes boucles de contrôle et les différents estimateurs, permettant au système d'être plus réactif et dynamique.

Les principales contributions de mon travail de thèse, outre quelques améliorations et modifications des capteurs permettant d'augmenter leur bande passante, résident en :

- l'élaboration d'une procédure de calibration robuste, permettant une linéarisation parfaite des sorties de chaque paire de pixels (voir section 3.1),
- le développement d'un algorithme de fusion multi-paire permettant d'augmenter le champ visuel de chaque capteur (voir section 3.2),

- une combinaison originale de quatre capteurs élémentaires permettant, à l'aide d'une fusion appropriée (voir section 3.3), d'étendre la localisation de cible du cas 1-D (une dimension) au cas 2-D (localisation de la cible selon deux directions) et de donner une mesure de son orientation (angle de lacet). Le capteur est ainsi capable de localiser précisément la position angulaire d'une cible selon 3 directions : azimut, élévation et orientation de la cible.

Le principal but du système occulo-moteur est donc d'offrir au robot la possibilité de suivre du regard une cible, et ce indépendamment des mouvements réalisés par le robot. Autrement dit, le système de vision doit être complètement découplé du corps du robot de façon à permettre au robot de se mouvoir tout en gardant son œil fixé sur sa cible. Les cibles utilisées sont des cibles de type croix ou mire quatre quadrants (figures IV.1a et IV.1b respectivement).

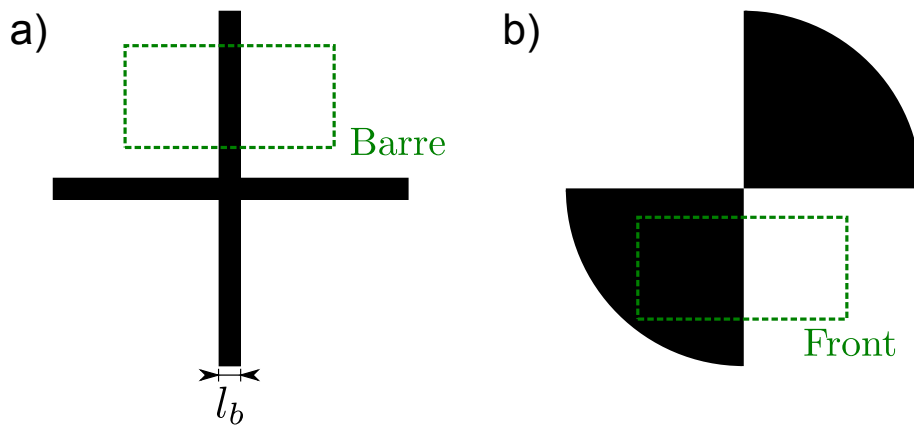


FIGURE IV.1 – a) Une cible de type *croix* composée de deux barres orthogonales de largeur l_b . b) Un cible de type *mire quatre quadrants* qui fait apparaître quatre fronts.

1.2 L'œil de l'insecte

La vision des insectes a été largement étudiée dans la littérature [Horridge, 1977, Franceschini et Kirschfeld, 1971, Snyder, 1979, Land et Collett, 1997, Stavenga, 2003a, Stavenga, 2003b]. Il existe également de nombreux ouvrages tels que *Animal Eyes* [Land et Nilsson, 2012] et très récemment *Visual Ecology* [Cronin et al., 2014] qui permettent d'aborder les grands principes liés à la vision des arthropodes.

1.2.1 L'œil composé

L'œil des insectes, que l'on nomme souvent œil composé ou encore œil à facettes, a la particularité d'être composé de plusieurs petits yeux élémentaires que l'on appelle *ommatidies* et qui regardent chacun dans des directions différentes (voir IV.2). Le nombre d'*ommatidies* varie selon les espèces, pouvant aller de quelques dizaines chez certaines mouches à plusieurs dizaines de milliers chez certaines libellules.

Les propriétés de chaque *ommatidie* peuvent également varier selon la région de l'œil, permettant ainsi une spécialisation de certaines régions de l'œil. Chez la mouche par exemple, la zone frontale des mâles présente un resserrement des axes optiques des différentes *ommatidies*, ce qui a pour effet d'améliorer l'acuité visuelle de cette zone. Cela permet notamment au mâle de repérer plus efficacement une femelle et d'améliorer ses capacités de poursuite lors du vol.

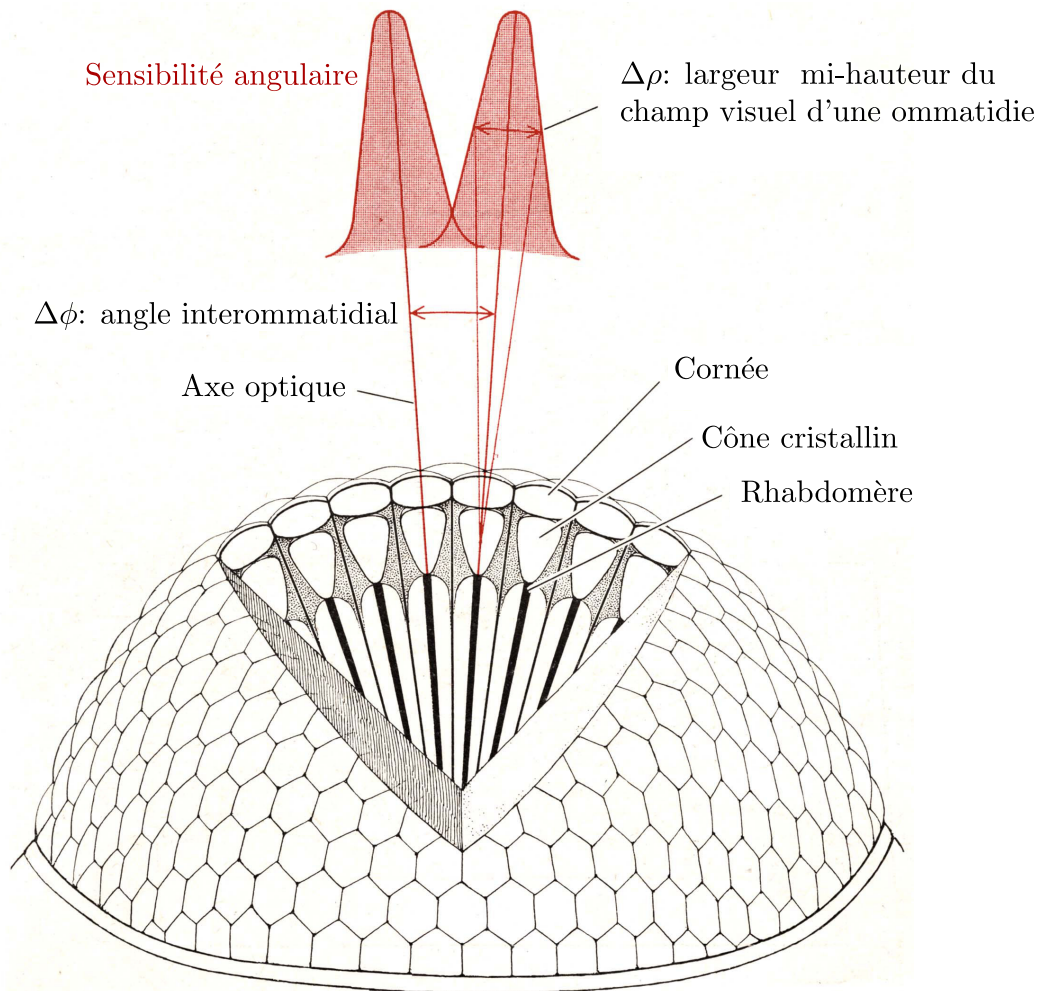


FIGURE IV.2 – Structure de l'œil composé de l'insecte. Chaque ommatidie est construite sur le même modèle avec la cornée, le cône cristallin réalisant une diffraction et le rhabdomère qui constitue la partie photosensible de l'œil et qui agit comme un guide d'onde. La combinaison de la diffraction et du guide d'onde confère à l'œil une sensibilité gaussienne de largeur à mi-hauteur $\Delta\rho$. La séparation des axes optiques est appelée "angle *inter-ommatidial*" et est noté $\Delta\phi$. Adapté à partir de [Horridge, 1977].

1.2.2 Sensibilité angulaire et échantillonnage spatial

Ce qui va nous intéresser dans ces travaux, ce sont les propriétés optiques des *ommatidies* de l'œil de l'insecte. En effet, les principes de fonctionnement de ces petits yeux élémentaires vont à l'encontre

de ce que l'on recherche en général pour un capteur optique. On constate notamment que la sensibilité angulaire de chaque *ommatidie* est gaussienne, ce qui traduit le fait que l'œil voit flou. Cette sensibilité gaussienne de l'œil résulte de deux phénomènes successifs :

- **Une diffraction** réalisée par le cône cristallin qui dépend du diamètre D de l'optique et de la longueur d'onde λ en question. Cette diffraction peut être assimilée à une sensibilité gaussienne de largeur à mi hauteur $\Delta\rho_d = \frac{\lambda}{D}$ comme décrit par [Snyder, 1979].
- **Un effet guide d'ondes** réalisé par les *rhabdomères* de chaque *ommatidie* qui mène également à une sensibilité gaussienne de largeur à mi-hauteur $\Delta\rho_r$.

La sensibilité angulaire finale $\Delta\rho$ de chaque *ommatidie* résulte donc de la convolution de ces deux phénomènes [Franceschini et Kirschfeld, 1971, Snyder, 1979, Land et Collett, 1997], et peut également être décrite par une loi de distribution gaussienne de largeur à mi hauteur $\Delta\rho \approx \sqrt{\Delta\rho_d^2 + \Delta\rho_r^2}$.

Chez la mouche, l'angle d'acceptance $\Delta\rho$ est en général proche de 4° , et permet, grâce au flou qui en résulte, d'effectuer un filtrage spatial passe-bas visant à supprimer les détails visuels superflus. L'échantillonnage spatial donnant lieu à un champ visuel plus large découle quant à lui d'un deuxième paramètre optique : l'angle *inter-ommatidial* qui n'est autre que l'angle qui sépare les axes optiques de deux *ommatidies* voisines. Cet angle *inter-ommatidial* est en général également de l'ordre de 4° . On peut donc dire que l'œil à "facettes" ou l'œil composé de la mouche contient autant d'éléments d'image (pixels) qu'il contient d'*ommatidies*.

On remarque cependant, que l'échantillonnage spatial de l'œil de l'insecte est assez grossier. Il existe donc d'autres mécanismes qui lui permettent d'accomplir les prouesses réalisées en vol comme l'évitement d'obstacle, l'atterrissage de précision, etc. La piste explorée dans ces travaux est l'hyperacuité (voir section 1.3), qui munirait l'œil de l'insecte d'une acuité bien supérieure à celle initialement contrainte par les paramètres optiques. C'est cette hyperacuité que l'on cherchera à reproduire au sein de l'œil artificiel décrit tout au long de ce chapitre.

1.2.3 Traitement de l'information visuelle

Dans cette partie, nous allons décrire brièvement comment l'information visuelle est codée chez l'insecte et surtout quels sont les mécanismes qui permettent d'en extraire les informations utiles.

Après avoir traversé la cornée et le cône cristallin, la lumière entre dans le *rhabdomère* où a lieu la photo-transduction (l'étape de codage traduisant l'absorption des photons en potentiel électrique). Mais cette photo-transduction est plus ou moins inhibée par un phénomène de migration de granules pigmentaire ([Kirschfeld et Franceschini, 1969, Lo et Pak, 1981, Satoh *et al.*, 2014]) permettant une auto-adaptation à la luminosité ambiante. Ce phénomène peut être assimilé à un filtrage passe-haut de fréquence de coupure très faible (0.1 Hz, [Franceschini, 1999]) et joue un rôle très important, rendant l'œil robuste aux variations d'éclairement. Une fois l'information lumineuse traduite en signaux

électriques, elle est transmise à des neurones constituant les ganglions optiques organisés en quatre structures distinctes : *lamina*, *medulla*, *lobula* et *lobula-plate*. Un autre filtrage passe-haut est alors réalisé par la *lamina* (neurone distal), qui permet de ne garder que les parties dynamiques des signaux visuels, plus utiles à la détection du mouvement ([Heisenberg, 1971, Juusola et French, 1997]).

Ces différentes étapes de traitement de l'information visuelle, vont être reproduites par le circuit électronique qui traite les signaux de sortie des photodiodes de l'œil artificiel (voir section 2.5.2).

1.3 Hyperacuité et vibration

L'hyperacuité a été définie par G. Westheimer en 1981 [Westheimer, 1981] puis reformulée en 2009 [Westheimer, 2009], comme étant la capacité d'un capteur à mesurer ou percevoir au delà de ses limites structurelles. C'est à dire, pour un capteur visuel, être capable de localiser un trait avec une résolution plus fine que celle imposée par l'angle interrécepteurs (angle $\Delta\phi$ de la IV.2). Autrement dit, la résolution du capteur est plus fine que celle imposée par l'échantillonnage spatial dudit capteur.

L'origine et les mécanismes de l'hyperacuité peuvent faire l'objet de longues discussions et il reste encore des questions sans réponses. Cependant, de nombreux invertébrés, telle que le crabe [Sandeman, 1978], le mollusque [Land, 1982] ou encore les araignées [Land, 1969] et [Kaps et Schmid, 1996], présentent un vibration rétinienne (ou balayage rétinien) qui semble jouer un rôle important vis-à-vis de l'hyperacuité. De même, les micros mouvements rétiniens ont fait l'objet d'étude chez l'homme, [Ko et al., 2010], visant à démontrer le lien existant entre micro-saccades et hyperacuité. Toutes ces études tendent à montrer, qu'en plus de prévenir l'adaptation des cellules des photo-récepteurs (adaptation visuelle, également appelé *phénomène de fondu*), les micro-mouvements rétiniens joueraient un rôle plus important, notamment dans l'hyperacuité [Fahle, 1991]. Une revue concernant les micro-saccades est proposée par [Martinez-Conde et al., 2013].

Les micro-mouvements présents dans l'œil de la mouche (voir synthèse de [Viollet, 2014]) ont été étudiés par [Hengstenberg, 1972] ou encore [Franceschini et Chagneux, 1997] et résultent de l'activité d'un muscle générant des mouvements de rétine (voir IV.3c. L'amplitude de ces micro-mouvements varie en fonction de la fréquence du signal d'excitation venant du système nerveux. La fréquence d'excitation du muscle varie de 50 Hz à 120 Hz (IV.3b) donnant lieu à une rotation des axes optiques de quelques degrés. [Hengstenberg, 1972] parle de 0.1-0.5°/10 Hz, ce qui amènerait à une vibration de 0.5-2.5° pour une fréquence d'excitation du muscle de 50 Hz comme observée par exemple chez la mouche *Calliphora erythrocephala* par [Kuiper et Leutscher-Hazelhoff, 1965]. La fréquence de vibration de la rétine obtenue est par contre plus faible, de l'ordre de quelques Hertz. Ce balayage visuel a été reproduit sur l'œil artificiel représenté par la IV.3e. Contrairement à l'insecte, ce n'est pas la rétine qui est mise en rotation, mais l'optique (c'est à dire la lentille). Cependant le phénomène de rotation des axes optiques obtenu est le même, et la translation des lentilles a été réglée pour obtenir une amplitude de vibration de 4° et une fréquence de 55 Hz (ce qui est beaucoup plus élevé que chez l'insecte mais qui

permet d'améliorer le taux de rafraîchissement du capteur).

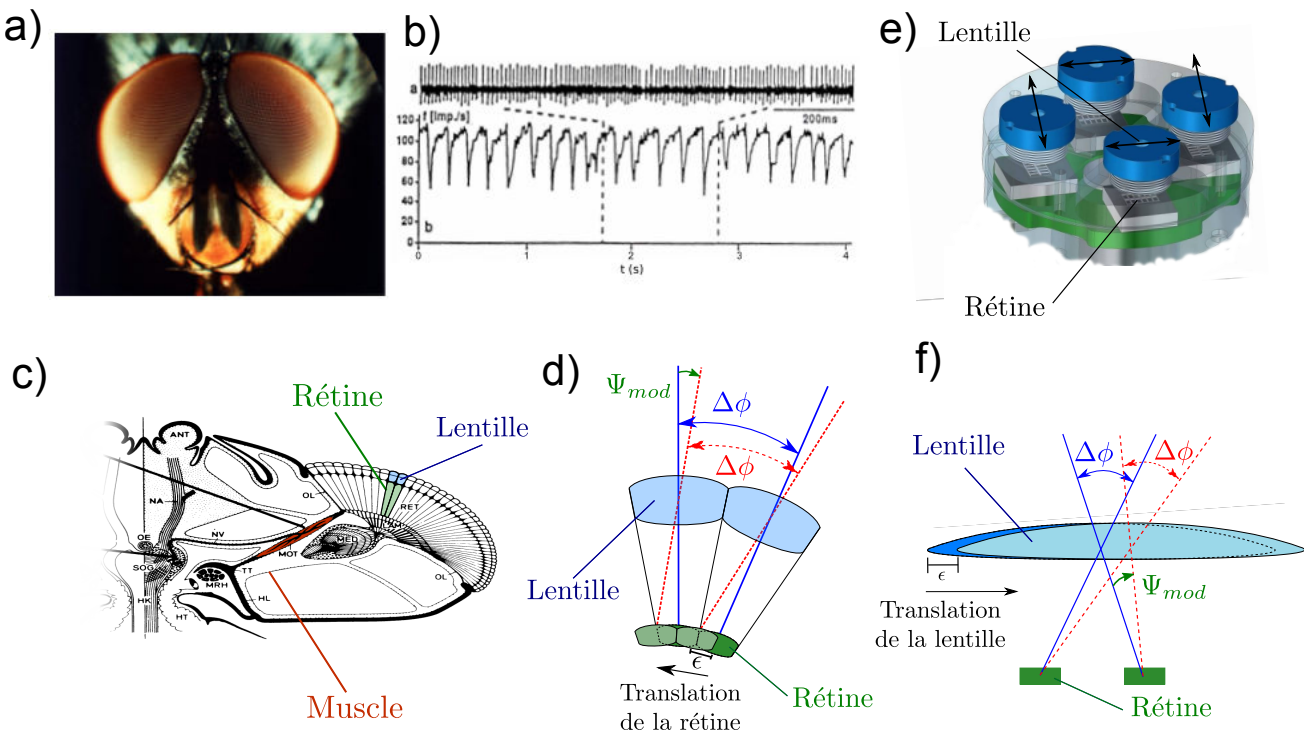


FIGURE IV.3 – a) Photographie des yeux composés d'une mouche domestique. b) Les signaux correspondant à l'activité du muscle responsable de la vibration rétinienne de l'œil d'une mouche domestique (d'après [Franceschini et Chagneux, 1997]). c) Coupe de la tête d'une mouche, vue du dessus, montrant le muscle orbito-tentorialis (MOT en rouge) responsable de la vibration rétinienne (adaptation de [Hengstenberg, 1972]). d) Schématisation de la rotation des axes optiques résultant de la vibration de la rétine chez la mouche. e) CAO de l'œil artificiel mettant en évidence la rotation des axes optiques de chaque œil grâce à la rotation du support des lentilles. f) Schéma représentant la rotation des axes optiques induite par la translation de la lentille au dessus du capteur LSC.

1.4 L'œil du robot

L'œil du robot qui a été réalisé pendant mes travaux de thèse vise à reproduire les paramètres optiques décrits précédemment (voir section 1.2). Pour cela, l'architecture du capteur doté d'hyperacuité décrit par [Kerhuel *et al.*, 2012], a été reprise et quelques modifications ont été apportées à cet élément de base pour augmenter son champ visuel.

Initialement le capteur de type VODKA développé par [Kerhuel *et al.*, 2012] était constitué d'une seule paire de photo-récepteurs (photodiodes) assimilable à 2 *ommatidies*. Ce capteur reposait alors sur une matrice de photodiodes (LSC) de chez iC-Haus dont seules les 4 photodiodes du milieu étaient utilisées et reliées 2 à 2 pour améliorer la sensibilité du capteur ; donnant finalement accès à une paire

de photodiodes. Cette paire de photodiodes était ensuite placée sous une optique¹ volontairement dé-focalisée afin de reproduire les propriétés de flou observées chez la mouche. Puis l'œil était animé d'une vibration, donnant lieu à une rotation des axes optiques de quelques degrés, qui une fois accompagnée d'un traitement adapté, donne lieu à une hyperacuité (voir figure de principe IV.3f).

L'œil présenté ici, reprend donc ce principe, mais exploite cette fois-ci, toute la matrice de photodiodes offerte par le capteur LSC de chez iC-Haus. Comme le montre la figure IV.4, chacun des 6 pixels de l'œil est constitué de 2 photodiodes reliées ensemble (améliorant ainsi la sensibilité) placé sous une unique optique délibérément dé-focalisée faisant apparaître les propriétés de flou souhaitées. L'hyperacuité est obtenue grâce à une translation de l'optique au dessus de la matrice de LSC, qui génère une rotation des axes optiques.

Dans la suite de ce chapitre, je décrirai le système occulo-moteurs qui a été développé pendant ma thèse et qui consiste en la combinaison de quatre yeux élémentaires soumis à une vibration périodique. La section 2 décrit en détail la conception de la partie mécanique et électronique de l'œil ainsi que les procédures de réglages permettant d'atteindre les paramètres optiques souhaités. Ensuite la section 3 expose les traitements qui permettent de passer des signaux photodiodes à une localisation de cibles selon 3 orientations. Un simulateur ainsi qu'un modèle simplifié de l'œil seront présentés en section 4 afin de mettre en place les différents asservissements liés à la vision. Et enfin, les résultats expérimentaux concernant la localisation et la poursuite de cible seront abordés dans la section 5 ainsi que la robustesse du système face à différents paramètres (vitesse de défilement de la cible, luminosité et angle de lacet de la cible).

2 Description de l'œil

2.1 Principes de fonctionnement

L'œil artificiel qui a été élaboré durant mes travaux de thèse, consiste en la combinaison de quatre capteurs dotés d'hyperacuité inspirés par les travaux de [Kerhuel *et al.*, 2012]. Ce type de capteur est capable de localiser avec une grande précision ($<0.1^\circ$) une cible de type front de contraste ou barre. L'idée ici, est de combiner quatre de ces capteurs de façon à pouvoir localiser une cible (de type croix ou mire quatre quadrants comme sur la figure IV.1) selon deux directions (ϕ et θ) et également remonter à son orientation ψ comme représenté par la figure IV.5a.

Cet œil est ensuite monté sur un système à double cardans imbriqués permettant de l'orienter selon deux directions afin de reproduire le découplage tête-corps existant chez la plupart des êtres vivants dotés de vision. Ce découplage est décrit par la figure IV.6.

1. L'optique est celle du capteur *CMOS Camera Module - 640 × 480 pixels SEN-00637* dont l'imageur initial a été retiré.

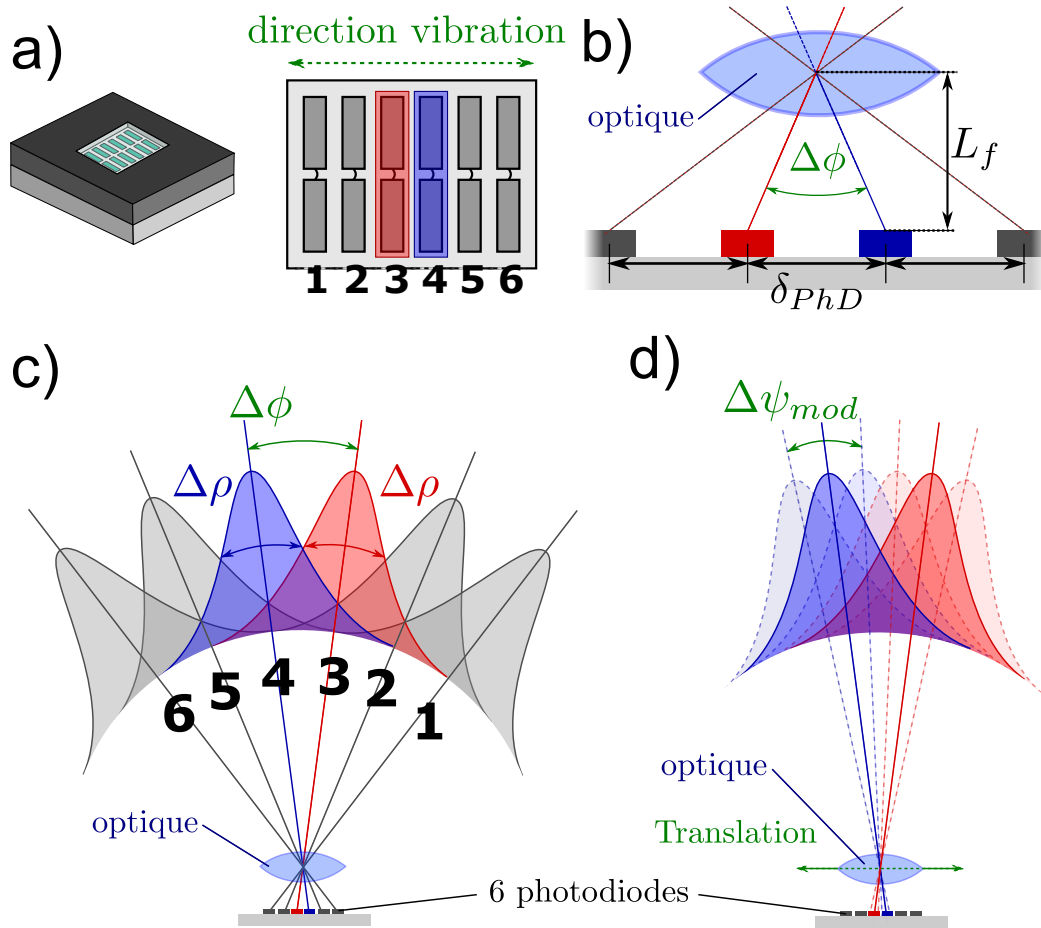


FIGURE IV.4 – Schéma de principe de l’œil de 6 pixels. a) Le capteur LSC de chez iC-Haus. La matrice de photodiodes du capteur. Les photodiodes sont reliées 2 à 2 verticalement de façon à améliorer la sensibilité du capteur. b) Distances optiques caractéristiques du capteur. L_f est la distance focale de la lentille, et δ_{PhD} est la distance séparant deux cellules photo-réceptrices (égale à $420\ \mu\text{m}$). c) Représentation schématique des paramètres optiques d’un œil de 6 pixels mettant en évidence l’angle d’acceptance $\Delta\rho$, et l’angle inter-ommatidial $\Delta\phi$. d) Représentation schématique de l’amplitude de vibration $\Delta\psi_{mod}$ obtenue par translation de l’optique.

2.2 Caractérisation et réglages optiques

De façon à reproduire au mieux la vision de l’insecte, il est nécessaire de se rapprocher le plus possible des paramètres optiques de l’œil composé de la mouche, c’est à dire un angle d’acceptance $\Delta\rho$ de 4° ici égal à l’angle inter-ommatidial $\Delta\phi$.

Pour ajuster ces paramètres, la seule variable d’ajustement est la distance entre la lentille et le capteur (noté L_f). Malheureusement, le fait qu’il n’y ait qu’une seule variable d’ajustement entraîne un couplage de ces deux paramètres, ne permettant pas de les régler de façon indépendante. La propriété de flou est donc obtenue en dé-focalisant volontairement la lentille, c’est-à-dire en réduisant ou en augmentant la distance focale par rapport à celle préconisée par les constructeurs de la lentille. Ce

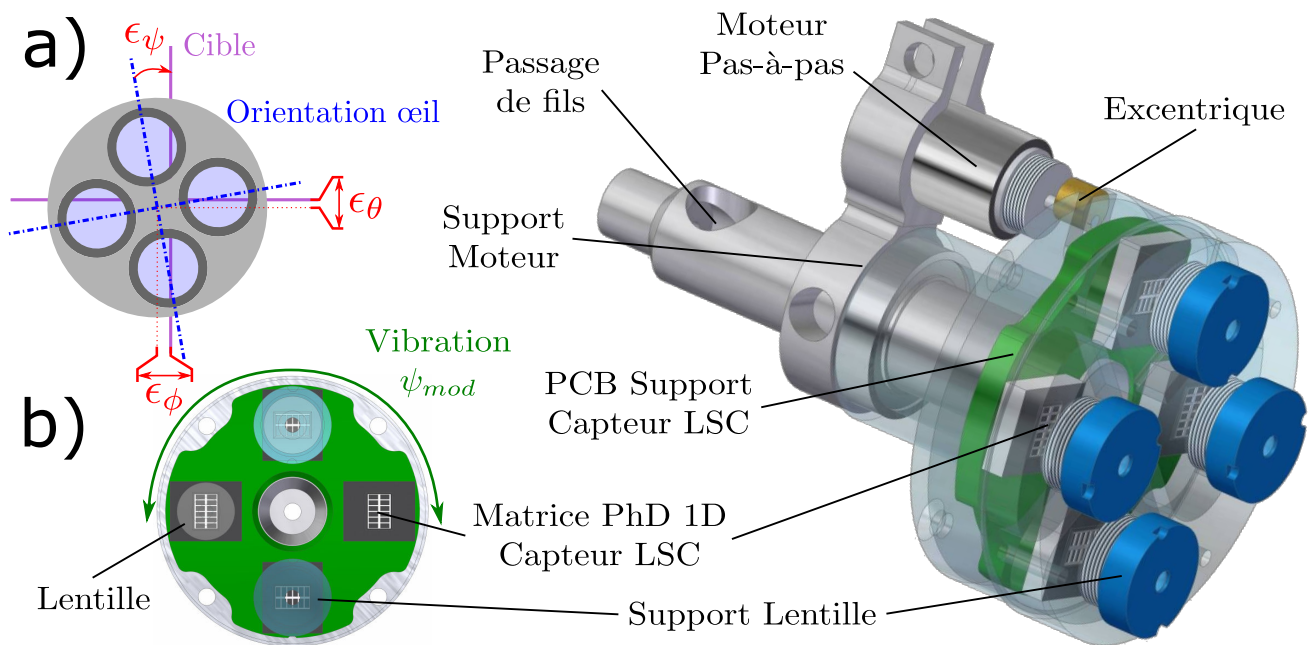


FIGURE IV.5 – a) Principe de localisation d'une cible : point de vue de l'œil. Chacun des yeux élémentaires mesure une erreur différente permettant de remonter à l'orientation de la cible par rapport à l'œil. b) Vue CAO de l'œil et de sa vibration : la pièce servant de support à l'optique tourne sur elle même de quelques degrés, générant ainsi une micro-translation des lentilles au dessus de chaque capteur.

réglage va donc être réalisé manuellement par itération, jusqu'à obtenir des réglages proches de ceux souhaités.

La méthode utilisée ici pour caractériser la sensibilité des photo-récepteurs est la méthode classique utilisée en électrophysiologie sur les photo-récepteurs d'invertébrés : on fait pivoter le capteur autour de son axe vertical devant une source lumineuse ponctuelle fixe et on enregistre la sortie des deux photodiodes. On place donc successivement chacun des quatre yeux face à une source ponctuelle modulée à 50 Hz². Cette source consiste en une simple LED blanche dont l'intensité lumineuse est modulée grâce à une modulation en courant. Ensuite on fait tourner le capteur sur lui même, c'est à dire autour d'un axe passant par la lentille du capteur et orthogonal à l'axe lumineux. La rotation du capteur est effectuée grâce à un moteur pas-à-pas configuré en mode micro pas (chaque pas correspond à une rotation de $0.9^\circ/32$), effectuant 10 pas par seconde. A cette vitesse, une identification complète balayant tout le champ visuel prend environ 2 minutes.

La figure IV.7a montre le dispositif expérimental permettant d'identifier et d'ajuster les paramètres

2. L'utilisation d'une source modulée est indispensable à cause du filtrage analogique passe haut (dû au filtre analogique passe-bande, voir section 2.5). Sans cette modulation, l'auto-adaptation des pixels opère et la source lumineuse n'est plus vue, ce qui rend impossible la caractérisation.

4. En réalité le montage était plongé dans un noir quasi total mais la photographie a été prise avec un temps d'exposition long pour rendre visible le montage.

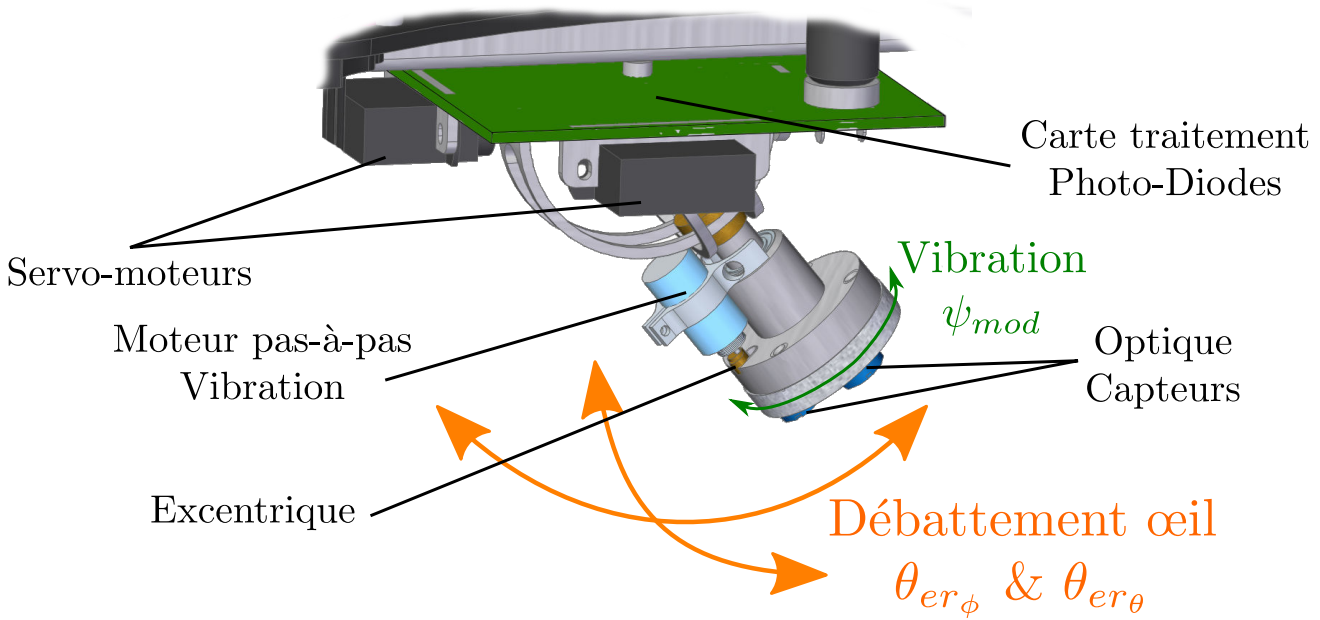


FIGURE IV.6 – Vue CAO de l’œil complet avec ses différents degrés de libertés. La vibration est réalisée à l’aide d’un unique moteur pas-à-pas et d’un système d’excentrique permettant d’appliquer un déplacement périodique identique pour chacune des quatre lentilles.

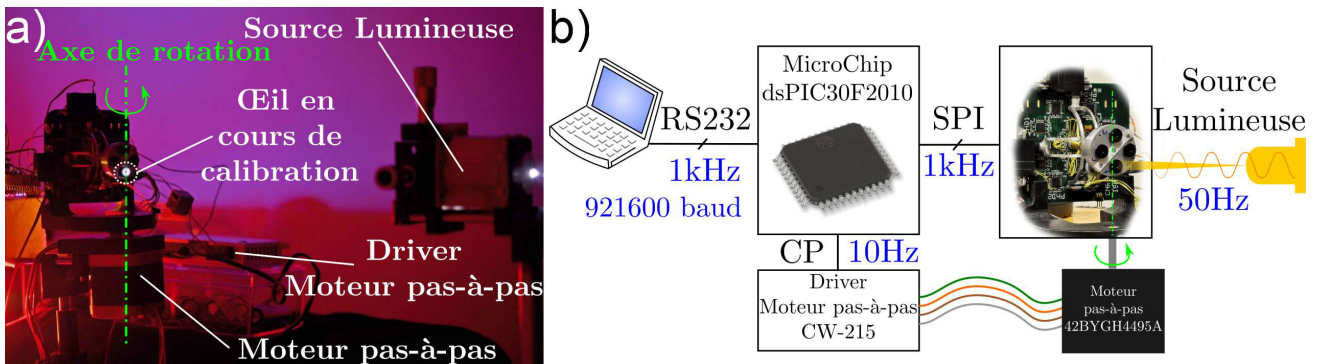


FIGURE IV.7 – a) Photographie⁴ du montage pendant une identification des paramètres optiques $\Delta\phi$ et $\Delta\rho$. Une source lumineuse modulée à 50 Hz est placé en face de l’optique de l’œil qui tourne lentement ($0.28^\circ \cdot s^{-1}$). b) Schéma du montage nécessaire à l’identification. Un dsPIC récupère les signaux photodiodes numérisés via une liaison SPI avec l’œil et les transmet à un PC via RS232 pour qu’ils soient traités. Simultanément, le dsPIC déclenche les pas du moteur en envoyant des "Clock Pulse" (CP) au driver moteur.

optiques de chaque œil. On fait pointer la source lumineuse sur un premier œil, que l’on fait tourner avec le moteur pas-à-pas, on identifie les paramètres optiques, on réajuste la lentille si besoin et on recommence jusqu’à obtenir les paramètres souhaités. Puis on passe à l’œil suivant, et le processus recommence jusqu’à avoir ajusté les quatre yeux. La figure IV.7f montre les différents éléments composant le dispositif d’identification. Les signaux photodiodes sont numérisés en continu par l’œil, puis envoyé

par SPI à 1 kHz à un micro contrôleur dsPIC30F2010 de chez MicroChip qui commande également les pas du moteur pas-à-pas en envoyant des "Clock Pulse" (CP) au driver. Le micro-contrôleur envoie ensuite les données photodiodes à un PC par liaison RS232 pour qu'elles soient traitées.

L'ajustement des paramètres optiques est réalisé en vissant ou dévissant l'optique, ce qui permet d'ajuster la distance séparant le capteur de la lentille. La figure IV.8a, montre les signaux photodiodes bruts reçus après numérisation par les convertisseurs analogiques-numériques (CAN). Ces signaux sont ensuite démodulés afin d'en extraire l'enveloppe correspondant à la sensibilité gaussienne de chaque pixel. La figure IV.8b, montre le résultat de la démodulation pour deux réglages de lentille différents (en bleu et en rouge). Pour chacun des réglages, les angles $\Delta\rho$ de chaque pixel ainsi que les angles $\Delta\phi$ de chaque pair de pixels sont affichés. Étant donné que les signaux de chaque pixel n'ont pas la même amplitude, ils sont normalisés afin de voir si les paramètres optiques sont acceptables. On se fixe une tolérance de 5%, et comme le montre la figure IV.8c, les angles $\Delta\phi$ et $\Delta\rho$ ont été réglés de façon à ce que l'intersection de deux gaussiennes normalisées se fasse à 5% près à mi-hauteur.

Une fois que les paramètres optiques ont été ajustés, une goutte de vernis bloquant est placée sur le pas de vis de l'optique de façon à ce que celle-ci ne se dérègle pas.

Remarque IV.1: On remarque qu'il n'est pas possible d'obtenir un angle $\Delta\phi$ et un angle $\Delta\rho$ unique au sein d'un même œil. Ceci s'explique par les distorsions de l'optique qui ne sont pas uniformes, le fait que le capteur LSC n'est pas disposé de façon parfaitement parallèle à la lentille et qu'il n'est pas non plus complètement centré. Cette disparité entre les différentes paires d'un même œil nécessite la mise en place d'une phase de calibration qui sera décrite dans la section 3.2.1.

La table IV.1 résume les paramètres optiques identifiés pour chaque photo-récepteur de chacun des quatre yeux. On remarque donc qu'il n'est pas évident d'obtenir des paramètres optiques identiques pour chaque œil, pour les raisons citées dans la remarque IV.1. Cependant, comme le montre la figure IV.8c, les réglages ont été faits de façon à avoir $\Delta\phi$ et $\Delta\rho$ le plus proche possible. Au final, la moyenne des angles d'acceptance $\Delta\rho$ est de 3.73° avec un écart type de 0.16° pour l'œil 1, 3.67° avec un écart type de 0.14° pour l'œil 2, 3.56° avec un écart type de 0.11° pour l'œil 3 et 3.76° avec un écart type de 0.15° . De même, la moyenne des angles inter-ommatidiaux $\Delta\phi$ est de 3.81° avec un écart type de 0.08° pour l'œil 1, 3.92° avec un écart type de 0.10° pour l'œil 2, 3.91° avec un écart type de 0.10° pour l'œil 3 et 3.97° avec un écart type de 0.16° .

2.3 Design du mécanisme de vibration

Pour mettre en œuvre l'hyperacuité, il est nécessaire d'appliquer une rotation périodique des axes optiques de quelques degrés pour chaque *ommatidies*. Comme montré dans la figure IV.3f, la rotation des axes optiques peut être obtenue grâce à une simple translation de l'optique au dessus du capteur.

PhD	Œil 1		Œil 2		Œil 3		Œil 4	
	$\Delta\rho_i^k$ [deg]	$\Delta\phi_{i,i+1}^k$ [deg]	$\Delta\rho_i^k$ [deg]	$\Delta\phi_{i,i+1}^k$ [deg]	$\Delta\rho_i^k$ [deg]	$\Delta\phi_{i,i+1}^k$ [deg]	$\Delta\rho_i^k$ [deg]	$\Delta\phi_{i,i+1}^k$ [deg]
1	3.60		3.63		3.46		3.63	
		3.74		3.97		3.80		3.85
2	3.91		3.71		3.52		3.88	
		3.88		3.88		3.97		4.11
3	3.54		3.83		3.68		3.82	
		3.82		4.02		3.91		4.02
4	3.94		3.80		3.46		3.94	
		3.91		3.97		3.85		4.11
5	3.74		3.57		3.71		3.66	
		3.74		3.77		4.05		3.77
6	3.66		3.46		3.54		3.60	
Moyenne	3.73	3.81	3.67	3.92	3.56	3.91	3.76	3.97
Écart type	0.16	0.08	0.14	0.10	0.11	0.10	0.15	0.16

TABLE IV.1 – Ce tableau décrit la sensibilité angulaire $\Delta\rho$ pour chaque photodiode de chaque œil. On note $\Delta\rho_i^k$ l'angle d'acceptance de la photodiode i de l'œil k . Le tableau référence également les 5 angles inter-ommatidiaux de chaque œil, et l'on note $\Delta\phi_{i,i+1}^k$, l'angle séparant l'axe optique des photodiodes i et $i + 1$ de l'œil k .

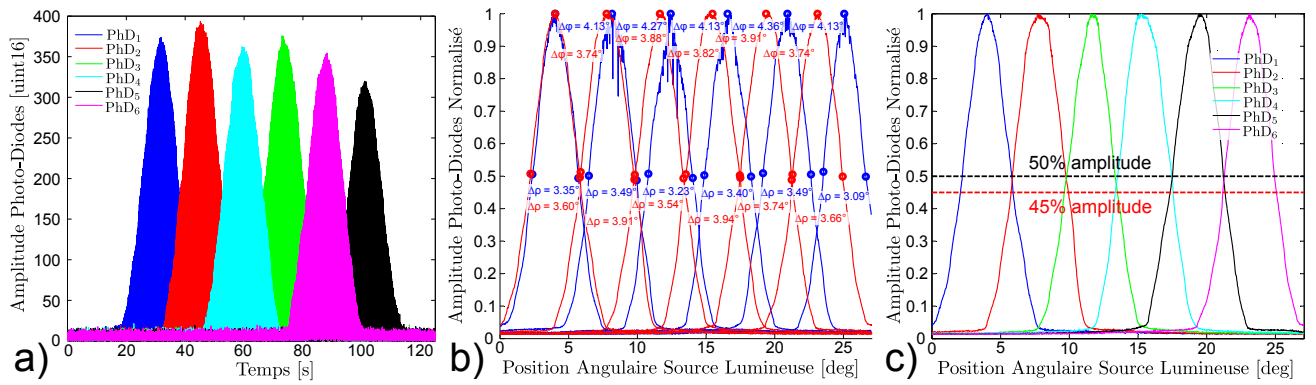


FIGURE IV.8 – a) La sensibilité angulaire de chacun des photo-récepteurs (photodiodes notées PhD_i) d'un œil obtenu en faisant tourner lentement l'œil face à une source lumineuse ponctuelle. On constate que la sensibilité est proche d'une gaussienne et que l'amplitude maximale n'est pas la même pour tout les photo-récepteurs. b) Enveloppe des gaussiennes de la figure a), avec les paramètres optiques identifiés ($\Delta\rho$ est la largeur de la gaussienne à mi-hauteur, et $\Delta\phi$ est l'angle séparant le maximum de deux gaussiennes voisines). Les courbes bleues montrent l'identification de ces paramètres avant de régler l'optique. Les courbes rouges montrent le meilleur compromis qui a été atteint ($\Delta\phi \approx \Delta\rho$) en dévissant presque au maximum l'optique. c) L'enveloppe normalisée des gaussiennes rouges de la figure b). Les réglages optiques ont été choisis de façon à ce qu'une fois normalisées, les gaussiennes se croisent à mi-hauteur à 5% près.

2.3.1 Design du système excentrique

Il a été choisi d'actionner la vibration des quatre yeux avec le même mécanisme. Cela présente un certain nombre d'avantages :

- l'amplitude de la vibration obtenue pour chacun des yeux est strictement identique,
- la vibration est synchrone pour chaque œil,
- un seul actionneur est nécessaire, rendant le système plus fiable et plus compact.

Le mécanisme en question consiste en un moteur pas-à-pas muni d'un excentrique sur son axe, générant un mouvement de va-et-vient sinusoïdale de quelques degrés au dessus des capteurs optiques. L'excentrique qui tourne en continu est placé dans un trou oblong et la rotation est guidée par deux roulements à billes. La figure IV.9a donne une vision schématique du système de vibration vue de dessus et du fonctionnement de l'excentrique. La figure IV.9b montre le déplacement obtenu au niveau de l'optique, et la modulation qui en résulte (rotation des axes optiques d'amplitude ψ_{mod}).

Le design et le dimensionnement de l'excentrique, dépend de la modulation ψ_{mod} souhaitée. Les figures IV.9a-b et IV.10a-c définissent les grandeurs géométriques permettant son dimensionnement.

Grâce à la figure IV.10c, on en déduit le déplacement Δ_{vib} nécessaire :

$$\Delta_{vib} = L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right) \quad (IV.1)$$

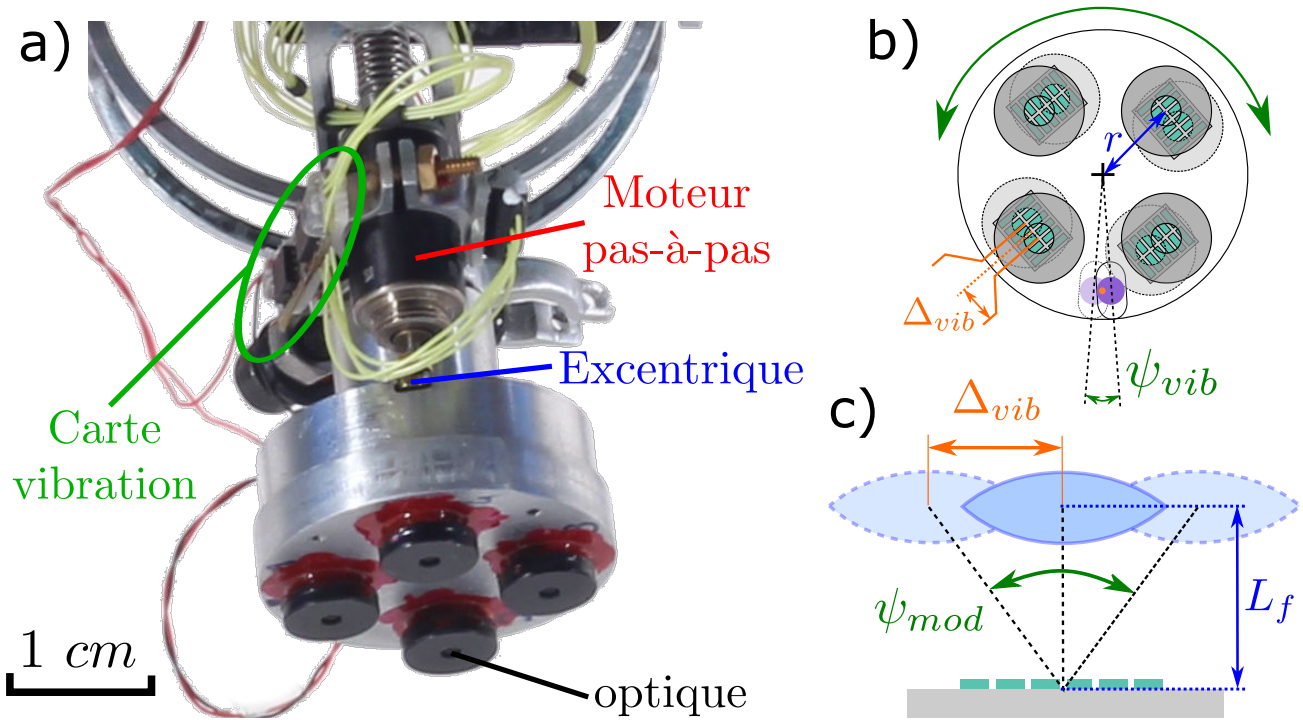


FIGURE IV.9 – a) Photographie de l'œil avec son système de vibration. b) Vibration obtenue grâce à un excentrique. Ce dernier est représenté en violet et son axe de rotation en orange. La vibration d'amplitude ψ_{vib} génère un déplacement Δ_{vib} des optiques au-dessus des capteurs. c) La modulation ψ_{mod} obtenue par la translation Δ_{vib} de l'optique dépend de la distance L_f séparant l'optique du capteur.

A partir de la figure IV.9c, on peut en déduire le dimensionnement de l'excentrique en fonction du déplacement des optiques Δ_{vib} :

$$\psi_{vib} = 2 \tan^{-1} \left(\frac{\Delta_{vib}}{r_c} \right) = 2 \tan^{-1} \left(\frac{L_f}{r_c} \cdot \tan \left(\frac{\psi_{mod}}{2} \right) \right) \quad (IV.2)$$

$$\epsilon_e = \frac{\Delta_e}{\cos \left(\frac{\psi_{vib}}{2} \right)} \quad (IV.3)$$

Il est donc nécessaire de connaître Δ_e , que l'on obtient d'après le théorème de Thalès :

$$\Delta_e = \frac{r_e - h_e}{r_c} \cdot \Delta_{vib} \quad (IV.4)$$

Finalement h_e s'obtient grâce au triangle (O_e, O_h, O_m) :

$$\begin{cases} h_e = \Delta_e \cdot \tan \left(\frac{\psi_{vib}}{2} \right) \\ \Delta_e = \frac{r_e - h_e}{r_c} \Delta_{vib} \end{cases}$$

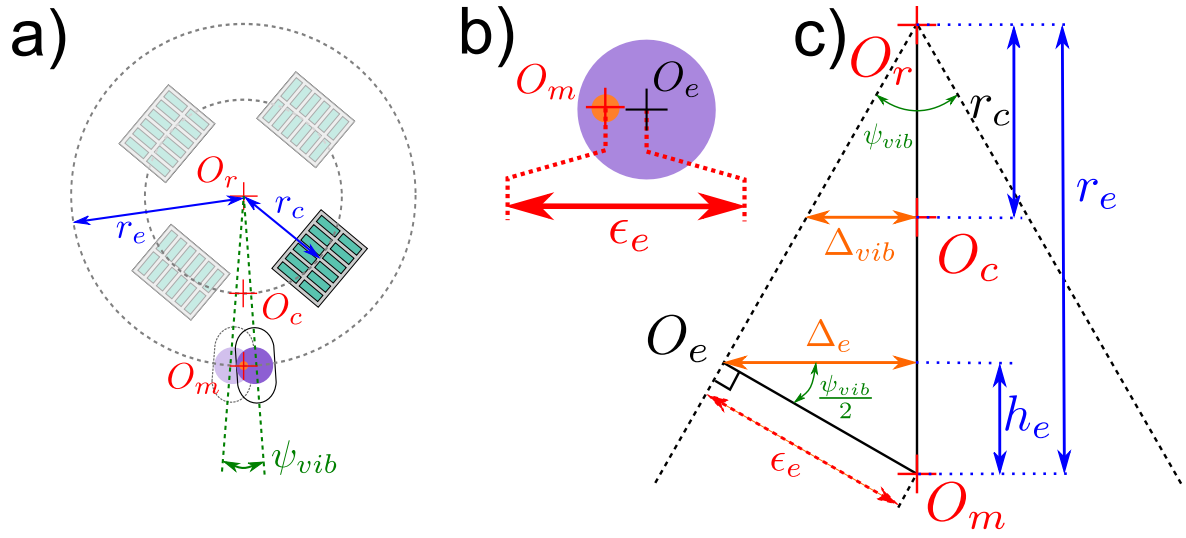


FIGURE IV.10 – a) Définition du centre de rotation des optiques O_r , du centre de l'optique O_c , et du centre de l'axe moteur O_m . b) L'excentrique consiste en un cylindre percé à une distance ϵ_e de son centre O_e . L'axe du moteur est fixé dans ce trou O_m générant ainsi un déplacement total sinusoïdal, d'amplitude $2\epsilon_e$. c) Lien entre la dimension de l'excentrique ϵ_e et le déplacement Δ_{vib} de l'optique.

$$\Leftrightarrow \Delta_e = \frac{r_e \cdot \Delta_{vib}}{r_c + \Delta_{vib} \cdot \tan\left(\frac{\psi_{vib}}{2}\right)} \quad (\text{IV.5})$$

Puis il suffit de tout exprimer en fonction de la modulation souhaitée sur les axes optiques. On injecte (IV.5) dans (IV.3) :

$$\epsilon_e = \frac{r_e \cdot \Delta_{vib}}{\cos\left(\frac{\psi_{vib}}{2}\right) \cdot \left(r_c + \Delta_{vib} \cdot \tan\left(\frac{\psi_{vib}}{2}\right)\right)} \quad (\text{IV.6})$$

Puis on utilise (IV.1) et (IV.2) que l'on injecte dans (IV.6) :

$$\epsilon_e = \frac{r_e \cdot L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right)}{\cos\left(\frac{2 \tan^{-1}\left(\frac{L_f}{r_c} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)\right)}{2}\right)} \cdot \frac{1}{\left(r_c + L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right) \tan\left(\frac{2 \tan^{-1}\left(\frac{L_f}{r_c} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)\right)}{2}\right)\right)}$$

$$\begin{aligned}
\Leftrightarrow \epsilon_e &= \frac{r_e \cdot L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right)}{\cos\left(\tan^{-1}\left(\frac{L_f}{r_c} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)\right)\right)} \\
&\quad \cdot \frac{1}{\left(r_c + L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right) \tan\left(\tan^{-1}\left(\frac{L_f}{r_c} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)\right)\right)\right)} \\
\Leftrightarrow \epsilon_e &= \frac{r_e \cdot L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right)}{\cos\left(\tan^{-1}\left(\frac{L_f}{r_c} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)\right)\right)} \\
&\quad \cdot \frac{1}{r_c + \frac{L_f^2}{r_c^2} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)^2} \\
\Leftrightarrow \epsilon_e &= \frac{r_e \cdot L_f \cdot \tan\left(\frac{\psi_{mod}}{2}\right)}{\cos\left(\tan^{-1}\left(\frac{L_f}{r_c} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)\right)\right) \left(r_c + \frac{L_f^2}{r_c^2} \cdot \tan\left(\frac{\psi_{mod}}{2}\right)^2\right)} \tag{IV.7}
\end{aligned}$$

La distance entre l'axe de rotation du moteur et l'axe de rotation des optiques est de $r_e = 12.0$ mm et la distance entre chaque capteur et le centre de rotation des optiques est de $r_c = 7.5$ mm. L'amplitude de modulation ψ_{mod} souhaitée est de 4° (du même ordre de grandeur que l'angle inter-ommatidial $\Delta\phi$). La seule inconnue dans l'équation (IV.7) est donc la longueur L_f séparant la lentille du capteur. On peut déduire cette dernière inconnue en étudiant la géométrie du capteur lui même (voir figure IV.4b) :

$$L_f = \frac{\delta_{PhD}}{2 \tan\left(\frac{\Delta\phi}{2}\right)} \tag{IV.8}$$

L'identification des paramètres optiques visible sur la figure IV.8b nous montre que l'angle $\Delta\phi$ de la paire du milieu est de 3.82° . Sachant que $\delta_{PhD} = 420$ μm , on en déduit par application numérique de l'équation (IV.8), que $L_f = 6.33$ mm. De même par application numérique de l'équation (IV.2), on trouve que l'amplitude de vibration de la pièce soutenant les optiques doit être de $\psi_{vib} = 3.4^\circ$. Cette valeur théorique a été vérifiée par la mesure expérimentale de la vibration décrite dans le paragraphe 2.3.3.

2.3.2 Réglage de la fréquence

La vibration est obtenue à l'aide d'un moteur pas-à-pas miniature de chez Faulhaber (moteur AM0820-V-5-56 fournissant un couple de 0.65 mN \cdot m à 5V). Ce moteur est piloté grâce à une carte driver réalisée spécialement. La carte driver (ou carte de vibration) est équipée du driver moteur

A3901SEJTR et d'un micro-contrôleur PIC12F615. Le driver moteur pilote le moteur en courant et réalise un pas à chaque fois qu'il reçoit une impulsion du micro-contrôleur. Le micro-contrôleur quant à lui génère les impulsions de commande pour obtenir la fréquence de vibration désirée. Un aperçu de la carte driver est disponible sur la figure IV.11.

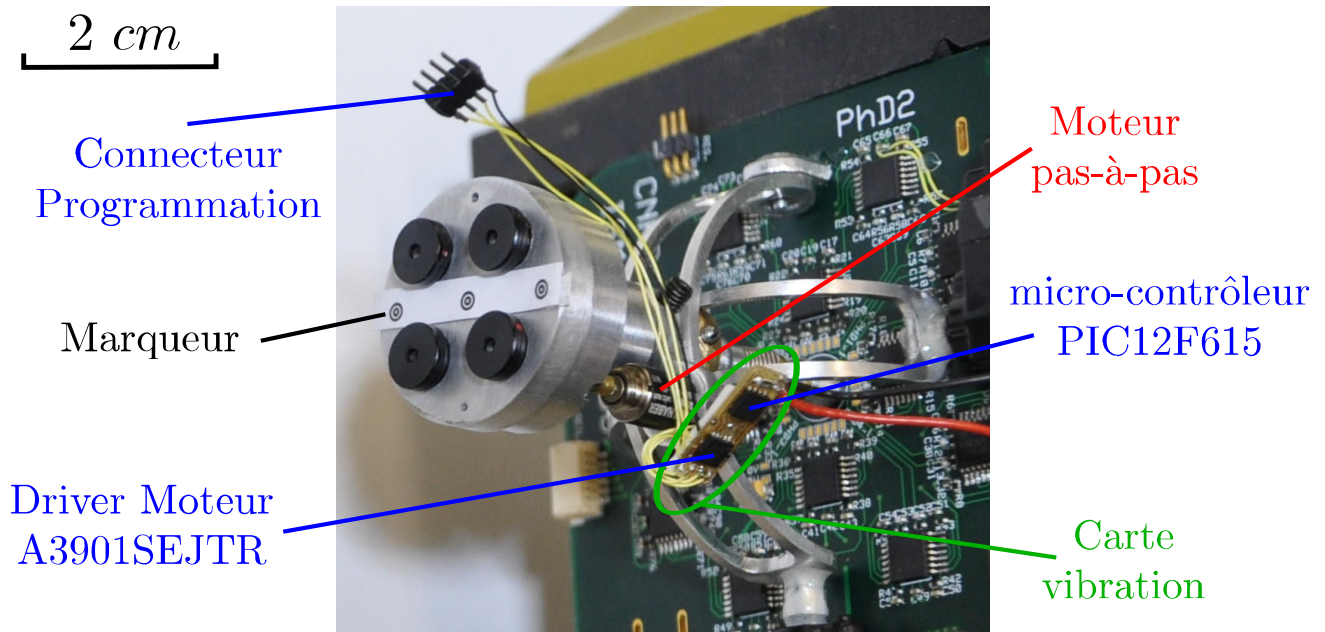


FIGURE IV.11 – La carte driver permettant de régler la fréquence de vibration de l'œil.

2.3.3 Vérification de l'amplitude de vibration

Pour vérifier l'amplitude de vibration et la fréquence obtenues pour ψ_{vib} , on place des marqueurs sur l'œil (voir figure IV.11) et on filme la vibration à l'aide d'une caméra rapide (la caméra DALSA Genie HM640).

La figure IV.12a montre le dispositif expérimental utilisé avec la caméra et l'œil équipé de marqueurs. Ensuite, grâce à la fonction Matlab DLTv5 (figure IV.12b), la vidéo est analysée de façon semi-automatique pour en extraire les coordonnées des marqueurs et en déduire la vibration mesurée $\bar{\psi}_{vib}$. L'amplitude de la vibration A_{vib} ainsi que la fréquence f_{vib} peuvent alors être vérifiées (voir figure IV.13a-b). La fréquence de vibration doit également être vérifiée, car même si on utilise un driver pour le moteur, il se peut que des pas soient ratés de temps en temps (à cause des frottements) ce qui pourrait affecter la fréquence de vibration finale. La vérification de la fréquence de vibration est effectuée à l'aide d'une FFT réalisée sur le signal $\bar{\psi}_{vib}$.

Remarque IV.2: La fréquence d'échantillonnage de la caméra ne peut pas dépasser 300 images/seconde, ce qui est insuffisant pour reproduire fidèlement le mouvement sinusoïdal de la vibration. Par conséquent, pour être sûr de pouvoir déterminer l'amplitude du signal, une fréquence de 240 Hz

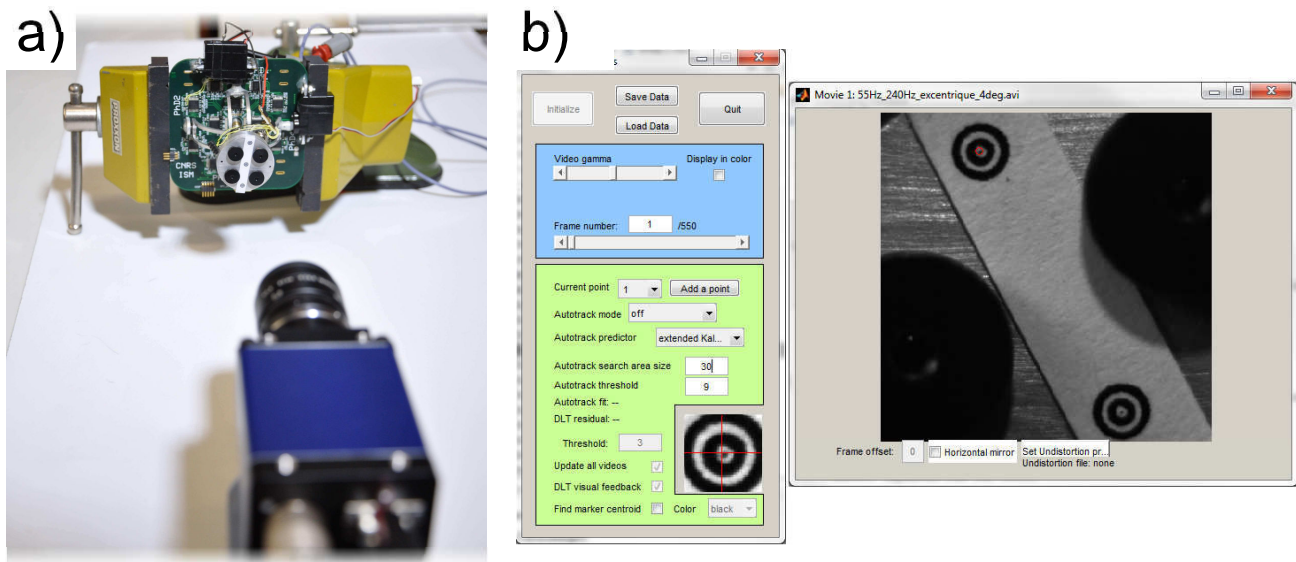


FIGURE IV.12 – a) Le dispositif permettant de quantifier l’amplitude de vibration ainsi que la fréquence. Une caméra DALSA Genie HM640 filme, à 240 images/seconde, l’œil qui est équipé de marqueurs. b) Aperçu de la fonction matlab DLTv5 permettant d’extraire la position des marqueurs de façon semi-automatique.

(qui n’est pas un multiple de la fréquence mesurée de 50 Hz) a été choisie. Ainsi un glissement des échantillons a lieu le long du signal, assurant de passer par le maximum, si la fenêtre d’observation est suffisamment grande.

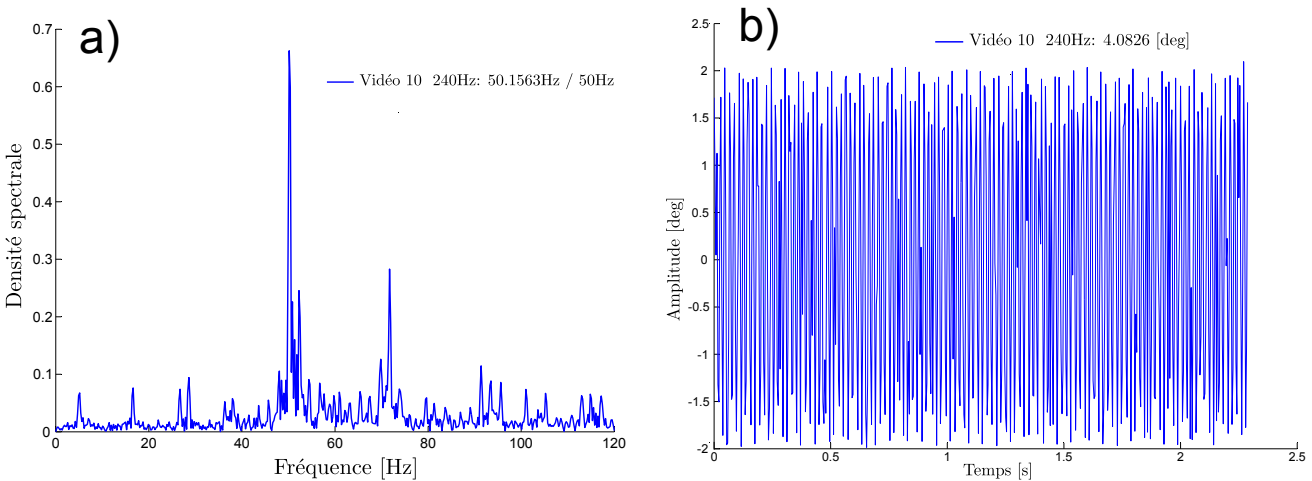


FIGURE IV.13 – a) La densité spectrale du signal de vibration $\bar{\psi}_{vib}(t)$. On observe un pic bien marqué à 50.1563 Hz, preuve de la qualité mécanique du système. b) Le signal $\bar{\psi}_{vib}(t)$ au cours du temps. Le signal est compris entre -1.9857° et 2.0970° , ce qui donne une amplitude crête à crête de 4.0826° .

Finalement, la fréquence de vibration de l’œil a été réglée à 55 Hz et l’amplitude obtenue pour ψ_{vib}

est de 4.08° au lieu des 3.4° visés (voir paragraphe 2.3.1). Cela entraîne donc une petite erreur sur la modulation obtenue sur les photo-récepteurs qui est de $\psi_{mod} = 4.73^\circ$ au lieu de 4° . Cette erreur reste peu importante, étant donnée que le but fixé était d'avoir une amplitude de vibration au moins égale à l'angle $\Delta\Phi$.

2.4 Conception du mécanisme de découplage

Malgré l'augmentation du champ de vision du capteur grâce à l'utilisation des 6 pixels de la matrice LSC (champ visuel d'environ $\pm 10^\circ$ contre $\pm 1^\circ$ dans [Kerhuel *et al.*, 2012]), le champ visuel reste trop restreint pour assurer la stabilisation d'un aéronef. En effet, pour un aéronef volant à 2 mètres du sol, un tel capteur ($FOV = \pm 10^\circ$) ne couvrirait que 70 centimètres au sol ($x = 2z \cdot \tan(FOV)$ avec x la "surface" couverte au sol et z la hauteur.)

Il est donc nécessaire de fournir à l'œil des degrés de liberté supplémentaires qui lui permettront de se réorienter afin de garder dans son champ visuel une cible, tout en se déplaçant. Les nombreux avantages apportés par un tel système de découplage sont discutés dans la section 6 du chapitre II.

2.4.1 Mécanisme de découplage

Le système de découplage proposé dans cette section fournit donc deux degrés de liberté à l'œil, lui permettant de se réorienter sur son axe de "roulis" et de "tangage". La vue CAO de la figure IV.6 met en évidence le système de double cardans orthogonaux imbriqués qui permet cette réorientation selon les deux axes. Chaque cardan en aluminium est actionné par un micro servomoteur de 4 g (micro servomoteur de chez HighTec) pouvant atteindre des vitesses de 800 degrés par seconde (1000 degrés par seconde à vide) et offrant un débattement de $\pm 50^\circ$. L'œil est fixé au bout d'un axe lié par une liaison rotule à son support et passant au milieu du système de cardan. De cette façon, l'orientation de l'œil est contrainte par celle des cardans. Le lien entre les cardans et l'axe est réalisé par une bague en laiton, qui glisse le long des cardans tout en verrouillant l'axe de "lacet" (la troisième rotation est laissée libre par la liaison rotule) grâce à une encoche rectangulaire. De cette façon, les seules rotations effectuées par l'œil sur l'axe de lacet sont exercées par le système de vibration. Enfin le contact entre la bague en laiton et les cardans est assuré par un système de piston-ressort, disposé à la base de l'axe de support de l'œil (entre l'axe et la rotule). Ce système à ressort permet d'assurer le contact permanent entre la bague en laiton et les cardans, tout en absorbant les éventuels défauts de courbure de ces derniers. Il permet également d'empêcher l'œil de bouger, quand celui-ci est orienté vers le bas. Le détail de ce mécanisme est donné par la figure IV.6, une vue éclatée de l'œil est disponible sur la figure IV.14 et la figure IV.15 donne une vision de l'œil une fois assemblé.

module	élément(s)	Masses [g]	total module[g]
Vision	carte vision	0.892 g	8.541 g
	coupole+lentilles	3.960 g	
	base vision+roulement	3.689 g	
	moteur pas-à-pas+support	3.762 g	
Découplage	bague laiton	1.174 g	13.645 g
	tube+roulement	2.469 g	
	support rotule	0.386 g	
	rotule	0.702 g	
	ressort	0.126 g	
	supports servomoteurs	0.657 g	
	guidage servomoteurs	0.034 g	
	grande cardan+roulement	1.254 g	
	petite cardan+roulement	1.043 g	
servomoteurs	5.8 g		
Traitement vision	carte électronique+fil	30.014 g	30.014 g
TOTAL			52.100g

TABLE IV.2 – Description des masses des différents éléments de l'œil

2.4.2 Assemblage complet

L'ensemble de l'œil ainsi que son assemblage a été confectionné dans le laboratoire, les seuls composants sur étagères sont les servomoteurs, le moteur pas-à-pas, les optiques et les capteurs LSC.

La figure IV.14 représente une vue éclatée de l'œil. On y retrouve l'ensemble des éléments mécaniques (IV.14a) décrits dans la section 2.4.1 ainsi que le détail des composants électroniques constituant le système de vision en lui même (IV.14b). La table IV.2 détaille la masse de chacun des composants de l'œil ainsi que sa masse totale. On remarque que la partie mécanique de l'œil, y compris les servomoteurs et la partie vision (capteurs et optiques) représentent seulement 42.6% de la masse totale, soit 22.186 g sur 52.1 g. La carte de traitement photodiodes représente donc un apport de poids non négligeable de 30.014 g.

Assemblage de la partie capteur : Chacun des 6 pixels de chaque LSC doit recevoir un traitement analogique, réalisé par une carte spécifique (voir section 2.5). Le "capteur", placé au bout de l'axe de découplage, est donc relié avec des fils souples. Au total 26 fils sont nécessaires : 24 fils transmettent les courants photodiodes et 2 autres alimentent le circuit avec une tension de 5V (voir figure IV.15). Les fils utilisés sont des fils très fins de 300 μm permettant de résister aux vibrations appliquées à l'œil. Chacun des fils va ensuite traverser l'axe creux qui sert de support à l'œil. Le bout de l'axe présente un trou permettant aux différents fils de déboucher sur la carte de traitement des signaux photodiodes présentée dans la partie suivante (voir section 2.5.) Comme on peut le voir sur la figure IV.15b-c, une fois la carte vision en place dans la coupole, l'étanchéité optique est assurée.

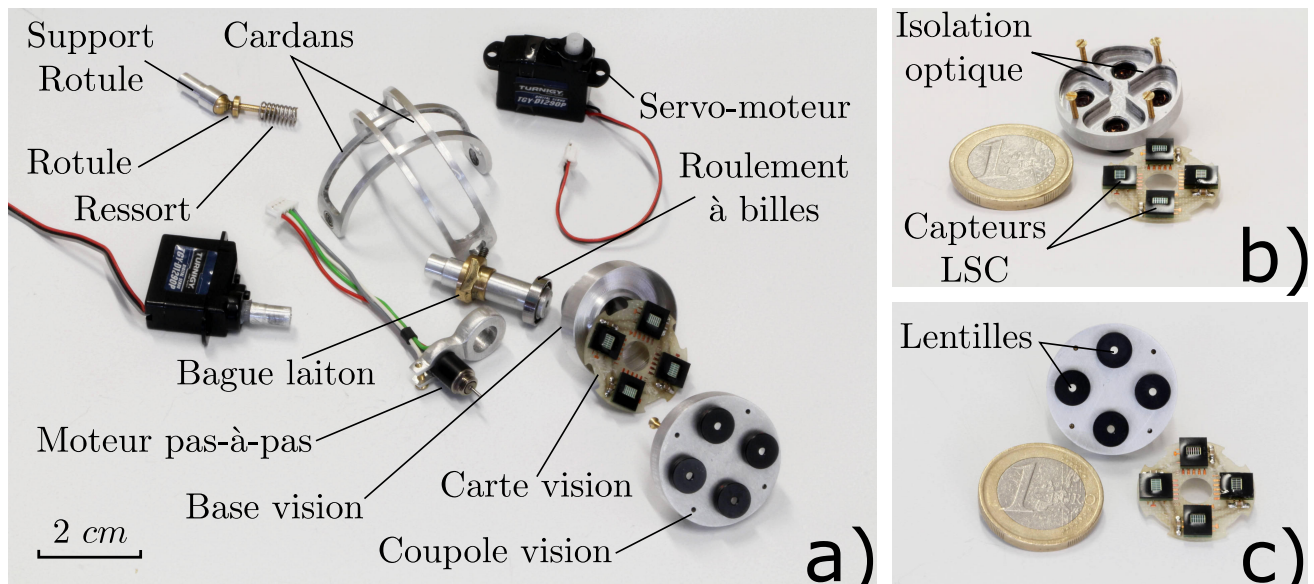


FIGURE IV.14 – a) Vue éclatée de l'œil et de ses différents composants. b) Vue de derrière de l'optique de l'œil. Cette pièce support dispose de parois qui permettent d'assurer l'étanchéité optique des différentes chambres optiques. Une fois assemblées, ces parois sont positionnées entre les capteurs LSC afin de les isoler les uns par rapport aux autres. c) Vue de face de l'optique qui vient fermer l'œil.

Assemblage de la partie mécanique : L'assemblage de la partie mécanique requiert une attention toute particulière et nécessite quelques réglages de précisions. En effet, il a été prévu sur la carte de traitement des photodiodes des emplacements destinés à accueillir les supports des servomoteurs (voir figure IV.17). Ces emplacements autorisent une translation latérale des supports de façon à pouvoir ajuster l'alignement de l'axe de chacun des servomoteurs avec la cardan et l'axe opposé. De même la hauteur du servomoteur par rapport à la carte de traitement peut être ajustée.

Ces ajustements sont réalisés de façon minutieuse lorsque le système est sous tension. Chaque servomoteur applique alors un mouvement de va et vient sur toute la plage de débattement et on recherche la position idéale, entraînant le moins de frottements et les mouvements les plus fluides. Une fois la position idéale trouvée, la position des servomoteur est fixée à l'aide d'une colle époxy.

De même, grâce à un système de 3 vis sur le support du moteur pas-à-pas, il est possible de régler finement la position de l'excentrique afin de limiter les frottements avec la coupole et d'optimiser son fonctionnement.

2.4.3 Identification des paramètres mécaniques

Une fois l'assemblage mécanique réalisé, il reste à identifier les dynamiques des servomoteurs de façon à pouvoir, par la suite, dimensionner les contrôleurs de vision. Pour cela, des potentiomètres sont temporairement placés sur l'axe des servomoteur. La tension des potentiomètres codant la position des servomoteurs est ensuite lue sous dSpace, avec lequel sont également générés les signaux de commande.

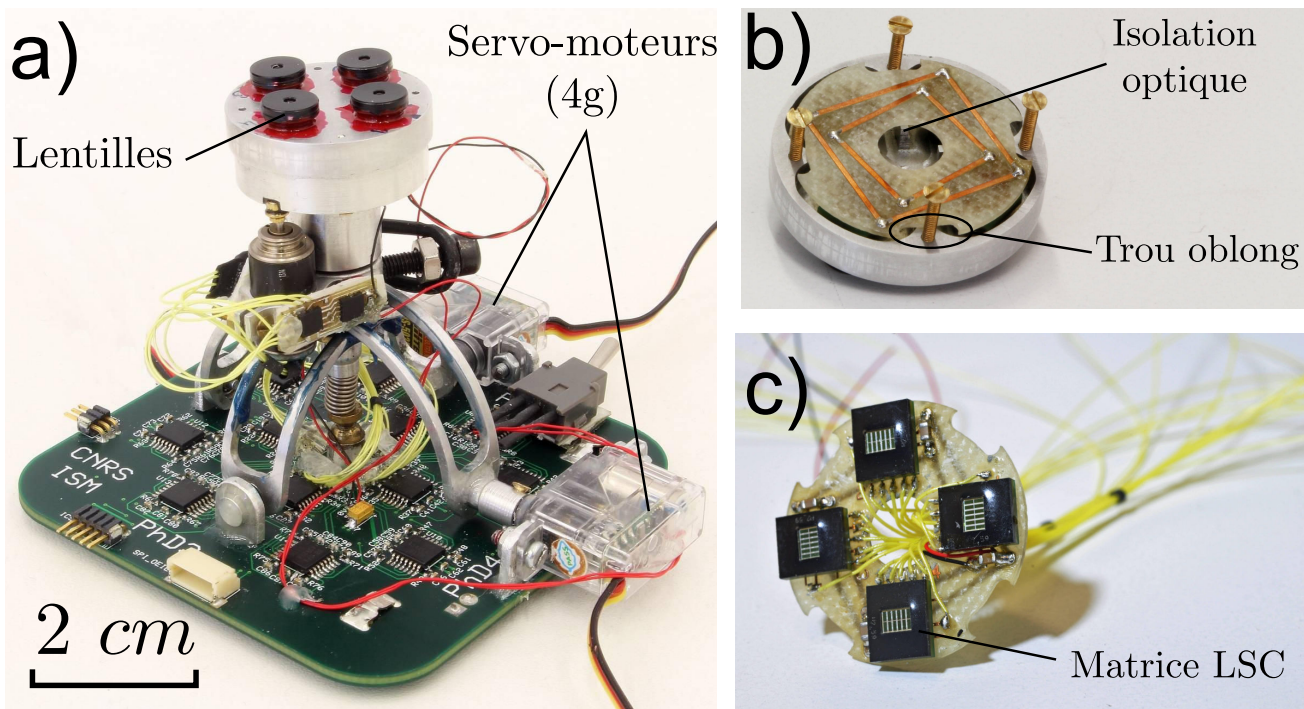


FIGURE IV.15 – a) Vue de l’œil assemblé. b) Visualisation de l’isolation optique une fois la carte vision en place. La carte des capteurs LSC est fixée sur l’axe central de l’œil, alors que l’optique est mise en mouvement grâce au système de vibration. Des trous oblongs permettent donc la rotation de l’optique par rapport aux rétines LSC d’environ 4 degrés. La carte supportant les LSC est placée au plus près de l’optique permettant ainsi d’assurer l’isolation optique grâce aux parois de la pièce en aluminium. c) La carte vision et ses 26 fils de 300 μm . Ces fils passent ensuite dans l’intérieur de l’axe principal de l’œil qui est creux à cet effet, et débouchent directement sur la carte de traitement électronique.

Le modèle des servomoteurs est identifié pour différentes réponses indicielles avec la méthode d’identification ARMAX (avec un ordre 2 pour $A(q)$, 2 pour $B(q) + 1$, 2 pour $C(Q)$ et 1 pour le retard). On identifie également la précision (le plus petit pas que le servomoteur est capable de réaliser) et les slew-rates⁵.

On constate dans un premier temps que le modèle n’est pas tout à fait le même dans un sens et dans l’autre. La figure IV.16 montre ses disparités et le tableau IV.3 résume les paramètres identifiés pour les différents modèles (montée, descente et modèle moyen). Chacun des modèles a été identifié avec la méthode ARMAX, soit sur une séquence de step de 5° en montée, soit en descente, soit en montée et descente pour le modèle moyen.

On remarque que les fonctions de transfert identifiées, ainsi que la réponse mesurée, présente un non minimum de phase, ce qui est assez inhabituel pour une contrôle en position. On suppose que cela est plutôt dû à une non linéarité, apparaissant pour de fortes accélérations sur le potentiomètre. Cela

5. Le slew-rate est une saturation sur la vitesse qui se traduit comme une pente maximale au-delà de laquelle le système ne peut pas aller. Ce paramètre a une importance car il révèle une saturation de l’actionneur et rend le modèle non-linéaire. On recherchera en général à avoir le slew-rate le plus élevé possible pour un système.

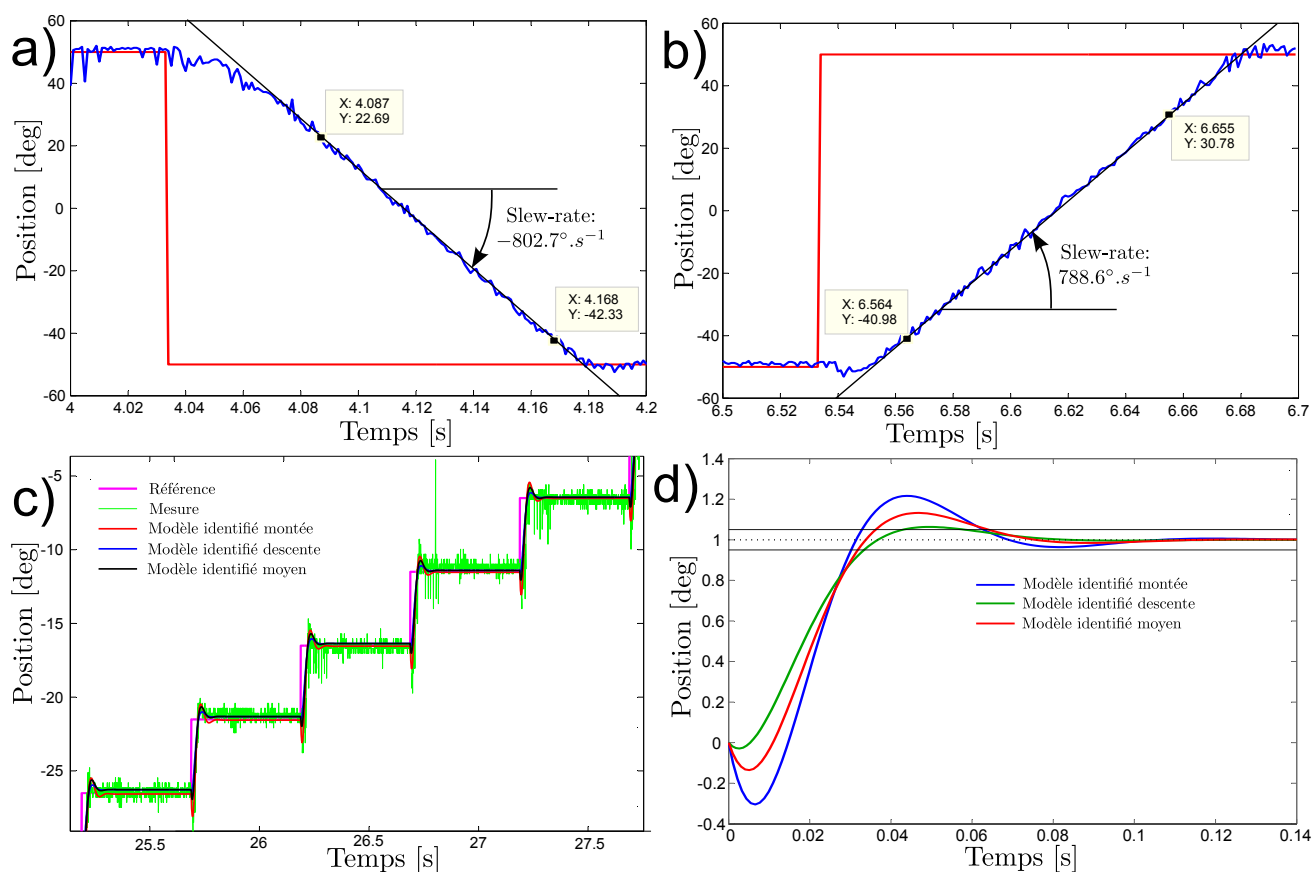


FIGURE IV.16 – a) Identification du slew-rate dans le sens négatif (ou "descente") pour un step de 100° . Le servomoteur atteint au maximum une vitesse de $802.7^\circ \cdot s^{-1}$. b) Identification du slew-rate dans le sens positif (ou "montée") pour un step de 100° . Le servomoteur atteint au maximum une vitesse de $788.6^\circ \cdot s^{-1}$. c) Comparaison des modèles dynamiques identifiés avec la mesure. d) Zoom sur la réponse indicielle identifiée pour la "montée", la "descente" et le modèle "moyen". Le système répond en moyenne en 65 ms avec un dépassement de l'ordre de 14%.

peut également être dû à un retard pur sur la boucle fermée.

On ne retiendra pour la suite que le modèle moyen, qui est un modèle du second ordre avec un temps de réponse à 95% de 0.065 seconde et un dépassement d'environ 14%. Enfin, on remarque une précision de 0.5° dans le meilleur des cas et 1° dans le pire des cas. La précision change un peu du fait des frottements secs qui sont plus ou moins importants selon l'orientation du servomoteur. Enfin, on note que ces servomoteurs sont très rapides, étant donné que même avec la charge importante que représente l'œil, ils arrivent à atteindre des vitesses de $800^\circ \cdot s^{-1}$ (slew-rate).

2.5 Conception de la carte de traitement des signaux visuels

Chaque capteur consiste en une matrice de 6 pixels comme décrit la figure IV.4. L'œil complet est donc composé de $6 \times 4 = 24$ photodiodes qui nécessitent chacune un traitement électronique individua-

	Montée	Descente	Moyen
Slew-Rate	$802.7^{\circ}.s^{-1}$	$788.6^{\circ}.s^{-1}$	$795.7^{\circ}.s^{-1}$
Dépassement	22%	6.5%	14%
Temps de réponse	62.9ms	58.2ms	64.1ms
Fonction de transfert	$\frac{-101.62(p - 91.39)}{p^2 + 95.61p + 9287}$	$\frac{-23.997(p - 333.8)}{p^2 + 118.8p + 8011}$	$\frac{-58.8233(p - 141.2)}{p^2 + 102.9p + 8309}$

TABLE IV.3 – Les paramètres identifiés pour les modèles des servomoteurs dans chacun des sens de fonctionnement. Le modèle retenu est le modèle moyen, qui sera utilisé pour réaliser les boucles d’asservissements liées à l’œil.

lisé.

2.5.1 Présentation générale

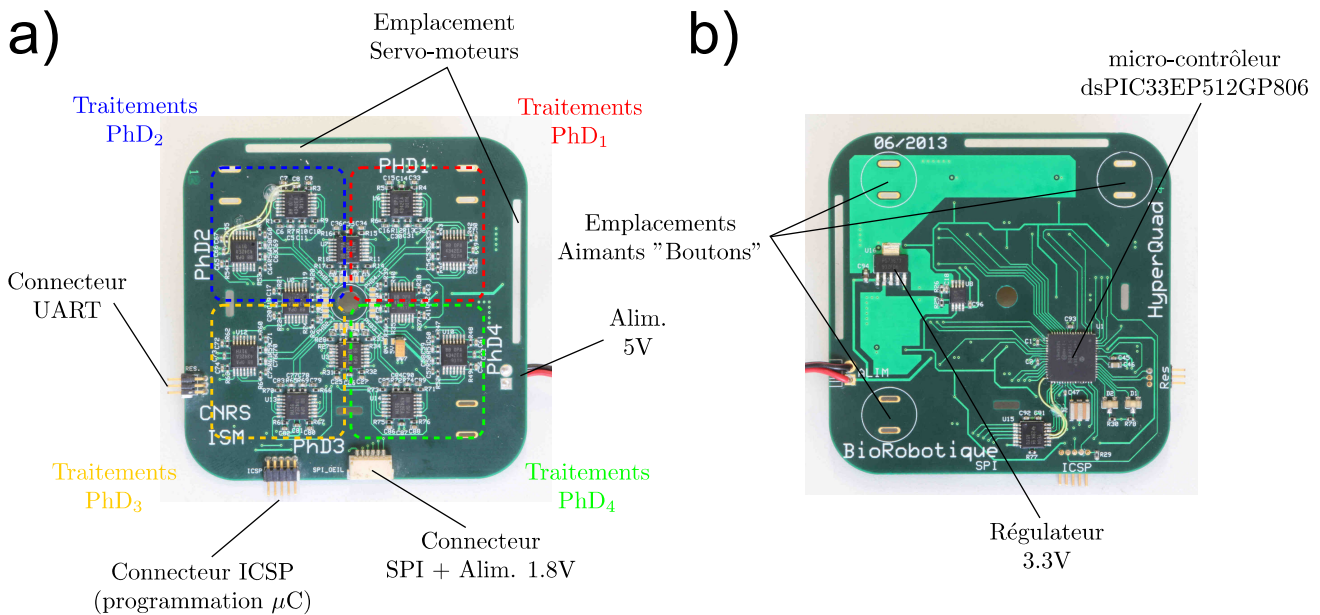


FIGURE IV.17 – a) Carte de traitement photodiodes vue de dessus. b) Carte de traitement photodiodes vue de dessous.

La carte électronique de traitement rattachée à l’œil est illustrée par la figure IV.17. Il s’agit d’une carte en PCB (circuit-imprimé) de quatre couches, permettant de réaliser le traitement analogique lié à chaque pixel. Elle dispose d’un micro-contrôleur 16-bits (le dsPIC33EP512GP806 de chez MicroChip) cadencé à 40 MHz, et disposant de 24 convertisseurs analogiques-numériques (CAN) afin de numériser les signaux photodiodes. Elle présente également une liaison SPI avec l’extérieur (en logique 1.8 V pour communiquer avec la Gumstix⁶ du robot), une liaison UART et une entrée d’alimentation 5 V. Comme

6. La Gumstix est le contrôleur haut niveau de type COM (Computer On Module) et fonctionne sous Linux (voir section 2.2 du chapitre III pour plus d’informations.)

on peut le voir sur la figure IV.17, elle présente des emplacements pour accueillir 2 servomoteurs qui permettront d'orienter l'œil selon 2 directions, ainsi qu'un trou central pour placer la liaison rotule correspondante (voir figure IV.15 dans la section 2.4 pour l'assemblage final). Elle dispose également de 3 emplacements pour accueillir des aimants "boutons" afin de faciliter son montage-démontage sur le robot.

2.5.2 Traitement Analogique

Le courant généré par une photodiode reste assez faible, ce qui nécessite de mettre en place un étage électronique d'amplification de courant, réalisant également le rôle de convertisseur courant-tension de façon à être en mesure de numériser les signaux (par des convertisseurs analogiques-numériques : CAN). De plus, le filtrage passe-bande existant aussi chez l'insecte (voir section 1.2) est reproduit grâce à un filtrage actif passe-bande. Le schéma électronique correspondant au traitement de chaque pixel est décrit par la figure IV.18

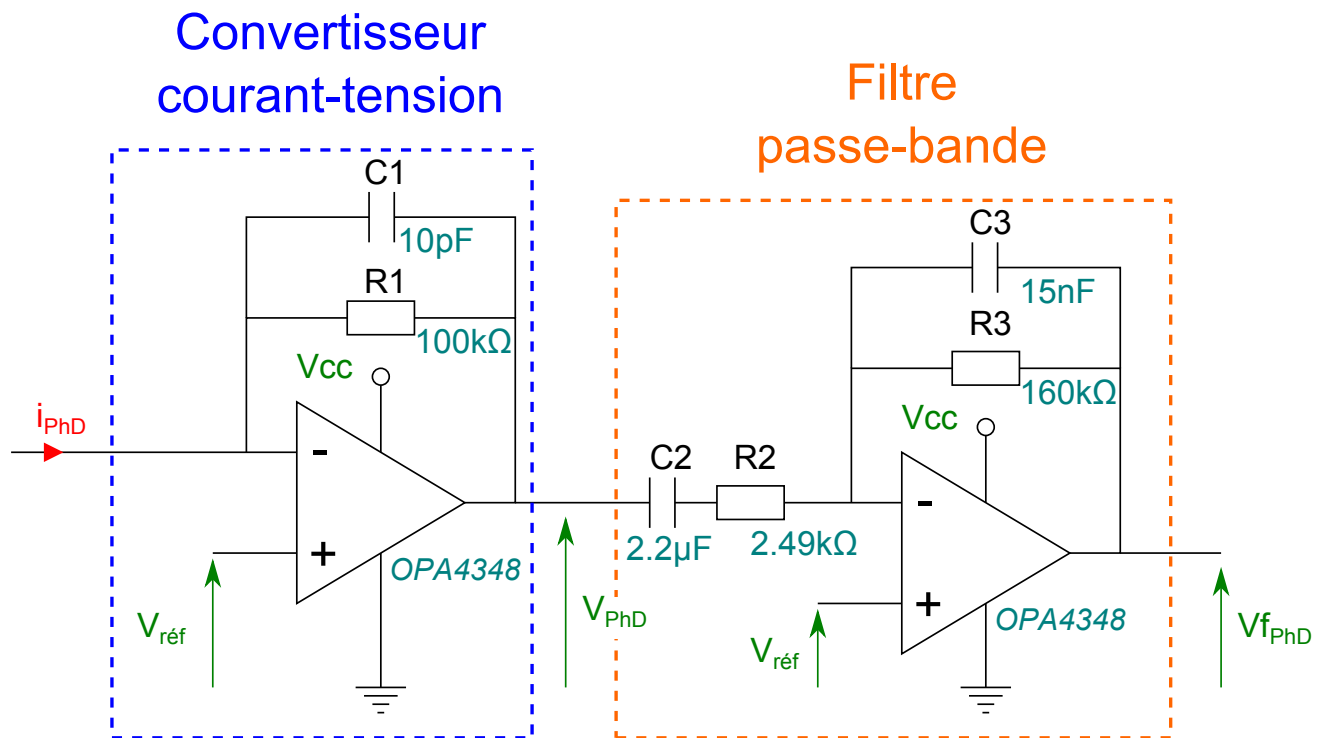


FIGURE IV.18 – Traitement électronique réalisé en parallèle pour chaque pixel. Le courant d'une photodiode i_{PhD} est amplifié, filtré passe-bas et converti en tension (V_{PhD}) par le premier étage. Le deuxième étage réalise la fonction de filtre passe-bande, grâce à un filtre passe-haut cascadié d'un filtre passe-bas. La tension V_{fPhD} est alors l'image filtrée de l'intensité lumineuse perçue par le pixel et est ensuite numérisé grâce à un convertisseur analogique-numérique 12-bits.

Le circuit a été dimensionné de façon à obtenir un filtrage passe-bande centré à 44 Hz avec une bande passante effective de 17 Hz à 113 Hz (définie à -3 dB par rapport au gain statique du système

de conversion courant-tension).

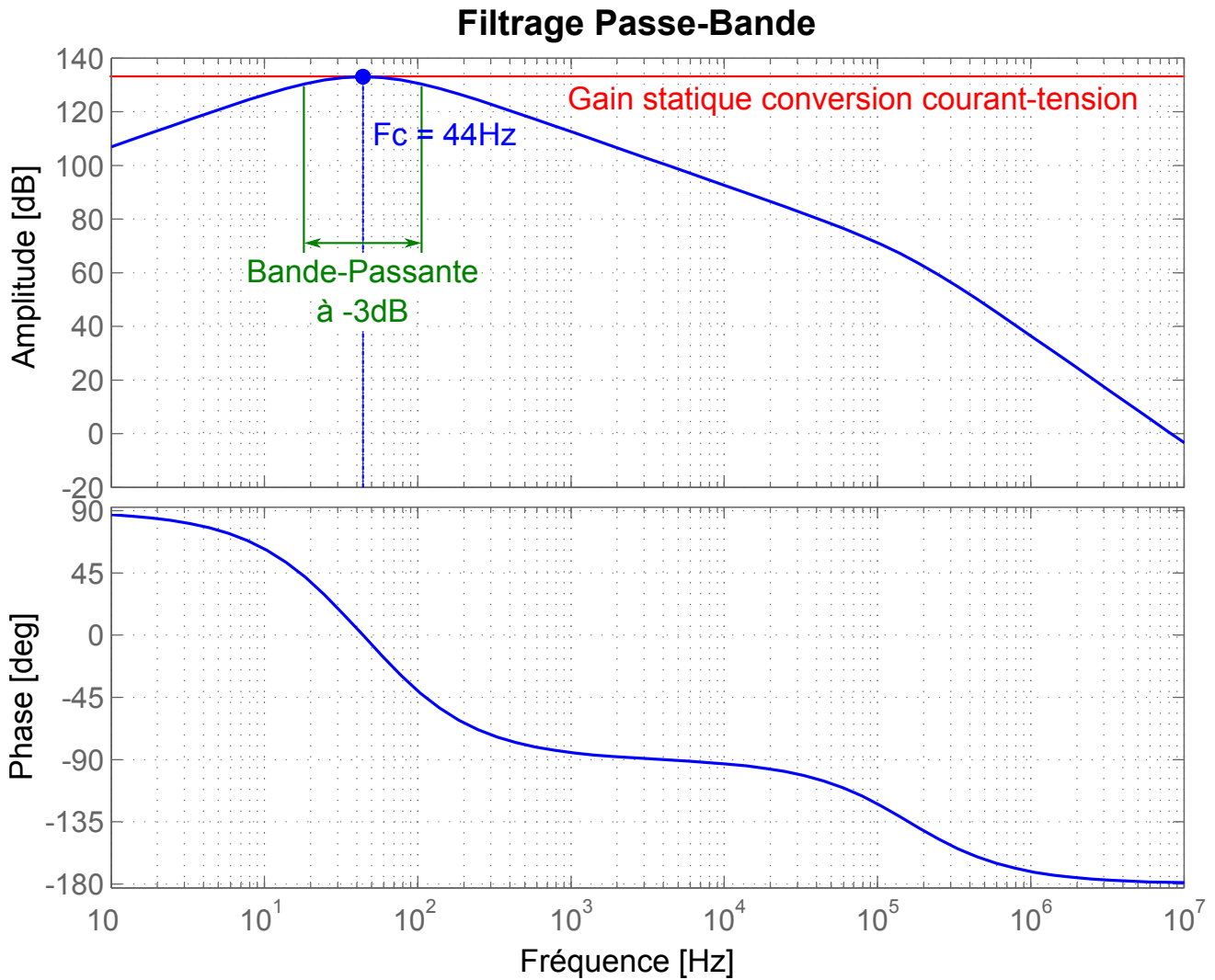


FIGURE IV.19 – Diagramme du Bode du filtrage analogique réalisé pour chaque pixel. Le gain statique de 133 dB correspond à l'amplification de courant, le filtrage passe-bande est centré en 44 Hz et la bande passante est [17 Hz-113 Hz].

Le diagramme de Bode du filtre obtenu est donné par la figure IV.19 et présente la fonction de transfert suivante :

$$V_{f_{PhD}}(p) = V_{réf}(p) + \frac{R_1}{R_1 C_1 p + 1} \cdot \frac{R_3 C_2 p}{(R_3 C_3 p + 1)(R_2 C_2 p + 1)} i_{PhD}(p) \quad (IV.9)$$

Avec $V_{f_{PhD}}$ la tension filtrée correspondante au courant i_{PhD} . $V_{réf}$ est une constante égale à $V_{cc}/2$ où V_{cc} est la tension d'alimentation du micro-contrôleur (ici 3.3 V).

Résistances	Valeur	Capacités	Valeur
R1	100k Ω	C1	10pF
R2	2.49k Ω	C2	2.2 μ F
R3	160k Ω	C3	15nF

La tension de référence $V_{réf}$ joue un rôle important car c'est elle qui permet de centrer le signal autour $V_{cc}/2$ et d'assurer l'utilisation des CAN sur toute leur plage de fonctionnement.

D'un point de vue pratique, l'effet passe-haut du filtrage passe-bande, permet de ne retenir que les variations des signaux visuels utiles pour la suite du processus. Le filtre passe bas, permet d'atténuer les bruits hautes fréquences et agit également comme filtre anti-repliement avant la numérisation des signaux.

3 Traitement des signaux visuels

Les signaux visuels de chacun des pixels, vont être étudiés par paire. En effet, il a été montré qu'une fois soumis à une vibration rétinienne, le traitement des pixels par paire permet de localiser très précisément des contrastes de type front ou barre ([Viollet et Franceschini, 2010], [Kerhuel *et al.*, 2012], [Juston *et al.*, 2014]). [Kerhuel *et al.*, 2012] ont notamment mis au point un algorithme de fusion de deux pixels voisins dénommé VODKA, permettant d'améliorer la résolution du capteur d'un facteur allant jusqu'à 900. Ce traitement a pour gros avantage de ne nécessiter que très peu de ressources calculatoires, et peut facilement être réalisé à des fréquences supérieures au kilo-Hertz (kHz) sur des cibles telle que des dsPIC 16-bits. [Juston *et al.*, 2014] ont ensuite proposé certaines évolutions permettant notamment de détecter le type de contraste vu (barre ou front) par le capteur afin d'adapter la réponse de ce dernier.

3.1 Traitement d'une paire de photo-récepteurs

3.1.1 Le traitement VODKA

Le traitement d'une paire de photo-récepteurs, élaboré dans [Kerhuel, 2009, Kerhuel *et al.*, 2012], qui est décrit par la figure IV.20, est assez simple : un filtre analogique réalise une présélection de la porteuse (fréquence de modulation de la vibration) ainsi que la fonction de filtre anti-repliement avant la numérisation des signaux. De même son effet passe haut réalise une pseudo dérivée des signaux photodiodes. La modulation en amplitude est ensuite épurée des fréquences parasites (dues notamment à l'éclairage artificiel des néons, etc.) grâce à un premier filtre sélectif numérique. Puis, la démodulation des signaux est réalisée à l'aide d'un redresseur double alternance (réalisé par une valeur absolue), suivie d'un filtrage passe bas. Finalement, la sortie du capteur codant la position d'un contraste, est obtenue en calculant le rapport de la différence des signaux démodulés sur la somme de ces derniers. La sortie

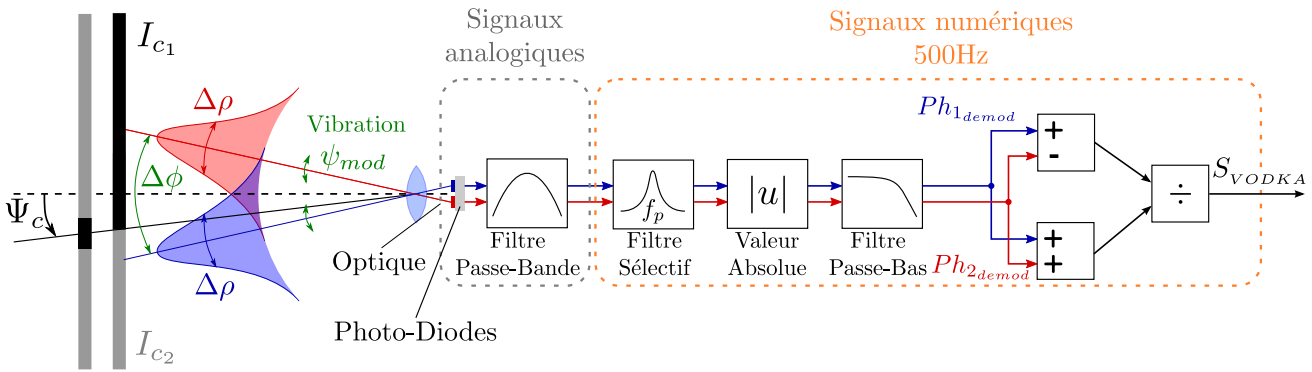


FIGURE IV.20 – Le traitement initial du capteur VODKA. Chacun des pixels balaye le contraste avec une modulation ψ_{mod} , un filtre passe-bas analogique agit comme filtre anti-repliement et pseudo-dérivateur. Ensuite une démodulation numérique est obtenue grâce à un filtrage sélectif, une valeur absolue et un filtre passe bas. Puis la sortie du capteur codant la position angulaire de contraste Ψ_c (ici barre ou front) est obtenue en effectuant la différence sur la somme des signaux démodulés. *Adaptée d'après [Kerhuel et al., 2012].*

caractéristique du capteur dépend bien évidemment de la nature du contraste qui est localisé, comme le montre la figure IV.21. Les travaux de thèse de [Juston, 2013] permettent ensuite de détecter le type de contraste observé, afin d'adapter la réponse du capteur (correction de l'inversion de pente, etc.)

On rappelle ici les équations de base du capteur VODKA menant aux caractéristiques décrites par la figure IV.21 (et utiles à l'élaboration du simulateur décrit en section 4, qui a servi à élaborer les algorithmes de fusion, calibration, etc.) :

Premièrement, on assimile la sensibilité angulaire d'un photo-récepteur à une gaussienne que l'on ne définit que sur une dimension :

$$s(\Psi) = A \cdot \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{\Psi^2}{2\sigma^2}} \quad (\text{IV.10})$$

Avec A un facteur d'échelle, Ψ qui représente l'azimut du photo-récepteur et σ la variance de la gaussienne.

$\Delta\rho$ étant défini comme la largeur à mi-hauteur de la gaussienne, on a :

$$\Delta\rho = \sigma \cdot 2\sqrt{2\ln 2} \quad (\text{IV.11})$$

Et ainsi, IV.10 devient :

$$s(\Psi) = A \cdot \frac{2\sqrt{\ln(2)}}{\Delta\rho\sqrt{\pi}} e^{-\frac{\Psi^2\Delta\rho^2}{4\ln(2)}} \quad (\text{IV.12})$$

La réponse d'une photodiode à un contraste résulte de la convolution entre sa fonction de sensibilité $s(\Psi)$ et la luminance du contraste se trouvant dans son champ de vision. Si l'on applique cela à un front de contraste situé à une position Ψ_c (voir figure IV.20), on obtient d'après [Juston et al., 2014] la

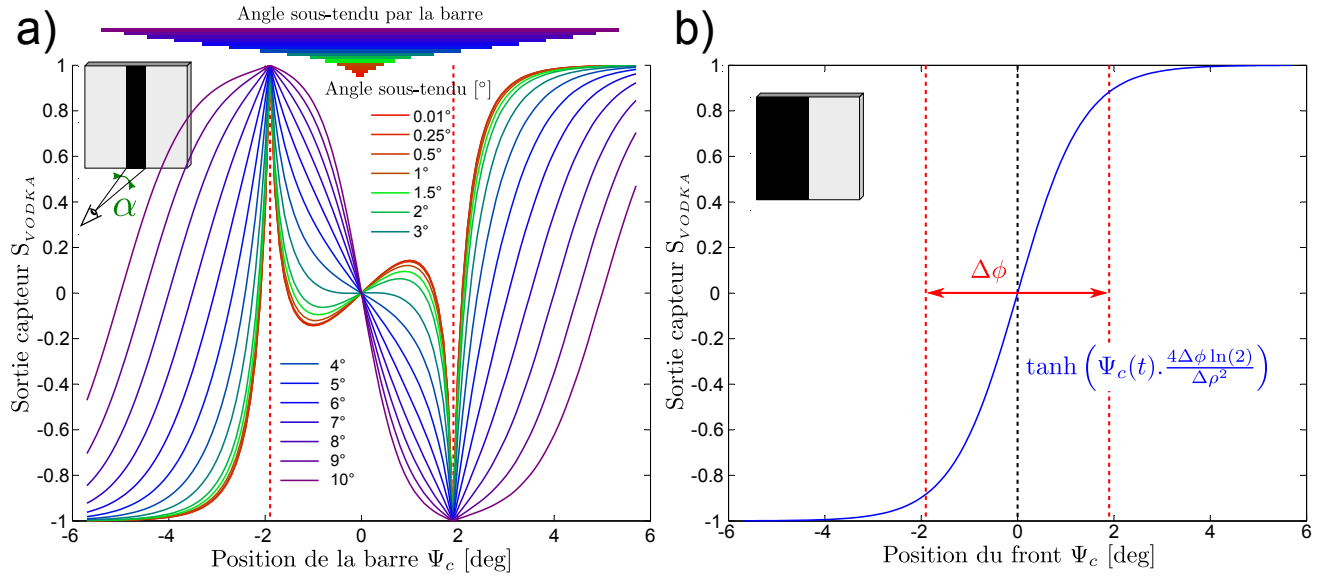


FIGURE IV.21 – a) La réponse caractéristique simulée du capteur VODKA a une barre de largeur L (ici $\Delta\phi = \Delta\rho = 3.8^\circ$). La réponse du capteur varie en fonction de l'angle sous tendu par la barre. On remarque que pour des angles sous tendus trop faibles, la caractéristique du capteur devient non monotone, ce qui empêche de déduire une mesure de la position de la barre à partir de la sortie capteur. *Adaptée de [Kerhuel, 2009]*. b) La réponse caractéristique simulée du capteur VODKA a un front de contraste. Sur les deux graphiques, les lignes pointillées rouges représentent l'intervalle $\left[-\frac{\Delta\phi}{2}; \frac{\Delta\phi}{2}\right]$ correspondant au champ de vision d'une paire de photo-récepteurs.

sortie décrite par :

$$\begin{aligned} PhD(\Psi_c) &= \int_{-\infty}^{\Psi_c} I_{c_1} \cdot s(\Psi) d\Psi + \int_{\Psi_c}^{+\infty} I_{c_2} \cdot s(\Psi) d\Psi \\ &= A \cdot \frac{I_{c_1} - I_{c_2}}{2} \cdot \text{erf}\left(\frac{2\sqrt{\ln(2)}}{\Delta\rho} \Psi_c\right) + A \cdot \frac{I_{c_1} + I_{c_2}}{2} \end{aligned} \quad (\text{IV.13})$$

Où I_{c_1} et I_{c_2} sont l'intensité du contraste (correspondant au produit de l'emissivité du contraste ϵ_{c_i} et de la luminosité ambiante l_a : $I_{c_i} = \epsilon_{c_i} \cdot l_a$). Et erf est définie comme étant la fonction "erreur" suivante :

$$\text{erf}(\Psi) = \frac{2}{\sqrt{\pi}} \int_0^{\Psi} e^{-x^2} dx \quad (\text{IV.14})$$

En présence d'une modulation Ψ_{mod} (entraînant une rotation sinusoïdale des axes optiques d'amplitude A), et en rappelant que les photo-détecteurs (PhD_1 et PhD_2) sont séparés d'un angle $\Delta\phi$, on

peut définir les réponses de chacun des photo-récepteurs :

$$PhD_1(\Psi_c(t)) = PhD \left(\Psi_c(t) + \Psi_{mod}(t) - \frac{\Delta\phi}{2} \right) \quad (IV.15)$$

$$PhD_2(\Psi_c(t)) = PhD \left(\Psi_c(t) + \Psi_{mod}(t) + \frac{\Delta\phi}{2} \right) \quad (IV.16)$$

La sortie du capteur VODKA, qui est décrite dans [Kerhuel, 2009, Kerhuel *et al.*, 2012], est donnée par l'équation suivante :

$$S_{VODKA}(t) = \frac{|P\dot{h}D_1(t)| - |P\dot{h}D_2(t)|}{|P\dot{h}D_1(t)| + |P\dot{h}D_2(t)|} \quad (IV.17)$$

Où $P\dot{h}D_{1,2}(t)$ sont les dérivées temporelles des signaux photodiodes obtenues grâce au filtre analogique passe-bande, qui réalise une pseudo dérivée pour les signaux ayant une fréquence inférieure à celle de la fréquence centrale.

Finalement, en posant comme hypothèses que la vibration Ψ_{mod} est sinusoïdale et que $\dot{\Psi}_c = 0$ (c'est à dire que la position du contraste varie très lentement devant la vitesse de vibration), on peut déduire la réponse du capteur pour un front de contraste :

$$S_{VODKA}(t) = \tanh \left(\Psi_c(t) \cdot \frac{4\Delta\phi \ln(2)}{\Delta\rho^2} \right) \quad (IV.18)$$

La réponse du capteur à un front de contraste est illustrée par la figure IV.21b. On remarque que cette sortie présente des propriétés très intéressantes :

1. La sortie ne dépend pas de l'intensité du contraste vu,
2. la sortie est strictement monotone sur tout le champ visuel,
3. seul les paramètres intrinsèques optiques sont nécessaires pour remonter à la position du contraste.

De même, on peut remonter à l'expression de la position angulaire d'une barre :

$$\begin{aligned} S_{VODKA}(t) &= \frac{|P\dot{h}D_1(t)| - |P\dot{h}D_2(t)|}{|P\dot{h}D_1(t)| + |P\dot{h}D_2(t)|} \\ &= \frac{\left| e^{\frac{\Psi_c(t)\Delta\phi}{2\sigma^2}} \cdot \sinh \left(\frac{\alpha}{2\sigma^2} \left(\Psi_c(t) - \frac{\Delta\phi}{2} \right) \right) \right| - \left| e^{-\frac{\Psi_c(t)\Delta\phi}{2\sigma^2}} \cdot \sinh \left(\frac{\alpha}{2\sigma^2} \left(\Psi_c(t) + \frac{\Delta\phi}{2} \right) \right) \right|}{\left| e^{\frac{\Psi_c(t)\Delta\phi}{2\sigma^2}} \cdot \sinh \left(\frac{\alpha}{2\sigma^2} \left(\Psi_c(t) - \frac{\Delta\phi}{2} \right) \right) \right| + \left| e^{-\frac{\Psi_c(t)\Delta\phi}{2\sigma^2}} \cdot \sinh \left(\frac{\alpha}{2\sigma^2} \left(\Psi_c(t) + \frac{\Delta\phi}{2} \right) \right) \right|} \end{aligned} \quad (IV.19)$$

Cette réponse plus complexe montre clairement une dépendance à l'angle sous tendu α de la barre. Cet angle α se définit comme suit :

$$\alpha = \tan^{-1} \left(\frac{l_b}{D} \right) \quad (IV.20)$$

Où l_b est la largeur de la barre (en mètres) et D la distance entre la barre et le capteur VODKA. La figure IV.21a, montre bien l'influence de cet angle sous-tendu. On remarque que pour des angles

sous-tendus trop faibles, la caractéristique du capteur n'est plus monotone, ce qui ne permet plus de remonter à la position de la barre (il y a plusieurs positions Ψ_c de barre qui donne la même sortie capteur.) Dans le cas d'une application à la stabilisation d'un drone, comme dans le chapitre V, cela ajoute une contrainte sur la largeur de la barre en fonction de la hauteur de vol D , comme l'indique l'équation (IV.20).

3.1.2 Adaptations et améliorations

La formulation initiale du capteur VODKA avait été pensée pour optimiser les performances statiques du capteur. Le champ visuel était seulement de quelques degrés ($\pm 1^\circ$), et la bande passante du capteur n'excédait pas 5 Hz.

Ici, le capteur est appliqué à la stabilisation en vol libre d'un aéronef de type quadrirotor. Il est donc important, afin d'assurer de bonnes performances, d'augmenter un peu la bande passante du système.

Pour cela, quelques modifications sont apportées au traitement du capteur VODKA :

- La fréquence de vibration est augmentée à 55 Hz, ainsi que le filtrage sélectif,
- Une étape de détection d'enveloppe est ajoutée à la phase de démodulation,
- La bande passante du filtre passe-bas, servant à supprimer les derniers résidus de la modulante restants, passe de 5 Hz à 40 Hz et est augmentée à l'ordre 6.

On rappelle ici la configuration des différents filtres utilisés par [Kerhuel *et al.*, 2012] que l'on compare aux nouveaux réglages permettant une augmentation non négligeable de la bande passante tout en maintenant un bon rapport signal sur bruit (RSB) :

Capteur	Filtre	Bande passante	Fonction de transfert
	Passe-Bande analogique	[20 Hz; 116 Hz]	$30. \frac{50.2\pi}{p + 50.2\pi} \cdot \frac{p}{p + 50.2\pi}$
VODKA	Filtre sélectif	40 Hz	$0.036. \frac{1 - z^{-2}}{1 - 1.68z^{-1} + 0.93z^{-2}}$
	Filtre Passe-Bas	5 Hz	$9.447.10^{-4} \frac{(1 + z^{-1})^2}{1 - 1.911z^{-1} + 0.915z^{-2}}$
	Passe-Bande analogique	[17 Hz; 113 Hz]	$64. \frac{66.2\pi}{p + 66.2\pi} \cdot \frac{p}{p + 29.2\pi}$
EyeRing	Filtre sélectif	55 Hz	$\frac{0.9282 - 1.69z^{-1} + z^{-2}}{1 - 1.69z^{-1} + 0.9282z^{-2}}$
	Filtre Passe-Bas	40 Hz	$\frac{0.3951z^{-1}}{(1 - 0.6049z^{-1})^6}$

TABLE IV.4 – Comparaison des différents filtres utilisés pour le traitement des photo-récepteurs pour le capteur VODKA et le nouveau capteur EyeRing. Le filtrage analogique reste quasi-inchangé, mais le filtrage sélectif passe de 40 Hz à 55 Hz dû à une augmentation de la fréquence de vibration. Le filtrage passe bas quant à lui, passe d'une bande passante de 5 Hz à 40 Hz et l'ordre du filtre est augmenté à 6 (au lieu de 2) pour assurer un bon rejet des résidus de la modulante.

La figure IV.22 résume le nouveau processus de localisation d'un contraste au niveau d'une paire de photo-récepteurs, et montre l'évolution de signaux réels au cours des différentes étapes du processus.

L'étape de détection d'enveloppe est réalisée selon l'algorithme suivant :

Entrée : $PhD(t)$ /* Le signal dont l'enveloppe doit être détectée	*/
Sortie : $Env(t)$ /* L'enveloppe détectée	*/
Constante : $\tau = \frac{2}{2\pi f_{mod}}$ /* Constante de temps de la décroissance exponentielle	*/
si $PhD(t) < Env(t - 1)$ alors	
/* Décroissance exponentielle (Euler)	*/
$Env(1) = Env(t - 1) + \frac{T_s}{\tau} (PhD(t) - Env(t - 1));$	
sinon	
/* Sélection de l'enveloppe	*/
$Env(t) = PhD(t);$	

Algorithme 1 : Détection de l'enveloppe supérieure

L'ajout de la détection d'enveloppe permet d'améliorer le rapport signal sur bruit et donc de rendre le filtre passe bas qui suit moins sélectif.

La figure IV.23 montre que la détection d'enveloppe permet d'obtenir un signal démodulé plus proche de l'enveloppe initiale. Le signal a donc une amplitude plus grande, et avec une amplitude

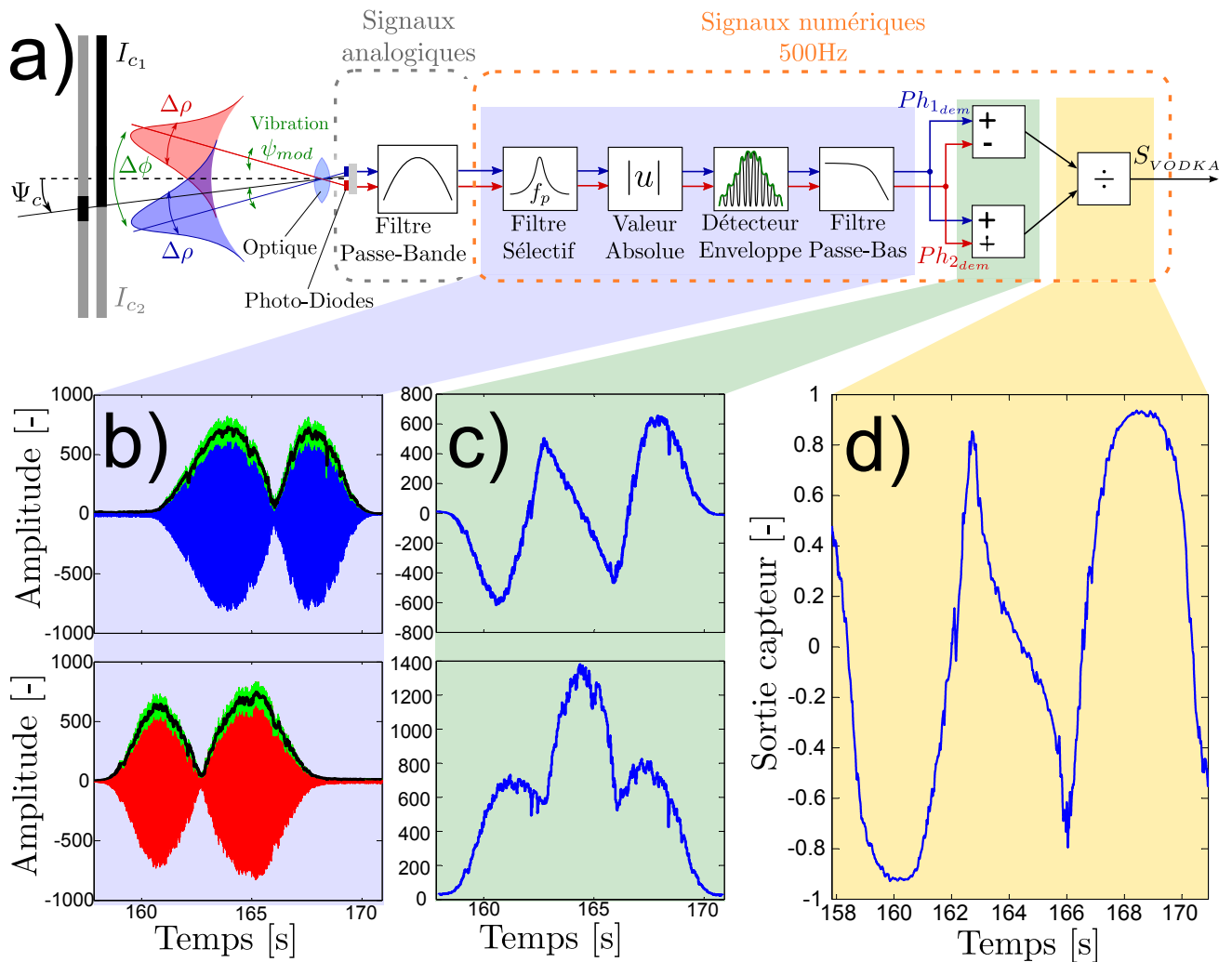


FIGURE IV.22 – a) La chaîne de traitement d'une paire de photo-récepteurs. Une vibration de l'optique permet de moduler la position du contraste, puis un premier filtre analogique effectue une pseudo-dérivée et agit également comme filtre anti-repliement. La porteuse est ensuite extraite à l'aide d'un filtre passe bande, une valeur absolue combinée à un détecteur d'enveloppe permet après un filtrage passe bas d'extraire l'enveloppe de chaque photo-récepteur ($Ph_{i_{dem}}$). b) Visualisation de la démodulation, les signaux bleus et rouges représentent les signaux avant application de la valeur absolue. Les signaux verts sont obtenus grâce au détecteur d'enveloppe une fois la valeur absolue appliquée et les courbes noires représentent l'enveloppe finale après application du filtre passe-bas. c) En haut, la différence des signaux démodulés $Ph_{1_{dem}} - Ph_{2_{dem}}$ et en bas la somme des signaux $Ph_{1_{dem}} + Ph_{2_{dem}}$. d) La sortie du capteur obtenue pour une barre de largeur 2 cm situé à 1.32 m (soit un angle sous tendu $\alpha = 0.87^\circ$) résultant de la différence sur la somme des signaux démodulés.

résiduelle de la modulation qui est du même ordre que celle des traitements VODKA ; le rapport signal sur bruit se voit donc amélioré de 39% ($RSB \times (1260 - 904)/904$). Le retard induit par la démodulation a également été réduit de 37%, et est maintenant de 65 ms. L'augmentation de la bande passante permet de continuer à localiser le contraste de façon satisfaisante, même lorsque celui-ci se "déplace rapidement"

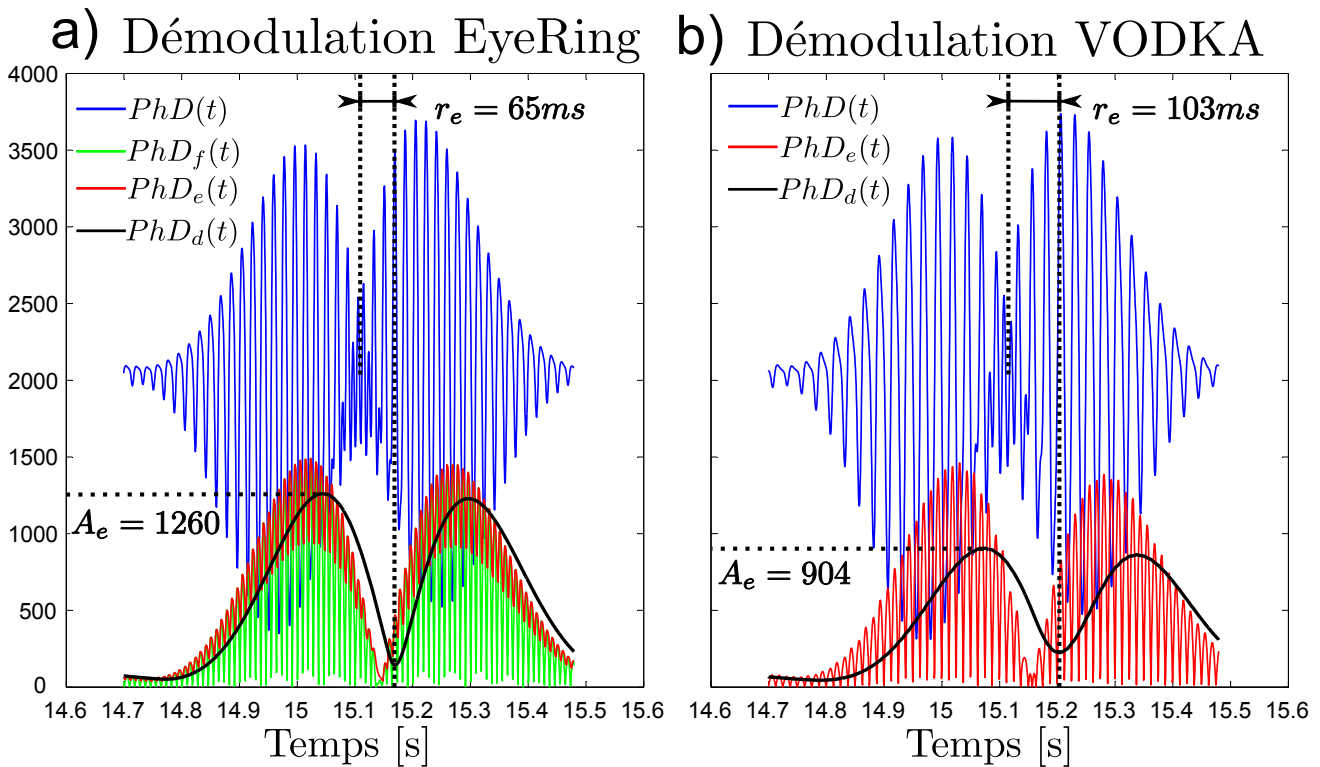


FIGURE IV.23 – Démodulation des signaux photodiodes (simulation) pour une barre défilant à $15.5^\circ \cdot \text{s}^{-1}$. a) En bleu le signal photodiode en sortie du CAN et modulé à 55 Hz, en vert le même signal une fois passé dans le filtre sélectif et la valeur absolue appliquée. La courbe rouge correspond à l'enveloppe identifiée par le détecteur d'enveloppe. La courbe noire représente l'enveloppe filtrée par le filtre passe-bas d'ordre 6. La démodulation complète entraîne un retard de 65 ms et l'amplitude du signal démodulé est 40% plus grande que celle obtenue avec la démodulation VODKA classique. b) Le signal photodiode modulé à 40 Hz est en bleu. La courbe rouge est l'enveloppe obtenue par application d'une valeur absolue, elle est ensuite filtrée à 5 Hz au second ordre pour obtenir la courbe noire. La démodulation entraîne un retard de 103 ms, soit 37% plus élevé qu'avec la méthode exposée en a).

dans le champ visuel. Les figures IV.24a-b montrent la réponse du capteur avec les deux traitements, lorsqu'une barre défile à différentes fréquences de défilement. Ici une fréquence de 1 Hz correspond à une vitesse de défilement de $7.5^\circ \cdot \text{s}^{-1}$. On remarque, que la bande passante a été nettement améliorée, étant donné que EyeRing répond encore avec 40% d'amplitude à une stimulation de 10 Hz (cercle violet de la figure IV.24a) alors que la réponse est "plate" pour le (cercle violet de la figure IV.24a). On remarque également sur la figure IV.24a que le retard constant de $r_{EyeRing} = 65 \text{ ms}$ est de moins en moins négligeable plus la fréquence augmente. En effet pour des vitesses élevées telle que $75^\circ \cdot \text{s}^{-1}$ cela correspond à une erreur de près de 5° par rapport à la position réelle de la barre.

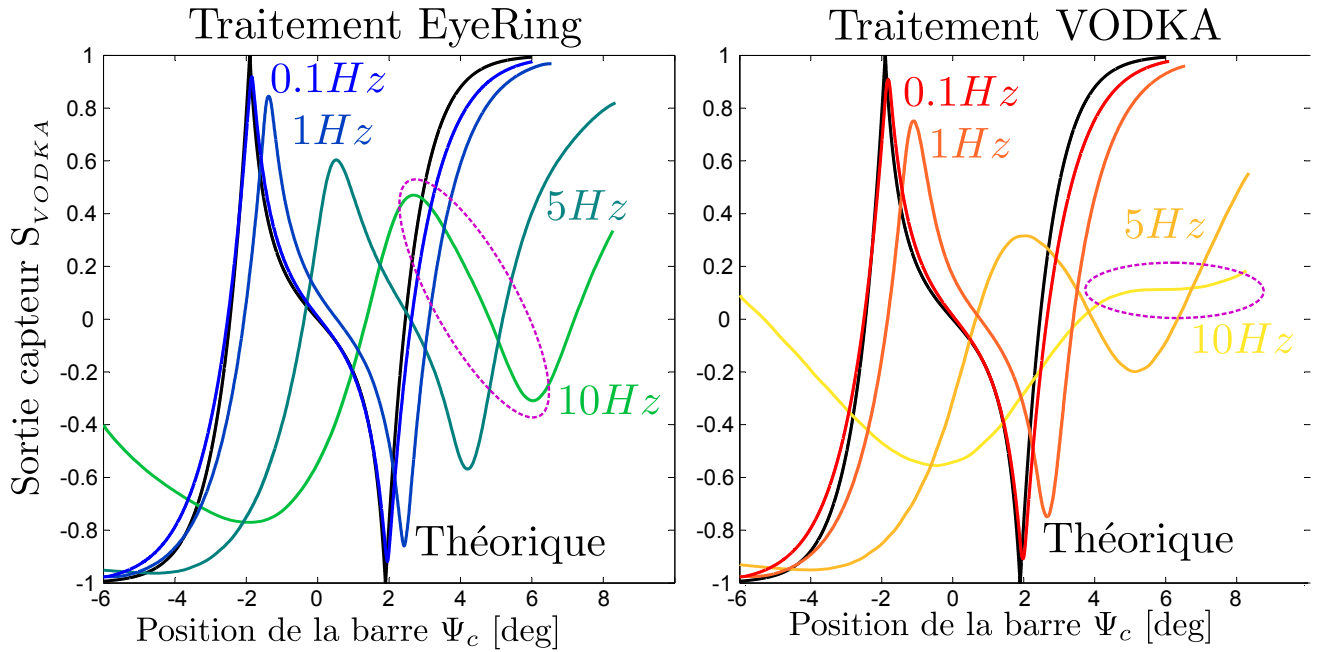


FIGURE IV.24 – Réponses simulées du capteur VODKA pour différentes fréquences de défilement d'une barre ; la barre fait des allers-retours de forme triangulaire devant le capteur. Une fréquence de 10 Hz correspond à une vitesse de défilement de $75^\circ \cdot s^{-1}$. a) Réponses du capteur avec la nouvelle démodulation en fonction de la position de la barre. La réponse du capteur est très proche de la réponse théorique pour des fréquences de défilement faibles. Plus la fréquence augmente et plus la caractéristique se voit déformée et en retard par rapport à la position réelle (le retard est ici en degré ; le retard en seconde reste le même). A 10 Hz, le capteur répond encore avec une amplitude de 40% de la réponse théorique. b) Avec les anciens traitements, le capteur n'arrive pas à répondre correctement à partir de 5 Hz. Pour une fréquence de défilement de la barre à 10 Hz, on voit que le capteur n'est plus en mesure de répondre étant donné que la réponse est totalement "plate". Les cercles violets mettent en évidence la réponse des deux capteurs pour la même vitesse de défilement ($75^\circ \cdot s^{-1}$).

3.1.3 Influences des asymétries des paramètres optiques

Dans la section 2.2, j'ai montré qu'il était très difficile voir impossible d'obtenir les mêmes paramètres optiques pour chacune des paires de photo-récepteurs, d'un point de vue pratique. Les caractéristiques théoriques du capteur en réponse à une barre, ou un front de contraste présentées dans les figures IV.21a et IV.21b se voient donc déformées et les équations (IV.18) et (IV.19) deviennent plus complexes. En

repartant de (IV.17), on obtient pour la réponse à un front :

$$S_{1,2VODKA}(t) = \frac{\left| k_1 \frac{\dot{\Psi}_c(t)}{\sigma_1 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_1^2}} \right| - \left| k_2 \frac{\dot{\Psi}_c(t)}{\sigma_2 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) + \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_2^2}} \right|}{\left| k_1 \frac{\dot{\Psi}_c(t)}{\sigma_1 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_1^2}} \right| + \left| k_2 \frac{\dot{\Psi}_c(t)}{\sigma_2 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) + \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_2^2}} \right|} \quad (\text{IV.21})$$

Où σ_1 et σ_2 sont obtenus grâce à l'équation (IV.11) en utilisant respectivement $\Delta\rho_1$ et $\Delta\rho_2$. k_1 et k_2 ne sont plus égaux dans le cas non idéal, étant donné que les fonctions de sensibilité angulaires n'ont pas toutes la même amplitude. $\Delta\phi_{1,2}$ correspond à l'angle inter-ommatidial de la paire des photodiodes PhD_1 et PhD_2 . La sortie pour une autre paire de photodiodes PhD_i, PhD_j est obtenue en remplaçant les indices 1 et 2 par i et j .

On remarque alors qu'il n'est plus possible de faire apparaître la tangente hyperbolique comme précédemment.

De même en repartant de (IV.17), on obtient pour la réponse à une barre :

$$S_{1,2VODKA}(t) = \left(\left| k_1 \frac{\dot{\Psi}_c(t)}{\sigma_1 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_1^2} + \frac{\alpha}{4}} \sinh \left(\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2} \right) \frac{\alpha}{2\sigma_1^2} \right) \right| - \left| k_2 \frac{\dot{\Psi}_c(t)}{\sigma_2 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) + \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_2^2} + \frac{\alpha}{4}} \sinh \left(\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2} \right) \frac{\alpha}{2\sigma_2^2} \right) \right| \right) \cdot \left(\left| k_1 \frac{\dot{\Psi}_c(t)}{\sigma_1 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_1^2} + \frac{\alpha}{4}} \sinh \left(\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2} \right) \frac{\alpha}{2\sigma_1^2} \right) \right| + \left| k_2 \frac{\dot{\Psi}_c(t)}{\sigma_2 \sqrt{2\pi}} e^{-\frac{\left(\Psi_c(t) + \frac{\Delta\phi_{1,2}}{2}\right)^2}{2\sigma_2^2} + \frac{\alpha}{4}} \sinh \left(\left(\Psi_c(t) - \frac{\Delta\phi_{1,2}}{2} \right) \frac{\alpha}{2\sigma_2^2} \right) \right| \right)^{-1} \quad (\text{IV.22})$$

Où α est l'angle sous tendu par la barre et X^{-1} représente l'opération $\frac{1}{X}$.

On remarque également que l'expression de la sortie du capteur se complique dans le cas de paramètres optiques non complètement maîtrisés.

Cela rend la construction d'une table permettant d'exprimer la position du contraste en fonction des paramètres optiques et de la sortie du capteur plus compliquée étant donné que la vitesse du contraste intervient (dérivée de la position du contraste, notée $\dot{\Psi}_c(t)$). Contrairement au cas idéal, il n'est donc plus possible de créer une table $\Psi_c(t) = f(S_{i,jVODKA}(t), \Delta\rho_1, \Delta\rho_j, \Delta\phi_{i,j}k_i, k_j)$, car la sortie est de la forme :

$$\Psi_c(t)(t) = f\left(\dot{\Psi}_c(t), S_{i,jVODKA}(t), \Delta\rho_1, \Delta\rho_j, \Delta\phi_{i,j}k_i, k_j\right) \quad (\text{IV.23})$$

La solution explorée ici consiste à construire expérimentalement une table (par calibration) pour des vitesses de défilement de contraste très faibles de façon à déterminer la caractéristique statique du capteur.

3.2 Fusion multi-paires d'un œil

La partie précédente a montré que la réponse du capteur n'est pas linéaire en fonction de la position du contraste. De plus, chaque paire de photodiodes a un axe optique principal décalé de $\Delta\phi$ (environ 3.8°) par rapport à ses paires voisines. Mais l'amplitude de réponse du capteur est seulement de 1 (voir figure IV.21), il y a donc en plus de la non linéarité, un facteur d'échelle qui apparaît. Il n'est donc pas possible d'utiliser la réponse du capteur directement pour obtenir une mesure de la position de la barre. De même en fonction de la position du contraste, il est nécessaire de sélectionner la bonne paire, c'est à dire celle qui pointe sur le contraste.

Le but de cette partie est donc de fournir une méthode permettant :

1. De linéariser la sortie de chaque paire de photo-récepteur de façon à obtenir une véritable mesure de position (voir section 3.2.1),
2. De détecter la paire qui voit le mieux le contraste, et de façon plus générale, détecter si une cible est présente où non dans le champ visuel (voir section 3.2.2),
3. De fusionner la mesure des différentes paires tout en assurant la continuité de la mesure (voir section 3.2.3).

3.2.1 Procédure de calibration

La méthode de calibration choisie consiste donc à faire tourner l'œil lentement sur lui même en face du contraste à calibrer, puis à enregistrer la réponse du capteur pour ce contraste. Ainsi une table prenant en compte l'ensemble des défauts du capteurs, permet de tenir compte du fait que :

- la sensibilité du capteur n'est pas parfaitement gaussienne, comme supposé dans l'équation (IV.11),
- $\Delta\rho_i \neq \Delta\rho_j$,

— des sensibilités angulaires qui ne sont pas les mêmes pour chaque photodiode.

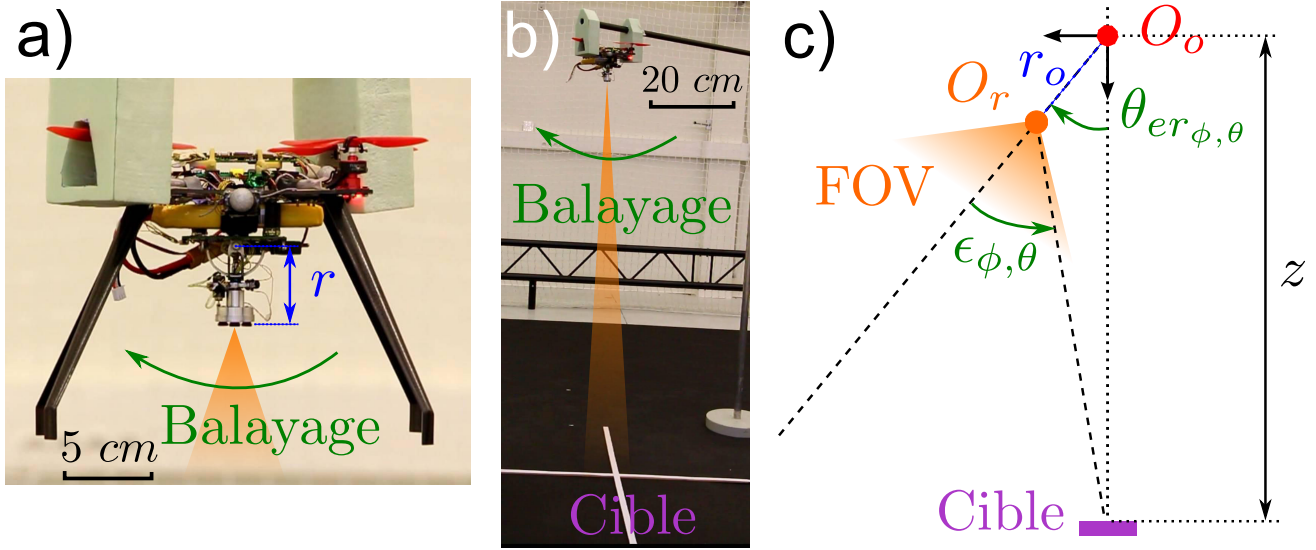


FIGURE IV.25 – a) Zoom sur le robot et son œil lors du processus de calibration des paires de photodiodes. b) Le robot est suspendu au dessus de la cible qui est placée à la verticale en dessous du robot. Ici $z = 1.32$ m. L'œil effectue un balayage latéral permettant de faire parcourir à la cible tout le champ visuel de l'œil. c) Représentation schématique des différentes grandeurs permettant de remonter à la relation géométrique entre l'orientation de l'œil θ_{er_ϕ} ou θ_{er_θ} et la position de la cible Ψ_{c_ϕ} ou Ψ_{c_θ} dans le champ visuel de l'œil.

Pour réaliser cette calibration, on décide de balayer très lentement le champ visuel de l'œil grâce à une rotation des servomoteurs sur un axe puis sur l'autre. Ainsi on peut calibrer 2 des 4 yeux la première fois et les deux autres la fois suivante. La figure IV.25 montre une vue du dispositif de calibration. La position angulaire du contraste est ensuite déduite géométriquement. En supposant que la position de repos de l'œil (servomoteurs en position centrale) pointe sur le centre du contraste, on peut remonter à la position angulaire de la cible par rapport à l'œil :

$$\Psi_{c_{\phi,\theta}} = \theta_{er_{\phi,\theta}} + \tan^{-1} \left(\frac{r_o \sin(\theta_{er_{\phi,\theta}})}{z - r_o \cos(\theta_{er_{\phi,\theta}})} \right) \quad (\text{IV.24})$$

Où r_o est le rayon de l'œil, c'est à dire la distance séparant le centre de rotation de l'œil O_o de l'origine O_r du plan où se situent les capteurs en eux-mêmes et z est la distance verticale entre le centre de rotation de l'œil et la cible. Si on fait l'hypothèse que la distance r_o séparant l'axe de rotation de l'œil des lentilles est négligeable devant la distance z le séparant de la cible, alors on obtient $\Psi_{c_{\phi,\theta}} = \theta_{er_{\phi,\theta}}$.

La procédure de calibration va se faire en deux étapes :

1. **Normalisation des signaux photodiodes :** On identifie un facteur d'échelle pour chacune des photodiodes qui sera ensuite systématiquement appliqué avant même d'effectuer la démodulation. Ce facteur d'échelle permet de faire disparaître les disparités des réponses en amplitude des

différentes photodiodes.

2. **Construction de la table de calibration :** Pour chacune des paires de photodiodes, on construit une table de correspondance entre la sortie de la paire et la position réelle du contraste (Ψ_{c_ϕ} ou Ψ_{c_θ}) selon l'axe en cours de calibration.

La première étape de normalisation est indispensable car elle permet de ne pas perdre la propriété d'insensibilité à l'intensité du contraste. En effet, il a été démontré que le capteur VODKA présente une insensibilité à l'intensité du contraste, c'est à dire aux conditions de luminance (voir 2.5.7 de [Kerhuel, 2009]). Cette propriété remarquable découle du fait que dans les équations (IV.10) et (IV.12) l'amplitude A est considérée comme étant identique pour les deux photo-récepteurs. Or, on a montré lors de la caractérisation (voir figure IV.8), que cette hypothèse n'était pas vérifiée, entraînant donc la perte de cette insensibilité comme le démontre l'équation (IV.21) pour le front et l'équation (IV.22) pour la barre. Cette étape de normalisation permet donc d'avoir le même facteur d'échelle A pour chaque photodiodes et donc de retrouver cette propriété. La figure IV.26 montre les réponses des photodiodes de l'œil n°3 en présence d'une barre avant calibration en a) puis après calibration en b).

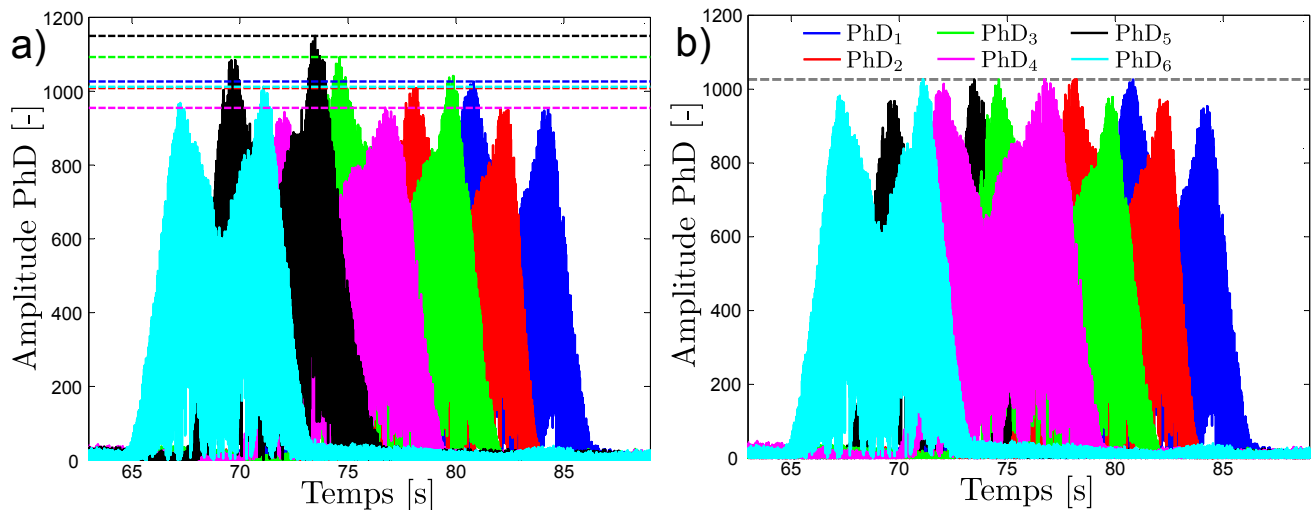


FIGURE IV.26 – Exemple de réponse pour l'œil n°3. a) La réponse non calibrée de chacune des photodiodes. En trait de couleur plein, la réponse des photodiodes, et en pointillés le maximum atteint pour la photodiode correspondante. On observe un écart d'amplitude d'environ 20% selon les photodiodes. b) La réponse des photodiodes une fois normalisée. le trait en pointillé correspond au maximum atteint par chacune des réponses : la normalisation est effectuée arbitrairement pour correspondre à l'amplitude de PhD_1 .

La deuxième étape est un peu plus complexe. Durant le balayage, le contraste va passer successivement dans le champ de vision de chacune des paires. La partie monotone de la réponse caractéristique d'une paire se trouvant, dans le cas d'un front ou d'une barre, dans l'intervalle $[-\frac{\Delta\phi}{2}; +\frac{\Delta\phi}{2}]$ (voir figure IV.21), il est donc nécessaire de sélectionner la paire qui voit le contraste. La méthode de sélection de paires ainsi que de présence de contraste est discutée dans la section 3.2.2. Une fois la zone d'intérêt

de chaque paire sélectionnée, il est encore nécessaire de filtrer la réponse de façon à éliminer le bruit et en extraire une caractéristique monotone.

Le processus permettant de déterminer une caractéristique monotone étant la plus proche possible de la caractéristique réelle du capteur est réalisé comme suit. Le signal de chaque LPU (pour "Local Processing Unit") est filtré sans retard à l'aide d'un algorithme de filtrage basé sur FFT (Fast Fourier Transform). L'utilisation d'un filtrage sans retard est capitale, de façon à ne pas introduire d'offset dans la table entre la position réelle du contraste et la sortie de la LPU. En effet un retard temporelle dû à un filtrage classique, introduirait un biais sur la mesure égale à l'angle parcouru par le contraste (lors de la calibration) durant ce retard. On recherche alors la première fréquence de coupure du filtre permettant de rendre la caractéristique strictement monotone. Pour chaque fréquence de coupure, on recherche le minimum et le maximum local de la caractéristique et on teste la monotonie. L'algorithme 2 décrit ce processus de recherche et la figure IV.27 illustre cette étape et décrit la caractéristique avant filtrage, puis la caractéristique monotone obtenue après filtrage. Cette procédure est répétée pour les 5 LPU de chacun des 4 yeux afin d'obtenir une calibration de l'œil complète.

```

/* Parcours de chaque œil                                     */
pour i ← 1 à 4 faire
  /* Parcours de chaque paire                                 */
  pour j ← 1 à 5 faire
    Monotone ← FAUX;
    /* Recherche de la zone correspondant à la paire j       */
    Index_1 ← TrouvePremierIndex(ZSL(t) == j);
    Index_2 ← TrouveDernierIndex(ZSL(t) == j);
    /* Initialisation automatique de la zone de recherche du min et max */
    Index_3 ← Index_1 - 0.1*(Index_2-Index_1);
    Index_4 ← Index_1 + 0.1*(Index_2-Index_1);
    Fc ← 3; /* fréquence de coupure à 3 Hz                    */
    tant que Monotone==FAUX faire
      Fc ← Fc-0.05;
      LPU_f(Index_3,Index_4) ← Filtre(LPUi,j(Index_3,Index_4), Fc);
      Index_5 ← TrouveMax(LPU_f(Index_3 :Index_4));
      Index_6 ← TrouveMin(LPU_f(Index_3 :Index_4));
      si EstMonotone(LPU(Index_5 :Index_6)) alors
        | Monotone ← VRAI;
      fin
    fin
  TableXi,j ← LPU_f(Index_5 :Index_6); /* Abscisse de la table */
  TableYi,j ← Ψc(Index_5 :Index_6); /* Ordonnées de la table */
fin
fin

```

Algorithme 2 : Détection et construction de la table d'une LPU

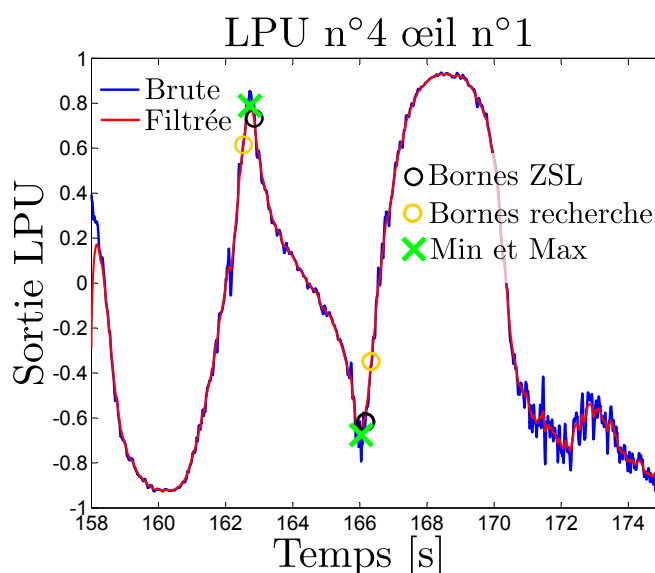


FIGURE IV.27 – Extraction de la caractéristique d’une LPU. En bleu la caractéristique brute mesurée pendant la calibration et en rouge la caractéristique filtrée de façon à être monotone. Les cercles noirs indiquent la zone sélectionnée par le ZSL. Cette zone est agrandie de 10% délimitée par les cercles jaunes. Les maximum et minimum (croix vertes) de cette zone délimitent la caractéristique complète du capteur.

A ce stade, chaque photodiode se voit attribuée un gain de correction d’amplitude et chaque LPU de chaque œil possède une table de calibration (voir figure IV.28). Il ne reste plus qu’à déterminer un critère permettant de sélectionner la bonne LPU (celle qui voit le mieux le contraste) et assurer la continuité de la mesure lors du passage d’une paire à l’autre.

3.2.2 Détecteur ZSL

Le détecteur ZSL est un point clé de la stratégie de fusion des différentes LPU. Originellement décrit par [Viollet et Franceschini, 2001] et [Kerhuel *et al.*, 2010], le ZSL a été entièrement modifié lors de mes travaux, de façon à l’adapter au cas multi-LPU et permettre deux choses :

1. sélectionner la bonne LPU quand le contraste est vu,
2. Détecter lorsqu’aucun contraste n’est vu et forcer la sortie à 0. L’importance de forcer la sortie à 0 en cas d’absence de contraste est discutée dans la partie 4.1 de [Manecy *et al.*, 2013b].

Le critère choisi pour passer d’une paire à une autre résulte de l’observation de la somme des signaux démodulés de chaque LPU présentés par la figure IV.29.

On observe sur la figure IV.29a que la LPU qui voit le contraste, est également la LPU qui possède la somme des ses signaux photodiodes la plus grande. Ce critère très simple permet donc de savoir quelle est la LPU qui voit le contraste. On remarque que les LPU voisines ont alors une somme également élevée, alors que les LPU les plus éloignées ont une somme proche de 0.

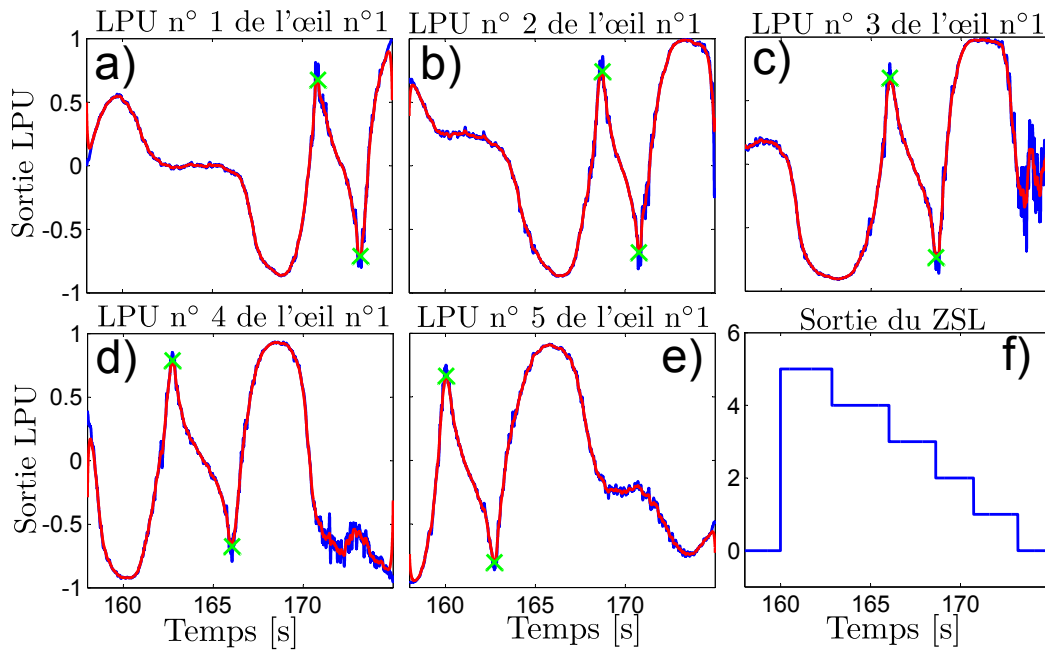


FIGURE IV.28 – a-e) Les caractéristiques statiques identifiées pour les différentes LPU de l’œil 1. En bleu les caractéristiques brutes et en rouge les caractéristiques filtrées et monotones. Les croix vertes représentent la zone sélectionnée pour construire la table. f) La sortie du ZSL donne l’indice de la paire à sélectionner, un indice 0 traduit le fait que le contraste est vu par aucune LPU.

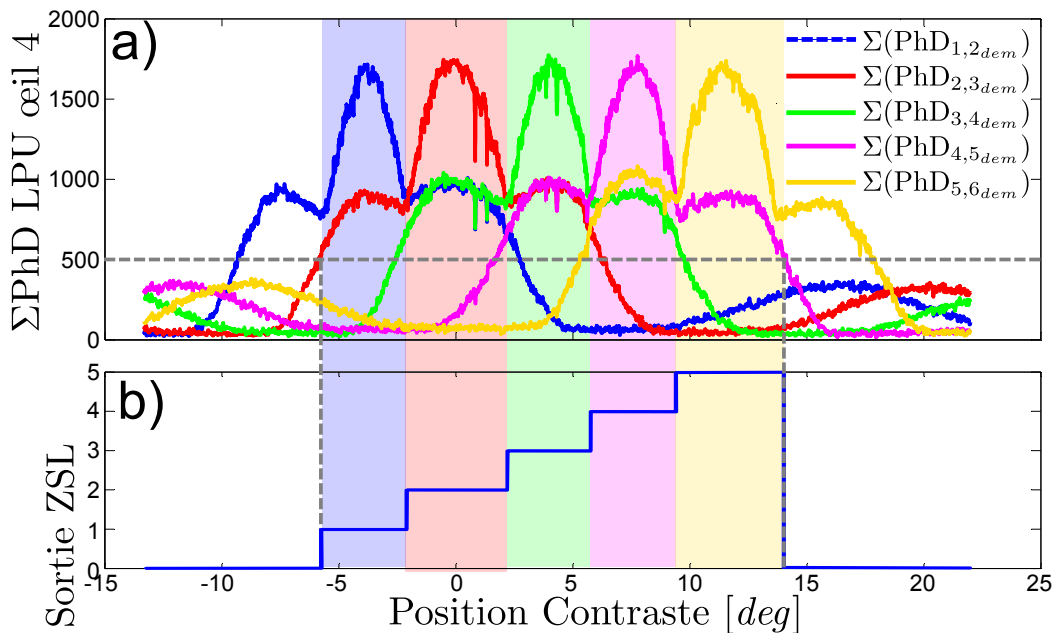


FIGURE IV.29 – a) La somme des signaux photodiodes démodulés de chacune des LPU d’un œil. La LPU choisie, correspond à celle dont la somme est maximum et dont au moins une des sommes voisines est supérieure à un seuil (représentée en pointillé gris). b) Sortie du ZSL en fonction de la position du contraste.

La fonction ZSL est donc réalisée comme suit :

```

S1 ← 500;
S2 ← 400;
IndexMax ← TrouveMax( $\Sigma PhD[](t)$ );
ZSL(t) ← IndexMax;
suivant IndexMax faire
  |
  | cas où 1
  | | si  $\Sigma PhD[IndexMax](t) < S1 + S2$  ou  $\Sigma PhD[IndexMax + 1](t) < S1 + S2$  alors
  | | | ZSL(t) ← 0;
  | | fin
  | | fin
  | | cas où 2, 3, 4
  | | | si  $\Sigma PhD[IndexMax + 1](t) < S1$  et  $\Sigma PhD[IndexMax - 1](t) < S1$  alors
  | | | | ZSL(t) ← 0;
  | | | fin
  | | | fin
  | | cas où 5
  | | | si  $\Sigma PhD[IndexMax](t) < S1 + S2$  ou  $\Sigma PhD[IndexMax - 1](t) < S1 + S2$  alors
  | | | | ZSL(t) ← 0;
  | | | fin
  | | fin
  | fin

```

3.2.3 Fusion multi-paires

La fusion multi-paires est donc obtenue simplement en sélectionnant la sortie de la LPU indiquée par le ZSL après avoir appliqué le facteur de correction. La figure IV.30 résume donc l'ensemble des étapes de la fusion. Une étape de détection d'enveloppe est ajoutée au traitement VODKA classique, améliorant ainsi le rapport signal sur bruit de près de 40%. Puis chaque signal démodulé se voit appliqué le facteur d'échelle déterminé pendant la première phase de calibration, permettant de rétablir une relative invariance à la luminosité ambiante. Puis, après avoir effectué la différence sur la somme des signaux démodulés, on détermine la position du contraste en appliquant la table obtenue lors de la deuxième phase de calibration. Pour finir, les sorties des différentes LPU sont fusionnées en appliquant une sélection grâce à la sortie du ZSL.

Pour conclure sur cette fusion de LPU, la figure IV.31 donne la mesure de position obtenue pour chacun des yeux après calibration. On remarque que chacun des yeux présente une très bonne linéarité

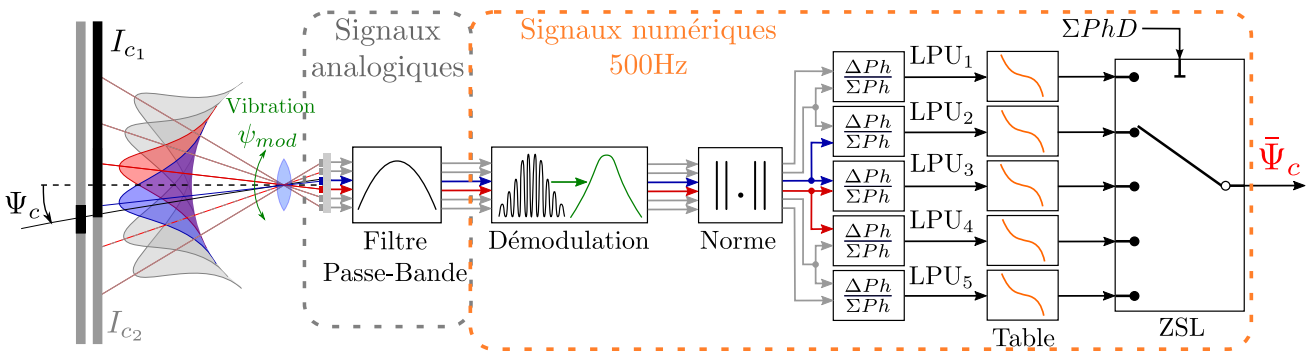


FIGURE IV.30 – Le diagramme complet permettant de passer des signaux des 6 photodiodes à la mesure de position. Le signal de chacune des photodiodes est donc démodulé, puis normalisé. Ensuite la différence sur la somme des signaux démodulés est opérée, formant ainsi 5 LPU (Local Processing Unit). Une table de correspondance est ensuite appliquée de façon à traduire la sortie du capteur en position. Enfin, le ZSL sélectionne la paire qui voit le mieux le contraste pour rediriger sa mesure vers la sortie fusionnée.

et que la sortie du capteur est très fidèle à la position réelle du contraste. On remarque cependant que le bruit présent sur la mesure est un peu plus important lors du passage d'une LPU à une autre, cela s'explique par le fait que le traitement VODKA présente une RSB plus faible proche des bornes de la caractéristique. De plus, le critère utilisé par le ZSL se base sur le $\max()$, qui est une fonction assez sensible aux bruits de mesures lorsqu'il est appliqué à des signaux d'amplitude proches; ce qui est le cas lorsque l'on est en bordure de deux LPU. Cela peut donc entraîner la sélection de la mauvaise paire pendant un bref instant, or la caractéristique à gauche et à droite du champ de vision d'une LPU (FOV_{LPU}) étant assez raide, l'erreur de mesure peut être importante. Ce bruit ajouté par la fonction ZSL ne dérange cependant pas pour avoir un comportement pleinement satisfaisant du capteur, d'autant que celui-ci a pour but d'asservir la position du contraste à 0. En fonctionnement normal, il n'y aura donc que très peu de transition entre LPU, étant donné que la zone de fonctionnement sera autour de la position 0.

Au final, les yeux présentent un champ visuel total compris entre 18.3° et 22° . La sortie du capteur est parfaitement linéaire et la précision est comprise entre 0.076° et 0.091° (écart type). La table IV.5 donne un résumé de la précision de chacun des yeux et de chacune de leur LPU respective.

3.3 Fusion des 4 yeux

Chacun des quatre yeux fournit donc une mesure de la position angulaire d'un contraste avec une précision inférieure à 0.1° sur un champ de vision complet d'environ 20° . Chaque œil peut localiser un contraste du type front, ou barre, et les yeux sont disposés sur un cercle chacun espacé de 90° avec ses voisins. On conçoit donc aisément, qu'il est alors possible de localiser précisément des cibles de type croix ou mire quatre cadrans (voir IV.1), si chaque œil "observe" une des 4 barres de la croix ou un des

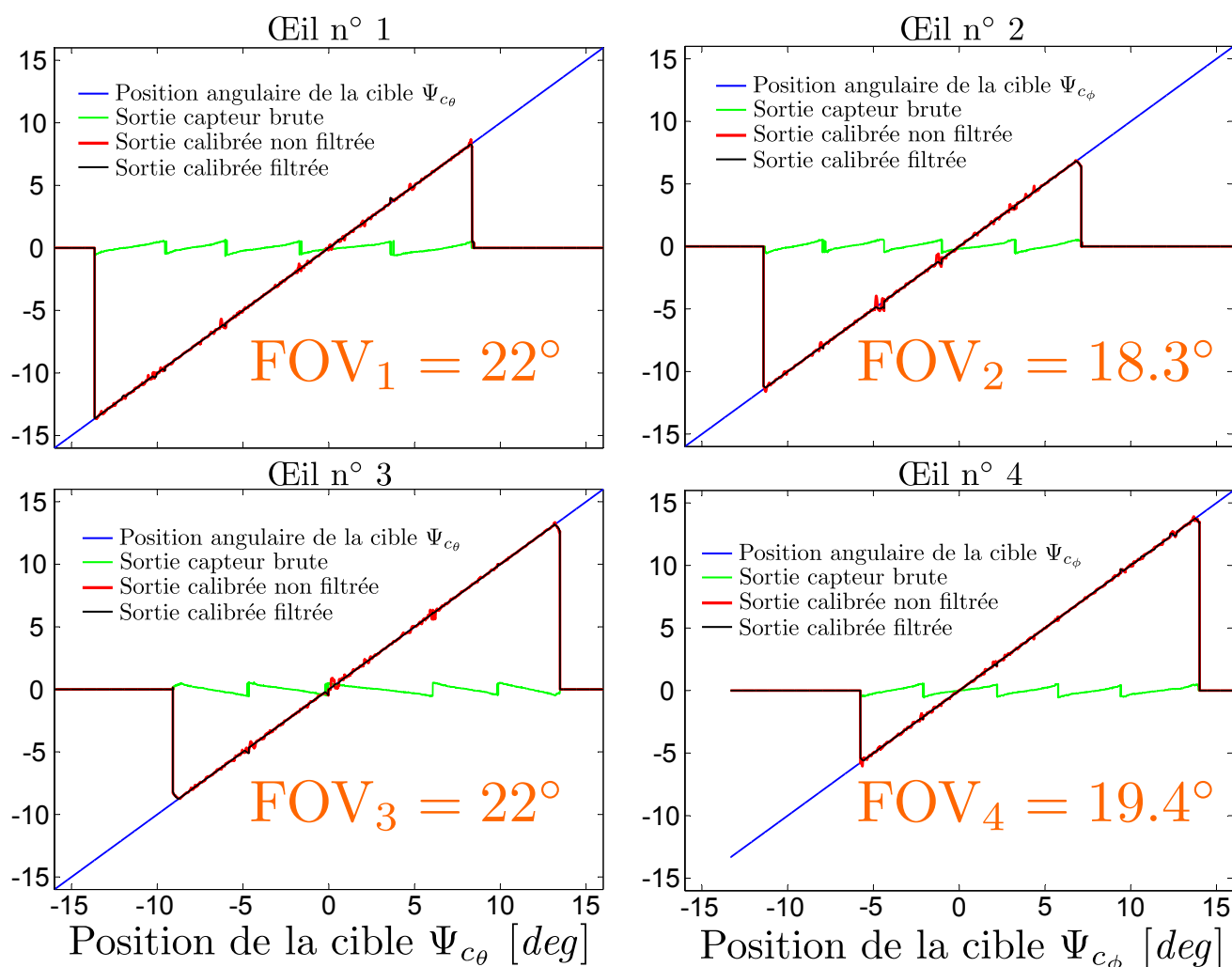


FIGURE IV.31 – Réponse réelle de chacun des yeux pour un balayage de -16° à $+16^\circ$ à $1.67^\circ \cdot \text{s}^{-1}$. Pour chaque œil, la position réelle du contraste est affichée en bleue, la sortie brute de la LPU avant la table est affichée en vert. L'application de la table, permet d'obtenir la mesure en rouge parfaitement linéarisée. En noire, on fait apparaître la mesure filtrée (en post-traitement par FFT une fois encore) ce qui montre qu'en absence de bruit la sortie est parfaitement confondue avec la position réelle, prouvant la fiabilité des différentes tables de calibrations.

quatre fronts de la mire.

On présente donc sur la figure IV.32, la mesure effectuée par chacun des yeux si l'on place une croix orientée de façon arbitraire en dessous de l'œil. On définit le repère orthonormé $(O, \vec{x}, \vec{y}, \vec{z})$ tel que O soit le centre du cercle sur lequel sont disposés les yeux, \vec{x} pointe vers l'œil 2 et \vec{y} vers l'œil 3. On suppose que la cible se trouve à une distance D sur l'axe \vec{z} (c'est à dire face à l'œil, la direction \vec{z} du repère pointant dans la même direction que le regard). On suppose que cette croix est orientée d'un angle ϕ_c selon le roulis, θ_c selon le tangage et d'un angle ψ_c en lacet, par rapport à l'œil. Si l'on note \bar{e}_{r_i} la position angulaire du contraste dans le champ de vision de l'œil i , on peut en déduire les 3

	Erreurs (Œil 1 <i>milli-degrés</i> [m°])					Erreurs (Œil 2 <i>milli-degrés</i> [m°])				
	1	2	3	4	5	1	2	3	4	5
σ_{LPU}	94	57	76	93	76	111	71	57	66	116
Min_{LPU}	-395	-731	-286	-397	-233	-640	-226	-375	-212	-272
Max_{LPU}	502	282	601	405	434	383	577	150	345	679
Moy_{LPU}	4.3	0.6	2.0	1.1	0.3	-15.0	4.4	-2.9	1.3	23.1
FaH_{LPU}	40.0	67.7	50.4	42.1	49.5	35.9	54.3	71.2	60.4	32.6
σ_{oe}			81.3					88.4		
Min_{oe}			-731					-640		
Max_{oe}			601					679		
Moy_{oe}			1.4					2.6		
FaH_{oe}			47.0					44.4		
	Erreurs (Œil 3 <i>milli-degrés</i> [m°])					Erreurs (Œil 4 <i>milli-degrés</i> [m°])				
	1	2	3	4	5	1	2	3	4	5
σ_{LPU}	71	61	107	111	76	53	72	68	58	106
Min_{LPU}	-373	-163	-747	-1083	-373	-181	-200	-231	-272	-331
Max_{LPU}	282	289	537	916	352	206	631	487	331	456
Moy_{LPU}	-5.1	2.6	-2.7	-0.5	-1.9	-0.9	3.1	3.4	2.7	0.5
FaH_{LPU}	53.9	65.2	36.4	34.7	53.1	72.4	57.3	58.9	70.3	35.5
σ_{oe}			91.1					75.9		
Min_{oe}			-1083					-331		
Max_{oe}			916					631		
Moy_{oe}			-1.5					1.8		
FaH_{oe}			43.0					52.4		

TABLE IV.5 – Dans cette table, on présente l’erreur de positionnement pour un balayage complet de -16° à 16° à une vitesse de 1.67° par seconde. La rotation est réalisée par les servomoteurs comme pour la calibration, et les signaux mesure de position sont échantillonnée à 1kHz. Toutes les erreurs de position sont exprimées en milli-degrés (noté [m°]) et sont calculées sur la plage correspondant au champ de vision (FOV) de chaque œil, qui est visible sur la figure IV.31. Pour chaque œil, on précise l’écart type de l’erreur de positionnement (noté σ), l’erreur minimum (notée Min_{LPU}), l’erreur maximum Max_{LPU} , l’erreur moyenne Moy_{LPU} et le facteur d’hyperactivité (noté FaH) pour chaque LPU de l’œil ainsi que les mêmes caractéristiques pour l’œil fusionné. L’écart type est calculé par $\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i^n - E[x])^2}$ où n est le nombre d’échantillon, x_i est l’échantillon i et $E[x]$ est la moyenne de la série. Le facteur d’hyperactivité d’une LPU correspond au rapport entre la précision obtenue pour la LPU j de l’œil i (écart type σ_{LPU}) et l’angle inter-ommatidial correspondant $\Delta\phi_{i,j}$: $\text{FaH}_{LPU,i,j} = \frac{\Delta\phi_{i,j}}{\sigma_{i,j}}$. Et le facteur d’hyperactivité FaH_{oei} de l’œil complet i est le rapport entre l’angle inter-ommatidial moyen $E[\Delta\phi_i]$ et son écart type total σ_i . Au final, on peut dire que les 4 yeux vibrants ont une acuité améliorée d’un facteur allant de 43.0 à 72.4 par rapport à leur acuité initiale.

orientations de la cible. Les deux premières orientations sont triviales :

$$\bar{\phi} = \frac{1}{2}(\bar{\epsilon}_{r_2} + \bar{\epsilon}_{r_4}) \quad (\text{IV.25})$$

$$\bar{\theta} = \frac{1}{2}(\bar{\epsilon}_{r_1} + \bar{\epsilon}_{r_3}) \quad (\text{IV.26})$$

L'orientation selon le lacet dépend, quant à elle, de la géométrie de l'œil, ainsi que de la distance qui sépare l'œil de la cible. Elle est obtenue en moyennant la mesure de l'orientation faite par le couple d'yeux 1,3 et le couple 2,4 :

$$\bar{\psi} = \frac{1}{2} \left[\tan^{-1} \left(\frac{D \cdot \tan \left(\frac{\bar{\epsilon}_{r_1} - \bar{\epsilon}_{r_3}}{2} \right)}{r_c} \right) + \tan^{-1} \left(\frac{D \cdot \tan \left(\frac{\bar{\epsilon}_{r_2} - \bar{\epsilon}_{r_4}}{2} \right)}{r_c} \right) \right] \quad (\text{IV.27})$$

Où r_c est le rayon du cercle sur lequel sont disposés les yeux et D est la distance entre l'œil et la cible (projetée sur l'axe Oz).

3.4 Fusion généralisée pour n yeux

Le concept précédent peut également être étendu à un œil ayant non plus 4 mais n yeux. On commence par introduire la définition de branche pour une cible :

Définition IV.1: On nomme **branche**, une des droites concourantes formant une cible. On suppose qu'une cible présente b droites qui sont toutes concourantes en un même point. Ainsi, une croix dispose de 2 branches, une mire quatre cadrans également. Des exemples de cibles sont donnés dans la figure IV.32a-d. Dans notre cas, on se restreint à l'étude de cible où le nombre de branches b est une puissance de 2 ($b = 2^k$, où $k \in \mathbb{N}$) afin d'assurer que l'angle séparant les branches soit un sous multiple de $\pi/2$.

Les intérêts d'ajouter des yeux supplémentaires sont multiples :

- Selon l'orientation de la cible par rapport à l'œil, dans le cas où il n'y a que 4 yeux, la cible peut ne pas être vue car il y a une branche tout les 90° et le champ visuel d'un œil est de l'ordre de 20° . L'utilisation d'yeux supplémentaires pallie à ce problème.
- Les yeux supplémentaires apportent plus de redondance et donc une fusion de meilleure qualité, moins sensible au bruit de mesure de chaque œil.

On se place exclusivement dans le cas d'un œil complet composé d'un nombre pair de petits yeux élémentaires. On cherche à reconstruire la position angulaire de la cible par rapport à l'œil, c'est à dire l'angle $\bar{\psi}$ traduisant son orientation selon le lacet, ou le "cap", l'angle $\bar{\phi}$ correspondant à la position angulaire de la cible sur l'axe de roulis et l'angle $\bar{\theta}$ sur l'axe de tangage. On définit le repère orthonormé $(O, \vec{x}, \vec{y}, \vec{z})$ tel que O soit le centre du cercle sur lequel sont disposés les yeux, \vec{x} pointe vers l'œil 1 et \vec{y} vers l'œil $n/4$. En se référant à la figure IV.32e, on peut déduire des mesures locales de chacun des yeux élémentaires, la position angulaire de la cible. On suppose cependant que l'œil se situe déjà plus ou moins au dessus de la cible, c'est à dire que au moins deux branches distinctes sont visibles par au moins deux yeux chacune.

Remarque IV.3: L'utilisation de b branches impliquent cependant une restriction sur la mesure de

l'angle ψ , qui ne peut être estimé qu'à un modulo $[\frac{\pi}{b}]$. Cela découle du fait que les cibles sont construites par symétrie de rotation d'angle $\frac{\pi}{b}$.

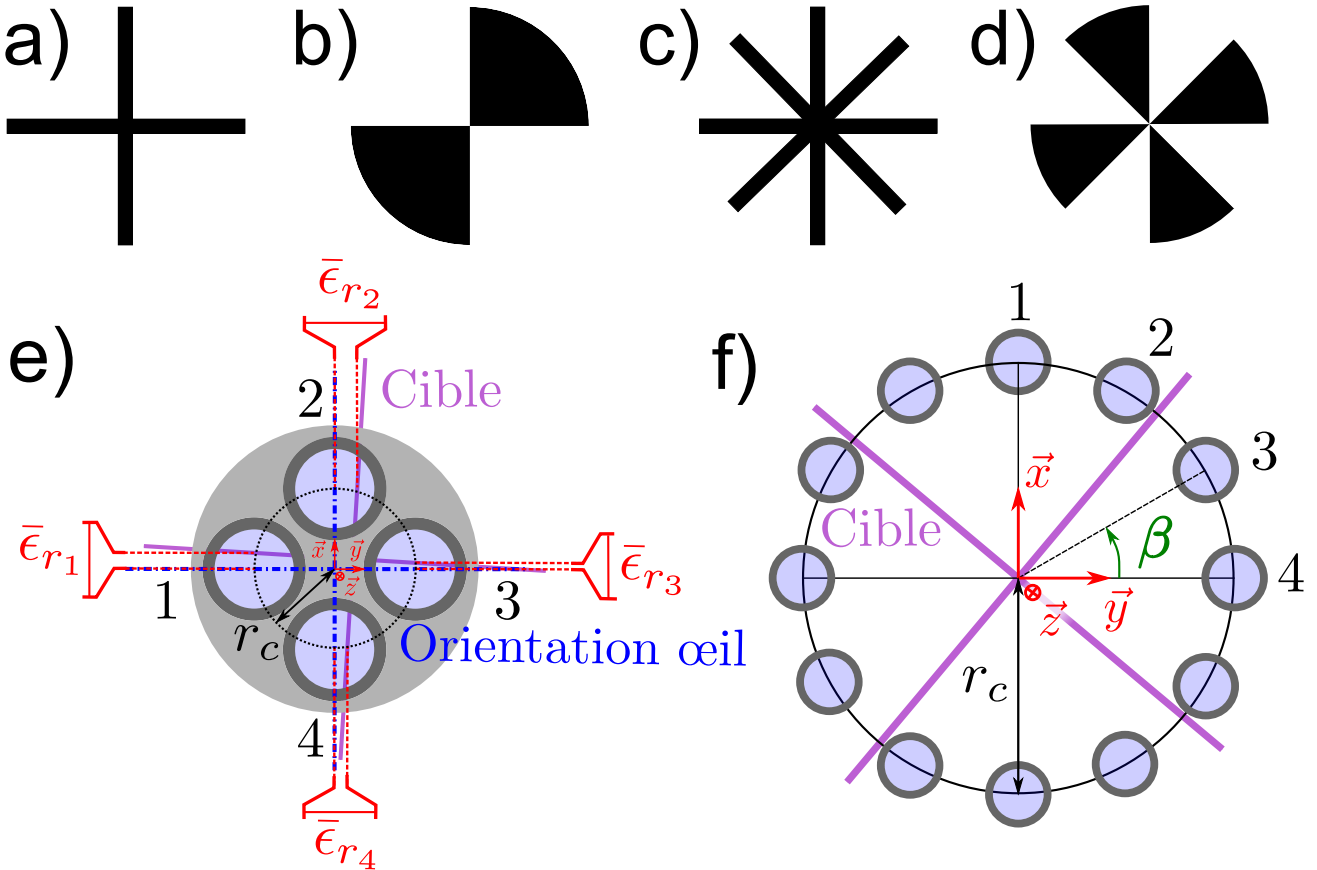


FIGURE IV.32 – a) Cible à 2 branches de type croix. b) Cible à 2 branches de type mire. c) Cible à 4 branches de type croix. d) Cible à 4 branches de type mire. e) Principe de localisation d'une cible. En rouge, on fait apparaître les mesures locales de la position angulaire de la cible dans le champ de vision d'un œil. f) Vue de dessus de l'œil et de son repère orthonormé $(O, \vec{x}, \vec{y}, \vec{z})$. β correspond à l'angle séparant deux yeux, tous disposés sur un cercle de rayon r_c .

On suppose ici que l'œil est composé de n yeux, avec n une puissance de 2. On commence par déterminer, à l'aide de tous les yeux voyants une des branches de la cible, l'angle de lacet $\bar{\psi}$. Une fois cet angle déterminé, on recherche le cap de la cible par rapport à celui de l'œil, c'est à dire l'indice $Index_{Cap}$ correspondant à l'œil étant le mieux "aligné" avec la cible. Puis on détermine l'angle $\bar{\phi}$ et $\bar{\theta}$ en utilisant les yeux : $Index_{Cap} + k \frac{n}{2 \cdot b}$ ou $k \in \mathbb{N} | 1 < k < n/2$. Ainsi la position angulaire de la cible peut être déterminée par l'algorithme suivant :

```

Moy $\bar{\psi}$   $\leftarrow$  0;
n $\bar{\psi}$   $\leftarrow$  0;
pour  $i \leftarrow 1$  à  $n/2$  faire
    si  $ZSL(i) > 0$  et  $ZSL(i + n/2) > 0$  alors
        /* L'angle  $\bar{\psi}$  mesuré par l'œil  $i$  et l'œil diamétralement opposé  $i + n/2$  */
         $\bar{\psi} \leftarrow \tan^{-1} \left( \frac{D \cdot \tan(\bar{\epsilon}_r(i) - \epsilon_r(i + n/2))}{r_c} \right) + \beta \cdot \frac{i-1}{n} \bmod \pi/2;$ 
        /* Mise à jour de l'angle moyen en conséquence */
        Moy $\bar{\psi}$   $\leftarrow \frac{Moy_{\bar{\psi}} \cdot n_{\bar{\psi}} + \bar{\psi}}{n_{\bar{\psi}} + 1};$ 
        n $\bar{\psi}$   $\leftarrow$  n $\bar{\psi}$  + 1;
    fin
fin
/* Recherche de l'indice de l'œil aligné avec le cap */
Index $_{Cap}$   $\leftarrow$  arrondi  $\left( \frac{\bar{\psi}}{\beta} \right) \bmod n;$ 
/* Recherche de l'indice de l'œil opposé au cap */
Index $_{CapOpp}$   $\leftarrow$   $\left( Index_{Cap} + \frac{n}{2} \right) \bmod n;$ 
/* Calcul de l'angle de roulis */
 $\bar{\phi} \leftarrow \frac{1}{2} (\epsilon_r(Index_{Cap}) + \epsilon_r(Index_{CapOpp}));$ 
/* Recherche de l'indice de l'œil orthogonal au cap à gauche */
Index $_{CapOrtG}$   $\leftarrow$  Index $_{Cap} + \frac{n}{4};$ 
/* Recherche de l'indice de l'œil orthogonal au cap à droite */
Index $_{CapOrtD}$   $\leftarrow$  Index $_{Cap} + \frac{3n}{4};$ 
/* Calcul de l'angle de tangage */
 $\bar{\theta} \leftarrow \frac{1}{2} (\epsilon_r(Index_{CapOrtG}) + \epsilon_r(Index_{CapOrtD}));$ 

```


4 Simulateur du système occulo-moteur

Dans cette section, on décrit le simulateur permettant de générer les sorties des photodiodes des yeux à partir d'un environnement visuel simple. Ce simulateur a été utilisé pour mettre en œuvre, tester et valider les algorithmes de fusion multi-paires (choix du critère pour le ZSL) ainsi que les algorithmes de calibration présentés précédemment. Ce simulateur a été d'une grande utilité et a permis de créer les différents algorithmes alors que l'œil était encore en construction. Il a également permis de mieux comprendre certains phénomènes et d'améliorer la conception de l'œil final.

4.1 Modélisation générique du capteur

On commence par élaborer un modèle permettant de décrire la réponse d'une photodiode et de son système de traitement électronique en interaction avec son environnement visuel (fronts et barres contrastés).

4.1.1 Environnement et signaux visuels

L'environnement visuel se résume en un couple de vecteurs représentant n fronts de contrastes successifs. Ce couple est composé d'un premier vecteur de dimension n codant l'intensité lumineuse du contraste I entre deux fronts. Le second vecteur de dimension $n - 1$ code la position X_I de chaque front, c'est à dire la position correspondant à un changement d'intensité dans le référentiel inertiel \mathcal{I} . On fait ici l'hypothèse que l'ensemble des contrastes est situé sur un même plan.

Dans un premier temps, on considère que le capteur visuel se situe face à ce plan à une distance z_{oeil_k} et forme un angle d'incidence Ψ_e . Puis, dans la partie 4.2, le concept sera étendu au cas d'étude final dans lequel l'œil est monté en dessous d'un quadrirotor (on décrira comment obtenir les angles θ_{c_i} à partir de la position et l'attitude du robot et de l'orientation de l'œil). On rappelle que la sensibilité d'une photodiode se décrit par une sensibilité angulaire gaussienne décrite par l'équation (IV.12). La figure IV.33 décrit l'environnement représenté par le simulateur ainsi que les différentes grandeurs géométriques.

La sortie de la photodiode est obtenue en faisant la convolution entre cette sensibilité angulaire et la luminance des contrastes présents dans le champ de vision. On détermine donc par géométrie la position angulaire θ_{c_i} de chacun des fronts de contraste dans le référentiel de la photodiode :

$$\theta_{c_i}(t) = \tan^{-1} \left(\frac{x_{oeil_k} - X_{I_i}}{z_{oeil_k}} \right) - \Psi_e + \Psi_{vib}(t) \quad (\text{IV.28})$$

Où θ_{c_i} est la position angulaire du $i^{\text{ème}}$ front, x_{oeil_k} représente la position de la photodiode, et X_{I_i} est la position cartésienne du $i^{\text{ème}}$ front et $\Psi_{vib}(t)$ représente la modulation générée par la vibration d'amplitude 4° et de fréquence 55 Hz.

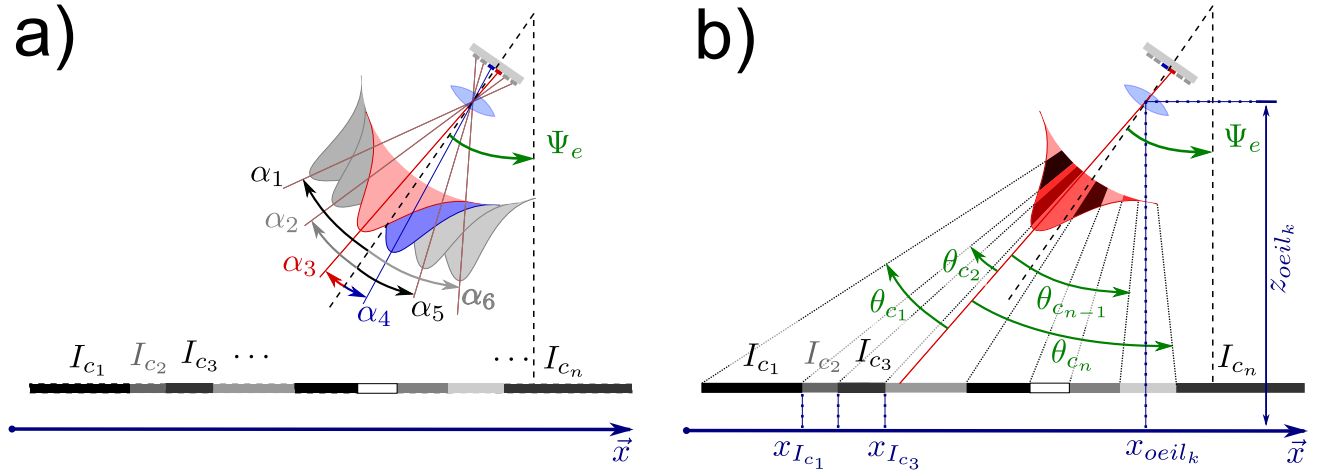


FIGURE IV.33 – a) Description schématique d'un œil du simulateur, composé de 6 photodiodes, face à un contraste. Le contraste est supposé plan et constitué d'une succession de fronts. L'angle Ψ_e décrit l'orientation de l'axe principal de l'œil par rapport à la verticale, et les angles α_j décrivent la position angulaire de l'axe optique de chaque photodiode par rapport à cet axe principal. b) Le signal d'une photodiode est obtenu par convolution entre la sensibilité angulaire de la photodiode et l'intensité lumineuse du contraste. Pour cela, on détermine la position angulaire θ_i de chacun des contrastes dans le repère de chaque photodiode j .

On en déduit ensuite la sortie de chacune des 6 photodiodes d'un œil en rappelant que que celle-ci sont espacées d'un angle $\Delta\phi$. En supposant que les photodiodes sont disposées de part et d'autre de l'axe principal de l'angle d'incidence Ψ_e , on en déduit le signal de la photodiode j de l'œil k :

$$\begin{aligned}
 PhD_j^k(t) &= \int_{-\infty}^{\theta_{c_1}(t)} I_{c_1} \cdot s(\Psi - \alpha_j) d\Psi + \int_{\theta_{c_1}(t)}^{\theta_{c_2}(t)} I_{c_2} \cdot s(\Psi - \alpha_j) d\Psi + \dots + \int_{\theta_{c_{n-1}}(t)}^{+\infty} I_{c_n} \cdot s(\Psi - \alpha_j) d\Psi \\
 &= A \cdot \frac{I_{c_1}}{2} \cdot (1 + \operatorname{erf}(A \cdot (\theta_{c_1}(t) - \alpha_j))) \\
 &+ A \cdot \frac{I_{c_2}}{2} \cdot (\operatorname{erf}(A \cdot (\theta_{c_2}(t) - \alpha_j)) - \operatorname{erf}(\theta_{c_1}(t) - \alpha_j)) + \dots \\
 &+ A \cdot \frac{I_{c_n}(t)}{2} \cdot (1 - \operatorname{erf}(A \cdot (\theta_{c_{n-1}}(t) - \alpha_j)))
 \end{aligned} \tag{IV.29}$$

Avec $s(\Psi)$ la sensibilité angulaire gaussienne des photodiodes décrite par l'équation (IV.12), α_j la position angulaire de l'axe optique de la photodiode j dans le référentiel de l'œil (de sorte que $\Psi = \Psi_e - \alpha_j$ avec $\alpha_j \in \left\{ \frac{-5\Delta\phi}{2}, \frac{-3\Delta\phi}{2}, \frac{-\Delta\phi}{2}, \frac{\Delta\phi}{2}, \frac{3\Delta\phi}{2}, \frac{5\Delta\phi}{2} \right\}$), $A = \frac{2\sqrt{\ln(2)}}{\Delta\rho}$ et I_{c_i} l'intensité du contraste i . La fonction erf correspond à la fonction erreur décrite précédemment par l'équation (IV.14). Le coefficient A permet de passer d'une intensité lumineuse de contraste en lux à un courant en Ampère.

L'intensité lumineuse d'un contraste I_c est obtenue en faisant le produit entre l'intensité du contraste i_c (codée dans $[0, 1]$, 0 représentant un contraste nul, c'est à dire une couleur uniforme et 1 un contraste maximum, c'est à dire un front noir/blanc ou blanc/noir) et l'ambiance lumineuse \mathcal{L} exprimée en lux

(lx) :

$$I_c = i_c \cdot \mathcal{L} \quad (\text{IV.30})$$

4.1.2 Modélisation des traitements électroniques

Comme expliqué dans la section 2.5, la carte électronique regroupe un circuit d'amplification de courant, un circuit trans-impédance, un filtre passe-bande analogique et enfin des convertisseurs analogiques-numériques (CAN). Le simulateur consiste donc en une fonction de transfert continue, suivi d'un CAN implémenté sous la forme d'un BOZ (Bloqueur d'Ordre Zéro), d'un quantificateur puis d'une saturation, et d'un bruit aléatoire additif d'amplitude B (de l'ordre de ± 50). Ce bruit découle principalement de deux phénomènes :

- un bruit dit électronique, provenant des circuits analogiques, mais qui reste très faible,
- un bruit résultant du fait que l'arrière plan de la cible n'est pas parfaitement uniforme (croix blanche sur fond noir non parfaitement uniforme). Cela entraîne donc, lors du balayage par la vibration, une faible modulation d'amplitude des signaux photodiodes, même lorsqu'une paire ne voit absolument pas la cible. Cela est dû à la présence de contrastes faibles présents sur l'arrière plan. Ce bruit dépend donc de la luminosité ambiante étant donné que la luminosité influe sur l'intensité de ces contrastes.

La figure IV.34 montre sous forme de schéma bloc le simulateur permettant de générer la réponse d'une photodiode en fonction de son environnement. Le premier bloc correspond au module "géométrique" permettant de déterminer la quantité de lumière atteignant la photodiode. Le second bloc correspond au filtre analogique passe-bande, et le dernier ensemble correspond au CAN.

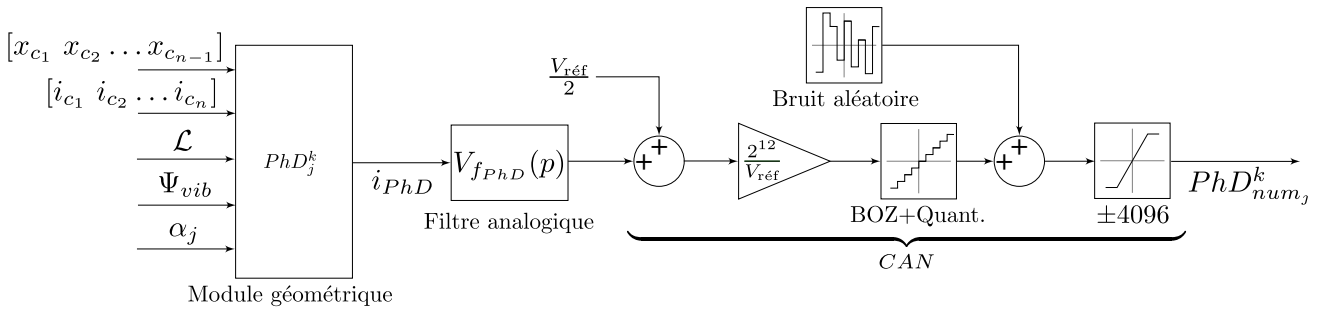


FIGURE IV.34 – Le simulateur de photodiode prend en entrée un vecteur indiquant la position des contrastes (dans le référentiel de l'œil k), un vecteur indiquant leur intensité, la luminance de l'environnement \mathcal{L} , le signal de modulation représentant la vibration Ψ_{vib} , et le décalage angulaire α_j de l'axe optique de la photodiode j par rapport à l'axe principal de l'œil k . Le résultat est un signal numérique (représenté en entier 16 bits) représentant la valeur qui aurait été lue par le micro-contrôle vision en sortie des convertisseurs analogiques numériques (CAN). Le bloc "BOZ+Quant" représente le bloqueur d'ordre zéro (discrétisation liée à l'échantillonnage) et la quantification (arrondi lors du passage en entiers) opérée par le CAN

4.1.3 Identification des différents paramètres

Afin de représenter au mieux le comportement final de l'œil, les angles α_j sont obtenus grâce au tableau IV.1 décrivant les angles inter-ommatidiaux réels entre chaque photodiodes.

Le coefficient A , est obtenu empiriquement en plaçant l'œil face à une barre blanche de 2 cm, positionnée sur un fond noir pour une ambiance lumineuse \mathcal{L} de 1000 lx. On relève alors l'amplitude maximale obtenue sur les signaux photodiodes lorsque l'œil vibre (le maximum est trouvé en déplaçant la barre lentement face à l'œil). Une fois ce maximum identifié (compris entre 0 et 2048 étant donné que les signaux sortant des CAN sont codés sous 12 bits et centrés en 2048), on recherche le coefficient A permettant d'obtenir cette même amplitude dans le simulateur pour les mêmes conditions d'éclairage pour une intensité de contraste de 1 (blanc/noir).

L'amplitude du bruit additif B est obtenue également lors de cette expérience en deux étapes. En désactivant la vibration et en relevant l'amplitude du bruit résiduel sur les signaux photodiodes, on obtient l'amplitude du bruit électronique. Et, en activant la vibration mais en plaçant l'œil face à un fond noir dépourvu de cible, on peut identifier l'amplitude de bruit en fonction de la luminosité ambiante. L'identification de ce niveau de bruit a été effectuée dans les conditions réelles d'expérience, c'est à dire en présence de l'éclairage artificiel des néons de l'arène de vol et le système Vicon allumé, car ces deux systèmes (assez énergétiques) génèrent du bruit dans certaines plages de fréquences (voir études de l'influence de ces systèmes dans la partie 5).

4.2 Modélisation complète du système oculo-moteur

Comme nous l'avons présenté précédemment, le simulateur nécessite de connaître la position x_{c_i} du contraste i dans le référentiel de l'œil k . Afin d'être le plus réaliste possible, on se place dans le cas où le système de vision est placé sous le quadrirotor, et on cherche à déterminer la position des fronts de contraste dans le référentiel de chacun des yeux en fonction de la position du robot, de son attitude et de l'orientation de son système de vision (la position articulaire de l'œil par rapport au robot). On décrit donc, dans la figure IV.35, les repères successifs permettant de construire la matrice de passage permettant d'exprimer les coordonnées d'une cible dans le référentiel \mathcal{O}_k de œil k . Pour cela, on construit le modèle cinématique complet, permettant de déterminer la position de chacun des quatre yeux, en fonction de la position du robot, de son attitude et de la configuration articulaire du système oculomoteur.

La position du robot est décrite dans le repère inertiel \mathcal{I} par la position de son centre de gravité G qui est également le centre du repère fixe lié au robot. On reprend donc le formalisme décrit dans le chapitre III pour passer du repère fixe \mathcal{B} au repère inertielle \mathcal{I} . On considère que le centre de rotation de l'œil est situé à une distance h en dessous du centre de gravité du robot dans le repère fixe (c'est à dire sur l'axe z_B). Ce centre de rotation permet de définir le centre d'un nouveau repère \mathcal{O}_R qui n'est

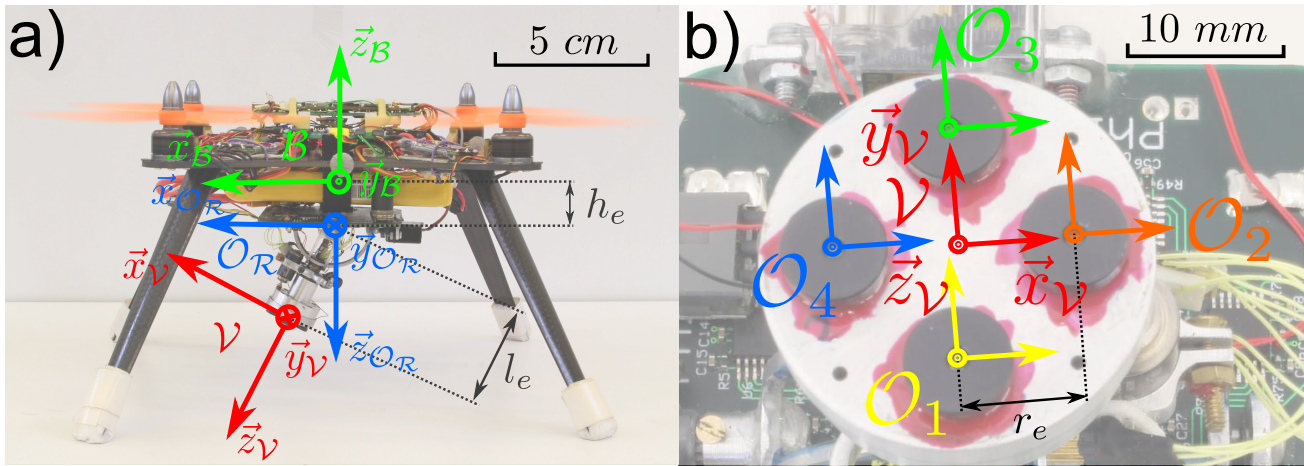


FIGURE IV.35 – a) Description du repère fixe du robot noté \mathcal{B} . Le repère \mathcal{O}_R est obtenu après une translation de distance h_e sur l'axe z_B et une rotation de 180° autour de l'axe x_B afin de l'orienter vers le bas. On définit ensuite le repère de la vision noté \mathcal{V} , qui est obtenu en appliquant dans un premier temps les rotations d'angles θ_{er_ϕ} et θ_{er_θ} (respectivement autour des axes $x_{\mathcal{O}_R}$ et $y_{\mathcal{O}_R}$) et dans un deuxième temps une translation de distance l_e le long du nouvel axe \vec{z} . b) On obtient ensuite les quatre repères \mathcal{O}_1 , \mathcal{O}_2 , \mathcal{O}_3 et \mathcal{O}_4 en appliquant des translations permettant de se placer au centre de l'optique de chacun des yeux.

autre que le repère \mathcal{B} traduit d'une distance $-h$ sur l'axe z_B . L'orientation de l'œil est décrite dans ce repère par deux angles θ_{er_ϕ} et θ_{er_θ} représentant respectivement l'orientation de l'œil selon l'axe de rotation $x_{\mathcal{O}_R}$ et $y_{\mathcal{O}_R}$. De ces deux angles, on déduit une matrice de rotation R_e qui permet de définir un nouveau repère \mathcal{O}'_R de sorte que $\mathcal{O}'_R = R_e \cdot \mathcal{O}_R$. On suppose ensuite que le plan comprenant les 4 lentilles, est distant d'une longueur l_e du centre de rotation de ce dernier repère \mathcal{O}'_R . On obtient alors un dernier repère \mathcal{V} qui est celui de la vision, et dans lequel on peut exprimer la position x_{oeil_i} de chacun des yeux :

$$\begin{cases} {}_V X_{oeil_1} = (r_e \ 0 \ 0)^T \\ {}_V X_{oeil_2} = (0 \ -r_e \ 0)^T \\ {}_V X_{oeil_3} = (-r_e \ 0 \ 0)^T \\ {}_V X_{oeil_4} = (0 \ r_e \ 0)^T \end{cases} \quad (\text{IV.31})$$

On obtient finalement 4 nouveaux repères (\mathcal{O}_1 , \mathcal{O}_2 , \mathcal{O}_3 et \mathcal{O}_4) dont le centre représente le centre de l'optique de chaque œil. On cherche donc à exprimer la position $X_{I_{c_i}}$ d'un contraste i initialement exprimé dans \mathcal{I} dans ces repères \mathcal{O}_k . Pour cela, on applique la transformation suivante :

$${}_{\mathcal{O}_k} X_{I_{c_i}} = T_{\mathcal{V} \rightarrow \mathcal{O}_k} \cdot T_{\mathcal{O}_R \rightarrow \mathcal{V}} \cdot T_{\mathcal{B} \rightarrow \mathcal{O}_R} \cdot T_{\mathcal{I} \rightarrow \mathcal{B}} \cdot \mathcal{I} X_{I_{c_i}} \quad (\text{IV.32})$$

Où $\mathcal{I} X_{I_{c_i}} = (x_{I_{c_i}} \ y_{I_{c_i}} \ z_{I_{c_i}})^T$ représente la position du contraste i exprimée dans le repère inertiel

\mathcal{I} . Les matrices de transformation en coordonnées homogènes permettant de passer d'un repère \mathcal{A} à un repère \mathcal{B} sont notées $T_{\mathcal{A} \rightarrow \mathcal{B}}$ et sont données par :

$$T_{\mathcal{I} \rightarrow \mathcal{B}} = \begin{pmatrix} R^T & -R^T \cdot \mathcal{I} X_{Robot} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (\text{IV.33})$$

$$T_{\mathcal{B} \rightarrow \mathcal{O}_{\mathcal{R}}} = \begin{pmatrix} R_{\vec{x}_{180^\circ}} & R_{\vec{x}_{180^\circ}} \cdot \begin{pmatrix} 0 \\ 0 \\ h_e \end{pmatrix} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (\text{IV.34})$$

$$T_{\mathcal{O}_{\mathcal{R}} \rightarrow \nu} = \begin{pmatrix} R_e^T & - \begin{pmatrix} 0 \\ 0 \\ l_e \end{pmatrix} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (\text{IV.35})$$

$$T_{\mathcal{O}_k \rightarrow \nu} = \begin{pmatrix} I_{3 \times 3} & -\nu X_{oeil_k} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (\text{IV.36})$$

Où R est la matrice de rotation décrivant l'attitude du robot comme décrit dans le chapitre III, $R_{\vec{x}_{180^\circ}}$ représente un rotation de 180° autour de l'axe \vec{x} du repère $\mathcal{O}_{\mathcal{R}}$, et R_e représente la matrice de rotation décrivant l'orientation de l'œil.

Une fois que l'on dispose de la position ${}_{\mathcal{O}_k} X_{I_{c_i}} = \left({}_{\mathcal{O}_k} x_{I_{c_i}} \quad {}_{\mathcal{O}_k} y_{I_{c_i}} \quad {}_{\mathcal{O}_k} z_{I_{c_i}} \right)^T$ du contraste i dans le référentiel d'un l'œil k on peut aisément remonter à sa direction angulaire θ_{c_i} nécessaire au calcul de des signaux photodiodes :

$${}_{\mathcal{O}_k} \theta_{c_i} = \tan^{-1} \left(\frac{{}_{\mathcal{O}_k} x_{I_{c_i}}}{{}_{\mathcal{O}_k} z_{I_{c_i}}} \right) \quad \text{si } k \in [1, 3] \quad (\text{IV.37})$$

$${}_{\mathcal{O}_k} \theta_{c_i} = \tan^{-1} \left(\frac{{}_{\mathcal{O}_k} y_{I_{c_i}}}{{}_{\mathcal{O}_k} z_{I_{c_i}}} \right) \quad \text{si } k \in [2, 4] \quad (\text{IV.38})$$

4.3 Asservissement visuel

L'asservissement visuel consiste à maintenir la cible au centre du champ visuel de l'œil. Le système de vision est actionné par des servomoteurs agissant sur des axes orthogonaux. On peut considérer que les systèmes de positionnement des axes \vec{x} et \vec{y} sont découplés. On aborde alors le problème d'asservissement visuel sur chacun des axes de façon indépendante, chaque axe a donc son propre contrôleur. Le système d'asservissement visuel est alors composé d'un actionneur de type servomoteur, et d'un système de traitement que l'on peut assimiler à un retard pur de 65 ms. L'identification du servomoteur a été faite dans la partie 2.4.3 et nous avons vu que ce dernier pouvait être assimilé à un système du second ordre. L'entrée de commande du servomoteur est la même que la plupart des matériels de modélisme. Elle

correspond à la durée du signal PPM (exprimée ici en dixième de microsecondes) codant la consigne de position et est comprise entre 10000 et 20000 ; sachant que 10000 correspond à la position -50° , 15000 correspond à la position milieu 0° et 20000 correspond à la position $+50^\circ$. La figure IV.36 décrit donc le système complet en boucle fermée avec la structure de correcteur qui a été choisie.

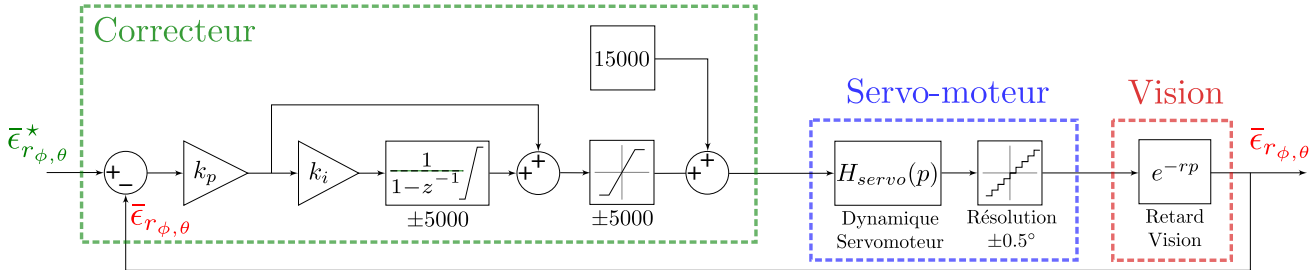


FIGURE IV.36 – La boucle d’asservissement visuel est constituée d’un simple correcteur PI saturé avec un terme de pré-positionnement d’actionneur en sortie. Le système de vision à asservir peut être assimilé à un second ordre traduisant la dynamique du servomoteur et un retard pur modélisant le système de vision (capteur et traitement associé).

Dans les faits, pour déterminer les coefficients du correcteur PI, le retard pur est approché par une approximation de Padé du 1^{er} ordre $\left(e^{-rp} \simeq \frac{1 - \frac{r}{2}p}{1 + \frac{r}{2}p} \right)$. Le correcteur est ensuite conçu sur le système discrétisé et les coefficients sont choisis de façon à obtenir des marges suffisantes ; c’est à dire une marge de module supérieure à 0.5, une marge de phase supérieure à 60° et une marge de gain supérieure à 5 dB. La figure IV.37 montre le digramme de Bode du transfert de boucle ainsi que des fonctions de sensibilité. On cherche également à limiter l’influence du bruit de mesure sur la commande et donc à avoir un transfert entre le bruit de mesure et la commande ne dépassant pas un gain de 2. En effet, on souhaite que le bruit de commande reste en dessous de la précision du servomoteur qui est de l’ordre de 0.5° . Or, le bruit de mesure du capteur est strictement inférieur à 0.2° (voir section 5). En garantissant un transfert inférieur à 13 en gain ($20 \cdot \log(2) \simeq 13$ dB), on garantit donc de ne pas générer de bruit parasite sur le servomoteur (souvent appelé "grognement") et donc d’améliorer sa durée de vie et optimiser son fonctionnement.

Ici nous avons donc choisi $k_i = 0.15$ et $k_p = 0.25$, qui permettent de respecter le cahier des charges, tout en garantissant un temps de réponse en boucle fermée de 0.1 seconde et un rejet de perturbation de commande en moins de 0.2 seconde.

4.4 Résultats de simulations du système occulo-moteur

On présente ici quelques résultats de simulation des signaux photodiodes, afin de discuter de l’influence de la vitesse de la cible et de la luminosité ambiante sur la qualité de la mesure.

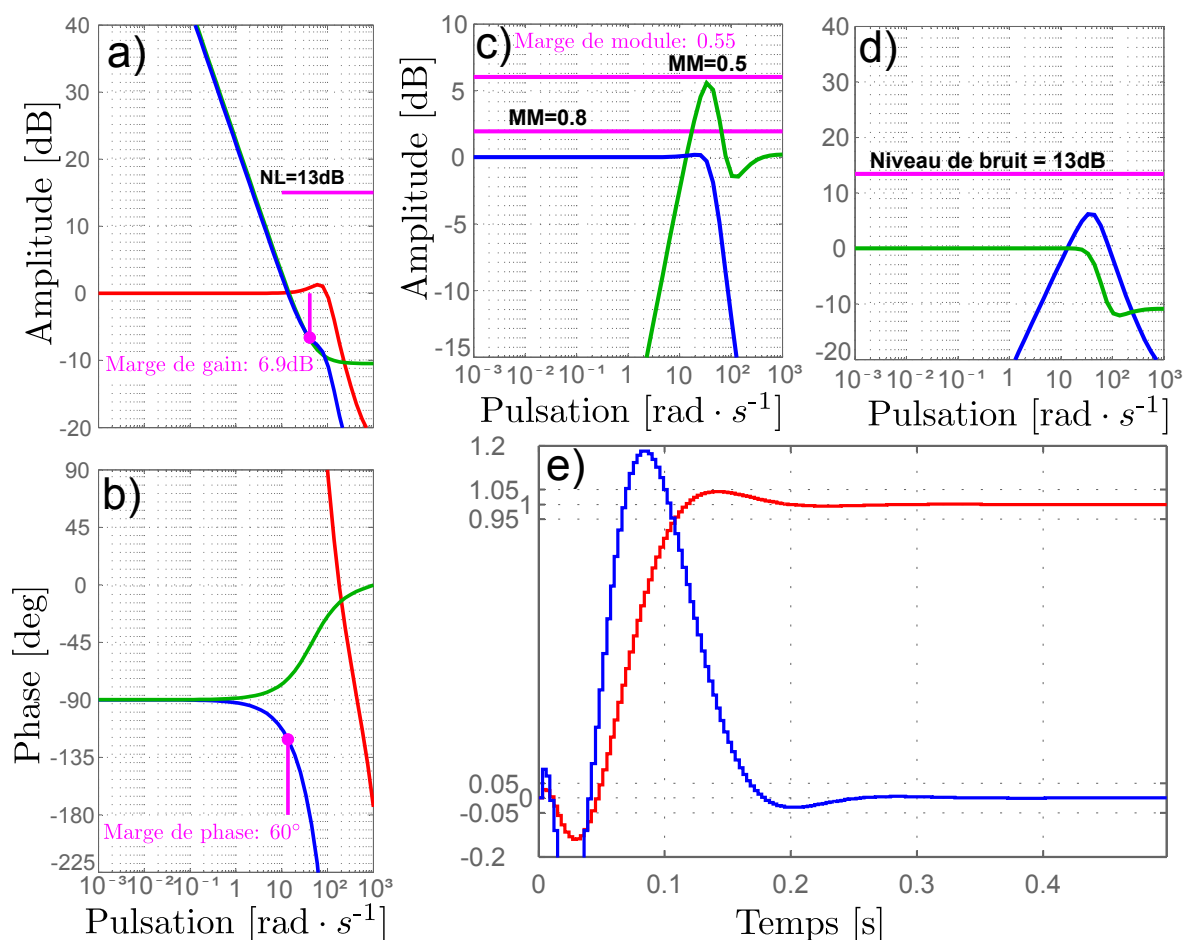


FIGURE IV.37 – a) Amplitude du diagramme de Bode du procédé en rouge, du contrôleur en vert et du transfert de boucle ouverte en bleu. b) Phase du diagramme de Bode avec également en rouge le procédé, en vert le contrôleur et en bleu le transfert de boucle ouverte final. c) Amplitude des diagrammes de Bode du transfert du bruit de la perturbation de sortie vers la sortie en vert et de la boucle fermée en bleu. d) Amplitude des diagrammes de Bode du transfert de la perturbation de commande sur la sortie en bleu, et du bruit de mesure vers la sortie en vert. On voit bien que ni la courbe verte ni la bleue ne dépassent le niveau de bruit de 13 dB du cahier des charges. e) Réponse en boucle fermée à un échelon de consigne en rouge et réponse à une perturbation de commande en bleu. Le temps de réponse final est de 0.1 seconde et le temps de rejet de 1.6 seconde.

4.4.1 Influence de la vitesse de la cible

Le protocole de calibration étant réalisé pour des vitesses de balayage du capteur très faible (on cherche à identifier la caractéristique statique), il est important de tester à partir de quelle vitesse de cible (vitesse angulaire) la sortie du capteur n'est plus fiable. En effet, nous avons présenté précédemment dans la partie 3.1 et plus précisément dans la figure IV.24, que la caractéristique du capteur se voit atténuée et déphasée plus la vitesse de défilement de la cible augmente. On réalise donc une simulation durant laquelle une cible effectue des aller-retours de plus en plus rapides face à l'œil. La figure IV.38a

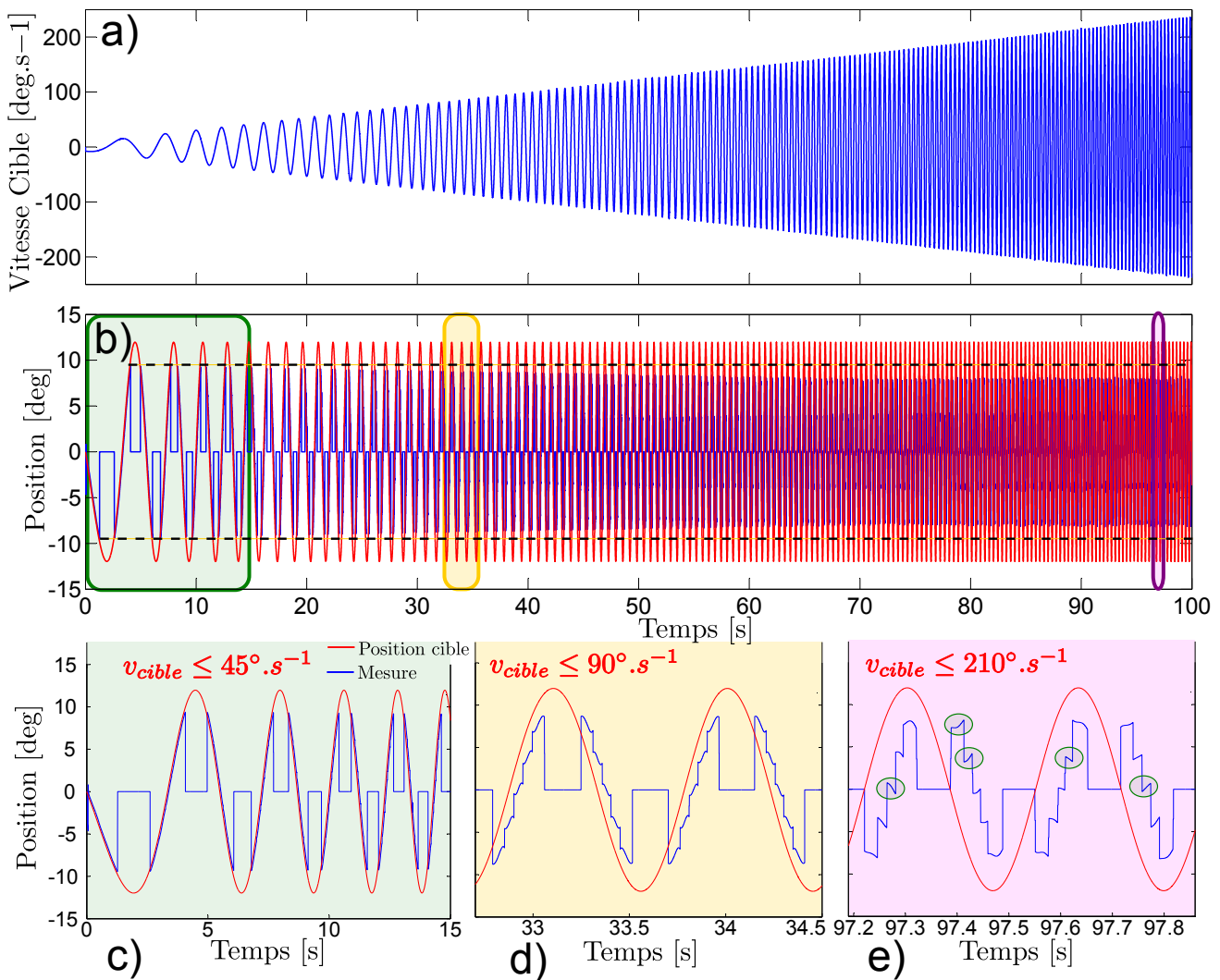


FIGURE IV.38 – a) Évolution de la vitesse de la cible au cours de la simulation. La cible effectue des allers-retours face à l’œil avec une vitesse qui augmente progressivement de $10^\circ \cdot s^{-1}$ à plus de $200^\circ \cdot s^{-1}$ (valeur crête). b) En rouge, la position réelle de la cible, et en bleu, la position mesurée par le capteur (simulée avec des conditions de bruits réalistes). En noir, on représente la plage de mesure déterminée par la ZSL pour des vitesses faibles. c) Zoom sur la mesure de position dans le cas de vitesse faible (encadré vert de la figure b), d) Zoom sur la mesure de position pour des vitesses devenant trop grandes et faisant apparaître des premières distorsions. e) Zoom sur la mesure pour des vitesses trop grandes entraînant même des inversions de la caractéristique du capteur (voir ellipses vertes).

décrit l’évolution de la vitesse angulaire de la cible dans le référentiel de l’œil. La vitesse de maximum lors d’un aller-retour commence seulement quelques degrés par seconde pour finir à plus de 220 degrés par seconde. La figure IV.38b présente la mesure de position délivrée par le capteur pour cette vitesse de défilement croissante. On observe notamment que pour des vitesses faibles, le ZSL permet une plage de mesure complète, qui se réduit d’autant plus que la vitesse augmente (voir l’évolution de l’enveloppe de la courbe bleue par rapport aux lignes noires discontinues). Cela s’explique par la bande passante

limitée du capteur qui entraîne une atténuation du signal de mesure lorsque la cible se déplace trop rapidement. Les figures IV.38c, d et e, présentent une vue plus précise de la mesure pour différentes vitesses. On observe que pour des vitesses faibles ($\leq 45^\circ \cdot \text{s}^{-1}$), le positionnement est de très bonne qualité, présentant seulement un retard de 65 ms comme le montre la figure IV.38c. Par contre, sur la figure IV.38d, on voit apparaître des sauts lors de la transition d'une paire à l'autre, ainsi que des déformations du signal, lorsque l'on atteint des vitesses proches de $90^\circ \cdot \text{s}^{-1}$. Ce phénomène est dû au fait que la caractéristique du capteur est atténuée et que la table permettant de linéariser sa sortie n'est plus parfaitement adaptée. Les valeurs extrêmes des tables des différentes paires ne sont alors plus atteintes, ce qui génère un saut lors de la transition d'une paire à une autre (voir également les explications liées à la figure IV.24 de la section 3.1). Pour des vitesses supérieures à $200^\circ \cdot \text{s}^{-1}$, on observe même une inversion de la pente à certains endroits de la caractéristique du capteur (voir figure IV.38e). Cela s'explique par le fait que le retard introduit par les différents filtrages (environ 65 ms) n'est alors plus négligeable. La sélection de la paire réalisée par le ZSL a alors lieu de façon déphasée, ce qui entraîne l'utilisation de la mauvaise partie de la caractéristique (celle ayant une pente inversée). En effet, si on regarde la caractéristique du capteur sur la figure IV.24, on observe, que de part et d'autre de la zone centrale servant à la mesure de position, la caractéristique a une pente inversée. Dans le cas où le déphasage est trop important (pour des vitesses de cibles élevées), la paire est sélectionnée en retard, ce qui entraîne l'utilisation de la mauvaise partie de la caractéristique de sortie du capteur VODKA.

On retiendra donc que la sortie du capteur reste correcte et suffisamment fiable pour des vitesses allant jusqu'à environ $100^\circ \cdot \text{s}^{-1}$. Ceci sera validé expérimentalement dans la partie suivante, et notamment dans la partie 5.2, où l'on montrera que l'œil arrive à poursuivre du regard des cibles allant à des vitesses supérieures à $100^\circ \cdot \text{s}^{-1}$, malgré la détérioration de la mesure.

4.4.2 Influence de la luminosité ambiante

Pour tester l'influence de la luminosité ambiante, on se propose de faire varier la luminosité ambiante de 1000 lx à 100 lx et de voir l'influence que cela a sur la qualité de la mesure. Pour cela, on simule la réponse fusionnée de l'œil pour une cible (croix blanche sur fond noir de 2 cm de large) effectuant des allers-retours de 60 cm d'amplitude (sur l'axe \vec{x} seulement) à une distance de 1.5 m de l'œil. On obtient alors un déplacement angulaire sinusoïdal de la cible à une fréquence de 0.1 Hz et d'amplitude légèrement supérieure au champ visuel de l'œil. Dans cette configuration, les yeux 1 et 3 voient donc la cible en mouvement alors que les yeux 2 et 4 voient une cible immobile.

La figure IV.39a présente la réponse fusionnée des yeux 1 et 3. On observe que la mesure de position fonctionne bien et est indépendante de la luminosité ambiante. On observe également que le ZSL fonctionne bien pour une luminosité suffisante, étant donnée que lorsque la cible sort du champ visuel de l'œil, le signal mesuré retombe à zéro (comme par exemple entre $t = 21$ s et $t = 24$ s, sur les

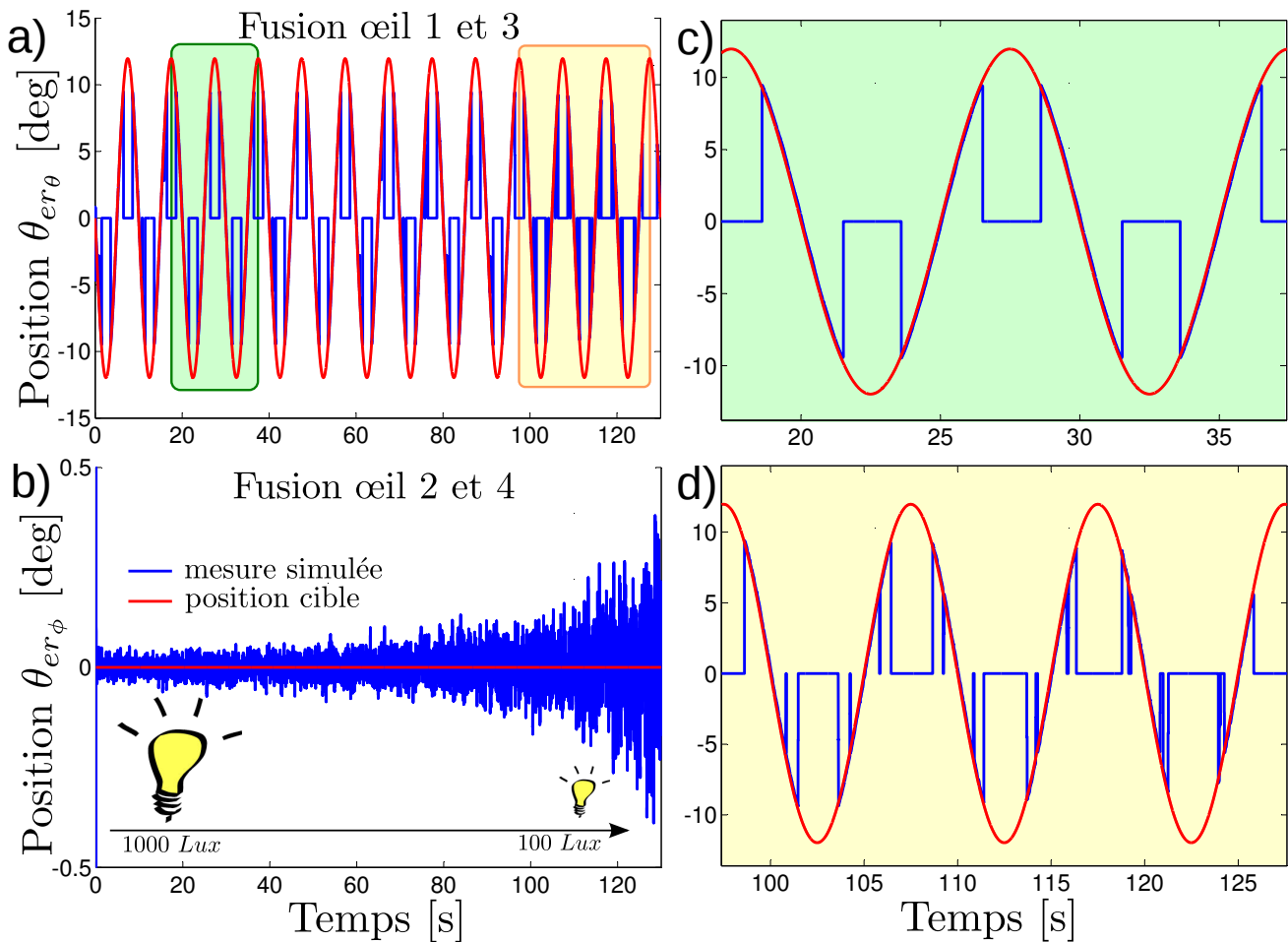


FIGURE IV.39 – Étude de l'influence de la luminosité sur la mesure de position, lors d'une diminution linéaire de la luminosité ambiante de 1000 lx à 100 lx. Les mesures simulées du capteur sont représentées par une courbe bleue alors que la position réelle de la cible correspond à la courbe rouge. a) Mesure en position pour la même diminution de luminosité, mais avec une cible effectuant des aller-retours devant le capteur. On observe, qu'avec la diminution de lumière, le fonctionnement du ZSL se détériore. b) Mesure de position du capteur pour une cible immobile. On observe que le bruit de mesure augmente de plus en plus au fur et à mesure que la lumière diminue. c) Zoom sur la mesure de position du capteur pour une luminance élevée. On observe que la fonction ZSL marche de façon optimale, étant donné que le signal retombe à zéro lorsque la cible sort du champ de vision de l'œil et que la mesure est continue lorsque la cible est dans le champ de vision. d) Zoom sur la mesure de position lors d'une luminosité faible. On observe que la fonction ZSL effectue de "faux" retour à zéro, et que la plage de mesure se voit réduite. Cependant la mesure en elle-même reste bonne et précise.

figures IV.39b-c). Cependant, en observant la réponse simulée de la sortie fusionnée des yeux 2 et 4 on voit que plus la luminosité diminue, plus le rapport signal sur bruit se détériore, comme le montre la figure IV.39b. Cela s'explique par le fait que le bruit additif est en partie dû à l'électronique, et que son amplitude est à peu près constante ; lorsque la luminosité diminue, l'amplitude des signaux photodiodes diminue également, ce qui détériore le rapport signal sur bruit (RSB) et entraîne une augmentation

du niveau de bruit sur la sortie finale. On note malgré tout que le filtrage passe-haut analogique permet de rendre le traitement quasiment insensible au changement d'ambiance lumineuse, au sens où aucun biais ni aucune distorsion n'apparaissent sur la mesure. Cette propriété est intéressante car elle permet d'utiliser le capteur dans une grande gamme de lumière, sans avoir à effectuer de réglages particuliers (seul le RSB est influencé par l'ambiance lumineuse). La figure IV.39c montre un zoom sur le fonctionnement de la mesure et du ZSL lors de conditions lumineuses suffisantes. La mesure reste très fidèle à la position réelle de la cible avec un retard inférieur à 65 ms. De même, dès lors que la cible sort du champ visuel de l'œil, la fonction ZSL le détecte et force la sortie à 0. En revanche, on voit que lorsque la luminosité commence à devenir faible, les amplitudes des signaux photodiodes ne sont plus suffisantes et la fonction ZSL est fortement perturbée (faux retours à zéro et diminution de la plage de mesure.) En effet, les seuils utilisés dans le ZSL sont fixes et une baisse de luminosité entraîne une détection plus tardive et par conséquent une diminution de la plage de mesure. De même, cela entraîne un retour à zéro lors du passage d'une paire à l'autre car les amplitudes des signaux photodiodes passent en dessous du seuil lors des transitions.

Remarque IV.4: Il est important de ne pas exposer le capteur à de trop forte luminosité afin de ne pas atteindre les saturations de ce dernier (saturation des CAN), ce qui aurait pour effet de rendre le signal de sortie inutilisable.

Pour conclure sur cette simulation, on observe que le capteur peut être utilisé sur plus d'une décade de luminance, grâce à son filtrage analogique passe-haut, qui rend la mesure quasiment insensible aux variations d'ambiance lumineuse (hormis une diminution du RSB). Par contre, la fonction ZSL est plus sensible à l'ambiance lumineuse et entraîne de "faux" retour à 0 lorsque la lumière devient faible. Cela pourrait être amélioré grâce à des seuils adaptatifs.

5 Résultats expérimentaux

On présente maintenant les résultats obtenus avec l'œil réel. L'ensemble des résultats présentés ici, ont été obtenus alors que le système Vicon⁷ ainsi que les Néons de la salle d'expérimentation étaient actifs. Ces deux systèmes étant des sources de lumières pulsées très énergétiques, il est nécessaire d'étudier leurs influences sur les mesures de l'œil. Une étude détaillée est donc proposée en annexe B. Cette annexe permet notamment de démontrer l'efficacité et la robustesse des traitements réalisés par l'œil. Je montre dans cette annexe, que ces traitements permettent d'être très robustes, aussi bien à la luminance ambiante qu'aux bruits parasites pouvant découler d'un système de lumière pulsée.

Les sections suivantes présentent les précisions obtenues pour les mesures de position en présence

7. Le système Vicon est le système de capture du mouvement qui permet de localiser avec une précision submillimétrique des marqueurs dans l'arène de vol ayant servie aux expérimentations. Une présentation plus détaillée de ce système est disponible dans la section 2.4 du chapitre III ainsi que dans l'annexe A.

du système Vicon et des néons.

5.1 Mesure de position angulaire

5.1.1 Précision statique

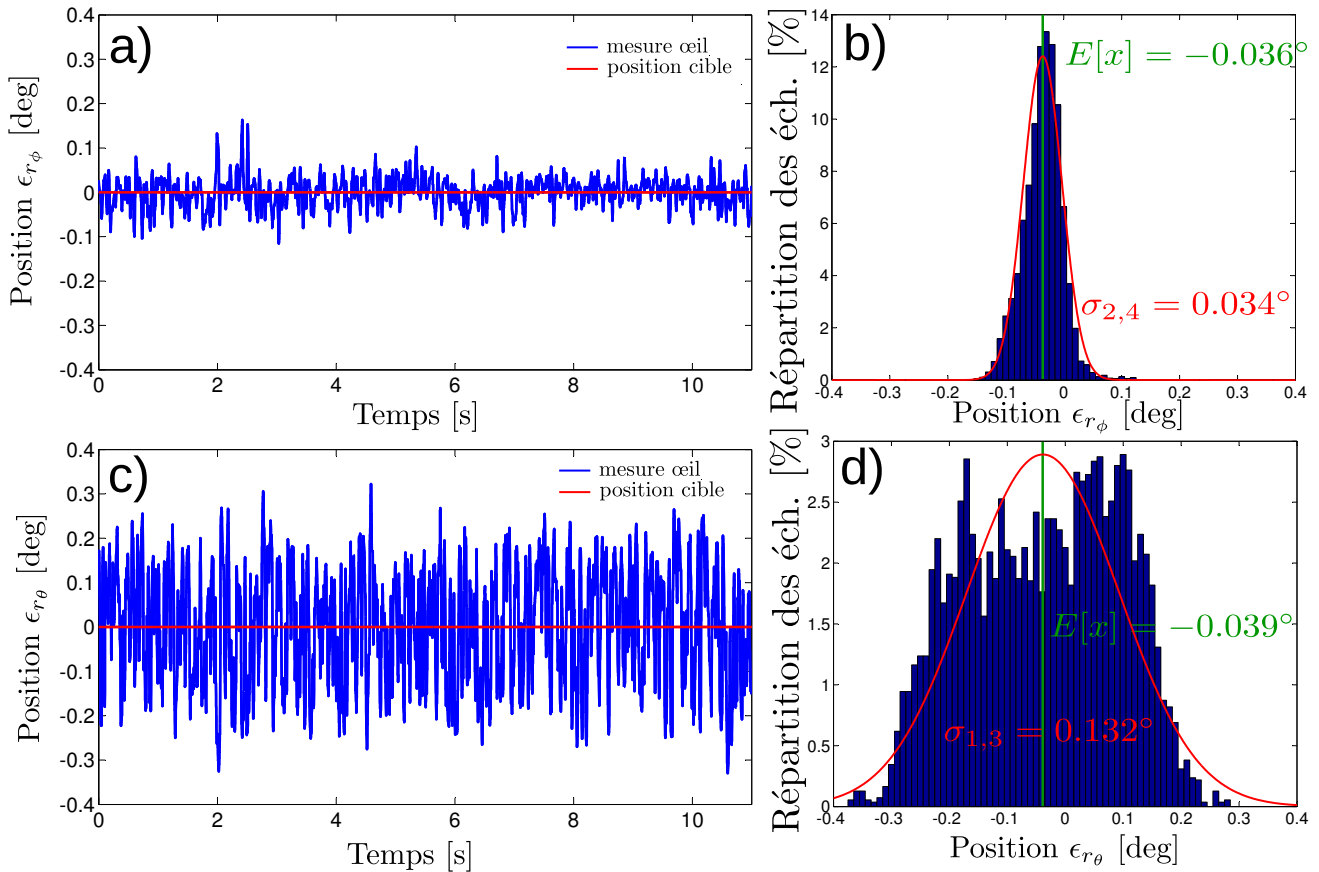


FIGURE IV.40 – a et b) Mesure fusionnée des yeux 2 et 4 et de la mesure fusionnée des yeux 1 et 3 respectivement, pour une croix positionnée à 1.32 m face à l’œil. La courbe bleue représente la position mesurée et la courbe rouge la position réelle de la cible. c et d) Répartition en pourcentage des échantillons de mesure pour les mesures de position $\bar{\epsilon}_{r_\phi}$ et $\bar{\epsilon}_{r_\theta}$. La courbe rouge représente la gaussienne présentant le même écart type que l’enregistrement de mesure. La droite verte représente la valeur moyenne de la série.

Les figures IV.40a et IV.40b représentent la précision de la mesure $\bar{\epsilon}_{r_\phi}$ autour de l’axe \vec{x} (mesure fusionnée des yeux 2 et 4). Les résultats pour la mesure $\bar{\epsilon}_{r_\theta}$ (autour de l’axe \vec{y}) découlant de la fusion des yeux 1 et 3 sont présentés dans les figures IV.40c et IV.40d. Ces mesures ont été obtenues en plaçant l’œil face à une croix blanche sur fond noir de 2 cm de largeur positionnée à 1.32 m à l’aplomb (comme sur la figure IV.25). On détermine donc la précision statique du capteur comme étant l’écart type des mesures obtenues. On obtient alors une distribution gaussienne pour le premier axe avec une précision

de 0.034° et une erreur maximale de 0.164° . Pour le second axe, la distribution n'est pas vraiment gaussienne et la précision est seulement de 0.132° avec une erreur maximale de 0.330° .

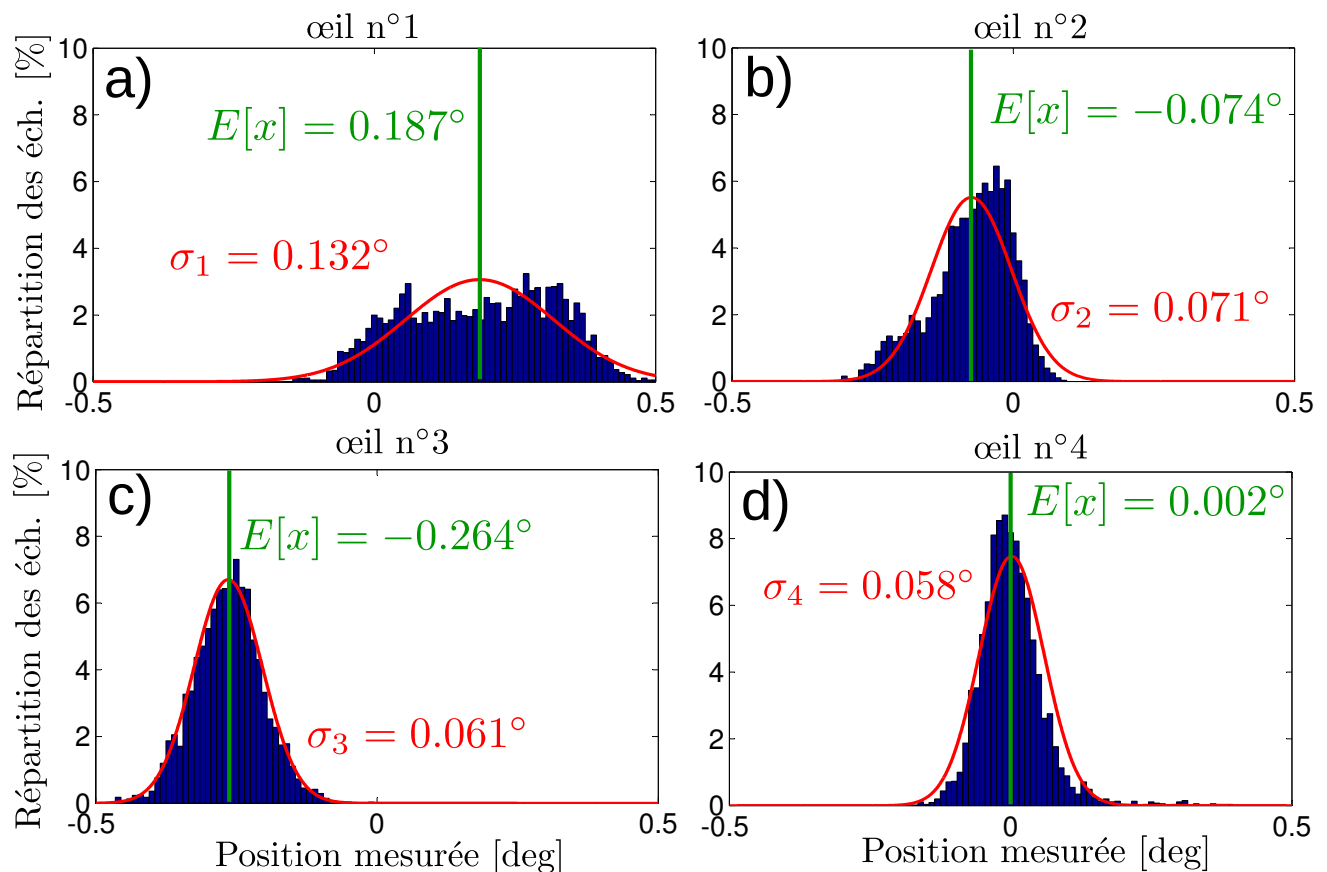


FIGURE IV.41 – Pour a), b), c) et d) en bleu, le pourcentage d'échantillon correspondant à une mesure de position. En rouge, la gaussioïde ayant les mêmes caractéristiques statistiques (écart type et valeur moyenne) et en vert la valeur moyenne de la série de mesure. Ces mesures ont été obtenues après de nombreux vols du quadrirotor et de son système de vision, en se replaçant dans les mêmes conditions que lors de la phase de calibration (effectuée plusieurs semaines auparavant). a) Répartition des échantillons mesurés par l'œil n°1. Cette répartition laisse supposer que l'optique de l'œil a légèrement bougé lors des nombreux essais en vol, ce qui est confirmé par l'écart type des mesures qui est 2 fois supérieur à celui des autres yeux. b) Répartition des échantillons mesurés par l'œil n°2. c) Répartition des échantillons mesurés par l'œil n°3. d) Répartition des échantillons mesurés par l'œil n°4.

La plus mauvaise qualité de la mesure sur le deuxième axe s'explique en observant plus en détail la précision de chacun des yeux. En effet, on présente sur les figures IV.41a-d la répartition des mesures pour chacun des yeux. Ces mesures ont été obtenues après de nombreux vols du quadrirotor et de son système de vision, en se replaçant dans les mêmes conditions que lors de la phase de calibration (effectuée plusieurs semaines auparavant). On observe que les précisions sont légèrement meilleures que celles annoncées dans la table IV.5 (sauf pour l'œil 1) car il s'agit ici de mesures prises pour une position fixe de l'œil alors que la table IV.5 avait été obtenue lors d'un mouvement de balayage. Ici on étudie

donc la répartition des échantillons de mesures pour une cible (croix blanche de 2 cm d'épaisseur sur un fond noir) placée à l'aplomb de l'œil. Les moins bons résultats obtenus sur l'œil 1 s'expliquent par le fait que ces mesures ont été réalisées "longtemps" après la calibration présentée dans la table IV.5, c'est à dire après plusieurs vols du quadrirotor complet. On peut donc supposer que les réglages de l'œil 1 ont été quelque peu altérés durant ces vols (vibrations, etc.) En effet, la précision réduite par rapport aux réglages annoncés initialement, laisse penser que la lentille a quelque peu bougée. Cependant, on remarque que les autres yeux conservent de bonnes performances de localisation étant donné que l'écart type des mesures reste très faible. Les valeurs moyennes non centrées, s'expliquent par le fait que la cible n'est pas placée exactement à la même position que lors de la calibration initiale.

5.1.2 Précision en mouvement simple

Ici on réalise un balayage sinusoïdal sur un axe puis l'autre. On observe sur les figures IV.42a et IV.42c que la mesure de position fonctionne de façon satisfaisante, ainsi que le ZSL qui force bien la sortie du capteur à 0 lorsque la cible sort du champ de vision. Les courbes fournies par les figures IV.42b et IV.42d représentent la position mesurée par le capteur en fonction de la position de la cible. On observe alors que la linéarité du capteur est tout à fait correcte pour l'axe \vec{x} et un peu moins bonne sur l'axe \vec{y} pour les mêmes raisons que celles énoncées précédemment à propos de la précision de la mesure $\bar{\epsilon}_{r_\theta}$. On observe également un défaut des servomoteurs qui entraîne un offset de quelques degrés sur la mesure selon le sens de déplacement du servomoteur. Cela est dû à un jeu mécanique (zone morte) qui apparaît lorsque le servomoteur change de sens de rotation. On mesure donc un jeu moyen d'environ 1.9° sur la mesure de $\bar{\epsilon}_{r_\phi}$ et $\bar{\epsilon}_{r_\theta}$.

5.2 Poursuite de cible

Afin de valider expérimentalement la poursuite de cible, on réalise deux types de tests :

- On déplace manuellement la cible selon différents types de mouvements échelons et déplacements pseudo-sinusoïdales. Ce test permet de vérifier le bon fonctionnement de l'asservissement sur une cible dont les mouvements sont aléatoires. Cela permet également de vérifier ce fonctionnement pour des vitesses angulaires de cible élevées.
- On génère des échelons de consigne sur l'erreur rétinienne désirée, ce qui est équivalent à réaliser un échelon sur la position de la cible. Ce test permet de quantifier la précision de l'asservissement et d'estimer les temps de réponses.

La figure IV.43 présente l'évolution de l'erreur rétinienne lorsque l'œil traque une cible en mouvement. Ces enregistrements correspondent à la vidéo `Poursuite_Cible_Seq1.avi` sur laquelle on déplace une croix blanche de 2cm d'épaisseur de façon aléatoire et selon les deux axes de l'œil. On observe que malgré des mouvements rapides atteignant plus de $100^\circ.s^{-1}$, l'œil parvient à maintenir la cible dans

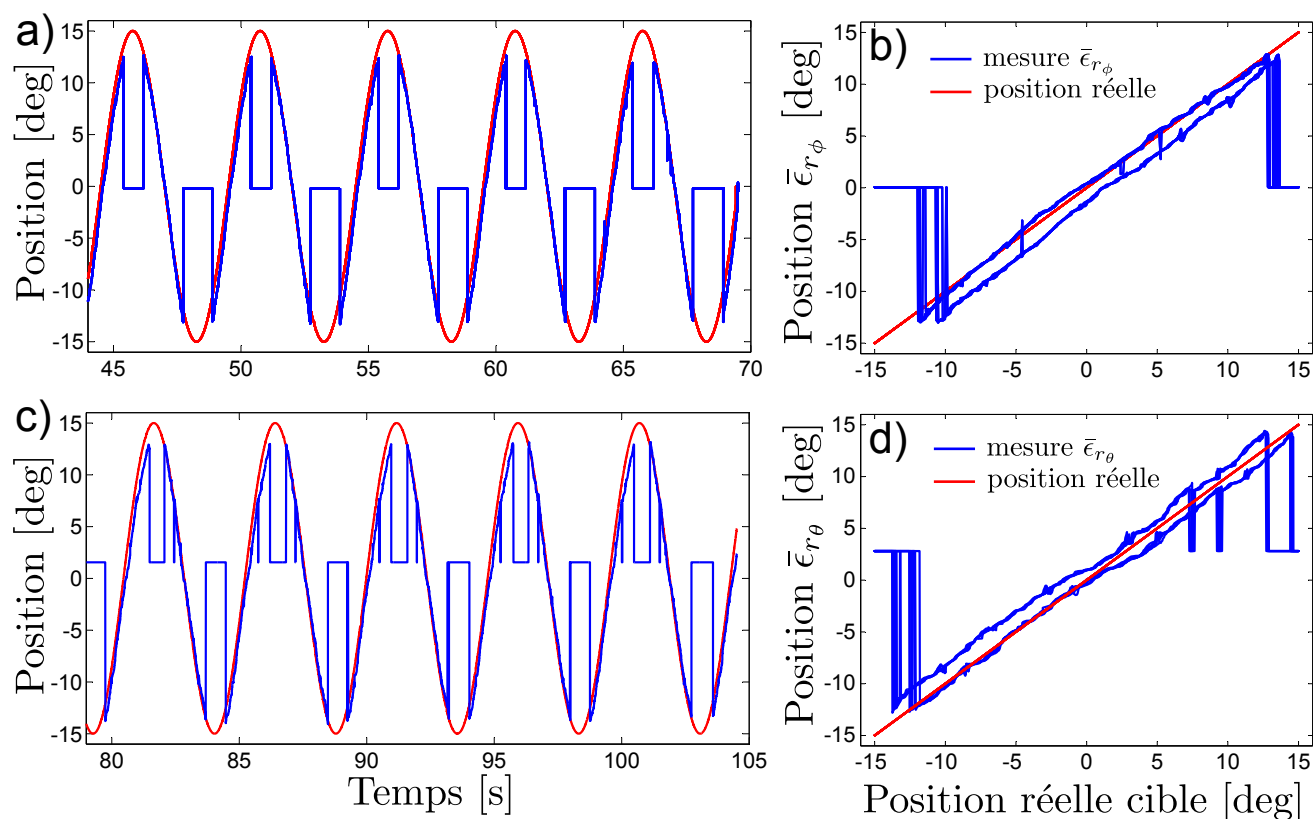


FIGURE IV.42 – a) En rouge la position réelle de la cible, et en bleu, la position mesurée $\bar{\epsilon}_{r_\phi}$ par le capteur sur l'axe \vec{x} . b) En bleu la position mesurée en fonction de la position de la cible, et rouge la droite "identité" représentant la position de la cible en fonction d'elle-même. On observe, pour plusieurs allers-retours de l'œil face à la cible un hystérésis dû à un jeu mécanique. c) En rouge la position réelle de la cible, et en bleu, la position mesurée $\bar{\epsilon}_{r_\theta}$ par le capteur sur l'axe \vec{y} . d) En bleu, la position mesurée en fonction de la position de la cible, et en rouge la droite "identité". Le même phénomène d'hystérésis dû au jeu du servomoteur est visible autour de l'axe \vec{y} dans le cas où l'œil effectue un balayage sinusoidal de la cible.

son champ de vision. En effet, on observe que l'erreur rétinienne est maintenue proche de zéro tout au long de l'expérience, même si elle peut atteindre près de 10° lorsque la vitesse de la cible atteint des pointes de $200^\circ \cdot s^{-1}$ (figure IV.43a et b à $t = 60$ s). Lors d'échelon de la cible de près de 20° , l'erreur maximale atteinte est de 4° et est annulée au bout de seulement 412 ms (figure IV.43a et b à $t = 40$ s).

La figure IV.44 présente l'évolution de l'erreur rétinienne pour une séquence similaire correspondant à la vidéo *Poursuite_Cible_Seq2.avi*. Lors de cette séquence, des aller-retours de la cible pour des vitesses comprises entre $-150^\circ \cdot s^{-1}$ et $+150^\circ \cdot s^{-1}$ sont effectués pendant 5 secondes durant lesquelles l'erreur rétinienne reste contenue à moins de $\pm 10^\circ$. On observe donc que les capacités de poursuite de l'œil sont très bonnes et que les phénomènes de distorsions des mesures décrits précédemment dans la partie 4.4.1 n'ont que peu de répercussions sur l'asservissement visuel.

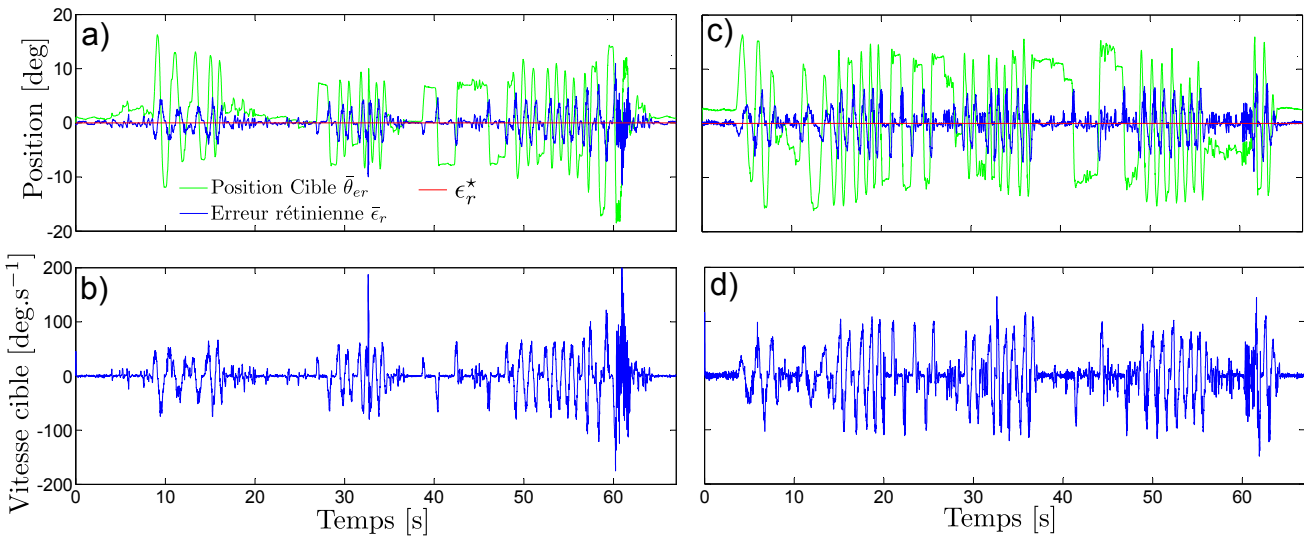


FIGURE IV.43 – Enregistrement correspondant à la vidéo `Poursuite_Cible_Seq1.avi` a) Poursuite de la cible sur l’axe de roulis (fusion œil 2 et 4). En vert, la position de la cible dans le champ de vision du robot. En bleu, l’erreur rétinienne correspondant à l’erreur de poursuite. En rouge la consigne d’erreur rétinienne. b) La vitesse de déplacement de la cible sur l’axe de roulis. c) Poursuite de la cible sur l’axe de tangage (fusion œil 1 et 3). En vert, la position de la cible dans le champ de vision du robot. En bleu, l’erreur rétinienne correspondant à l’erreur de poursuite. En rouge la consigne d’erreur rétinienne. d) La vitesse de déplacement de la cible sur l’axe de tangage.

Bien que non mesuré ici, on observe sur les vidéos que la cible n’a pas un angle de lacet constant durant ces expériences. En effet, celui-ci évolue approximativement entre $[-10^\circ, 10^\circ]$ sans aucune influence sur la poursuite de la cible. Le fait de fusionner deux yeux opposés, permet de reconstruire une mesure indépendante de l’angle de lacet. Ainsi, on montre que même si la cible n’est pas parfaitement alignée selon son angle de lacet avec l’œil, cela ne l’empêche pas de la verrouiller dans son champ de vision.

La figure IV.45 quant à elle, présente la réponse à des échelons de consigne d’erreur rétinienne. La figure IV.45a montre la séquence dans son ensemble, alors que les figures IV.45b et c montrent précisément sur deux échelons différents. Cela permet de quantifier le temps de réponse en boucle fermée de l’asservissement visuel. On observe donc que la boucle visuelle répond à des échelons de consigne en moins de 350 ms. De plus, le rejet de perturbations de commandes s’effectue en moins de 150 ms. On remarque cependant que malgré une précision de mesure inférieure à 0.1° (amplitude du bruit de mesure de la courbe bleue), le positionnement en boucle fermée n’est précis qu’à $\pm 0.5^\circ$. Ceci s’explique par la présence de frottement sec sur le système de cardans, ne permettant pas un positionnement très fin des servomoteurs. Cela résulte en une faible oscillation autour de la position de consigne, ce phénomène est bien visible sur les figures IV.45b et IV.45c.

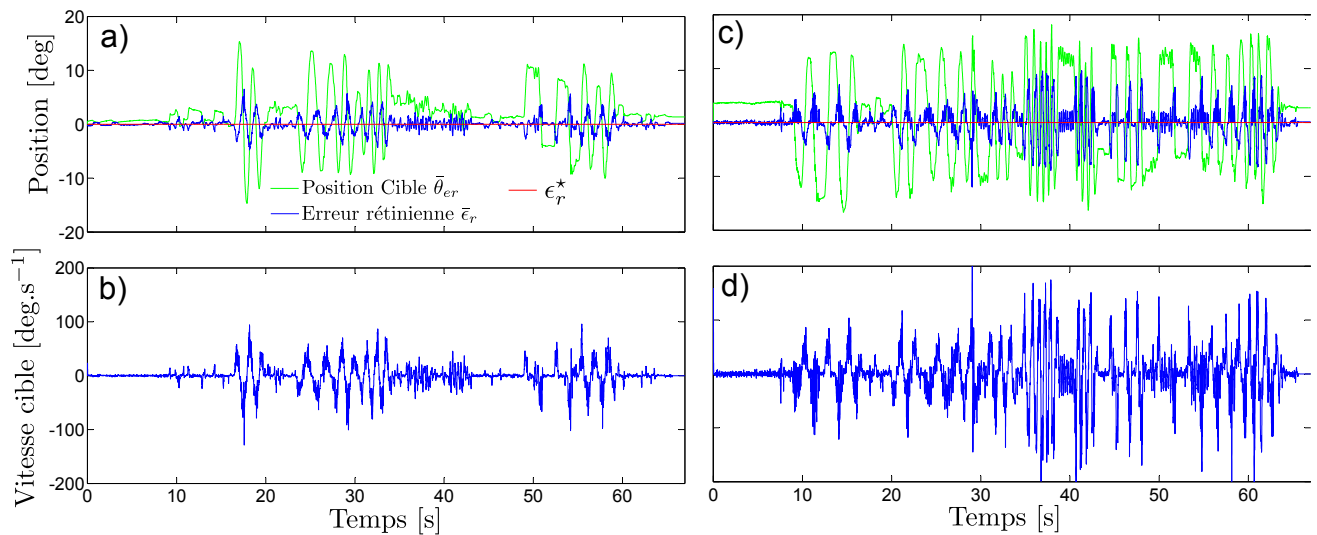


FIGURE IV.44 – Enregistrement correspondant à la vidéo `Poursuite_Cible_Seq2.avi` a) Poursuite de la cible sur l’axe de roulis. En vert, la position de la cible dans le champ de vision du robot. En bleu, l’erreur rétinienne correspondant à l’erreur de poursuite. En rouge la consigne d’erreur rétinienne. b) La vitesse de déplacement de la cible sur l’axe de roulis. c) Poursuite de la cible sur l’axe de tangage. En vert, la position de la cible dans le champ de vision du robot. En bleu, l’erreur rétinienne correspondant à l’erreur de poursuite. En rouge la consigne d’erreur rétinienne. d) La vitesse de déplacement de la cible sur l’axe de tangage.

6 Conclusion du chapitre

Nous avons, dans ce chapitre, décrit entièrement les différentes étapes de conception d’un œil bio-inspiré de la vision de la mouche. Cette œil se compose de 4 petits yeux élémentaires de 6 pixels chacun, dont l’optique est volontairement défocalisée. Nous avons montré qu’un traitement par paire de ces pixels combiné à une vibration active et périodique de l’optique, permettait à chacun des yeux de localiser finement une barre ou un front de contraste. L’utilisation d’une fusion appropriée des ces quatre yeux permet ensuite de reconstruire la position angulaire de la cible selon trois angles : roulis, tangage et lacet. Nous avons montré ainsi, qu’une croix pouvait être localisée avec une précision de 0.1° sur un champ de vision total de 20° selon les deux axes de roulis et tangage (les résultats concernant le lacet seront présentés dans le chapitre V). Enfin, nous avons montré qu’avec un mécanisme de découplage adapté, il était possible de poursuivre une cible du regard, même lorsque celle-ci se déplace à des vitesses proches de $200^\circ \cdot s^{-1}$. Finalement, nous disposons d’un système occulo-moteur complet, capable de traquer une cible sur 2 axes et plus de 100° de champ de vision ($\pm 50^\circ$ grâce au mécanisme de découplage et $\pm 10^\circ$ supplémentaire de champ de vision de l’œil en lui-même.)

On peut donc dire que l’œil est capable de traquer des cibles se déplaçant sur 2 dimensions et allant jusqu’à $200^\circ \cdot s^{-1}$ et ce, sur plus de 100° de champ de vision. De plus, la localisation de la cible est obtenue avec seulement 24 pixels avec une précision de 0.1° et un rafraîchissement pouvant aller jusqu’à

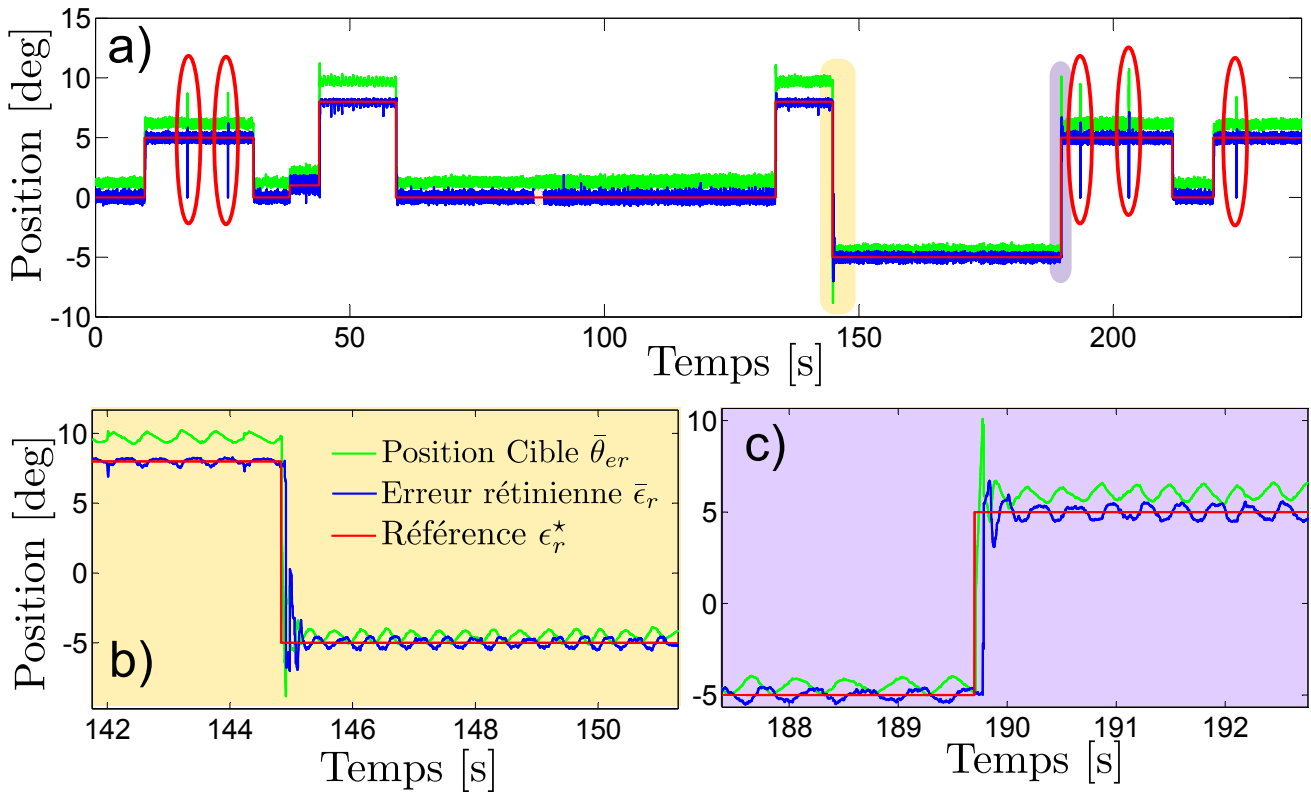


FIGURE IV.45 – a) En rouge la consigne d’erreur rétinienne, en bleu l’erreur rétinienne mesurée et en vert la position du servomoteur permettant d’obtenir cette erreur rétinienne. Les ellipses rouge pointent les instants où une perturbation de commande est appliquée. b) Zoom sur la réponse en descente de l’asservissement visuel. Le temps de réponse à 95% est de 338 ms pour un échelon de -13° . En rouge la consigne, en bleu l’erreur mesurée et en vert la position du servomoteur. Les oscillations autour de la position de consigne en "régime établi" sont dues aux frottements secs qui dégradent la précision du servomoteur (entre 0.5 et 1° de précision selon la position du servomoteur). c) Le temps de réponse à un échelon de 10° est également de 327 ms. Les temps de rejets de perturbations (entourés en rouge) sont quant à eux de 147 ms en moyenne.

1 kHz.

Chapitre V

Stabilisations bio-inspirées du quadrirotor X4-MaG

Sommaire

1	Introduction	172
1.1	État de l'art	172
1.2	Motivations	174
2	Description du robot complet	175
2.1	Assemblage complet	175
2.2	Modèle pour l'estimation	176
2.3	Contrôle de position et d'attitude	180
3	Stabilisation visuelle et réflexes	180
3.1	La stabilisation visuelle chez l'insecte	182
3.2	Stabilisation visuelle chez X4-MaG	183
3.3	Résultats et discussions	184
4	Stabilisation bio-inspirée	186
4.1	Asservissement en position basé vision	187
4.2	Stabilisation partielle d'attitude sans accéléromètre, décollage et atterrissage automatique	199
4.3	Stabilisation complète d'attitude bio-inspirée et asservissement en position	205
5	Application à la poursuite d'un Rover	217
5.1	Poursuite du Rover basée vision	217
5.2	Atterrissage sur cible mouvante	219
6	Discussions finales et perspectives de travail	221

1 Introduction

Dans ce dernier chapitre, nous allons décrire le quadrirotor entièrement assemblé avec son système de vision bio-inspirée. Nous montrerons qu'il est alors possible de concevoir de nouvelles stratégies d'estimation et de stabilisation d'attitude, qui ne reposent pas sur une centrale inertielle 9 axes, comme on le trouve généralement dans la littérature.

Après avoir passé en revue les techniques d'estimations d'attitude classiques, nous reprendrons le modèle de quadrirotor exposé dans le chapitre III et y apporterons quelques modifications afin d'y intégrer le système de vision. Nous discuterons alors des avantages structurels que présente ce système et montrerons expérimentalement son efficacité.

Puis, nous décrirons en détail les différentes versions d'estimateurs et de contrôleurs permettant de rendre le système complet autonome de façon progressive :

- Nous étudierons, dans un premier temps, un EKF permettant d'estimer la position et les vitesses de translation du robot sur la base de la vision. Nous évaluerons alors les performances de l'asservissement en position qui en résulte, en utilisant toutes les données inertielles disponibles.
- Nous montrerons ensuite, comme une extension du chapitre II, qu'il est possible de se passer des accéléromètres en utilisant seulement le gyromètre 3 axes et l'orientation du regard. On présentera un nouvel EKF capable de reconstruire l'attitude (hormis le lacet) ainsi que les positions et vitesses de translation. Nous montrerons alors que les performances sont suffisamment bonnes pour effectuer les tâches de décollage et atterrissage automatiques, ainsi que la poursuite de cible.
- Enfin, nous nous intéresserons au cas spécial de l'estimation du lacet, et présenterons les difficultés particulières liées à cet axe. Nous montrerons qu'il est possible sur la base de la vision, d'estimer cet angle et d'obtenir un système entièrement autonome, capable de poursuivre une cible mobile, placée sur un robot rover à l'aide uniquement d'un gyromètre 3 axes, d'un système de vision de 24 pixels doublé d'une mesure proprioceptive et d'un altimètre.

Enfin, nous concluons sur les développements envisageables dans le futur, liés à ces nouvelles stratégies de guidage par le regard.

1.1 État de l'art

Dans la plupart des applications aéronautiques, l'estimation d'attitude et de position est un point clé. Dans ce cadre, les centrales inertielles à base de micro-systèmes (MEMS) se sont, depuis de nombreuses années, imposées comme une solution efficace permettant d'estimer précisément l'attitude des engins volants. L'attitude complète, constituée des angles de roulis, tangage et lacet (ou cap), est alors le plus souvent reconstituée grâce à l'association de trois types de capteurs :

- Gyromètres : 3 axes permettant de mesurer la vitesse de rotation selon chacun des axes. On utilise, le plus souvent, les informations hautes fréquences de ces capteurs, qui sont fusionnées avec les

composantes basses fréquences d'accéléromètres et de capteurs magnétiques, afin de reconstituer l'attitude complète, ainsi que les vitesses de rotation.

- Accéléromètres : 3 axes, permettant de mesurer les accélérations du mobile selon chacun des axes. On utilise le plus souvent les informations basses fréquences de ces capteurs pour déterminer l'attitude du mobile (roulis et tangage).
- Magnétomètres ou compas : permettant de mesurer le champ magnétique terrestre selon 3 axes orthogonaux. On utilise le plus souvent les informations basses fréquences de ces capteurs, pour déterminer le cap du mobile (lacet).

Selon l'application, il existe différentes technologies pour ces capteurs, plus ou moins précises. Les centrales inertielles d'avions présentent, par exemple, très peu de dérives mais sont très coûteuses et volumineuses. Dans le domaine du drone ou des micro aéronefs, on préférera donc des technologies peu coûteuses, peu volumineuses et de faible consommation. Dans ce cadre, les centrales inertielles sont souvent réalisées à base de MEMS et présentent des précisions relativement médiocres. Ainsi, une centrale inertielle bas coût (quelques dizaines d'euros) présentera des dérives de l'ordre de quelques dixièmes de degrés par seconde pour les gyromètres (contre quelques dizaines de degrés par heure pour des centrales d'avions). Par exemple la centrale MPU6050 de chez *InvenSense* utilisé sur X4-MaG présente des biais de l'ordre de quelques degrés par seconde ($2\text{-}3^\circ \cdot \text{s}^{-1}$).

Face à ces dérives, il est nécessaire de mettre en place des filtres efficaces permettant, grâce à une fusion appropriée des différents capteurs, de délivrer une estimation non biaisée de l'attitude et des taux de rotations. La fusion de donnée des centrales inertielles a été intensément étudiée depuis de nombreuses années. Et il existe de nombreux filtres (linéaires ou non), permettant de reconstruire des estimations non biaisées. [Crassidis *et al.*, 2007] passent en revue les méthodes d'estimation d'attitude non linéaires qu'ils considèrent comme prometteuses, notamment les méthodes dérivées du filtre de Kalman étendu (EKF). Des exemples de filtres de Kalman peuvent être trouvés dans [Marins *et al.*, 2001] ou [Tayebi *et al.*, 2007]. [Hoffmann *et al.*, 2010] ont également appliqué le filtre à des capteurs MEMS bas-coût. [Castellanos *et al.*, 2005b] ont également utilisé un filtre de Kalman étendu, dans lequel ils ajoutent également un capteur de pression, permettant de contrer les effets de la force centrifuge, influençant les accéléromètres. D'autres filtres, comme le filtre complémentaire, présentent de très bonnes performances et ce, même pour des taux de rafraîchissements faibles. Différentes versions et mise en œuvre de ces filtres sont disponibles dans [Mahony *et al.*, 2005], [Hamel et Mahony, 2006] ou encore dans [Mahony *et al.*, 2008] et ont l'avantage de fournir également une estimation des biais des gyromètres, afin d'en déduire la vitesse de rotation réelle du système. Ces dernières années ont également vu apparaître de plus en plus de techniques d'estimation d'attitude reposant sur la vision. [Shabayek *et al.*, 2012] résument les principales techniques qui peuvent se baser sur des caméras catadioptriques et sur la poursuite du point de fuite [Bazin *et al.*, 2008]. D'autres détectent l'horizon par des techniques de segmentation (basées sur les couleurs [Thurrowgood *et al.*, 2009], [Mondragón *et al.*, 2010a], [Mondragón *et al.*, 2010b] ou la

polarisation [Shabayek *et al.*, 2012]) pour remonter à l'attitude. D'autres techniques consistent à utiliser trois antennes GPS, dont la disposition est connue et à déterminer l'attitude en utilisant la phase des signaux ([Peng *et al.*, 2000], [Chiang *et al.*, 2002]). Une autre approche originale consiste à utiliser le rapport signal sur bruit du GPS pour remonter à l'attitude ([Lightsey et Madsen, 2003]).

Les informations inertielles peuvent également être utilisées pour estimer la position et la vitesse de translation d'un mobile dans le référentiel inertiel. Cependant, étant donné les dérives inhérentes à l'intégration des accéléromètres, il est nécessaire de recalibrer régulièrement les estimations, à l'aide de capteurs supplémentaires ne présentant pas de dérives. [Barton, 2012] présente les capteurs les plus souvent utilisés dans le domaine des drones, pour arriver à une estimation de position non biaisée. Cela est généralement effectué à l'aide d'un capteur GPS qui offre un moyen efficace de corriger les dérives ([Kingston et Beard, 2004], [Wendel *et al.*, 2006], [Nemra et Aouf, 2010]). Dans le cas où les signaux GPS ne sont pas accessibles, la vision est souvent utilisée comme un moyen efficace pour l'estimation de position. Certains utilisent des systèmes de capture du mouvement donnant de bonnes précisions comme [Gurdan *et al.*, 2007]. Et d'autres choisissent d'utiliser de simples caméras CCD extérieures pour localiser le drone ([Kim *et al.*, 2010]). [Mori *et al.*, 2007b] utilisent quant à eux, deux caméras pointant sur des cibles elliptiques afin de reconstruire la position et l'attitude d'un micro hélicoptère. [Zhang *et al.*, 2009b] utilisent cinq marqueurs, qui sont repérés par une caméra, pour estimer le lacet et la position d'un quadrirotor, alors qu'une IMU délivre les vitesses de rotation et les angles de roulis et tangage. [Yang *et al.*, 2013b] ont choisi d'utiliser une caméra monoculaire et de l'associer aux accéléromètres pour réaliser les tâches de décollage et atterrissage automatiques.

1.2 Motivations

Dans cette thèse, on cherche à se rapprocher un maximum, d'un point de vue sensori-moteur, du vol de la mouche, qui ne dispose que de son œil composé, d'un gyromètre et de fines soies innervées à la base de son cou, pour mesurer l'orientation de sa tête par rapport à son corps. Hors, chez la plupart des insectes comme la mouche, on ne trouve pas de capteurs capable de mesurer la gravité en vol. Comme nous l'avons montré dans le chapitre II, il est possible de stabiliser un birotor sans accéléromètres, en se basant uniquement sur les informations d'un gyromètre et de l'orientation du regard, verrouillé sur une cible placée au sol. En effet, l'orientation du regard va pouvoir être utilisée comme "inclinomètre", remplissant alors le même rôle que les accéléromètres dans l'ensemble des estimateurs précédemment cités.

Bien que cela ne prouve pas que la mouche utilise un "pilote automatique" semblable à celui que nous allons décrire, cela permet de valider une stratégie de stabilisation qui est plausible d'un point de vue biologique. De plus, ne pas utiliser d'accéléromètres permet de s'affranchir de l'influence des accélérations latérales, ou de la force centrifuge sur l'estimation d'attitude. En effet, dans le cas où l'on se base uniquement sur une IMU pour estimer l'attitude d'un aéronef, on fait l'hypothèse que les

accélérations latérales restent faibles et que, par conséquent, la mesure des accéléromètres correspond à la verticale. Ce phénomène n'apparaît donc pas si on utilise l'orientation du regard dans le cas d'une cible immobile.

2 Description du robot complet

Dans cette section, on décrit brièvement le robot complet ainsi que le modèle simplifié, utilisé pour le contrôle et les différents estimateurs.

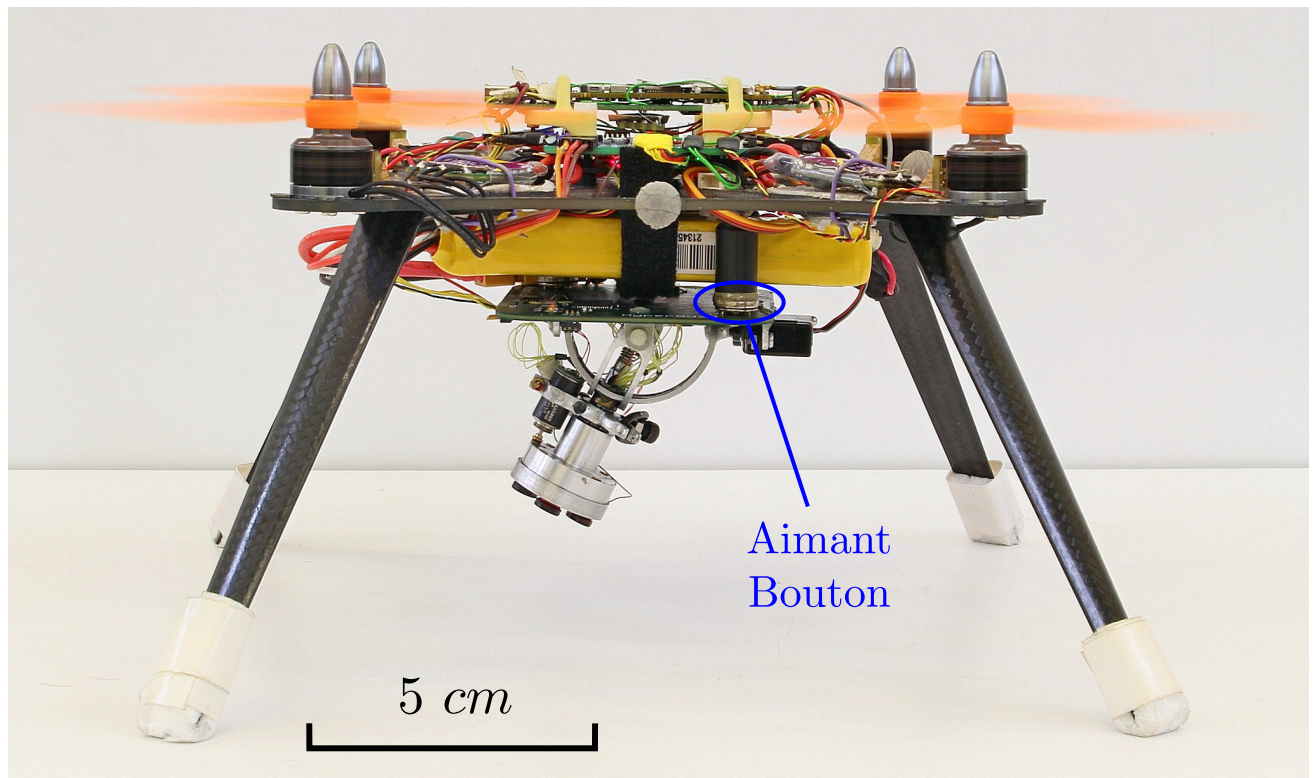


FIGURE V.1 – Le robot X4-MaG complètement assemblé et équipé de son œil. Le robot pèse au total 395 g, pour une envergure de 30 cm et une autonomie d'environ 9 min. Le robot dispose d'un microcontrôleur bas niveau Arduino (NanoWii) équipé de la centrale inertielle MPU6050 et d'un contrôleur haut niveau (Gumstix Overo). Son œil de 24 pixels lui permet de localiser une cible de type croix avec une précision de 0.1° et de la poursuivre du regard sur plus de 100° selon deux directions. Le robot dispose d'un mode de pilotage manuel, d'un mode automatique reposant sur le système de capture du mouvement Vicon et d'un mode basé uniquement sur son système de vision, ses gyromètres et un altimètre.

2.1 Assemblage complet

Le robot complet est obtenu en ajoutant au quadricoptère décrit dans le chapitre III, l'œil décrit dans le chapitre IV. De façon à simplifier l'assemblage, la carte support de l'œil est dotée de trois "aimants

boutons" mâle. Le robot, quant à lui, dispose de trois petites colonnettes en PVC au bout desquelles sont également placés des "aimants boutons" femelles. Ainsi, l'œil peut être facilement "clipsé/déclipsé" selon le besoin, et cela permet également d'avoir une sorte de "fusible" mécanique. En effet, en cas de crash et d'efforts trop importants sur l'œil, ce dernier se détache "naturellement" prévenant ainsi la casse des pièces sensibles. Ce système s'est montré particulièrement efficace lors des différentes phases de mises aux points. Quatre pieds en carbone (HobbyKing) ont également été ajoutés, afin de poser le robot sans endommager l'œil, étant donné que celui-ci est situé en dessous du robot.

La figure V.1 présente donc le robot complet avec son œil. Le robot complet avec ses pieds et son œil pèse au total 395 g pour une envergure totale de 30 cm et une hauteur de 17 cm. Avec cette configuration, il dispose d'environ 8 à 9 min d'autonomie de vol.

2.2 Modèle pour l'estimation

Nous allons maintenant décrire le modèle que nous utilisons pour concevoir l'ensemble des contrôleurs et estimateurs du robot X4-MaG final.

2.2.1 Hypothèses

Le modèle complet du quadrirotor est donné dans le chapitre III. Ce modèle complet est ici simplifié pour le contrôle et un certain nombre d'hypothèses simplificatrices sont faites :

- On suppose que le centre de gravité du robot appartient au plan de poussée des rotors.
- On néglige les couples générés par les mouvements de l'œil sur le corps du robot. En réalité, lors de mouvements rapides et de grandes amplitudes de l'œil, l'anti-couple généré par l'œil ne serait plus négligeable, et il serait utile d'ajouter un terme de feedforward sur le contrôle d'attitude du robot afin de contrer cet effet. Ceci s'avérerait notamment utile lors de saccades de l'œil en cas de changement rapide de cible (voir section 5.1 de chapitre II), mais ce point ne sera pas traité ici.
- L'effet dû au flapping est assimilé à une simple terme de frottements, proportionnels à la vitesse de translation du robot (cette hypothèse est régulièrement faite pour le modèle de contrôle des multirotores : [Mahony *et al.*, 2012]).
- La mesure de position des servomoteurs actionnant l'œil n'étant pas disponible, un capteur fictif prenant en compte les dynamiques identifiées de ces servomoteurs sera utilisé.
- Le délai introduit par la démodulation sur la mesure de position de la cible par l'œil est négligé dans les modèles d'estimateurs (mais pris en compte dans le réflexe de fixation visuelle).
- Les dynamiques des accéléromètres et gyromètres, liées au filtrage analogique du premier ordre de la MPU6050 (20 Hz), sont négligées.

- On considère que la dynamique des rotors (dynamique de la boucle fermée) fixe directement la dynamique de la poussée et des couples. On approche cette dynamique par un premier ordre de constante de temps τ_{mot} (voir l'identification de la boucle fermée dans la section 2.5.3 du chapitre III).
- Dans tout le chapitre, on considérera que les orientations de l'œil (θ_{er_ϕ} et θ_{er_θ}) ainsi que les erreurs rétinienne (ϵ_{r_ϕ} et ϵ_{r_θ}) sont exprimées dans le repère de l'œil, et que l'attitude du robot (ϕ , θ et ψ) est exprimée dans le repère inertielle.
- On supposera également que l'angle de lacet de cible est nulle dans le repère inertielle.
- Enfin, on supposera que le robot a un angle de lacet proche de celui de la cible (ce qui est bien le cas quand celui ci est asservi sur celui de la cible), ce qui permettra de supposer que la position angulaire de la cible θ_{t_θ} selon l'axe X est uniquement liée à l'orientation en tangage de l'œil θ_{er_θ} et à l'erreur rétinienne ϵ_{r_θ} associée. Et la position angulaire de la cible θ_{t_ϕ} selon l'axe Y uniquement lié au roulis de l'œil θ_{er_ϕ} et à l'erreur rétinienne ϵ_{r_ϕ} associée. Voir équations (V.4), (V.5) et (V.6).

2.2.2 Modèle des capteurs et des actionneurs

Gyromètres : On suppose dans toute cette partie un modèle classique de gyromètre. A savoir, que la mesure $\bar{\Omega}$ est entachée d'un biais $b_{\bar{\Omega}}$ constant (ou variant lentement) et d'un bruit additif $\eta_{\bar{\Omega}}$ centré, c'est à dire de moyenne nulle :

$$\begin{cases} \bar{\Omega} = \Omega + b_{\bar{\Omega}} + \eta_{\bar{\Omega}} \\ \dot{b}_{\bar{\Omega}} = 0 \end{cases} \quad (\text{V.1})$$

Dans cette partie, on cherchera donc à estimer en ligne le biais $b_{\bar{\Omega}}$ afin de déterminer les vitesses de rotation réelles Ω du robot.

Accéléromètre : De même que pour les gyromètres, on suppose que la mesure \bar{a} des accéléromètres est entachée par un biais constant $b_{\bar{a}}$ et un facteur d'échelle T (considérés purement constants ici) et d'un bruit $\eta_{\bar{a}}$ centré.

$$\bar{a} = T(a + b_{\bar{a}}) + \eta_{\bar{a}} \quad (\text{V.2})$$

Ici, on considère que les biais $b_{\bar{a}}$ et le facteur d'échelle T des accéléromètres sont constants et qu'ils ont été identifiés une fois pour toute (grâce à la méthode décrite dans la section 5.1 du chapitre III), ce qui permet de les compenser.

L'œil : On décrit également les relations géométriques permettant de décrire la position angulaire de la cible en fonction des orientations de l'œil, de l'attitude et de la position du robot (par extension des

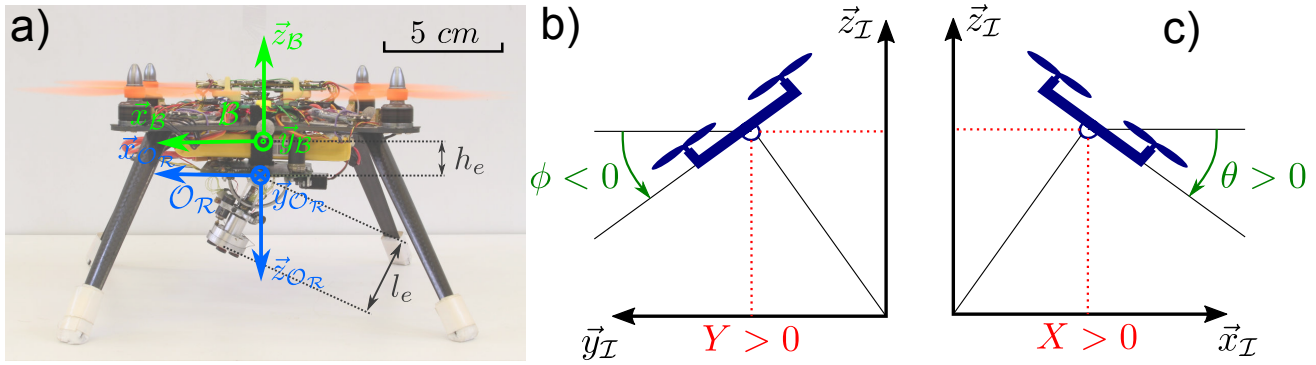


FIGURE V.2 – Localisation de la cible et repères utilisés pour l'estimation ; on suppose ici un angle de lacet entre la cible et le robot nul. a) Description des repères du robot \mathcal{B} et de l'œil \mathcal{O}_R . Dans ce chapitre, pour simplifier les estimations, les distances h_e et l_e seront négligées et supposées nulles. b) Rappel du lien entre l'angle de roulis et la position Y pour un angle $\theta_{er_\phi} = 0^\circ$ et une erreur rétinienne $\epsilon_{r_\phi} = 0^\circ$. c) Rappel du lien entre l'angle de tangage et la position X pour un angle $\theta_{er_\theta} = 0^\circ$ et une erreur rétinienne $\epsilon_{r_\theta} = 0^\circ$.

travaux présentés dans le chapitre II). On obtient donc :

$$\begin{cases} \theta_{t_\phi} = \underbrace{\theta_{er_\phi} - \phi}_{\theta_{g_\phi}} + \epsilon_{r_\phi} = \tan^{-1} \left(\frac{Y}{Z} \right) \\ \theta_{t_\theta} = \underbrace{\theta_{er_\theta} + \theta}_{\theta_{g_\theta}} + \epsilon_{r_\theta} = \tan^{-1} \left(\frac{X}{Z} \right) \\ \theta_{t_\psi} = \underbrace{-\psi}_{\theta_{g_\psi}} + \epsilon_{r_\psi} \end{cases} \quad (\text{V.3})$$

Où $\theta_{t_{\phi,\theta}}$ représente la position angulaire de la cible par rapport au robot dans le repère inertiel selon les axes de roulis et de tangage. θ_{t_ψ} représente l'orientation de la cible par rapport à celle du robot. ϕ , θ et ψ sont respectivement les angles de roulis, tangage et lacet, ϵ_{r_ϕ} , ϵ_{r_θ} et ϵ_{r_ψ} sont les erreurs d'orientations de l'œil (c'est à dire l'écart entre la position angulaire de la cible $\theta_{t_{\phi,\theta,\psi}}$ et la position angulaire du regard $\theta_{g_{\phi,\theta,\psi}}$ exprimée dans le repère inertiel \mathcal{I}). Enfin X , Y et Z décrivent les positions cartésiennes du robot dans le repère inertiel.

On suppose ici que la cible reste toujours dans le champ visuel de l'œil et qu'elle est presque alignée avec l'œil (l'écart entre l'angle de lacet de la cible et du robot est petit, et on peut supposer que le roulis et le tangage sont découplés). Par conséquent, on peut raisonnablement supposer que les erreurs rétinienne mesurées correspondent aux écarts entre l'orientation de l'œil et l'orientation de la cible,

auxquelles s'ajoute un bruit centré. On peut donc construire le modèle de mesure suivant :

$$\bar{\theta}_{t_\phi} = \hat{\theta}_{er_\phi} - \hat{\phi} + \underbrace{\epsilon_{r_\phi} + \eta_{\bar{\epsilon}_{r_\phi}}}_{\bar{\epsilon}_{r_\phi}} \quad (\text{V.4})$$

$$\bar{\theta}_{t_\theta} = \hat{\theta}_{er_\theta} + \hat{\theta} + \underbrace{\epsilon_{r_\theta} + \eta_{\bar{\epsilon}_{r_\theta}}}_{\bar{\epsilon}_{r_\theta}} \quad (\text{V.5})$$

$$\bar{\theta}_{t_\psi} = -\hat{\psi} + \underbrace{\epsilon_{r_\psi} + \eta_{\bar{\epsilon}_{r_\psi}}}_{\bar{\epsilon}_{r_\psi}} \quad (\text{V.6})$$

Où $\bar{\theta}_{t_{\phi,\theta,\psi}}$ représentent respectivement la position angulaire de la cible mesurée selon les axes \vec{x} , \vec{y} et \vec{z} . $\bar{\epsilon}_{r_{\phi,\theta,\psi}}$ représentent les erreurs rétinienne mesurées sur chaque axe, que l'on considère comme étant la vraie erreur rétinienne $\epsilon_{r_{\phi,\theta,\psi}}$ entachée d'un bruit $\eta_{\bar{\epsilon}_{r_{\phi,\theta,\psi}}}$. $\hat{\phi}$, $\hat{\theta}$ et $\hat{\psi}$ sont les angles d'attitude estimés. Et $\hat{\theta}_{er_{\phi,\theta}}$ représente l'orientation du regard estimée (car on ne la mesure pas) selon les axes \vec{x} et \vec{y} , où l'estimation est faite comme suit :

$$\begin{cases} \hat{\theta}_{er_\phi}(p) = H_{\text{servo}}(p) \cdot \theta_{er_\phi}^*(p) \\ \hat{\theta}_{er_\theta}(p) = H_{\text{servo}}(p) \cdot \theta_{er_\theta}^*(p) \end{cases} \quad (\text{V.7})$$

Où $H_{\text{servo}}(p)$ est la fonction de transfert entre la consigne de position $\theta_{er_{\phi,\theta}}^*$ et la position réelle du servomoteur $\theta_{er_{\phi,\theta}}$. Cette dynamique a été identifiée selon la méthode décrite dans la section 2.4.3 du chapitre IV et la fonction de transfert est donnée dans le tableau IV.3.

Remarque V.1: L'amplitude du bruit sur les erreurs rétinienne est obtenue dans le chapitre IV (section 5.1). L'amplitude de bruit η_{r_ψ} de la mesure $\bar{\epsilon}_{r_\psi}$ sera étudiée dans la section 4.3.1 de ce chapitre. En effet, la qualité de la mesure étant beaucoup plus mauvaise sur cet axe (voir explications), une attention particulière sera accordée à cette mesure $\bar{\epsilon}_{r_\psi}$.

2.2.3 Modèle complet

On peut donc, suite aux hypothèses précédentes, déduire le modèle complet du quadrirotor X4-MaG :

$$\begin{cases} \dot{\xi} = v \\ \dot{v} = \frac{1}{m} (R.T.\mathcal{B}\vec{z} - K_v.v) \\ \dot{R} = R.\Omega_\times \\ I\dot{\Omega} = -\Omega \times I\Omega + \Gamma \end{cases} \Leftrightarrow \begin{cases} \dot{\xi} = v \\ \dot{v} = \frac{T}{m} \cdot \begin{pmatrix} \sin(\phi) \sin(\psi) + \cos(\phi) \cos(\psi) \sin(\theta) \\ \cos(\phi) \sin(\psi) \sin(\theta) - \sin(\phi) \cos(\psi) \\ \cos(\phi) \cos(\theta) \end{pmatrix} - \frac{K_v}{m} . v \\ \dot{\Omega} = \Omega \\ I\dot{\Omega} = -\Omega \times I\Omega + \Gamma \end{cases} \quad (\text{V.8})$$

$$\left\{ \begin{array}{l} T(p) = \frac{1}{(\tau_{\text{mot}}p + 1)^2} T^*(p) \\ \hat{\theta}_{er_\phi}(p) = \frac{z_{\text{servo}}p + 1}{\frac{1}{\omega_{\text{servo}}^2}p^2 + \frac{2\eta}{\omega_{\text{servo}}p + 1}} \cdot \theta_{er_\phi}^*(p) \end{array} \right. , \quad \left\{ \begin{array}{l} \Gamma(p) = \frac{1}{(\tau_{\text{mot}}p + 1)^2} \Gamma^*(p) \\ \hat{\theta}_{er_\theta}(p) = \frac{z_{\text{servo}}p + 1}{\frac{1}{\omega_{\text{servo}}^2}p^2 + \frac{2\eta}{\omega_{\text{servo}}p + 1}} \cdot \theta_{er_\theta}^*(p) \end{array} \right. \quad (\text{V.9})$$

Le système (V.8) représente le modèle dynamique du robot qui est assimilé à un corps rigide. $\xi = (X \ Y \ Z)^T$ est le vecteur de position dans le repère inertiel, v le vecteur de vitesse de translation dans le même repère. $T \cdot \mathcal{B} \vec{z}$ est la poussée des rotors orientée selon l'axe z du repère du robot. K_v est le terme de frottement approximant les frottements visqueux ainsi que le phénomène de flapping. R est la matrice de rotation associée au vecteur $\Theta = (\phi \ \theta \ \psi)^T$ permettant de passer du repère du robot au repère inertiel et R est décrite dans le chapitre III. Ω est le vecteur de vitesse de rotation, et Γ est le vecteur de couples appliqué au robot.

Le système d'équation (V.9) représente les dynamiques des différents actionneurs. Les couples Γ et poussée T découlent des grandeurs de commandes Γ^* et T^* (fournies par les contrôles d'attitude et de position), et sont assujettis à une dynamique du premier ordre, liés aux rotors. Les servomoteurs actionnant l'œil sont assimilés à un système du second ordre permettant d'obtenir les orientations réelles de l'œil à partir des positions de référence $\theta_{er_{\phi,\theta}}^*$, découlant du contrôle du regard (voir section 2.4.3 du chapitre IV).

2.3 Contrôle de position et d'attitude

Dans toute la suite de ce chapitre, on utilisera toujours le même ensemble de contrôleurs (contrôleurs d'attitude, de position et du regard). Ces contrôleurs nécessitent la connaissance complète des différents états du modèle, décrit par le système (V.8). On changera donc uniquement les estimateurs permettant de reconstruire ces états et on conservera les paramètres des différents contrôleurs à l'identique.

On choisit donc des paramètres permettant d'obtenir un comportement robuste.

3 Stabilisation visuelle et réflexes

Ces dernières années ont vu naître de nombreux systèmes de stabilisation pour les différents matériels de capture vidéo embarqués à bord de drone. Ces systèmes présentent la plupart du temps un découplage sur deux axes, en roulis et en tangage, bien que plus récemment, on ait vu apparaître des systèmes présentant également un découplage selon le lacet. Ces systèmes sont connus sous le nom de *Gimbal* (système mécatronique de stabilisation visuelle) ou encore de caméra *pan-tilt* et sont, la plupart du temps, destinés à la prise de vue dans des domaines divers et variés (événements sportifs, reportages, contrôle de grandes structures, etc.) Dans ce cadre, de nombreux systèmes ont été développés pour la célèbre *GoPro* qui allie la légèreté (environ 70g), un grand angle de vue et une image en haute résolution. La stabilisation de ces systèmes *Gimbal* est réalisée à l'aide de moteur brushless dédiés,

appelés moteurs *Gimbal*, offrant des couples importants et des vitesses de rotations faibles. Ceci permet donc de contrôler directement l'inclinaison de la caméra, sans passer par des étages de réduction. La compensation des mouvements du drone est réalisée grâce à une carte contrôleur dédiée embarquant une centrale inertielle supplémentaire (en plus de celle du pilote automatique du robot) et solidaire de la caméra. Cette carte GCU (pour Gimbal Control Unit) est souvent placée au plus près du centre de rotation, afin d'estimer directement l'attitude de la caméra et de la stabiliser autour d'une position souhaitée.

Ce genre de système est également utilisé dans le domaine de la recherche, afin de permettre la poursuite autonome de cible. Dans la plupart des cas, les caméras *Pan/tilt* ou *Gimbal* sont utilisées sur des drones à voilure fixe, du fait des fortes contraintes non-holonomes de ces systèmes. [Wang *et al.*, 2005] ont, par exemple, étudié en simulation des algorithmes permettant de générer des trajectoires pour un drone à voilure fixe, équipé d'une caméra *Gimbal*. Ils ont alors proposé des algorithmes permettant d'estimer la position de la cible et ont étudié la robustesse de ces algorithmes à la perte de cible. [Rafi *et al.*, 2006] ont également étudié la poursuite de cible mobile, à l'aide d'une aile volante équipée d'une caméra *Pan/tilt*. [Quigley *et al.*, 2005] ont également proposé une approche permettant d'estimer la position d'une cible, à partir d'un drone à voilure fixe, piloté manuellement et équipé d'une caméra *Gimbal* hémisphérique. Puis [Dobrokhodov *et al.*, 2006] ont démontré qu'il était possible de poursuivre de façon efficace, une cible évoluant à $4\text{-}5\text{m} \cdot \text{s}^{-1}$, tout en estimant sa position (précision de localisation de la cible d'environ 50m pour un drone évoluant à 400m d'altitude.) Les travaux de [Campbell *et Wheeler*, 2006] visent également à fournir un estimateur générique, permettant de géolocaliser une cible depuis le *SeaScan UAV* (drone à voilure fixe également) et ce, en présence de biais et d'incertitude sur le système de localisation du drone. Ils ont ensuite présenté une évolution de l'algorithme, pour une géolocalisation coopérative, via une flotte de drone dans [Campbell *et Whitacre*, 2007]. Mais on remarque également l'utilisation de caméra entièrement gyro-stabilisée sur des systèmes à voilure tournante. Par exemple, [Jakobsen *et Johnson*, 2005] ont développé un système *Gimbal* complet qui a été validé en HIL (Hardware-In-the-Loop) sur un drone de type hélicoptère (basé sur le Yamaha R-max).

Dans toutes ces études, le système de découplage est vu comme un moyen pour stabiliser un appareil de capture d'image autonome. C'est à dire que le système a une mission de "surveillance" ou de poursuite qui lui est propre et il n'intervient pas dans l'estimation de position ou d'attitude de l'engin volant lui-même. Dans le cas de mission de poursuite, le système sert également à estimer la position et les éventuels mouvements de la cible comme dans [Quigley *et al.*, 2005], [Dobrokhodov *et al.*, 2006] ou encore [Campbell *et Wheeler*, 2006].

Dans ma thèse, je me suis intéressé à ce que peut apporter l'orientation du regard, dans l'estimation des états du robot lui-même. Relativement peu de travaux ont été menés dans ce domaine, pour la reconstruction complète de l'attitude et de la position d'un drone. Par exemple, [Dusha *et al.*, 2007] ont montré qu'il était possible d'extraire une ligne d'horizon et d'estimer le roulis d'un engin à voilure

fixe. [Ettinger *et al.*, 2003] ont également montré qu'il était possible d'asservir l'attitude de micro drone à l'aide de la vision, en estimant l'attitude à partir de l'horizon. Puis [Démonceaux *et al.*, 2006] ont introduit l'utilisation des caméras catadioptriques pour estimer l'attitude d'un engin aérien. Mais il a fallu attendre les travaux de [Mondragón *et al.*, 2010a] pour voir apparaître l'utilisation de système *pan/tilt* dans l'estimation d'attitude. Dans ces travaux, le système *pan/tilt* permet de maintenir parallèle l'axe vertical d'une caméra catadioptrique avec celui d'un hélicoptère, afin d'estimer l'attitude de l'appareil. [Mondragón *et al.*, 2010b] présentent une solution complète, permettant d'estimer l'attitude du UAV COLIBRI (hélicoptère autonome) à partir d'une caméra catadioptrique, d'estimer la position et l'altitude de l'appareil à partir de caméras stéréos, et de poursuivre une cible à partir d'une caméra *pan/tilt*. Dans ces travaux, l'autonomie est donc obtenue en associant différents systèmes de caméras, ayant chacun une mission précise (estimation d'attitude, odométrie ou poursuite de cible).

Dans mes travaux de thèse, l'idée était donc d'utiliser un unique système de vision pour réaliser à la fois toutes ces missions. L'utilisation d'un système de découplage mécanique pour la vision, n'a donc pas pour but la stabilisation d'une image, mais du regard en général. J'ai cherché donc à utiliser les capacités de poursuite qu'offre un tel système, pour suivre une cible du regard et utiliser la mesure de l'orientation du regard, pour en déduire des informations sur la position et l'attitude du drone simultanément. Pour cela, il a été décidé de découper le problème en deux fonctions indépendantes :

- **Une fonction de poursuite de cible**, implantée sous la forme d'un réflexe bas niveau que l'on appelle réflexe de fixation visuelle (VFR, pour Visual Fixation Reflex) et qui guide le regard.
- **Une fonction de stabilisation du regard**, également implantée sous la forme d'un réflexe, nommé réflexe vestibulo-oculaire (VOR). Ce dernier permet de compenser tous les mouvements que fait le robot, de façon à ne pas impacter la tâche de poursuite de cible.

Le regard évolue donc librement par rapport au robot et on utilise la mesure de son orientation (sa direction) pour en déduire l'attitude et la position du robot, grâce à la connaissance de son modèle. On parle de réflexes, car il n'y a pas de prise de décision ou de changement de comportement en fonction de la situation. Les réflexes sont donc des mécanismes au cœur de cette stratégie et nous allons décrire en détail leur mise en œuvre.

3.1 La stabilisation visuelle chez l'insecte

Chez l'insecte, comme la mouche, mais également chez la plupart des êtres vivants, il existe une stabilisation du regard efficace, permettant d'améliorer la perception de l'environnement. Les mécanismes de stabilisation du regard chez l'homme ont été largement étudiés dans la littérature et il a été montré qu'il s'agissait d'un comportement réflexe, appelé VOR (Réflexe Vestibulo-Occulaire). Un réflexe similaire existe chez la mouche et réagit de façon antagoniste aux rotations qui sont perçues par l'insecte et permettent une stabilisation du regard selon l'angle de roulis [Hengstenberg, 1988], tangage [Van Hataren et Schilstra, 1999] et lacet [Liske, 1977]. Par exemple, chez la mouche, les vitesses

de rotations sont perçues grâce à des balanciers, qui battent en opposition de phase avec les ailes et dont le plan de battement est modifié par la force de Coriolis ([Pringle, 1948]). Cette modification est perçue, puis codée, afin d'en déduire une mesure des vitesses angulaires. A partir de cette information de vitesse, un réflexe induit une rotation inverse de la tête. Dans le cas de l'insecte, la compensation se fait en vitesse. Cela a été vérifié, car si l'insecte est placé dans le noir et qu'on le soumet à des mouvements de rotations de faibles amplitudes, il compense ces rotations mais on voit apparaître un biais au cours du temps. La tête initialement horizontale se penche peu à peu. Cela étant dû à l'accumulation d'erreurs de compensation, liées, soit à une erreur de mesure de la vitesse, soit à une compensation imparfaite. Par contre, si la mouche est placée face à un horizon artificiel, ce biais n'apparaît plus. On en déduit que la vision permet de "recaler" cette dérive. Cela se confirme dans le cas où l'on place la mouche face à un horizon artificiel et que l'on incline ce dernier. La mouche tourne alors la tête pour maintenir l'horizon toujours horizontal dans son champ de vision, il s'agit du deuxième réflexe appelé VFR (réflexe de fixation visuelle). On constate donc la complémentarité de ces deux réflexes.

3.2 Stabilisation visuelle chez X4-MaG

J'ai donc décidé d'implémenter ces deux mêmes réflexes chez le robot X4-MaG. Le VFR est réalisé, comme décrit dans la section 4.3 du chapitre IV, en asservissant l'erreur rétinienne selon chaque axe autour de 0 :

$$\begin{cases} \theta_{er\phi_VFR}^*(t) = k_p \cdot \left(\tilde{\epsilon}_{r\phi}(t) + k_i \int_0^t \tilde{\epsilon}_{r\phi}(t) dt \right) \\ \theta_{er\theta_VFR}^*(t) = k_p \cdot \left(\tilde{\epsilon}_{r\theta}(t) + k_i \int_0^t \tilde{\epsilon}_{r\theta}(t) dt \right) \end{cases} \quad (\text{V.10})$$

Où $\tilde{\epsilon}_{r\phi}(t) = \epsilon_{r\phi}^*(t) - \bar{\epsilon}_{r\phi}(t)$ et $\tilde{\epsilon}_{r\theta}(t) = \epsilon_{r\theta}^*(t) - \bar{\epsilon}_{r\theta}(t)$ et on fixe $\epsilon_{r\phi}^*(t) = \epsilon_{r\theta}^*(t) = 0 \forall t$.

Le VOR, quant à lui, est réalisé un peu différemment de celui de la mouche. En effet, étant donné que nous utilisons des servo-moteurs qui sont commandés en position, le VOR n'est pas implanté sous la forme $\Omega_{er}^* = -\bar{\Omega}_{er}$, mais consiste en une consigne d'orientation opposée à l'attitude du robot :

$$\begin{cases} \theta_{er\phi_VOR}^*(t) = -\hat{\phi}(t) + \theta_{erG\phi}^*(t) \\ \theta_{er\theta_VOR}^*(t) = \hat{\theta}(t) + \theta_{erG\theta}^*(t) \end{cases} \quad (\text{V.11})$$

Où $\hat{\phi}(t)$ et $\hat{\theta}(t)$ sont les estimations des angles de roulis et de tangage du robot. $\theta_{erG\phi}^*(t)$ et $\theta_{erG\theta}^*(t)$ sont des angles permettant d'orienter le regard dans une direction donnée, ce qui peut être vu comme un offset. Ici on choisit $\theta_{erG\phi}^*(t) = \theta_{erG\theta}^*(t) = 0 \forall t$. Au final, la stabilisation visuelle et la poursuite de la cible sont obtenues en combinant ces deux réflexes :

$$\begin{cases} \theta_{er\phi}^*(t) = \theta_{er\phi_VOR}^*(t) + \theta_{er\phi_VFR}^*(t) \\ \theta_{er\theta}^*(t) = \theta_{er\theta_VOR}^*(t) + \theta_{er\theta_VFR}^*(t) \end{cases} \quad (\text{V.12})$$

D'un point de vu pratique, les servomoteurs, permettant d'actionner l'œil, ne peuvent pas être pilotés à plus de 333Hz. Ces réflexes sont donc rafraichis toutes les 3ms.

3.3 Résultats et discussions

Afin de valider l'efficacité du VOR réalisant la stabilisation visuelle, on a appliqué manuellement de fortes perturbations de roulis et de tangage de façon aléatoire et on a vérifié que l'œil pointe bien toujours dans la direction verticale. Afin d'éviter les interactions avec le réflexe de fixation visuelle (VFR), ce dernier a été désactivé.

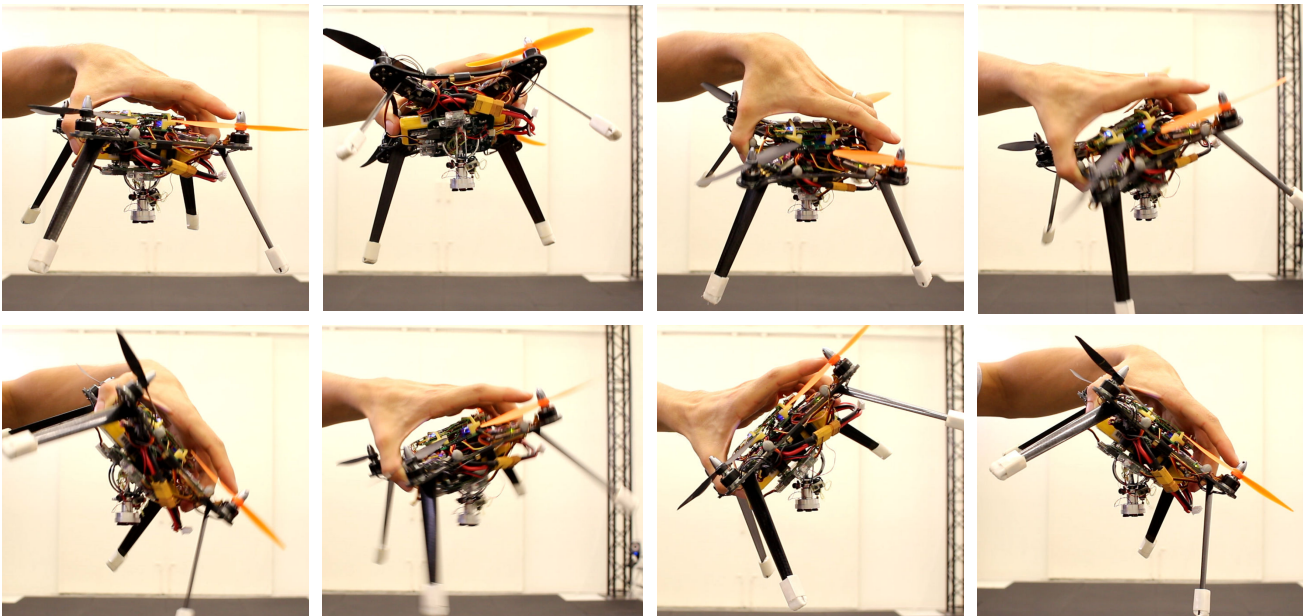


FIGURE V.3 – Séquence photographique mettant en évidence le VOR en charge de stabiliser le regard (orientation absolue de l'œil). On observe que malgré les mouvements rapides du robot, l'œil maintient son regard orienté selon la direction verticale.

La figure V.3 montre une séquence de photographies, démontrant l'efficacité du VOR. Cette séquence correspond à la vidéo `VOR.avi`. Il n'a pas été possible de placer des marqueurs Vicon sur l'œil lui même, du fait de sa petite taille, et de sa haute réflexivité, perturbant la localisation des marqueurs placés sur l'œil (qui est en bonne partie réalisé en aluminium). Par conséquent, nous ne disposons pas de l'orientation réelle de l'œil mais uniquement de celle du robot. Cependant, en étudiant finement la vidéo, et la figure V.3, on observe que malgré les mouvements très rapides de rotation infligés au robot, l'œil maintient très bien la verticale (voir ligne verticale jaune sur la figure V.3). Certains clichés extraits de la vidéo sont légèrement flous, montrant de fait les mouvements de rotation rapides. On observe cependant que la l'orientation verticale de l'œil est bien maintenue constante.

Lors de travaux préliminaires, j'avais mis en œuvre le même réflexe sur un birotor, comme décrit dans l'article [Colonnier *et al.*, 2015a]. La stabilisation visuelle était alors réalisée exactement de la

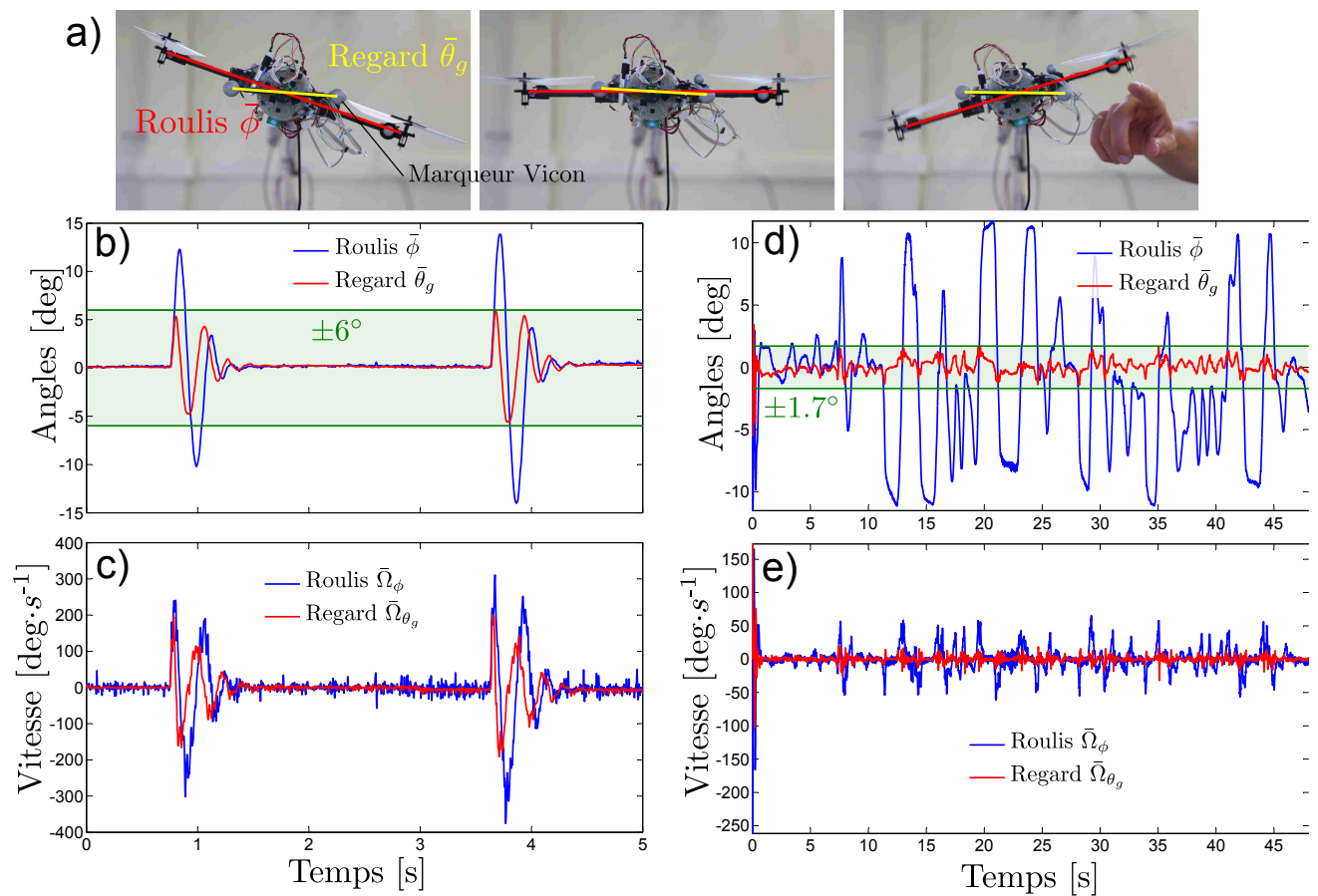


FIGURE V.4 – a) Série de photographies montrant la stabilisation visuelle lors de perturbations en roulis. D'après [Colonnier et al., 2015a] b) En bleu le roulis et en rouge l'orientation du regard lors de perturbations manuelles en roulis. c) Évolution des vitesses de rotation du roulis et du regard lors des perturbations manuelles. d) Évolution du roulis et du regard lors d'une séquence de poursuite d'un panneau texturé situé en dessous du robot. e) Évolution des vitesses angulaires du regard et du roulis lors de la poursuite du panneau.

même manière, et le robot avait été équipé de façon à pouvoir mesurer, à l'aide du Vicon, l'angle de roulis ϕ du robot ainsi que l'orientation du regard θ_g . Le servomoteur utilisé présentait des performances du même ordre que ceux utilisés sur X4-Mag. La figure V.4 montre la mesure de l'orientation du regard $\bar{\theta}_g$ et la mesure de l'angle de roulis $\bar{\phi}$ lors de perturbations et lors de mouvements complexes en roulis. Des marqueurs Vicon avaient été placés au bout de tiges, orientées de façon orthogonales par rapport au regard, permettant de mesurer l'angle du regard. De même, le robot était équipé de marqueurs, permettant de mesurer son orientation selon l'axe de roulis (voir figure V.4a). Le robot était positionné au bout d'un bras rotatif, tournant sur le plan azimutal et autorisant des mouvements de pseudo-translation au robot. Sur les figures V.4b-c, le bras était verrouillé à une position (pas de translation possible du robot), et successivement deux perturbations en roulis de l'ordre d'une quinzaine de degrés étaient appliquées. On observe que l'orientation du regard $\bar{\theta}_g$ reste inférieure à 6° , alors que les vitesses de

rotation atteignent près de $300^\circ \cdot s^{-1}$. Le fait que le robot ne parvient pas à maintenir complètement son regard à la verticale ($\theta_g = 0^\circ$) s'explique par la dynamique du servomoteur. Les figures V.4d-e montrent l'évolution du roulis et du regard du robot, lorsque celui-ci poursuit un pattern visuel situé en dessous de lui. Le robot ajuste donc son angle de roulis pour poursuivre le panneau qui est déplacé manuellement, et le réflexe Vestibulo-Occulaire essaye de maintenir le regard à la verticale ($\theta_g = 0$). On observe donc que lors de déplacements plus "naturels" du robot, le réflexe donne des résultats tout à fait satisfaisants, car le regard est maintenu inférieur à 1.7° .

4 Stabilisation bio-inspirée

Nous allons donc maintenant présenter 3 estimateurs non linéaires différents, développés sous forme de filtres de Kalman étendus (EKF).

Pour cela, nous allons retirer progressivement des mesures, afin d'augmenter à chaque fois l'ordre de l'estimateur. Cette méthode itérative, nous permet de valider progressivement le modèle et facilite le réglage des différents estimateurs.

Les filtres de Kalman seront synthétisés sous forme discrète, nous présenterons donc pour chaque filtre la version discrétisé et plus ou moins réduite (en fonction du nombre d'états à estimer) du modèle donné par le système d'équations (V.8). Ces filtres permettront d'estimer à une fréquence de 400 Hz les différents états du système avec différentes données d'entrées. Afin de réduire l'ordre des EKF, on décide de représenter les dynamiques des actionneurs sous forme de fonctions de transfert discrètes à l'entrée des filtres.

Les fonctions de transfert discrètes sont obtenues pour une discrétisation par le modèle Euler Forward, pour une période d'échantillonnage de 2.5ms (soit une fréquence de 400Hz). On utilisera donc systématiquement le modèle suivant pour estimer l'orientation de l'œil (angles $\hat{\theta}_{er_\phi}$ et $\hat{\theta}_{er_\theta}$) :

$$\begin{cases} \frac{\hat{\theta}_{er_\phi}(z)}{\theta_{er_\phi}^*(z)} = \frac{-0.1048 + 0.1504z^{-1}}{1 - 1.7275z^{-1} + 0.7731z^{-2}} \\ \frac{\hat{\theta}_{er_\theta}(z)}{\theta_{er_\theta}^*(z)} = \frac{-0.1048 + 0.1504z^{-1}}{1 - 1.7275z^{-1} + 0.7731z^{-2}} \end{cases} \quad (V.13)$$

Ce qui nous donne les équations de récurrences suivantes pour l'estimation des orientations de l'œil :

$$\begin{cases} \hat{\theta}_{er_\phi}(k) = -0.1048 \theta_{er_\phi}^*(k) + 0.1504 \theta_{er_\phi}^*(k-1) + 1.7275 \hat{\theta}_{er_\phi}(k-1) - 0.7731 \hat{\theta}_{er_\phi}(k-2) \\ \hat{\theta}_{er_\theta}(k) = -0.1048 \theta_{er_\theta}^*(k) + 0.1504 \theta_{er_\theta}^*(k-1) + 1.7275 \hat{\theta}_{er_\theta}(k-1) - 0.7731 \hat{\theta}_{er_\theta}(k-2) \end{cases} \quad (V.14)$$

De même, à partir de la dynamique des moteurs et des consignes de couples Γ^* et de poussée T^* ,

on peut estimer les couples et la poussée réellement générés :

$$\begin{cases} \frac{\hat{T}(z)}{\hat{T}^*(z)} = \frac{0.0057 + 0.0053z^{-1}}{1 - 1.790z^{-1} + 0.801z^{-2}} \\ \frac{\hat{\Gamma}(z)}{\hat{\Gamma}^*(z)} = \frac{0.0057 + 0.0053z^{-1}}{1 - 1.790z^{-1} + 0.801z^{-2}} \end{cases} \quad (\text{V.15})$$

Ce qui nous donne les équations au récurrence suivante pour l'estimateur de poussée et de couples :

$$\begin{cases} \hat{T}(k) = 0.0057 T^*(k) + 0.0053 T^*(k-1) + 1.790 \hat{T}(k-1) - 0.801 \hat{T}(k-2) \\ \hat{\Gamma}(k) = 0.0057 \Gamma^*(k) + 0.0053 \Gamma^*(k-1) + 1.790 \hat{\Gamma}(k-1) - 0.801 \hat{\Gamma}(k-2) \end{cases} \quad (\text{V.16})$$

4.1 Asservissement en position basé vision

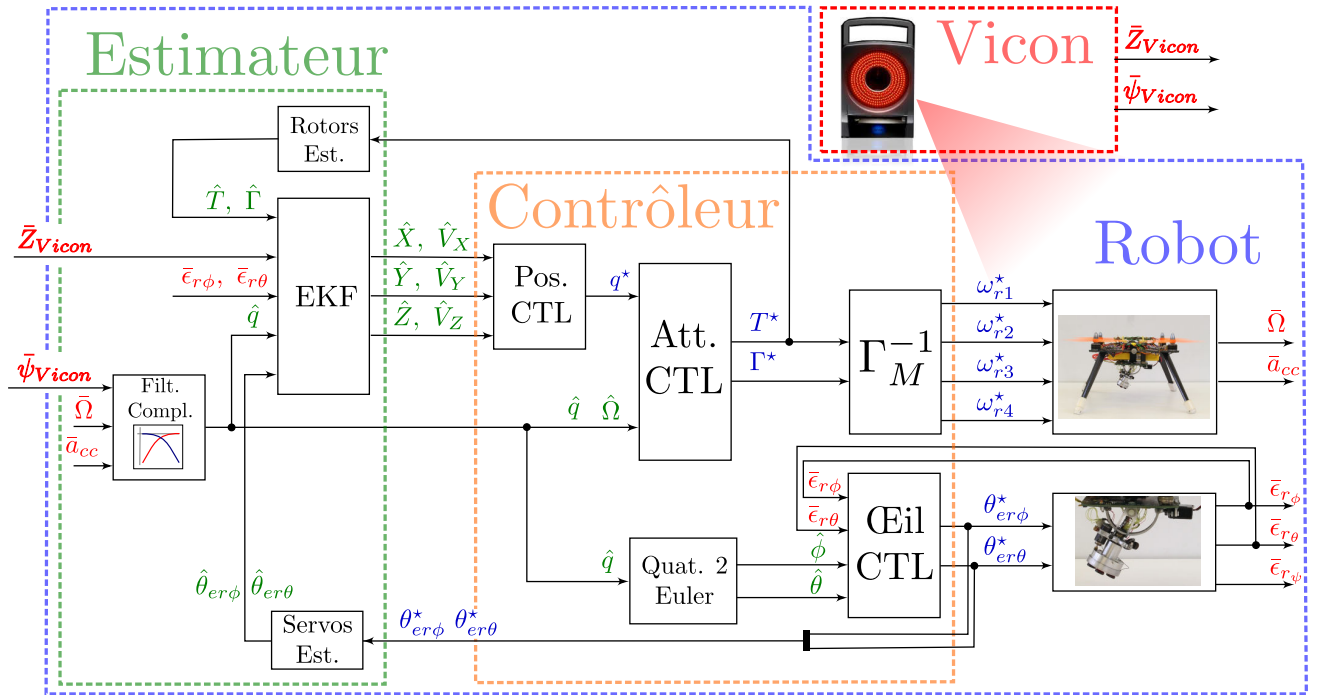


FIGURE V.5 – Diagramme en bloc du système complet. Le bloc **Filt. Compl.** réalise l'estimation d'attitude (III.64), **Servos Est.** estime la position réelle des servomoteurs à partir de leur modèle dynamique (V.14) ; Le bloc **Rotors Est.** estime la poussée générée à partir du modèle dynamique des rotors (V.16). Le bloc **EKF** correspond au filtre de Kalman étendu estimant la position et la vitesse du robot (V.25) à (V.32). **Pos. CTL** est le contrôleur de position (III.71) à (III.74), **Att. CTL** le contrôleur d'attitude (III.70) et **Œil CTL** est le contrôleur de regard (V.10). Enfin, **Quat. 2 Euler** permet d'extraire du quaternion d'attitude, les angles de roulis et de tangage et Γ_M^{-1} permet de passer de la consigne de couples et de poussée aux consignes rotors (III.62). Ici, le système Vicon permet de capturer l'angle de lacet ainsi que l'altitude du robot.

Dans un premier temps nous ne cherchons qu'à estimer la position et la vitesse du robot dans le

repère inertiel, c'est à dire $x = (X \ Y \ Z \ V_X \ V_Y \ V_Z)^T$. Le but principal est donc de vérifier que l'œil est bien capable de poursuivre la cible du regard pendant un vol du robot. Dans un premier temps, on simplifie donc le problème, en considérant que l'on a accès à l'attitude complète du robot, ainsi qu'à son altitude. L'attitude est déterminée grâce au filtre complémentaire décrit dans le chapitre III, dans lequel on fusionne les mesures des gyromètres, des accéléromètres et la mesure de lacet fournie par le Vicon. On récupère également la hauteur Z fournie par le Vicon, qui est fusionnée dans un EKF avec les orientations de l'œil, les erreurs rétinienne et l'attitude estimée afin de déterminer la position et la vitesse du robot. La figure V.5 résume les interactions des différents estimateurs et contrôleurs. Cette première version dépend donc des mesures de hauteur et de lacet fournies par le Vicon, et utilise également les accéléromètres pour l'estimation d'attitude.

Dans la suite, nous allons construire un filtre de Kalman étendu afin de reconstruire les différents états du système, mais avant cela on propose de vérifier l'observabilité du système. On décrit donc ici comment cette étude d'observabilité est menée :

Notions d'observabilité : On vérifie en premier lieu que le système non linéaire est bien observable, ce qui est nécessaire pour construire l'observateur d'état. Nous vérifions l'observabilité locale faible basée sur la *dérivée de Lie*. Soit le système non linéaire suivant :

$$\begin{cases} \dot{x} = f(x, u(t)) \\ y = h(x) \end{cases}$$

Où x représente le vecteur de dimension n des états du système, $u(t)$ représente le vecteur de dimension m des entrées de commande, f et h sont des fonctions différentiables telles que $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ et $h : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p$.

Définition V.1: Soit $h \in \mathbb{R}^p$, une fonction régulière, on appelle *dérivée de Lie* de h dans la fonction f , notée $\mathcal{L}_f h$, la dérivée le long de la courbe intégrale de f en $t = 0$:

$$\mathcal{L}_f h = \sum_{i=1}^n \frac{\partial h(x)}{\partial x_i} f_i(x) \quad \text{et} \quad \mathcal{L}_f^k h = \mathcal{L}_f(\mathcal{L}_f^{k-1} h)$$

Où $k > 0$ et $\mathcal{L}_f^0 h = h$.

On notera également : $d_x h = \frac{\partial h}{\partial x}$ et $d_x \mathcal{L}_f h = \frac{\partial}{\partial x}(\mathcal{L}_f h)$.

Théorème V.1. *Un système est dit localement faiblement observable si le rang de la matrice \mathcal{O} suivante est égal au rang n dudit système :*

$$\mathcal{O} = [d_x h \quad d_x \mathcal{L}_f h \quad \dots \quad d_x \mathcal{L}_f^{n-1} h]^T$$

Dans la suite, on vérifiera donc que, pour le point de fonctionnement étudié, le système est bien localement faiblement observable. Étant donné qu'il ne s'agit que d'une étude locale, cela ne garanti pas

forcément la convergence des observateurs que l'on va construire (en cas d'initialisation trop éloignée de la réalité par exemple). Une étude d'observabilité plus poussée serait nécessaire afin de prouver cette convergence, et de déterminer exactement l'espace observable, mais cela ne sera pas traité ici.

4.1.1 Modèle et estimations

Ici, on cherche juste à reconstruire la position et la vitesse de translation du robot, en se basant sur les mesures de position de la cible, effectuées par l'œil. On considère que l'on dispose de l'attitude complète, issue du filtre complémentaire décrit dans le chapitre III. On suppose également que l'altitude de vol est connue ; ici elle est fournie par le Vicon, mais elle pourrait être obtenue par un altimètre embarqué (un baromètre, ou un télémètre par exemple).

On cherche donc à estimer le vecteur d'état $x = (\xi \ v)^T$ décrit par :

$$\begin{pmatrix} \dot{\xi} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{V}_X \\ \dot{V}_Y \\ \dot{V}_Z \end{pmatrix} = \begin{pmatrix} V_X \\ V_Y \\ V_Z \\ \frac{T}{m}(\sin \phi \sin \psi + \cos \phi \cos \psi \sin \theta) - \frac{K_{vx}}{m} V_X \\ \frac{T}{m}(\cos \phi \sin \psi \sin \theta - \sin \phi \cos \psi) - \frac{K_{vy}}{m} V_Y \\ \frac{T}{m}(\cos \phi \cos \theta) - \frac{K_{vz}}{m} V_Z \end{pmatrix} \quad (\text{V.17})$$

Et on propose de s'intéresser à deux vecteur de mesure différents, donné dans la suite par les équations (V.19) (première approche) et (V.22) (deuxième approche).

Première approche : En sommant la mesure de l'erreur rétinienne et l'estimation de l'orientation du regard $\theta_{er\phi,\theta}$ et l'attitude (ϕ , ou θ selon l'axe), on obtient la mesure de la position angulaire de la cible θ_t :

$$\bar{y} = \begin{pmatrix} \bar{\theta}_{t\theta} \\ \bar{\theta}_{t\phi} \\ \bar{Z} \end{pmatrix} = \begin{pmatrix} \hat{\theta}_{er\theta} + \bar{\epsilon}_{r\theta} + \hat{\theta} \\ \hat{\theta}_{er\phi} + \bar{\epsilon}_{r\phi} - \hat{\phi} \\ \bar{Z}_{\text{Vicon}} \end{pmatrix} \quad (\text{V.18})$$

Où $\bar{\theta}_{t\phi}$ et $\bar{\theta}_{t\theta}$ représentent les mesures (fictives) des positions angulaires de la cible dans le référentiel inertiel. Elles sont déduites de l'estimation des orientations $\hat{\theta}_{er\phi}$ et $\hat{\theta}_{er\theta}$ de l'œil (obtenues grâce au système (V.13)), des erreurs rétinienne $\bar{\epsilon}_{r\phi}$ et $\bar{\epsilon}_{r\theta}$ mesurées par l'œil et des estimations des angles de roulis et de tangages $\hat{\phi}$ et $\hat{\theta}$.

D'après le système (V.3), ces mesures sont reliées aux système d'état par la relation suivante :

$$\bar{y} = h(x) = \begin{pmatrix} \tan^{-1}\left(\frac{X}{Z}\right) \\ \tan^{-1}\left(\frac{Y}{Z}\right) \\ Z \end{pmatrix} \quad (\text{V.19})$$

Observabilité du système : Pour étudier l'observabilité du système (V.17), on calcule successivement les dérivées de Lie du vecteur de mesure \bar{y} (V.19) le long des trajectoires. On obtient alors la matrice d'observabilité suivante :

$$\mathcal{O} = \begin{pmatrix} \frac{1}{Z \cdot \left(\frac{X^2}{Z^2} + 1\right)} & 0 & \frac{-X}{Z^2 \cdot \left(\frac{X^2}{Z^2} + 1\right)} & 0 & 0 & 0 \\ 0 & \frac{1}{Z \cdot \left(\frac{Y^2}{Z^2} + 1\right)} & \frac{-Y}{Z^2 \cdot \left(\frac{Y^2}{Z^2} + 1\right)} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{Z \cdot \left(\frac{X^2}{Z^2} + 1\right)} & 0 & \frac{-X}{Z^2 \cdot \left(\frac{X^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & \frac{1}{Z \cdot \left(\frac{Y^2}{Z^2} + 1\right)} & \frac{-Y}{Z^2 \cdot \left(\frac{Y^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{-2 \cdot Kx \cdot Vx}{Z \cdot m \cdot \left(\frac{X^2}{Z^2} + 1\right)} & 0 & \frac{Kz \cdot X}{Z^2 \cdot m \cdot \left(\frac{X^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & \frac{-2 \cdot Ky \cdot Vy}{Z \cdot m \cdot \left(\frac{Y^2}{Z^2} + 1\right)} & \frac{-Kz \cdot Y}{Z^2 \cdot m \cdot \left(\frac{Y^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & 0 & -Kz/m \\ 0 & 0 & 0 & \frac{4 \cdot Kx^2 \cdot Vx^2}{Z \cdot m^2 \cdot \left(\frac{X^2}{Z^2} + 1\right)} & 0 & \frac{-Kz^2 \cdot X}{Z^2 \cdot m^2 \cdot \left(\frac{X^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & \frac{4 \cdot Ky^2 \cdot Vy^2}{Z \cdot m^2 \cdot \left(\frac{Y^2}{Z^2} + 1\right)} & \frac{-Kz^2 \cdot Y}{Z^2 \cdot m^2 \cdot \left(\frac{Y^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & 0 & \frac{Kz^2}{m^2} \\ 0 & 0 & 0 & \frac{-8 \cdot Kx^3 \cdot Vx^3}{Z \cdot m^3 \cdot \left(\frac{X^2}{Z^2} + 1\right)} & 0 & \frac{Kz^3 \cdot X}{Z^2 \cdot m^3 \cdot \left(\frac{X^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & \frac{-8 \cdot Ky^3 \cdot Vy^3}{Z \cdot m^3 \cdot \left(\frac{Y^2}{Z^2} + 1\right)} & \frac{Kz^3 \cdot Y}{Z^2 \cdot m^3 \cdot \left(\frac{Y^2}{Z^2} + 1\right)} \\ 0 & 0 & 0 & 0 & 0 & \frac{-Kz^3}{m^3} \end{pmatrix} \quad (\text{V.20})$$

La matrice d'observabilité \mathcal{O} obtenue est bien de rang 6. On en déduit donc, que le système est localement faiblement observable et que l'on peut par conséquent estimer les différents états du système¹.

1. On remarque cependant que certains points de fonctionnement sont problématiques : si Z est égale à 0, certains termes tendent vers l'infini et inversement si Z devient très grand, certains tendent vers 0. Cela traduit une perte d'observabilité et on peut donc déjà prévoir que pour une altitude très proche de 0 ou très grande, les estimations seront erronées.

Ce qui nous donne, pour la construction du filtre de Kalman étendu (EKF), la matrice jacobienne de mesure suivante :

$$H = \frac{\partial h}{\partial x} = \begin{pmatrix} \frac{1}{z\left(\frac{x^2}{z^2}+1\right)} & 0 & -\frac{x}{z^2\left(\frac{x^2}{z^2}+1\right)} & 0 & 0 & 0 \\ 0 & \frac{1}{z\left(\frac{y^2}{z^2}+1\right)} & -\frac{y}{z^2\left(\frac{y^2}{z^2}+1\right)} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (\text{V.21})$$

On observe donc que cette matrice jacobienne n'est pas triviale et nécessite d'être recalculée à chaque itération en fonction des états courants.

Deuxième approche : On décide donc d'exprimer différemment le problème, de façon à obtenir une jacobienne du vecteur de mesure plus simple. Pour cela, on reconstruit à partir des mesures disponibles, une mesure fictive des positions latérales X et Y qui sont les deux premiers états du système. On obtient donc le vecteur de mesure suivant :

$$\bar{y} = \begin{pmatrix} \bar{Z}_{\text{Vicon}} \cdot \tan(\hat{\theta}_{er_\theta} + \bar{\epsilon}_{r_\theta} + \hat{\theta}) \\ -\bar{Z}_{\text{Vicon}} \cdot \tan(-\hat{\theta}_{er_\phi} - \bar{\epsilon}_{r_\phi} + \hat{\phi}) \\ \bar{Z}_{\text{Vicon}} \end{pmatrix} = h(x) = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (\text{V.22})$$

Observabilité du système : Pour étudier l'observabilité du système (V.17), on calcule successivement les dérivées de Lie du vecteur de mesure \bar{y} (V.22) le long des trajectoires. On obtient alors la

matrice d'observabilité suivante :

$$\mathcal{O} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{-2.Kx.Vx}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-2.Ky.Vy}{m} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{-Kz}{m} \\ 0 & 0 & 0 & \frac{4.Kx^2.Vx^2}{m^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{4.Ky^2.Vy^2}{m^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{Kz^2}{m^2} \\ 0 & 0 & 0 & \frac{-8.Kx^3.Vx^3}{m^3} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-8.Ky^3.Vy^3}{m^3} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{-Kz^3}{m^3} \end{pmatrix} \quad (\text{V.23})$$

La matrice d'observabilité \mathcal{O} obtenue est bien de rang 6. On en déduit donc, que le système est localement faiblement observable et que l'on peut par conséquent estimer les différents états du système.

Ce vecteur de mesure donne également lieu à une matrice Jacobienne beaucoup plus simple :

$$H = \frac{\partial h}{\partial x} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (\text{V.24})$$

Filtre de Kalman Étendu (EKF) : On construit donc le filtre de Kalman étendu suivant permettant d'estimer les états du système en se basant sur le modèle (V.17) et sur le vecteur de mesure (V.22) :

1. Étape de prédiction :

(a) Prédiction des états :

$$\underbrace{\begin{pmatrix} \hat{X}_{k|k-1} \\ \hat{Y}_{k|k-1} \\ \hat{Z}_{k|k-1} \\ \hat{V}_{Xk|k-1} \\ \hat{V}_{Yk|k-1} \\ \hat{V}_{Zk|k-1} \end{pmatrix}}_{\hat{x}_{k|k-1}} = \underbrace{\begin{pmatrix} \hat{X}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Xk-1|k-1} \\ \hat{Y}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Yk-1|k-1} \\ \hat{Z}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Zk-1|k-1} \\ \frac{\hat{T}_k}{m} (\sin \hat{\phi}_k \sin \hat{\psi}_k + \cos \hat{\phi}_k \cos \hat{\psi}_k \sin \hat{\theta}_k) - \frac{K_{vx}}{m} V_{Xk-1|k-1} \\ \frac{\hat{T}_k}{m} (\cos \hat{\phi}_k \sin \hat{\psi}_k \sin \hat{\theta}_k - \sin \hat{\phi}_k \cos \hat{\psi}_k) - \frac{K_{vy}}{m} V_{Yk-1|k-1} \\ \frac{\hat{T}_k}{m} (\cos \hat{\phi}_k \cos \hat{\theta}_k) - \frac{K_{vz}}{m} V_{Zk-1|k-1} \end{pmatrix}}_{f(\hat{x}_{k-1|k-1}, u_k)} \quad (\text{V.25})$$

Où $\hat{x}_{k|k-1}$ représente la prédiction *a priori* de l'état futur basée sur la fonction non linéaire f mettant en jeux les états corrigés (*a posteriori*) de l'itération précédente $\hat{x}_{k-1|k-1}$, et les entrées de commandes courantes u_k . Ici on considère comme entrées $u_k = \left(\hat{T}_k \quad \hat{\phi}_k \quad \hat{\theta}_k \quad \hat{\psi}_k \right)^T$

(b) Prédiction de la matrice de covariance de l'erreur :

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k \quad (\text{V.26})$$

Où $P_{k|k-1}$ est la prédiction *a priori* de la matrice de covariance de l'erreur, F_k est la jacobienne de la fonction d'états à l'instant k , $P_{k-1|k-1}$ est la matrice de covariance de l'erreur corrigée *a posteriori* de l'itération précédente, et Q_k est la matrice de covariance correspondant au bruit du procédé (c-à-d du système). Ici la matrice Q_k a été choisie diagonale. La matrice jacobienne F_k est donnée par :

$$F_k = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}_{k-1|k-1}, u_k} = \begin{pmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 - \Delta t \cdot \frac{K_{vx}}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 - \Delta t \cdot \frac{K_{vy}}{m} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 - \Delta t \cdot \frac{K_{vz}}{m} \end{pmatrix} \quad (\text{V.27})$$

2. Étape de correction et mise à jour :

(a) Calcul de l'erreur de prédiction (terme d'innovation) :

$$\tilde{y}_k = \bar{y}_k - h(\hat{x}_{k|k-1}) = \underbrace{\begin{pmatrix} \bar{Z}_{Viconk} \cdot \tan(\hat{\theta}_{er\theta k} + \bar{\epsilon}_{r\theta k} + \hat{\theta}_k) \\ -\bar{Z}_{Viconk} \cdot \tan(-\hat{\theta}_{er\phi k} - \bar{\epsilon}_{r\phi k} + \hat{\phi}_k) \\ \bar{Z}_{Viconk} \end{pmatrix}}_{\bar{y}_k} - \underbrace{\begin{pmatrix} \hat{X}_{k|k-1} \\ \hat{Y}_{k|k-1} \\ \hat{Z}_{k|k-1} \end{pmatrix}}_{h(\hat{x}_{k|k-1})} \quad (\text{V.28})$$

Où \tilde{y}_k correspond à l'erreur de prédiction à l'instant k , et est déterminée à l'aide de la mesure courante \bar{y}_k et de la mesure estimée $h(\hat{x}_{k|k-1})$ (à partir des états prédits $\hat{x}_{k|k-1}$).

(b) Mise à jour de la covariance du terme d'innovation :

$$S_k = H_k P_{k|k-1} H_k^T + R_k \quad (\text{V.29})$$

Où S_k est la covariance à l'instant k , H_k est la matrice jacobienne du vecteur de mesure et est définie par l'équation (V.24), $P_{k|k-1}$ est la matrice de covariance prédite et R_k est la matrice de covariance du bruit de mesure qui est prise ici diagonale.

(c) Calcul du gain du filtre :

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \quad (\text{V.30})$$

Où S_k^{-1} est la matrice inverse de S_k .

(d) Correction de la prédiction des états :

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k \quad (\text{V.31})$$

(e) Correction de la covariance de l'erreur :

$$P_{k|k} = (I_{6 \times 6} - K_k H_k) P_{k|k-1} \quad (\text{V.32})$$

Où $I_{6 \times 6}$ est la matrice identité de dimension 6×6 .

4.1.2 Résultats expérimentaux

Dans un premier temps, on vérifie que l'estimation fonctionne correctement. Pour cela, on conserve les asservissements décrits dans le chapitre III mais on utilise les estimations de positions et vitesses obtenues par l'EKF au lieu de celles mesurées par le Vicon. Ici, on estime toujours l'attitude à partir du filtre complémentaire qui se base sur l'IMU et l'angle de lacet délivré par Vicon. L'asservissement d'attitude est donc réalisé par le contrôleur géométrique en quaternion, qui utilise les estimés du filtre complémentaire et l'asservissement en position est réalisé par le contrôleur non linéaire proposé dans le chapitre III, mais utilise cette fois-ci, les estimations provenant de l'EKF.

Scénario : Le robot décolle en utilisant le contrôleur de position basé sur les mesures Vicon, afin de vérifier la convergence des estimations de l'EKF. Ensuite, on bascule lors du vol sur le contrôleur de position basé sur les estimations provenant de l'EKF, puis on déplace la cible afin d'évaluer les capacités de poursuite de X4-MaG. Puis le robot effectue un atterrissage automatique toujours grâce à son œil.

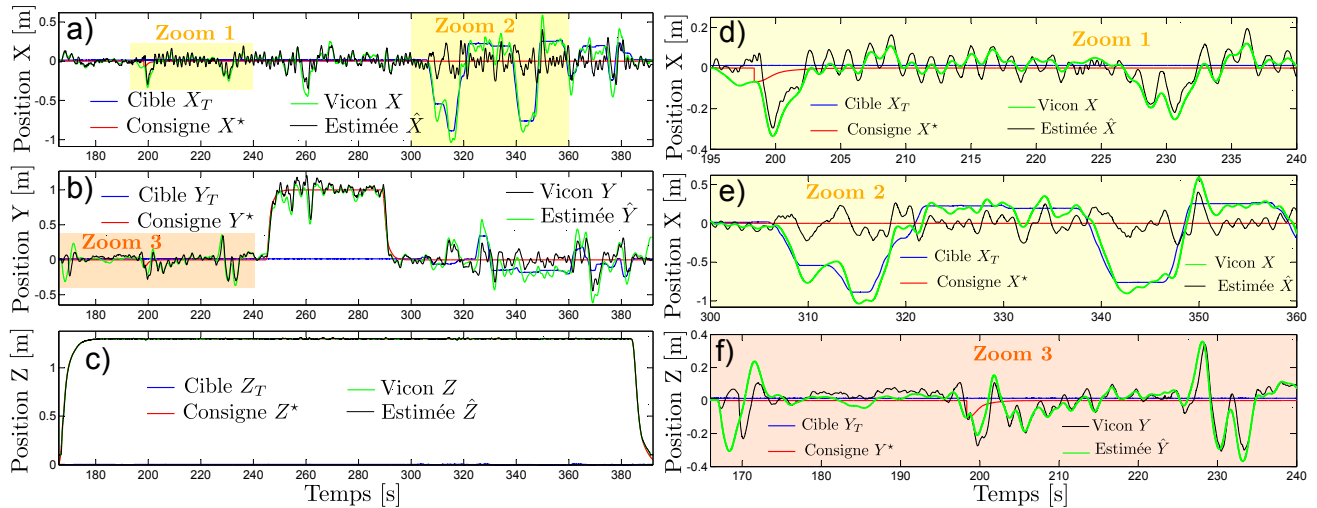


FIGURE V.6 – Asservissement en position basé vision. En rouge la consigne de position, en bleu la position de la cible, en vert la position mesurée par la Vicon (réalité terrain) et en noir la position estimée sur la base de la vision. a) Évolution des positions sur l'axe X , on observe bien à partir de $t = 300s$ que la consigne de position relative nulle entraîne le robot à poursuivre la cible. En maintenant sa position estimée proche de 0 (courbe noire), le robot (courbe verte) poursuit naturellement la cible (courbe bleue). b) Évolution de la position sur l'axe Y . c) Évolution de la position sur l'axe Z , le robot décolle, maintient son altitude tout au long du vol puis atterri. d) Zoom sur l'asservissement de position selon X (figure a) de $t = 195s$ à $t = 240s$. On observe bien que les zones mortes des servomoteurs (liées aux frottements) entraînent une "surestimation" des amplitudes de mouvements. e) Zoom sur l'asservissement de position X (figure a) de $t = 300s$ à $t = 360s$, lors de la poursuite de la cible. f) Zoom sur l'asservissement de position selon Y (figure b) de $t = 165s$ à $t = 240s$.

Résultats La figure V.6 montre l'évolution des positions du robot estimée par l'EKF et mesurée par le Vicon. Le robot décolle grâce au contrôleur se basant sur les mesures de position du Vicon. Puis à $t = 198s$, on bascule sur le contrôleur utilisant les estimations obtenues grâce à l'EKF. Cette transition génère un "sursaut" qui est bien visible sur les figures V.6d et f, car le robot réalise un écart de près de 50cm, s'expliquant par l'initialisation du nouveau contrôleur. En effet, les intégrateurs du nouveau contrôleur ne se trouvent pas dans le même état que le contrôleur précédent, ce qui génère un saut de commande et donc un déplacement vif du robot lors de la transition. Le nouveau contrôleur stabilise alors rapidement cela et ramène le robot autour de sa référence en moins de 3 secondes. Une fois le deuxième contrôleur activé, on remarque que des oscillations autour de la position de référence apparaissent. Ceci s'explique principalement par deux phénomènes que nous décrirons dans la section 4.1.3. La figure V.6d montre ces oscillations sur \hat{X} lors d'une phase de vol stationnaire et on voit que ce phénomène est moins marqué sur les estimations de \hat{Y} , comme le montre la figure V.6f. Durant cet essai, l'altitude est asservie en utilisant la mesure du Vicon \bar{Z}_{Vicon} , ce qui explique les très bonnes performances de poursuite sur l'axe z (voir figure V.6c). On observe également que le robot est capable, grâce à son œil, de se positionner précisément au voisinage de la cible. Par exemple, le robot se déplace

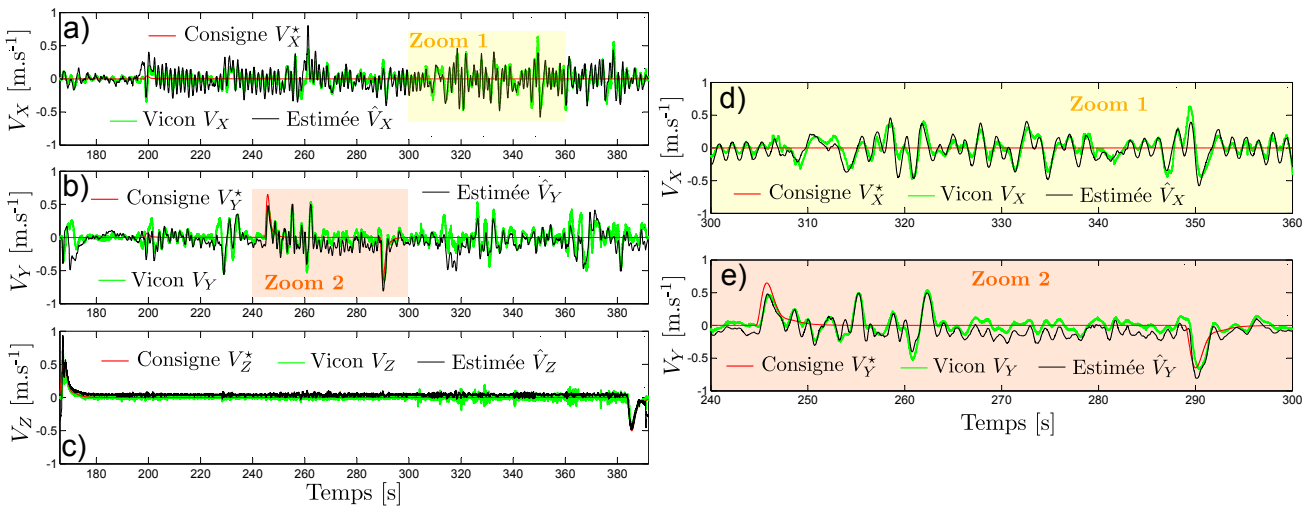


FIGURE V.7 – En rouge les consignes de vitesses, en noir les estimations obtenues à partir des mesures de l’œil et en vert les mesures de vitesses faites par le Vicon. a) De même que pour les estimations de positions, l’estimation de vitesse a tendance à surestimer les vitesses réelles. b) L’estimation est de meilleure qualité sur l’axe Y . c) L’estimation de vitesse selon Z est presque confondue avec la mesure du Vicon, étant donné la forte pondération de la mesure de l’altitude Z dans l’EKF. d) Zoom correspondant au temps $t = 300$ s à $t = 360$ s de l’estimation de vitesse \hat{V}_x . e) Zoom correspondant au temps $t = 300$ s à $t = 360$ s de l’estimation de vitesse \hat{V}_y .

à 1m de la cible entre $t = 245$ s et $t = 290$ s (voir figure V.6), et estime sa position avec une erreur maximale de 30 cm (atteinte lors de l’atterrissage) et maintient sa position autour de la référence sans erreur statique. Ensuite, on déplace manuellement la cible selon les directions X et Y, entre $t = 300$ s et $t = 380$ s. Puis, le robot effectue un atterrissage automatique sur la cible, en utilisant toujours uniquement son œil pour se positionner. Le robot réussit à atterrir sur la cible avec une précision de 1cm sur l’axe X et 30cm selon l’axe Y. On observe que la position reste asservie proche de 0 par rapport à la cible, jusqu’à ce que le robot passe en dessous de 15cm d’altitude par rapport à la cible. En effet, en dessous de cette altitude, la cible s’apparente de plus en plus à un front de contraste (vis à vis du champ de vision d’une paire), ce qui induit une perte de vue de la cible (la cible se trouve encore dans le champ de vision de l’œil, mais sa nature ayant changée de barre à front, la mesure n’est plus exploitable). Ceci est bien visible à la fin de l’enregistrement figure V.6b car la courbe noire (estimation de position basée vision) n’évolue plus et ne suit plus les variations de la position réelle (courbe verte). La cible ayant été "perdue de vue", le ZSL force la mesure de position angulaire de la cible à 0° , ce qui a pour effet d’annuler la poursuite de la cible, et donc la réorientation du regard. Sans cette réorientation du regard, la mesure de position déduite du regard n’évolue plus et fige donc l’estimation de position dans l’état de la dernière mesure de position connue.

Les figures V.7a-c montrent l’évolution des estimations de vitesses pour le même scénario. Les figures V.7d-e correspondent à un zoom des estimations de vitesses \hat{V}_x et \hat{V}_y . La figure V.8a montre l’évolution des angles de roulis, tangage et lacet durant ce même scénario, alors que les figures V.8b-c reportent

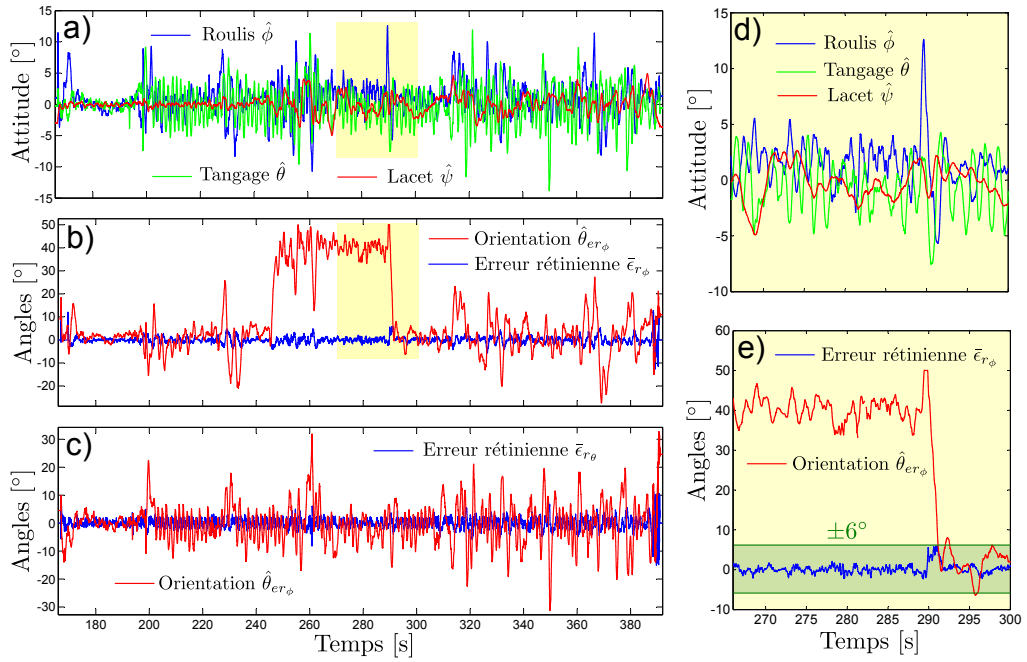


FIGURE V.8 – a) En bleu, l'angle de roulis, en vert le tangage et en rouge le lacet estimé par le filtre complémentaire. b) Evolution de l'orientation du regard selon l'angle de roulis en rouge et de l'erreur rétinienne en bleu. Entre $t = 250$ s et $t = 300$ s, le robot se place à 1 m de la cible d'où la réorientation du regard autour de 37° ($\theta_{er_\phi} = \tan^{-1}\left(\frac{Y}{Z}\right)$ où $Z = 1.3$ m et $Y = 1$ m). c) Évolution de l'orientation de l'œil selon l'angle de tangage en rouge et l'erreur rétinienne correspondante en bleu. d) Zoom sur l'attitude lors d'un déplacement latéral d'un mètre (correspondant à l'encadré jaune des figures a et b). e) Zoom sur l'orientation de l'œil et de l'erreur rétinienne lors d'un déplacement latéral.

l'erreur rétinienne mesurée ainsi que l'orientation de l'œil selon ses deux degrés de liberté. Les figures V.8d-e montre l'efficacité des réflexes Vestibulo-Oculaire et de fixation visuelle lors d'une translation d'un mètre du robot. Cette translation est amorcée par un rapide changement de l'angle de roulis, atteignant plus de 13° , qui est compensé par le réflexe VOR, ce qui n'engendre qu'une erreur rétinienne maximale de 6° (courbe bleue sur la figure V.8e). Puis, tout au long du déplacement, le VFR permet de poursuivre la cible durant toute la translation, d'ajuster automatiquement l'orientation de regard (courbe rouge sur la figure V.8e) et donc d'assurer une bonne estimation de la position et de la vitesse (comme le montrent les figures V.6b et V.7e).

4.1.3 Discussions

Durant cet essai, on observe que le robot oscille assez fortement autour de la position de référence. Cela est volontaire et ce comportement résulte d'une véritable démarche. En effet, les contrôleurs d'attitude et de position sont exactement les mêmes que ceux présentés dans le chapitre III, à la différence près que la masse et la matrice d'inertie ont été adaptées. Cependant, en ajoutant un estimateur de position et de vitesses dans la boucle, les performances de ces contrôleurs se voient dégradées. Pour

a)		$\hat{\theta}_{er_\phi}$	$\hat{\theta}_{er_\theta}$	$\bar{\epsilon}_{r_\phi}$	$\bar{\epsilon}_{r_\theta}$
Moyenne	$E(\cdot)$	9.54°	-0.29°	-0.0027°	-0.0084°
Écart Type	$\sigma(\cdot)$	17.14°	7.06°	1.31°	1.65°
Min	$\min(\cdot)$	-27.41°	-31.21°	-5.61°	-8.12°
Max	$\max(\cdot)$	50.10°	31.88°	5.90°	8.63°
b)		\tilde{X}	\tilde{Y}	\tilde{V}_X	\tilde{V}_Y
Écart Type	$\sigma(\cdot)$	4.41 cm	5.89 cm	9.78 cm · s ⁻¹	9.97 cm · s ⁻¹
Min	$\min(\cdot)$	-23.27 cm	-30.78 cm	-37.87 cm · s ⁻¹	-33.69 cm · s ⁻¹
Max	$\max(\cdot)$	12.00 cm	16.28 cm	40.09 cm · s ⁻¹	8.63 cm · s ⁻¹

TABLE V.1 – a) Moyenne, écart type, valeur minimale et maximale des erreurs rétinienne mesurées ainsi que de l'orientation du regard lors de l'essai (hors décollage et atterrissage étant donné que durant cette période, l'œil "ne voit pas" la cible). b) Erreurs d'estimations de positions et de vitesses durant l'essai (hors décollage et atterrissage). Où les erreurs d'estimation de positions sont obtenues comme suit $\tilde{X} = \bar{X} - (\hat{X} + \bar{X}_T)$ et $\tilde{Y} = \bar{Y} - (\hat{Y} + \bar{Y}_T)$ et les erreurs d'estimation de vitesses sont définies par $\tilde{V}_X = \bar{V}_X - (\hat{V}_X + \bar{V}_{X_T})$ et $\tilde{V}_Y = \bar{V}_Y - (\hat{V}_Y + \bar{V}_{Y_T})$.

cet essai, j'ai volontairement conservé les mêmes valeurs de gains pour les contrôleurs pour plusieurs raisons :

- Les oscillations sur l'attitude du robot permettent de mieux évaluer les performances du découplage corps-vision. En effet, on observe que malgré des oscillations crête à crête de l'ordre de 5° à 10° sur le tangage, la cible n'est jamais perdue de vue, et ce, même lors de translations rapides.
- Cela permet également de tester l'estimateur pour des dynamiques élevées.

Cet essai a donc permis de valider le bon fonctionnement de l'asservissement visuel en vol. On remarque également que les vibrations engendrées par la rotation des rotors ne perturbent pas le fonctionnement de l'œil, qui rappelons le, est un système reposant sur une vibration active. Une fois la cible dans le champ visuel de l'œil, celui-ci réussit à la maintenir dans son champ de vision très efficacement, grâce à ces réflexes. En effet, comme le présente le tableau V.1, on observe des erreurs rétinienne moyennes de $\bar{\epsilon}_{r_\phi}$ et $\bar{\epsilon}_{r_\theta}$ de quelques milli-degrés, des écarts types de seulement $\sigma(\bar{\epsilon}_{r_\phi} = 1.31)^\circ$ et $\sigma(\bar{\epsilon}_{r_\theta} = 1.65)^\circ$. De plus, on remarque que lorsque le robot effectue une translation d'un mètre en latéral, cela induit une rotation du servomoteur de plus de 50° sur l'axe de roulis en moins de 0.6s. Or, pendant cette rotation rapide (plus de $80^\circ \cdot \text{s}^{-1}$), le réflexe Vestibulo-Occulaire (VOR) permet de maintenir l'erreur rétinienne en dessous de 6°. On résume les caractéristiques de la poursuite visuelle dans le tableau V.1.

De même, on observe que les estimations de positions et de vitesses sont satisfaisantes, étant donné que l'erreur d'estimation en position présente une précision inférieure à 6 cm, et l'erreur d'estimation sur la vitesse reste inférieure à $10 \text{ cm} \cdot \text{s}^{-1}$ (en terme d'écart type). Cependant on observe qu'en vol stationnaire, pour une position réelle constante, de petites oscillations apparaissent sur l'estimation de position. Cela est principalement dû à un phénomène : les frottements secs liés au système de découplage

de l'œil, et la précision de l'asservissement de position des servomoteurs,

En effet, le réflexe de fixation visuelle tend à annuler l'erreur rétinienne, en augmentant progressivement la consigne d'orientation des servomoteurs (via le terme intégral du contrôleur). Mais, du fait des frottements secs, les servomoteurs réagissent par à-coup, provoquant un dépassement d'un côté ou de l'autre de la consigne. Or, la position réelle des servomoteurs n'est pas mesurée, elle est seulement estimée à partir des consignes $\theta_{er\phi,\theta}^*$. Les estimations de l'orientation des servomoteurs $\hat{\theta}_{er\phi,\theta}$ peuvent alors être biaisées de $DZ = \pm 0.5^\circ$, du fait de la zone morte. En supposant que l'estimation d'attitude, la mesure de l'erreur rétinienne et la mesure d'altitude sont parfaites (sans biais : $\hat{\phi} = \hat{\theta}$, $\bar{\epsilon}_{r\phi} = \bar{\epsilon}_{r\theta}$, $\bar{Z} = Z$), on peut déduire les bornes de l'erreur en position qui en résulte :

$$\begin{aligned} \theta_{er\phi} - DZ &\leq \hat{\theta}_{er\phi} \leq \theta_{er\phi} + DZ \\ \Leftrightarrow Z \tan(\phi + \epsilon_{r\phi} + \theta_{er\phi} - DZ) &\leq \bar{Z} \tan(\hat{\phi} - \bar{\epsilon}_{r\phi} - \hat{\theta}_{er\phi}) \leq Z \tan(\phi - \epsilon_{r\phi} - \theta_{er\phi} + DZ) \end{aligned}$$

4.2 Stabilisation partielle d'attitude sans accéléromètre, décollage et atterrissage automatique

4.2.1 Modèle et estimations

Ici on cherche à estimer en plus le roulis et le tangage. L'estimation de l'angle de lacet nécessite une attention particulière, et sera traitée dans la section suivante (voir section 4.3).

On utilise toujours l'altitude et la mesure de lacet fournies par le Vicon. Le reste de l'attitude, c'est à dire le roulis et le tangage ainsi que les vitesses de rotation correspondantes sont, contrairement au cas précédent, estimée par l'EKF. Ce dernier permet donc d'estimer l'attitude, les vitesses de rotation, la position et la vitesse du robot à partir de l'altitude, de la poussée estimée et des mesures de l'œil. La figure V.9 résume les interactions des différents estimateurs et contrôleurs. Contrairement à la version précédente, les mesures des accéléromètres ne sont plus nécessaires pour estimer l'altitude ; le recalage des dérives des gyromètres est réalisé grâce à l'orientation du regard.

On s'intéresse donc à un vecteur d'attitude réduit noté $\Theta_{\text{red}} = \begin{pmatrix} \phi & \theta \end{pmatrix}^T$ et à son vecteur de biais réduit $b_{\text{red}} = \begin{pmatrix} b_\phi & b_\theta \end{pmatrix}^T$ correspondant aux mesures des gyromètres $\bar{\Omega}_{\text{red}} = \begin{pmatrix} \bar{\Omega}_\phi & \bar{\Omega}_\theta \end{pmatrix}^T$. On cherche

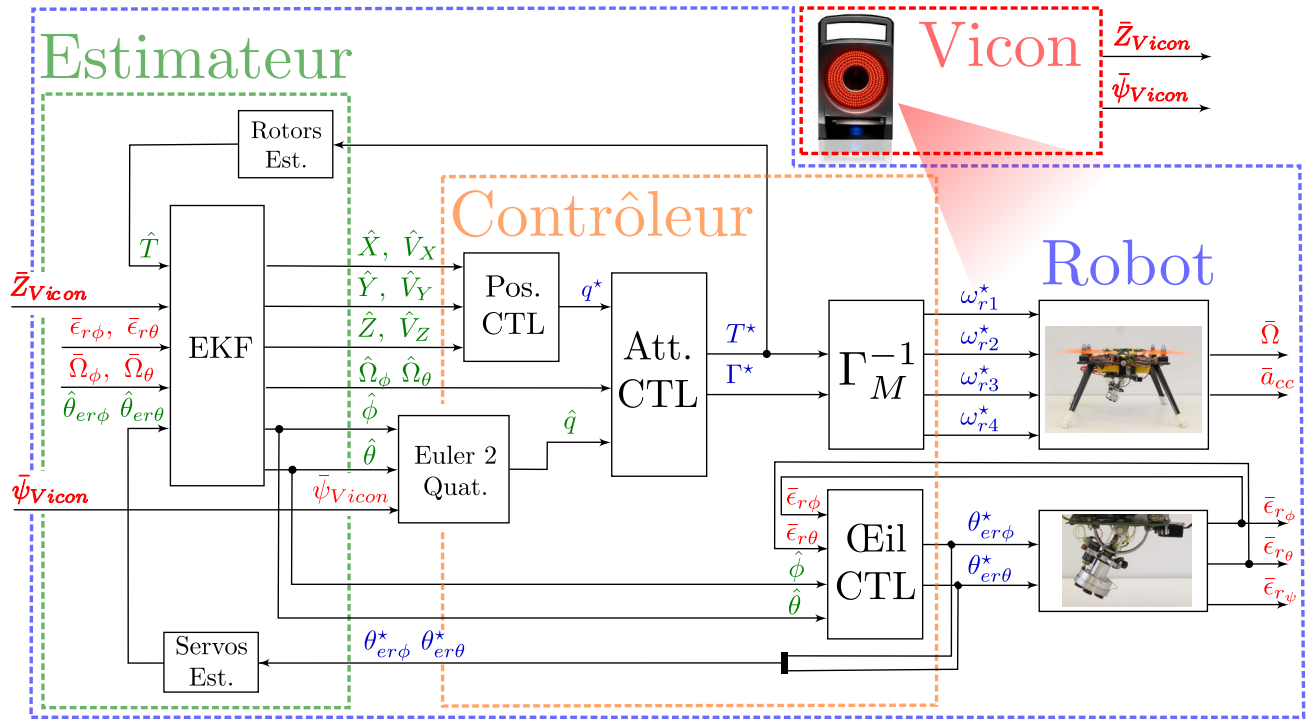


FIGURE V.9 – Diagramme en bloc du système complet. Le bloc **Servos Est.** estime la position réelle des servomoteurs à partir de leur modèle dynamique (V.14) ; Le bloc **Rotors Est.** estime la poussée générée à partir du modèle dynamique des rotors (V.16). Le bloc **EKF** correspond au filtre de Kalman étendu estimant la position et la vitesse du robot, le roulis et le tangage ainsi que les vitesses de rotation correspondantes (V.35) à (V.44). **Pos. CTL** est le contrôleur de position (III.71), **Att. CTL** le contrôleur d'attitude (III.70) et **Ëil CTL** est le contrôleur de regard (V.10). Enfin, **Quat. 2 Euler** permet d'extraire du quaternion d'attitude les angles de roulis et de tangage et Γ_M^{-1} permet de passer de la consigne de couple et de poussée aux consignes rotors. Ici, le système **Vicon** permet de capturer l'angle de lacet ainsi que l'altitude du robot.

donc à estimer le vecteur d'état $x = \left(\xi \quad v \quad \Theta_{red} \quad b_{red} \right)^T$ décrit par le modèle suivant :

$$\begin{pmatrix} \xi \\ v \\ \Theta_{red} \\ b_{red} \end{pmatrix} = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{V}_X \\ \dot{V}_Y \\ \dot{V}_Z \\ \dot{\phi} \\ \dot{\theta} \\ \dot{b}_\phi \\ \dot{b}_\theta \end{pmatrix} = \begin{pmatrix} V_X \\ V_Y \\ V_Z \\ \frac{T}{m} (\sin \phi \sin \psi + \cos \phi \cos \psi \sin \theta) - \frac{K_{vx}}{m} V_X \\ \frac{T}{m} (\cos \phi \sin \psi \sin \theta - \sin \phi \cos \psi) - \frac{K_{vy}}{m} V_Y \\ \frac{T}{m} (\cos \phi \cos \theta) - \frac{K_{vz}}{m} V_Z \\ \bar{\Omega}_\phi - b_\phi \\ \bar{\Omega}_\theta - b_\theta \\ 0 \\ 0 \end{pmatrix} \quad (V.33)$$

On dispose du vecteur de mesure décrit par l'équation suivante :

$$\bar{y} = \begin{pmatrix} \bar{Z}_{\text{Vicon}} \cdot \tan(\hat{\theta}_{er\theta} + \bar{\epsilon}_{r\theta} + \hat{\theta}) \\ -\bar{Z}_{\text{Vicon}} \cdot \tan(-\hat{\theta}_{er\phi} - \bar{\epsilon}_{r\phi} + \hat{\phi}) \\ \bar{Z}_{\text{Vicon}} \\ \bar{\Omega}_{\text{red}} \end{pmatrix} = h(x) = \begin{pmatrix} X \\ Y \\ Z \\ \Omega_{\text{red}} + b_{\text{red}} \end{pmatrix} \quad (\text{V.34})$$

Observabilité du système : Pour des raisons de taille, on ne détaille pas ici la matrice d'observabilité du système, mais on vérifie que cette matrice est bien de plein rang. La matrice d'observabilité \mathcal{O} obtenue pour le système (V.33), avec le vecteur de mesure \bar{y} (V.34), est bien de rang 10. On en déduit donc, que le système est localement faiblement observable et que l'on peut par conséquent estimer les différents états du système.

Filtre de Kalman Étendu (EKF) : On construit donc le filtre de Kalman étendu suivant permettant d'estimer les états du système :

1. Étape de prédiction :

(a) Prédiction des états :

$$\underbrace{\begin{pmatrix} \hat{X}_{k|k-1} \\ \hat{Y}_{k|k-1} \\ \hat{Z}_{k|k-1} \\ \hat{V}_{X_{k|k-1}} \\ \hat{V}_{Y_{k|k-1}} \\ \hat{V}_{Z_{k|k-1}} \\ \hat{\phi}_{k|k-1} \\ \hat{\theta}_{k|k-1} \\ \hat{b}_{\phi_{k|k-1}} \\ \hat{b}_{\theta_{k|k-1}} \end{pmatrix}}_{\hat{x}_{k|k-1}} = \underbrace{\begin{pmatrix} \hat{X}_{k-1|k-1} + \Delta t \cdot \hat{V}_{X_{k-1|k-1}} \\ \hat{Y}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Y_{k-1|k-1}} \\ \hat{Z}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Z_{k-1|k-1}} \\ \frac{\hat{T}_k}{m} (\sin \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_k + \cos \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_k \sin \hat{\theta}_{k-1|k-1}) - \frac{K_{vx}}{m} V_{X_{k-1|k-1}} \\ \frac{\hat{T}_k}{m} (\cos \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_k \sin \hat{\theta}_{k-1|k-1} - \sin \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_k) - \frac{K_{vy}}{m} V_{Y_{k-1|k-1}} \\ \frac{\hat{T}_k}{m} (\cos \hat{\phi}_{k-1|k-1} \cos \hat{\theta}_{k-1|k-1}) - \frac{K_{vz}}{m} V_{Z_{k-1|k-1}} \\ \hat{\phi}_{k-1|k-1} + \Delta t \cdot (\bar{\Omega}_\phi - \hat{b}_{\phi_{k-1|k-1}}) \\ \hat{\theta}_{k-1|k-1} + \Delta t \cdot (\bar{\Omega}_\theta - \hat{b}_{\theta_{k-1|k-1}}) \\ \hat{b}_{\phi_{k-1|k-1}} \\ \hat{b}_{\theta_{k-1|k-1}} \end{pmatrix}}_{f(\hat{x}_{k-1|k-1}, u_k)} \quad (\text{V.35})$$

(b) Prédiction de la matrice de covariance de l'erreur :

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k \quad (\text{V.36})$$

Avec Q_k une matrice diagonale et F_k la jacobienne donnée par :

$$F_k = \frac{\partial f}{\partial x} \Big|_{\hat{x}_{k-1|k-1}, u_k} = \begin{pmatrix} I_{3 \times 3} & \Delta t \cdot I_{3 \times 3} & 0_{3 \times 2} & 0_{3 \times 2} \\ 0_{3 \times 3} & F_{a3 \times 3} & F_{b3 \times 2} & 0_{3 \times 2} \\ 0_{2 \times 3} & 0_{2 \times 3} & I_{2 \times 2} & -\Delta t \cdot I_{2 \times 2} \\ 0_{2 \times 3} & 0_{2 \times 3} & 0_{2 \times 2} & I_{2 \times 2} \end{pmatrix} \quad (\text{V.37})$$

Où :

$$F_{a3 \times 3} = \begin{pmatrix} 1 - \Delta t \cdot \frac{K_{vx}}{m} & 0 & 0 \\ 0 & 1 - \Delta t \cdot \frac{K_{vy}}{m} & 0 \\ 0 & 0 & 1 - \Delta t \cdot \frac{K_{vz}}{m} \end{pmatrix} \quad (\text{V.38})$$

Et :

$$F_{b3 \times 2} = \Delta t \cdot \frac{\hat{T}_k}{m} \cdot \begin{pmatrix} \cos \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_k - \cos \hat{\psi}_k \sin \hat{\phi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} & \cos \hat{\phi}_{k-1|k-1} \cos \hat{\theta}_{k-1|k-1} \cos \hat{\phi}_k \\ -\cos \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_k - \sin \hat{\phi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} \sin \hat{\psi}_k & \cos \hat{\phi}_{k-1|k-1} \cos \hat{\theta}_{k-1|k-1} \sin \hat{\phi}_k \\ 0 & 0 \end{pmatrix} \quad (\text{V.39})$$

2. Étape de correction et mise à jour :

(a) Calcul de l'erreur de prédiction (terme d'innovation) :

$$\tilde{y}_k = \bar{y}_k - h(\hat{x}_{k|k-1}) = \underbrace{\begin{pmatrix} \bar{Z}_{Viconk} \cdot \tan(\hat{\theta}_{er\theta k} + \bar{e}_{r\theta k} + \hat{\theta}_k) \\ -\bar{Z}_{Viconk} \cdot \tan(-\hat{\theta}_{er\phi k} - \bar{e}_{r\phi k} + \hat{\phi}_k) \\ \bar{Z}_{Viconk} \end{pmatrix}}_{\tilde{y}_k} - \underbrace{\begin{pmatrix} \hat{X}_{k|k-1} \\ \hat{Y}_{k|k-1} \\ \hat{Z}_{k|k-1} \end{pmatrix}}_{h(\hat{x}_{k|k-1})} \quad (\text{V.40})$$

(b) Mise à jour de la covariance du terme d'innovation :

$$S_k = H_k P_{k|k-1} H_k^T + R_k \quad (\text{V.41})$$

(c) Calcul du gain du filtre :

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \quad (\text{V.42})$$

(d) Correction de la prédiction des états :

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k \quad (\text{V.43})$$

(e) Correction de la covariance de l'erreur :

$$P_{k|k} = (I_{10 \times 10} - K_k H_k) P_{k|k-1} \quad (\text{V.44})$$

4.2.2 Résultats expérimentaux

Scénario : Ici le robot décolle grâce au contrôleur *Vicon*, puis on bascule sur le contrôleur *vision* pour faire un vol stationnaire au dessus d'une cible se déplaçant lentement. Ce contrôleur *vision* se base sur les estimations délivrées par le filtre de Kalman étendu (EKF) donné par l'équation V.35, et qui estime les positions et vitesses du robot ainsi que l'angle de roulis, de tangage et les biais des gyromètres correspondants. Une fois encore, le but étant de valider le fonctionnement de l'estimateur, on conserve les gains des contrôleurs précédents, même si ceux-ci donnent lieu à des oscillations d'attitude plus ou moins importantes.

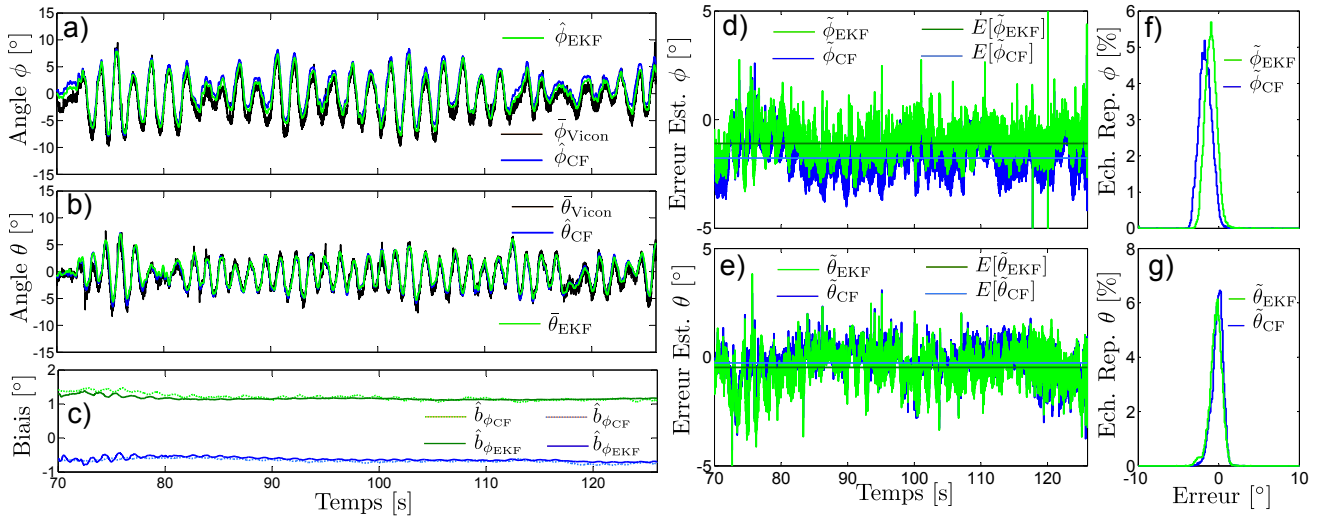


FIGURE V.10 – En vert l'estimation découlant de l'EKF fusionnant l'altitude, les informations de l'orientation du regard et le modèle dynamique, en bleu l'estimation faite par le filtre complémentaire (CF) se basant sur les accéléromètres et gyromètres, et en noir la mesure réelle effectuée par le Vicon. a) Comparaison des estimations de l'angle de roulis faite par l'EKF et le filtre complémentaire avec la réalité terrain (Vicon). b) Comparaison des estimations de l'angle de tangage faite par l'EKF et le filtre complémentaire avec la réalité terrain (Vicon). c) Estimations des biais des gyromètres faites par l'EKF et le filtre complémentaire relatif à l'axe de roulis (EKF en vert foncé et CF en pointillé vert clair, et relatif à l'axe de tangage (EKF en bleu foncé et pointillé bleu clair pour le CF). d) Erreurs d'estimation de l'angle de roulis pour l'EKF et le CF ainsi que l'erreur moyenne. e) Erreurs d'estimation de l'angle de tangage pour l'EKF et le CF ainsi que l'erreur moyenne. f) Répartition de l'erreur d'estimation pour les deux estimateurs de roulis. g) Répartitions de l'erreur d'estimation pour les deux estimateurs de tangage.

Résultats : On compare donc les estimations d'attitude et de biais des gyromètres réalisées par l'EKF à celles du filtre complémentaire présenté dans le chapitre III. La figure V.10 présente donc les résultats liés à l'estimation d'attitude alors que la figure V.11 montre les résultats liés à l'estimation de position et à la poursuite de cible.

On remarque que le filtre de Kalman estime précisément l'attitude et fournit des performances très similaires à celles du filtre complémentaire. En effet, les courbes d'estimation de l'EKF et du CF se superposent sur les figures V.10a-b et présentent des distributions d'erreurs similaires. On remarque également que la dispersion de l'erreur d'estimation pour les deux filtres est sensiblement identique (voir figure V.10c-d) ce qui s'explique par la mesure du Vicon qui est plus bruitée. On observe également sur la figure V.10c que les biais des gyromètres sont correctement estimés par l'EKF car ils convergent vers les mêmes valeurs que ceux estimés par le filtre complémentaire. De même, on observe que l'estimation de position fonctionne assez bien malgré le problème de "surestimation" des petites oscillations autour d'une position. L'estimation est cependant de suffisamment bonne qualité puisqu'elle permet au robot de se maintenir lors de son vol stationnaire, dans une enveloppe de ± 10 cm autour de la cible. De plus comme on peut le voir sur les figures V.11a-b, le robot est capable d'estimer son attitude et sa position et de poursuivre les mouvements de la cible.

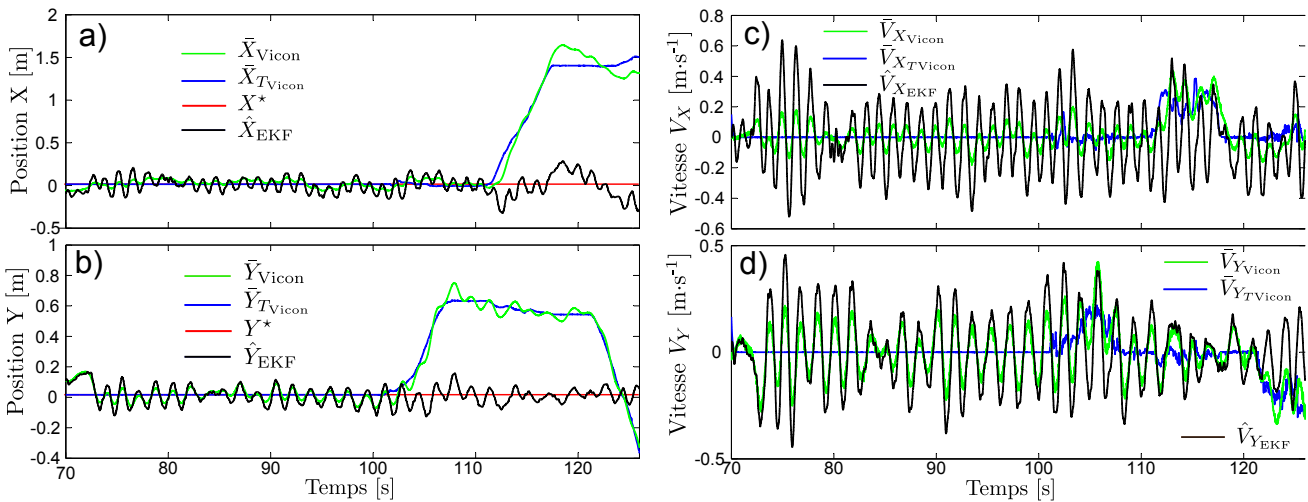


FIGURE V.11 – a et b) Estimation de la position \hat{X}_{EKF} et \hat{Y}_{EKF} en noir, en vert la position réelle du robot \bar{X}_{Vicon} et \bar{Y}_{Vicon} , en bleu la position de la cible \bar{X}_{TVicon} et \bar{Y}_{TVicon} et en rouge la consigne de position X^* et Y^* (position relative par rapport à la cible). c et d) En vert les vitesses de translation estimées \hat{V}_{XEKF} et \hat{V}_{YEKF} . En bleu les vitesses mesurées $\bar{V}_{XTVicon}$ et $\bar{V}_{YTVicon}$ de la cible.

4.2.3 Discussions

Au final, on décrit dans le tableau V.2 les erreurs d'estimations pour les différents états du système. L'EKF présente donc une précision inférieure à 0.7° en terme d'écart type sur l'estimation d'attitude.

Avec une erreur maximale de 3.44° qui correspond à l'erreur initiale, lorsque les biais des gyromètres n'avaient pas encore convergé. De même, les erreurs d'estimations de positions maximales restent inférieures à 16 cm et ont un écart types de 5.47 cm selon X , qui est le plus sensible au phénomène de frottements secs du système de vision. Dans le cas d'angle de lacet nul comme ici, les estimations de positions sur l'axe X sont liées à l'angle de roulis de l'œil, qui présente le plus de frottements secs impliquant des oscillations sur l'estimation de X et réduisant donc sa précision (voir explications données en section 4.1.3). En revanche, sur l'axe Y , les estimations dépendent de l'angle de tangage de l'œil qui présente moins de frottements, ce qui donne lieu à des estimations plus précises (écart type de 2.44 cm). De même, c'est sur l'axe X que les estimations de vitesses présentent le plus gros écart type, alors que la vitesse est estimée avec une précision inférieure à $9 \text{ cm} \cdot \text{s}^{-1}$ sur l'axe Y .

		$\tilde{\phi}$ [°]	$\tilde{\theta}$ [°]	\tilde{X} [cm]	\tilde{Y} [cm]	\tilde{V}_X [cm · s ⁻¹]	\tilde{V}_Y [cm · s ⁻¹]
Écart Type	$\sigma(\cdot)$	0.59	0.63	5.47	2.44	16.60	8.26
Min	$\min(\cdot)$	-2.69	-3.44	-15.4	-9.05	-48.13	-26.49
Max	$\max(\cdot)$	2.11	1.91	11.85	6.07	40.70	22.78

TABLE V.2 – Écart type, minimum et maximum des erreurs d'estimations des angles de roulis et tangage ainsi que des vitesses et positions. Les erreurs d'estimation des angles sont définies par $\tilde{\phi} = \bar{\phi} - \hat{\phi}$ et $\tilde{\theta} = \bar{\theta} - \hat{\theta}$. Les erreurs d'estimation de positions sont obtenues comme suit $\tilde{X} = \bar{X} - (\hat{X} + \bar{X}_T)$ et $\tilde{Y} = \bar{Y} - (\hat{Y} + \bar{Y}_T)$ et les erreurs d'estimation de vitesses sont définies par $\tilde{V}_X = \bar{V}_X - (\hat{V}_X + \bar{V}_{X_T})$ et $\tilde{V}_Y = \bar{V}_Y - (\hat{V}_Y + \bar{V}_{Y_T})$.

4.3 Stabilisation complète d'attitude bio-inspirée et asservissement en position

4.3.1 Le cas particulier de l'estimation du lacet

Comme précisé précédemment, le cas de l'estimation du lacet est un peu particulière. En effet, la mesure du lacet peut être obtenue directement de l'œil, comme décrit par le chapitre IV. On rappelle que cette mesure est alors obtenue en fusionnant l'erreur rétinienne des quatre yeux de la façon suivante :

$$\bar{\psi} = \bar{\epsilon}_\psi = f(\bar{\epsilon}_{r_i}) = \frac{1}{2} \left[\tan^{-1} \left(\frac{D \cdot \tan \left(\frac{\bar{\epsilon}_{r_1} - \bar{\epsilon}_{r_3}}{2} \right)}{r_c} \right) + \tan^{-1} \left(\frac{D \cdot \tan \left(\frac{\bar{\epsilon}_{r_2} - \bar{\epsilon}_{r_4}}{2} \right)}{r_c} \right) \right] \quad (\text{V.45})$$

Où r_c est le rayon du cercle sur lequel sont disposés les yeux et D est la distance entre l'œil et la cible (projetée sur l'axe Oz).

La mesure de l'angle $\bar{\psi}$ est alors obtenue en estimant la distance d , séparant l'intersection entre le plan de la cible et l'axe principal d'un œil. Puis, on déduit de cette distance et de la géométrie de l'œil (son rayon r_c), l'angle de lacet correspondant. Ce calcul est, en réalité, impacté par des imperfections

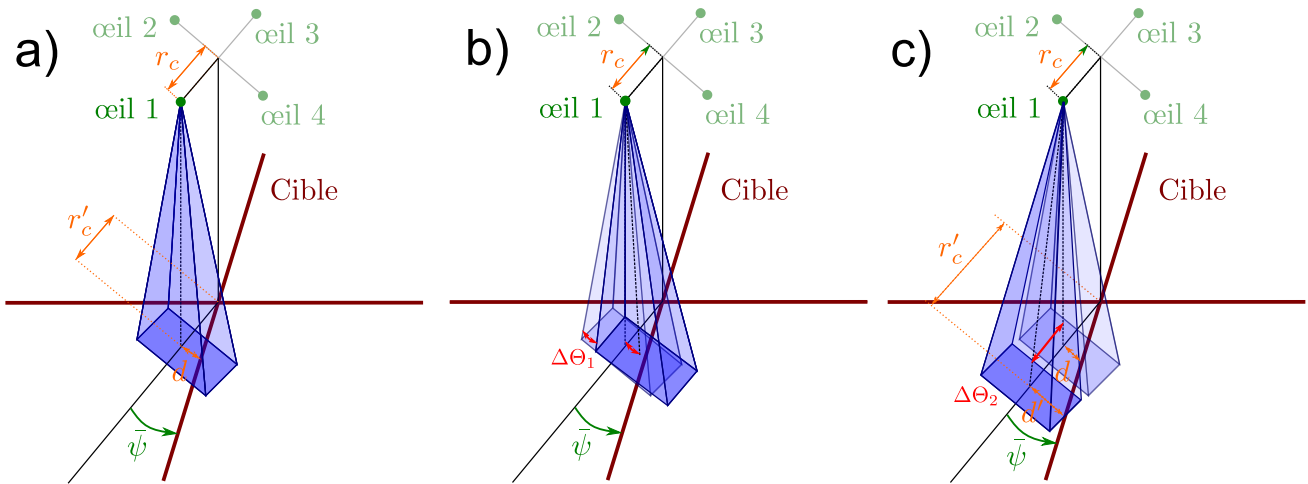


FIGURE V.12 – a) Schéma de principe de la mesure de l'angle de lacet $\bar{\psi}$. On présente ici la projection de l'axe optique principal d'un œil sur le plan de la cible, permettant de déterminer l'angle $\bar{\psi}$. Dans le cas idéal $r_c = r'_c$. b) Impact d'un défaut d'alignement latéral $\Delta\Theta_1$ de l'axe principal d'un œil. Ce phénomène n'est en réalité pas un problème, car la phase de calibration permet de déterminer ce défaut. c) Impact d'un défaut d'alignement longitudinal $\Delta\Theta_2$ de l'axe principal d'un œil. Le projeté r'_c du rayon r_c sur le plan contenant la cible, se voit donc agrandi ou réduit ($r_c \neq r'_c$), impliquant une erreur, étant donné que la grandeur mesurée d' n'est pas égale à d .

que nous citons par ordre d'importance :

1. Les mesures $\bar{\epsilon}_{r_i}$ présentent, en réalité, un bruit additif que l'on peut considérer comme gaussien (voir chapitre IV) et dont l'écart type maximal est inférieur à 100 m° (millidegrés). Bien que la fonction $f(\bar{\epsilon}_{r_i})$ ne transforme pas une gaussioïde en une autre gaussioïde (on ne peut pas directement déduire un écart type pour la mesure de $\bar{\psi}$), on peut déjà donner un ordre de grandeur de l'amplitude du bruit de mesure. En effet, en considérant $D = 1.3 \text{ m}$ et $r_c = 7.5 \text{ mm}$, on peut en déduire qu'une erreur de mesure de $\pm 0.1^\circ$ sur les erreurs rétinienne $\bar{\epsilon}_{r_i}$, donnerait une erreur de $\pm 16.8^\circ$. On remarque donc, que le très faible rayon r_c induisant un rapport $\frac{D}{r_c}$ élevé, implique une forte réduction de précision pour la mesure de l'angle $\bar{\psi}$.
2. Le calcul de la distance d est obtenu en moyennant la mesure de deux yeux opposés. Cette moyenne suppose que l'œil est parfaitement centré sur la cible.
Or, cela n'est pas forcément le cas, car ce centrage dépend de l'asservissement visuel. Et même si celui-ci fonctionne suffisamment bien pour poursuivre la cible, l'erreur rétinienne en roulis et tangage peut atteindre quelques degrés.
3. La conception de l'œil est supposée parfaite pour le calcul de $\bar{\psi}$, c'est à dire que l'axe optique principal de chacun des yeux est parfaitement perpendiculaire au plan comprenant les quatre optiques (V.12a).
Ceci est difficilement atteignable en réalité, et les axes principaux des différents yeux ne sont

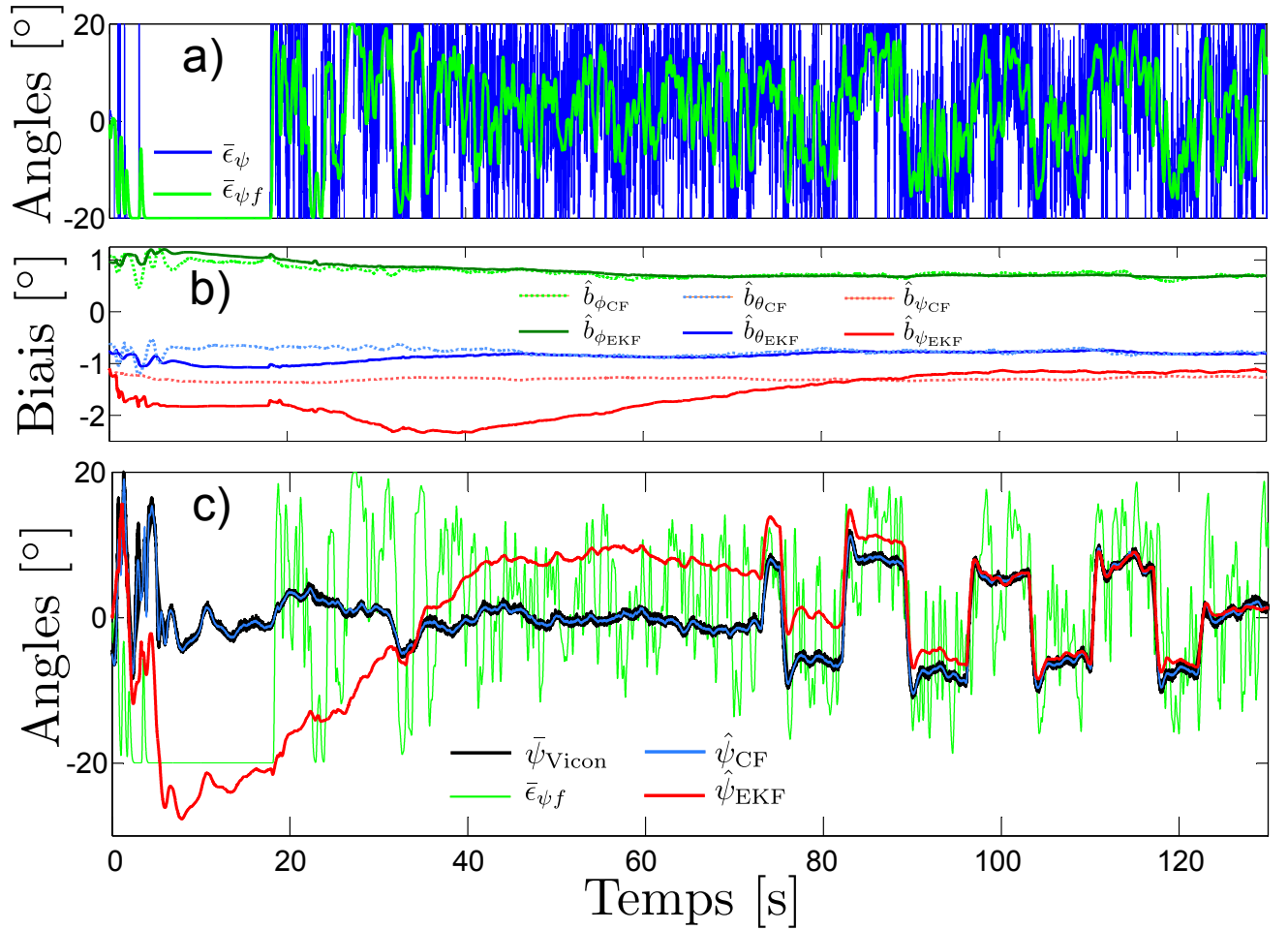


FIGURE V.13 – a) En bleu la mesure brute de l’erreur rétinienne $\bar{\epsilon}_\psi$ de l’œil qui est très fortement bruitée. En vert, la même sortie filtrée avec un filtre passe bas d’ordre 5 ayant une constante de temps de 50ms. b) Estimation des différents biais des gyromètres. En vert pointillé le biais en roulis estimé par le filtre complémentaire (CF), et en vert continu le biais estimé par le filtre de Kalman étendu (EKF). En bleu pointillé le biais en tangage estimé par le CF et en bleu continu le biais estimé par l’EKF. En rouge pointillé le biais en lacet estimé par le CF et en rouge continu le biais estimé par l’EKF. Entre $t = 5s$ et $t = 20s$, la mesure de l’erreur rétinienne est artificiellement bloquée afin de démontrer que le biais ne peut plus être estimé. c) Comparaison de l’angle de lacet mesuré par le Vicon (noir), de celui estimé par le filtre complémentaire se basant sur la mesure du Vicon (bleu), de l’erreur rétinienne filtrée (vert) et de l’estimation faite par l’EKF (rouge). Une fois que le biais du gyromètre correspondant a convergé, on estime correctement l’angle de lacet.

alors pas complètement parallèles, présentant un défaut d’alignement selon deux directions que l’on note $\Delta\theta_1$ et $\Delta\theta_2$. Les figures V.12b-c illustre ce défaut. Le premier défaut, lié à $\Delta\theta_1$, est transparent et complètement compensé par la méthode de calibration de l’œil. Mais le second défaut $\Delta\theta_2$ n’est pas pris en compte et il n’en résulte qu’un très léger décalage de l’axe optique. Ceci implique une surestimation ou une sous estimation de la distance d , liée au fait que le projeté r'_c n’est plus égal au rayon initial r_c (voir figure V.12c).

La figure V.13a illustre la forte dégradation de la mesure réelle, qui découle des trois imperfections décrites ci-dessus. On observe alors que la mesure brute (en bleue) est particulièrement bruitée du fait du rapport élevé $\frac{D}{r_c}$. On choisit donc de réduire l'impact des imperfections, en filtrant la mesure brute. Cela permet d'atténuer le bruit, issu des erreurs rétinienne et amplifié par le rapport $\frac{D}{r_c}$ (imperfection 1). De plus, en filtrant cette mesure, cela revient à admettre que la cible n'est pas située au centre du champ de vision à chaque instant (imperfection 2), mais que si on prend une fenêtre d'observation suffisamment large, cela devient vrai en moyenne. La mesure brute de l'angle $\bar{\psi}$, issue de la fusion des 4 yeux, est alors filtrée par un filtre passe bas d'ordre 5 et ayant une constante de temps de 50 ms. On obtient alors, comme le montre la figure V.13c, une mesure filtrée qui, bien que encore fortement bruitée, est centrée autour de la vraie valeur de ψ (courbe noire correspondant à la mesure Vicon). Le filtre de Kalman étendu, utilise ensuite la composante basse fréquence de cette mesure, pour estimer le biais du gyromètre correspondant. Cependant, la mesure étant très bruitée, on obtient une convergence beaucoup plus lente que pour les biais des deux autres gyromètres (voir figure V.13b), ce qui explique que l'angle estimé $\hat{\psi}$, mette plus de 90 secondes pour être estimé sans erreur.

On rappelle que l'angle de lacet mesurée ici par l'œil, correspond à l'angle de lacet relatif entre la cible et le robot. Sur les figures V.13, le robot était piloté par le contrôleur basé sur le Vicon et l'EKF tournait en parallèle sur le robot. L'écart de lacet entre la cible et le robot, était alors généré en réalisant des échelons sur la consigne de lacet du quadrirotor.

La bonne qualité d'estimation obtenue à partir de $t = 95$ s (malgré le filtrage fort de de la mesure de lacet de l'œil : $\bar{\epsilon}_\psi$), vient du fait que les mouvements de lacet sont réalisés par le robot et donc mesurés par le gyromètre, dont le biais est correctement estimé. On propose donc d'étudier la qualité de l'estimation, dans le cas où ce serait la cible qui modifierait son lacet et non le robot. On reprend donc les mesures réalisées par l'ensemble des capteurs lors de l'essai précédent, puis on ajoute artificiellement lors d'une simulation², une variation de l'angle de la cible $\bar{\psi}_T$. Les figures V.14a-c montrent donc l'évolution de l'estimation, pour différentes pondérations de la mesure $\bar{\epsilon}_\psi$ dans l'EKF. La pondération $W_{\bar{\psi}}$ (la non confiance en la mesure) étant diminuée d'un facteur 10 de a) à b) puis, encore d'un facteur 10 de b) à c). La figure V.14d représente, quant à elle, l'erreur d'estimation en fonction de la pondération choisie. Premièrement, on remarque que pour les mêmes valeurs initiales de l'EKF, la pondération de la mesure à une forte influence sur la trajectoire de convergence de l'estimateur (courbes rouges des figures V.14a-c). Naturellement, plus la pondération est faible (plus on a confiance en la mesure) et plus l'angle de lacet estimé converge rapidement vers la valeur réelle. Cependant, la mesure de lacet $\bar{\epsilon}_\psi$ étant très bruitée, une trop forte pondération implique une trop forte sensibilité au bruit sur l'estimation. En effet, la diminution de la pondération améliore l'estimation jusqu'à un certain point, puis la sensibilité au bruit devient trop grande pour des pondérations trop faibles (l'erreur d'estimation est plus faible

2. On choisit ici de faire cette étude en simulation, sur un jeu de données réelles, afin de mieux pouvoir comparer les résultats obtenus.

pour la courbe rouge que pour la courbe bleue sur la figure V.14d). On recherche donc la pondération la plus faible, qui permette à la fois d'obtenir une convergence rapide vers l'écart de lacet, tout en assurant une certaine insensibilité au bruit de mesure. La figure V.14c semble représenter un bon compromis, car l'estimée converge en 4 s vers le lacet de la cible (entre $t = 310$ s et $t = 314$ s) alors qu'il faut plus de 14 s sur la figure V.14a.

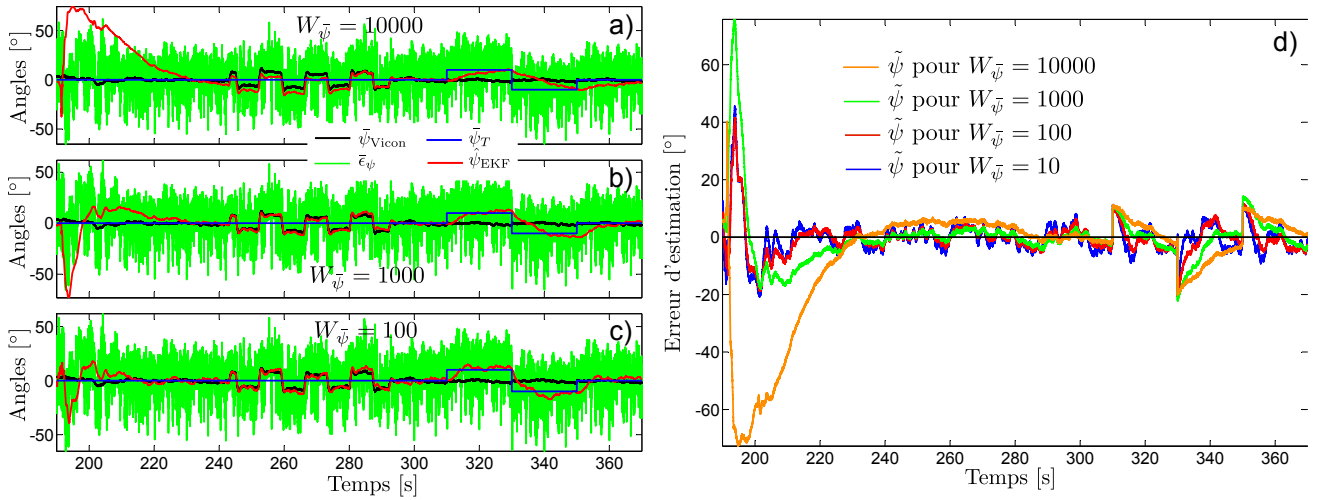


FIGURE V.14 – a-c) Évolution de l'estimation de l'angle de lacet en fonction de la pondération de la mesure de l'œil. En vert la mesure brute $\tilde{\epsilon}_{\psi}$ issue de la fusion des quatre yeux, en noir la mesure réelle de l'angle de lacet réelle du robot $\tilde{\psi}_{Vicon}$, en bleu l'orientation en lacet de la cible $\tilde{\psi}_T$ et en rouge l'estimation de l'angle de lacet du robot par rapport à la cible $\hat{\psi}_{EKF}$. d) L'erreur d'estimation en fonction de la pondération de la mesure. L'erreur d'estimation $\tilde{\psi}$ est définie par $\tilde{\psi} = \tilde{\psi}_{Vicon} - (\hat{\psi} - \tilde{\psi}_T)$.

Dans le cas de perturbations en lacet sur le robot comme le vent par exemple, on peut considérer que, grâce au gyromètre, cela n'aura que peu d'influence. En effet, on voit bien, que lorsque le biais a convergé, le gyromètre permet d'estimer parfaitement l'angle de lacet et ce, instantanément. Cependant, dans le cas où le lacet de la cible évoluerait, la convergence reste assez longue (environ 4 secondes). On en déduit donc, que les capacités de poursuite d'une cible effectuant des mouvements en lacet, reste assez limitées. De plus, on remarque que lors de l'initialisation du filtre, les valeurs de lacet peuvent être assez éloignées de la réalité de plus de 40° , tant que le biais du gyromètre n'est pas correctement estimé. Ce qui est bien supérieur au champ de vision de l'œil, sur l'axe de lacet, qui ne présente pas de découplage. On prendra donc soin de systématiquement attendre une quarantaine de secondes, avant de décoller, afin de laisser le temps à l'estimation du biais du gyromètre de converger. De même, le robot devra être placé avec un angle de lacet assez faible, par rapport à la cible, avant le décollage, afin de limiter les risques de perdre la cible de vue durant ce décollage. Enfin, on veillera à ce que les mouvements de la cible restent suffisamment lents selon le lacet, afin de laisser le temps à l'EKF d'estimer l'orientation de la cible.

4.3.2 Modèle et estimations sans accéléromètre

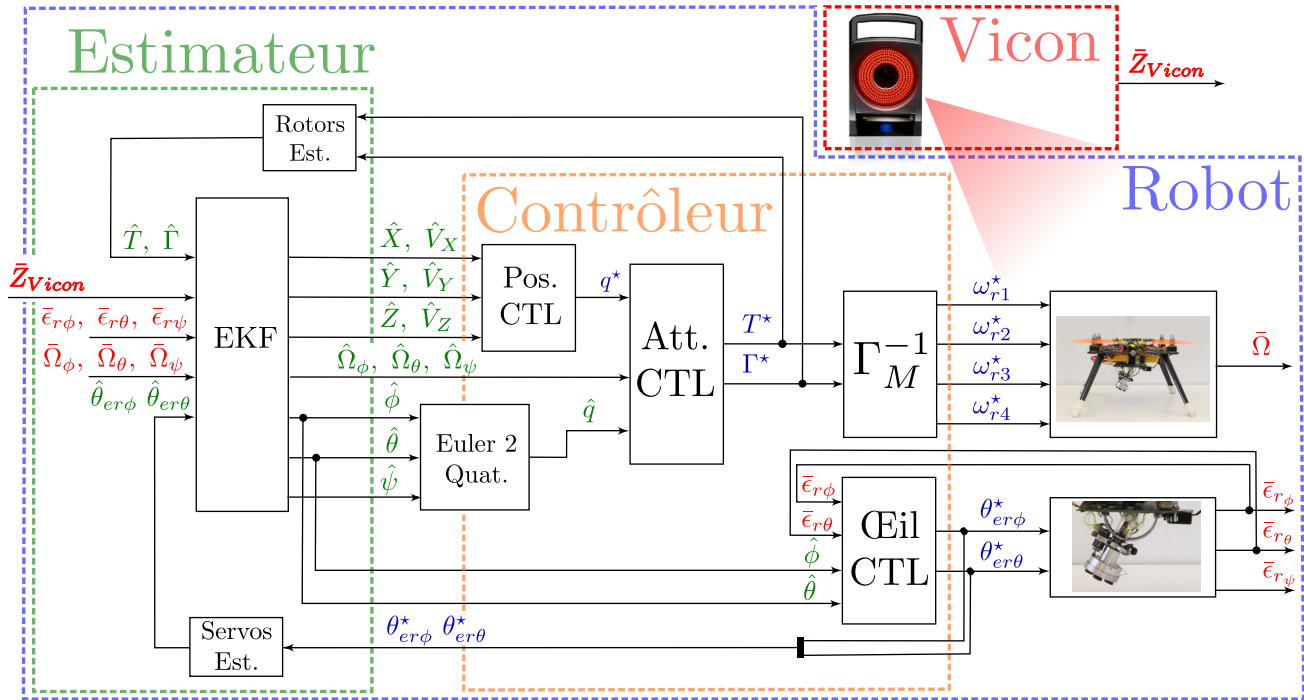


FIGURE V.15 – Diagramme en bloc du système complet. Le bloc **Servos Est.** estime la position réelle des servomoteurs à partir de leur modèle dynamique (V.14) ; Le bloc **Rotors Est.** estime la poussée générée à partir du modèle dynamique des rotors (V.16). Le bloc **EKF** correspond au filtre de Kalman étendu, estimant la position et la vitesse du robot, le roulis et le tangage, ainsi que les vitesses de rotation correspondantes (V.48) à (V.62). **Pos. CTL** est le contrôleur de position (III.71) à (III.74), **Att. CTL** le contrôleur d'attitude (III.70) et **Œil CTL** est le contrôleur de regard (V.10). Enfin, **Quat. 2 Euler** permet d'extraire du quaternion d'attitude les angles de roulis et de tangage et Γ_M^{-1} permet de passer de la consigne de couple et de poussée aux consignes rotors. Ici, le système Vicon ne fournit que l'altitude du robot.

Dorénavant, on utilise uniquement l'altitude fournie par le Vicon. L'attitude, les vitesses de rotation, les biais des gyromètres, les positions et les vitesses de translations sont estimés par l'EKF. Cette fois ci, la mesure des accéléromètres n'est plus prise en compte pour l'estimation des différents états. Seules les mesures des gyromètres, et les mesures de l'œil sont utilisées au niveau du robot, et le Vicon est utilisé pour émuler un altimètre. La figure V.15 résume les interactions des différents estimateurs et contrôleurs. Ici, le recalage des dérives des gyromètres est réalisé grâce à l'orientation du regard et les erreurs rétinienne mesurées par l'œil.

On s'intéresse donc au vecteur d'état complet noté $x = (\xi \ v \ \Theta \ \Omega \ b)^T$ décrit par le modèle

suivant :

$$\begin{pmatrix} \dot{\xi} \\ \dot{v} \\ \dot{\theta} \\ \dot{\Omega} \\ \dot{b} \end{pmatrix} = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{V}_X \\ \dot{V}_Y \\ \dot{V}_Z \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{\Omega}_\phi \\ \dot{\Omega}_\theta \\ \dot{\Omega}_\psi \\ \dot{b}_\phi \\ \dot{b}_\theta \\ \dot{b}_\psi \end{pmatrix} = \begin{pmatrix} V_X \\ V_Y \\ V_Z \\ \frac{T}{m}(\sin \phi \sin \psi + \cos \phi \cos \psi \sin \theta) - \frac{K_{vx}}{m} V_X \\ \frac{T}{m}(\cos \phi \sin \psi \sin \theta - \sin \phi \cos \psi) - \frac{K_{vy}}{m} V_Y \\ \frac{T}{m}(\cos \phi \cos \theta) - \frac{K_{vz}}{m} V_Z \\ \Omega_\phi \\ \Omega_\theta \\ \Omega_\psi \\ \Omega_\theta \Omega_\psi \frac{(I_y - I_z)}{I_x} + \Gamma_\phi \\ \Omega_\phi \Omega_\psi \frac{(I_z - I_x)}{I_y} + \Gamma_\theta \\ \Omega_\phi \Omega_\theta \frac{(I_x - I_y)}{I_z} + \Gamma_\psi \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (\text{V.46})$$

On dispose du vecteur de mesure décrit par l'équation suivante :

$$\bar{y} = \begin{pmatrix} \bar{Z}_{\text{Vicon}} \cdot \tan(-\hat{\theta}_{er_\theta} + \bar{\epsilon}_{r_\theta} + \hat{\theta}) \\ -\bar{Z}_{\text{Vicon}} \cdot \tan(-\hat{\theta}_{er_\phi} + \bar{\epsilon}_{r_\phi} + \hat{\phi}) \\ \bar{Z}_{\text{Vicon}} \\ \bar{\epsilon}_{r_\psi} \\ \bar{\Omega} \end{pmatrix} = h(x) = \begin{pmatrix} X \\ Y \\ Z \\ \psi \\ \Omega + b \end{pmatrix} \quad (\text{V.47})$$

Observabilité du système : Pour les mêmes raisons que précédemment (taille), on ne détaille pas la matrice d'observabilité, mais on vérifie que cette dernière est bien de plein rang. La matrice d'observabilité \mathcal{O} obtenue pour le système (V.46), avec le vecteur de mesure \bar{y} (V.47), est bien de rang 15. On en déduit donc, que le système est localement faiblement observable et que l'on peut par conséquent estimer les différents états du système.

Filtre de Kalman Étendu (EKF) : On construit donc le filtre de Kalman étendu suivant, permettant d'estimer les états du système :

1. Étape de prédiction :

(a) Prédiction des états :

$$\begin{pmatrix} \hat{X}_{k|k-1} \\ \hat{Y}_{k|k-1} \\ \hat{Z}_{k|k-1} \\ \hat{V}_{Xk|k-1} \\ \hat{V}_{Yk|k-1} \\ \hat{V}_{Zk|k-1} \\ \hat{\phi}_{k|k-1} \\ \hat{\theta}_{k|k-1} \\ \hat{\psi}_{k|k-1} \\ \hat{\Omega}_{\phi k|k-1} \\ \hat{\Omega}_{\theta k|k-1} \\ \hat{\Omega}_{\psi k|k-1} \\ \hat{b}_{\phi k|k-1} \\ \hat{b}_{\theta k|k-1} \\ \hat{b}_{\psi k|k-1} \end{pmatrix} = \begin{pmatrix} \hat{X}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Xk-1|k-1} \\ \hat{Y}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Yk-1|k-1} \\ \hat{Z}_{k-1|k-1} + \Delta t \cdot \hat{V}_{Zk-1|k-1} \\ \frac{\hat{T}_k}{m} (\sin \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_{k-1|k-1} + \cos \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1}) - \frac{K_{vx}}{m} V_{Xk-1|k-1} \\ \frac{\hat{T}_k}{m} (\cos \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} - \sin \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_{k-1|k-1}) - \frac{K_{vy}}{m} V_{Yk-1|k-1} \\ \frac{\hat{T}_k}{m} (\cos \hat{\phi}_{k-1|k-1} \cos \hat{\theta}_{k-1|k-1}) - \frac{K_{vz}}{m} V_{Zk-1|k-1} \\ \hat{\phi}_{k-1|k-1} + \Delta t \cdot \hat{\Omega}_{\phi} \\ \hat{\theta}_{k-1|k-1} + \Delta t \cdot \hat{\Omega}_{\theta} \\ \hat{\psi}_{k-1|k-1} + \Delta t \cdot \hat{\Omega}_{\psi} \\ \Omega_{\phi k-1|k-1} + \Delta t \cdot (\Omega_{\theta k-1|k-1} \Omega_{\psi k-1|k-1} \frac{(I_y - I_z)}{I_x} + \hat{\Gamma}_{\phi k}) \\ \Omega_{\theta k-1|k-1} + \Delta t \cdot (\Omega_{\phi k-1|k-1} \Omega_{\psi k-1|k-1} \frac{(I_z - I_x)}{I_y} + \hat{\Gamma}_{\theta k}) \\ \Omega_{\psi k-1|k-1} + \Delta t \cdot (\Omega_{\phi k-1|k-1} \Omega_{\theta k-1|k-1} \frac{(I_x - I_y)}{I_z} + \hat{\Gamma}_{\psi k}) \\ \hat{b}_{\phi k-1|k-1} \\ \hat{b}_{\theta k-1|k-1} \\ \hat{b}_{\psi k-1|k-1} \end{pmatrix} \\
\hat{x}_{k|k-1} \qquad \qquad \qquad f(\hat{x}_{k-1|k-1}, u_k)
\end{pmatrix} \tag{V.48}$$

(b) Prédiction de la matrice de covariance de l'erreur :

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k \tag{V.49}$$

Avec Q_k une matrice diagonale et F_k la jacobienne donnée par :

$$F_k = \frac{\partial f}{\partial x} \Big|_{\hat{x}_{k-1|k-1}, u_k} = \begin{pmatrix} I_{3 \times 3} & \Delta t \cdot I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & F_{a3 \times 3} & F_{b3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & \Delta t \cdot I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & F_{c3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} \end{pmatrix} \tag{V.50}$$

Où :

$$F_{a3 \times 3} = \begin{pmatrix} 1 - \Delta t \cdot \frac{K_{vx}}{m} & 0 & 0 \\ 0 & 1 - \Delta t \cdot \frac{K_{vy}}{m} & 0 \\ 0 & 0 & 1 - \Delta t \cdot \frac{K_{vz}}{m} \end{pmatrix} \tag{V.51}$$

$$F_{b3 \times 3} = \Delta t \cdot \frac{\hat{T}_k}{m} \cdot (F_{b:,1} \quad F_{b:,2} \quad F_{b:,3}) \tag{V.52}$$

Avec :

$$F_{b:,1} = \begin{pmatrix} \cos \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_{k-1|k-1} - \cos \hat{\psi}_{k-1|k-1} \sin \hat{\phi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} \\ - \cos \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_{k-1|k-1} - \sin \hat{\phi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} \sin \hat{\psi}_{k-1|k-1} \\ - \cos \hat{\theta}_{k-1|k-1} \sin \hat{\phi}_{k-1|k-1} \end{pmatrix} \quad (\text{V.53})$$

$$F_{b:,2} = \begin{pmatrix} \cos \hat{\phi}_{k-1|k-1} \cos \hat{\theta}_{k-1|k-1} \cos \hat{\phi}_{k-1|k-1} \\ \cos \hat{\phi}_{k-1|k-1} \cos \hat{\theta}_{k-1|k-1} \sin \hat{\phi}_{k-1|k-1} \\ - \cos \hat{\phi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} \end{pmatrix} \quad (\text{V.54})$$

$$F_{b:,3} = \begin{pmatrix} \cos \hat{\psi}_{k-1|k-1} \sin \hat{\phi}_{k-1|k-1} - \cos \hat{\phi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} \sin \hat{\psi}_{k-1|k-1} \\ - \sin \hat{\phi}_{k-1|k-1} \sin \hat{\psi}_{k-1|k-1} + \cos \hat{\phi}_{k-1|k-1} \cos \hat{\psi}_{k-1|k-1} \sin \hat{\theta}_{k-1|k-1} \\ 0 \end{pmatrix} \quad (\text{V.55})$$

Et :

$$F_{c3 \times 3} = \begin{pmatrix} 1 & \Delta t \cdot \Omega_{\psi_{k-1|k-1}} \cdot \frac{(I_y - I_z)}{I_x} & \Delta t \cdot \Omega_{\psi_{k-1|k-1}} \cdot \frac{(I_y - I_z)}{I_x} \\ \Delta t \cdot \Omega_{\psi_{k-1|k-1}} \cdot \frac{(I_z - I_x)}{I_y} & 1 & \Delta t \cdot \Omega_{\phi_{k-1|k-1}} \cdot \frac{(I_z - I_x)}{I_y} \\ \Delta t \cdot \Omega_{\theta_{k-1|k-1}} \cdot \frac{(I_x - I_y)}{I_z} & \Delta t \cdot \Omega_{\phi_{k-1|k-1}} \cdot \frac{(I_x - I_y)}{I_z} & 1 \end{pmatrix} \quad (\text{V.56})$$

2. Étape de correction et mise à jour :

(a) Calcul de l'erreur de prédiction (terme d'innovation) :

$$\tilde{y}_k = \bar{y}_k - h(\hat{x}_{k|k-1}) = \underbrace{\begin{pmatrix} \bar{Z}_{\text{Viconk}} \cdot \tan(\hat{\theta}_{er_{\theta k}} + \bar{\epsilon}_{r_{\theta k}} + \hat{\theta}_k) \\ -\bar{Z}_{\text{Viconk}} \cdot \tan(-\hat{\theta}_{er_{\phi k}} - \bar{\epsilon}_{r_{\phi k}} + \hat{\phi}_k) \\ \bar{Z}_{\text{Viconk}} \\ \bar{\epsilon}_{r_{\psi k}} \\ \bar{\Omega}_k \end{pmatrix}}_{\bar{y}_k} - \underbrace{\begin{pmatrix} \hat{X}_{k|k-1} \\ \hat{Y}_{k|k-1} \\ \hat{Z}_{k|k-1} \\ \hat{\psi}_{k|k-1} \\ \hat{\Omega}_{k|k-1} + \hat{\theta}_{k|k-1} \end{pmatrix}}_{h(\hat{x}_{k|k-1})} \quad (\text{V.57})$$

(b) Mise à jour de la covariance du terme d'innovation :

$$S_k = H_k P_{k|k-1} H_k^T + R_k \quad (\text{V.58})$$

Avec :

$$H_k = \begin{pmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{1 \times 3} & 0_{1 \times 3} & 0_{1 \times 2} & 1 & 0_{1 \times 3} & 0_{1 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & I_{3 \times 3} \end{pmatrix} \quad (\text{V.59})$$

(c) Calcul du gain du filtre :

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \quad (\text{V.60})$$

(d) Correction de la prédiction des états :

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k \quad (\text{V.61})$$

(e) Correction de la covariance de l'erreur :

$$P_{k|k} = (I_{15 \times 15} - K_k H_k) P_{k|k-1} \quad (\text{V.62})$$

4.3.3 Résultats expérimentaux

Scénario 1 : Dans un premier temps, afin de valider le fonctionnement de l'EKF complet, nous allons juste réaliser une trajectoire circulaire d'un mètre de rayon à 1.3 m au dessus de la cible et vérifier les estimations fournies par l'EKF. Pour ce premier essai, le robot est contrôlé à l'aide du contrôleur utilisant les mesures Vicon, et l'EKF tourne juste en parallèle à une fréquence de 400Hz.

Résultats 1 : La figure V.16 montre donc l'évolution des estimations au cours de cette trajectoire. Les estimations en positions sont plutôt de bonnes qualités, étant donné que l'erreur en position sur l'axe X reste inférieure à 12 cm et l'erreur sur l'axe Y inférieure à 20 cm. L'estimation d'attitude semble également satisfaisante et présente une erreur maximale de 2° pour le roulis, de 4° pour le tangage et de 6° pour le lacet.

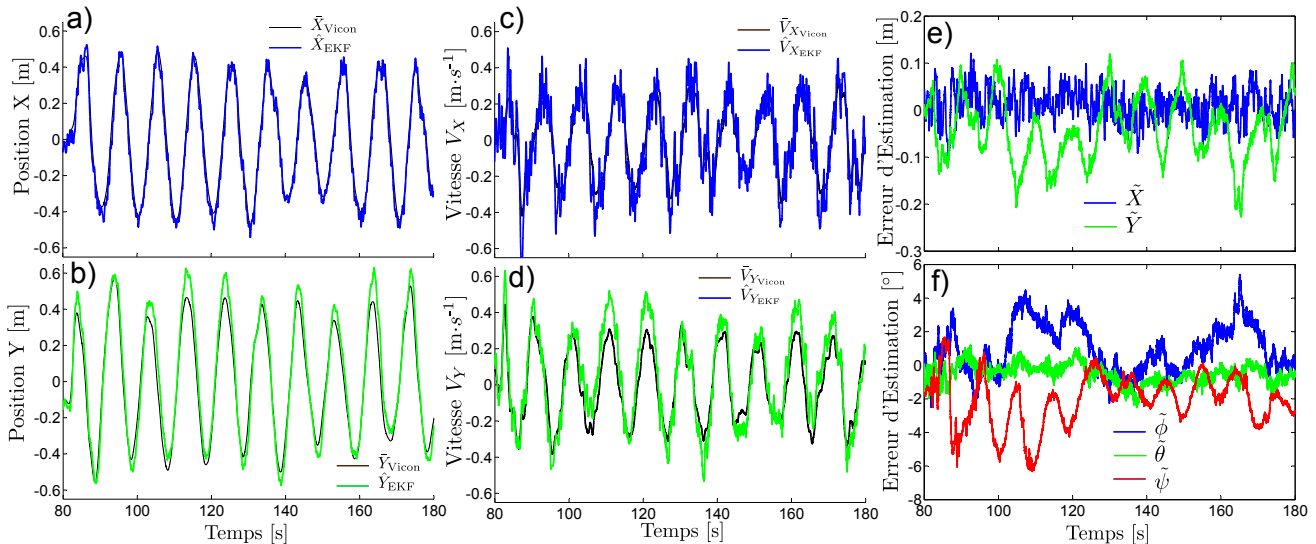


FIGURE V.16 – Erreur d'estimations du filtre de Kalman complet lors d'une trajectoire circulaire. a-b) Comparaison de la position estimée et de la position réelle mesurée (noir) lors d'une trajectoire circulaire à altitude constante de rayon 0.5m. c-d) Comparaison des vitesses estimées avec les vitesses réelles mesurées (noir) pour la même trajectoire. e) Erreur d'estimation de position. f) Erreur d'estimation des angles de roulis tangage et lacet.

		$\tilde{\phi}$ [°]	$\tilde{\theta}$ [°]	$\tilde{\psi}$ [°]	\tilde{X} [cm]	\tilde{Y} [cm]	\tilde{V}_X [cm · s ⁻¹]	\tilde{V}_Y [cm · s ⁻¹]
Moyenne	$E[\cdot]$	0.99	-0.52	-2.10	1.06	-4.15	0.48	-3.93
Écart Type	$\sigma(\cdot)$	1.47	0.57	1.53	3.42	6.79	9.14	9.32
Min	$\min(\cdot)$	-2.35	-2.37	-6.30	-10.17	-22.67	-28.10	-28.38
Max	$\max(\cdot)$	5.20	1.17	1.65	12.05	11.89	31.24	23.12

TABLE V.3 – Précisions obtenues pour les différentes estimations lors de la répétition d’une trajectoire circulaire.

Scénario 2 : Les résultats obtenus avec l’EKF complet étant suffisants, on décide donc de tester les capacités du robot en vol stationnaire au dessus d’une cible. Pour cela, le robot décolle grâce au contrôleur utilisant les données Vicon, et une fois positionné au dessus de la cible, on bascule sur le contrôleur utilisant les données estimées par l’EKF. Puis après une trentaine de secondes, on déplace manuellement la cible, afin de vérifier les capacités de poursuite du système complet.

Cette fois ci, nous sommes dans la configuration décrite par la figure V.15, c’est à dire que le robot est complètement autonome (on considère que la mesure de hauteur fournie par le Vicon pourrait provenir d’un baromètre, ou d’un télémètre embarqué). L’algorithme de localisation de la cible (algorithme de vision) est exécuté à une fréquence de 800Hz. La fusion de données permettant d’estimer les états du robot (EKF) fonctionne à 400Hz, tout comme le contrôle en attitude et la stabilisation visuelle (réflexes VOR et VFR). La boucle d’asservissement en position et en vitesse fonctionne, quant à elle, à 100Hz. Enfin, l’ensemble des données est supervisé et enregistré sur la station sol à une fréquence de 200Hz, via une connexion Wifi. Le robot est donc complètement autonome en terme de ressources calculatoires, car l’ensemble des calculs sont effectués à bord.

Résultats 2 : La figure V.17 montre donc les résultats de cette expérience. On observe, lors du passage sur le contrôleur reposant sur l’EKF, un écart du robot par rapport à sa position de référence. En effet, sur la figure V.17a, le robot s’éloigne jusqu’à plus de 50 cm, avant de revenir autour de sa position de référence. Cela s’explique, une fois encore, par l’initialisation du contrôleur (les différents intégrateurs doivent atteindre leur état d’équilibre). Une fois cette phase transitoire écoulée, le robot parvient à se maintenir au dessus de la cible, avec une précision inférieure à 20 cm. Et ce, même lorsque la cible est en déplacement, à partir de $t = 150s$. On observe toujours un transitoire assez long sur l’estimation de l’angle de lacet, qui de fait des couplages, entraîne une erreur sur les angles de roulis et de tangage qui finit par disparaître une fois que l’angle de lacet est correctement estimé (voir figures V.17c-e et g entre $t = 130s$ et $t = 160s$). On remarque bien que le contrôle d’attitude se base sur les données de l’EKF, étant donné que sur la figure V.17e, c’est l’angle $\hat{\psi}_{EKF}$ qui est maintenu autour de zéro. Ceci implique que, durant une bonne partie du vol, l’angle réel $\bar{\psi}_{Vicon}$ n’est en réalité pas égal à 0. Cela permet de mettre en évidence la robustesse du système bouclé, face à une erreur d’estimation

de plus de 10° sur l'angle de lacet.

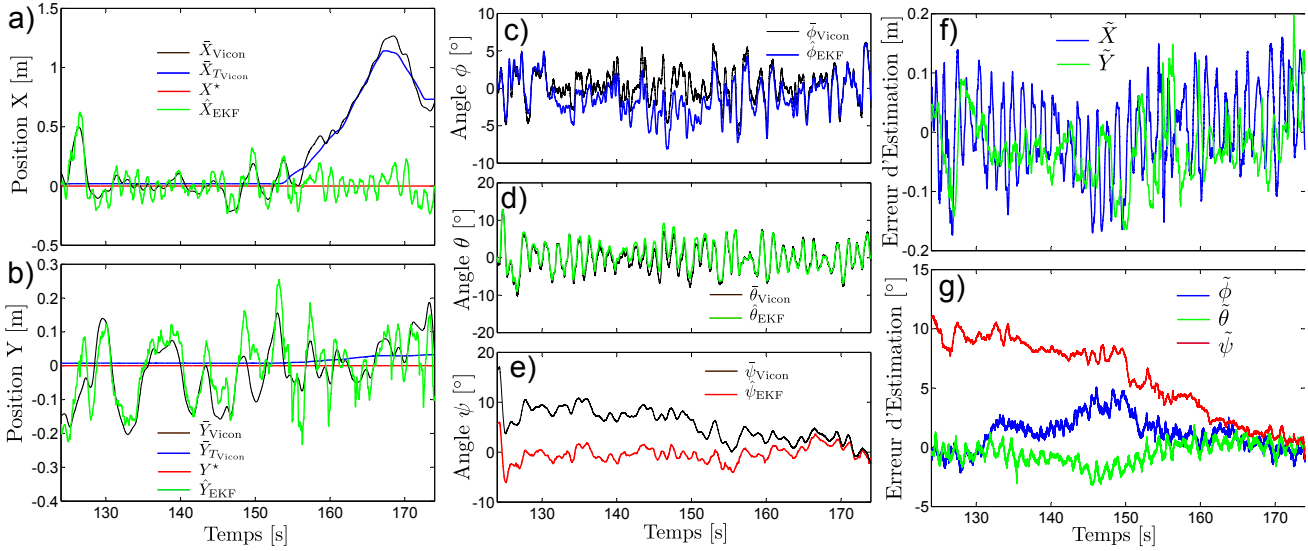


FIGURE V.17 – a) Estimation de la position \hat{X}_{EKF} en vert, consigne de position relative X^* en rouge, position du Rover \bar{X}_{TVicon} en bleu et position réelle du quadrirotor \bar{X}_{Vicon} en noir. b) Estimation de la position \hat{Y}_{EKF} en vert, consigne de position relative Y^* en rouge, position du Rover en bleu et position réelle du quadrirotor (en noir).

4.3.4 Discussions

J'ai donc montré dans cette partie que X4-MaG était capable d'accomplir à la fois la tâche de stabilisation, mais également de poursuite de cible, et ce, sans utiliser de centrale inertielle 9 axes (en se basant uniquement sur un gyromètre 3 axes peu précis, un œil à seulement 24 pixels et un découplage actionné de l'œil). J'ai également montré dans la partie 4.3.1 que l'estimation du lacet était un des points faibles, et que sa dynamique d'estimation était assez lente. Les craintes évoquées dans la partie précédente étaient liées au fait que pour une erreur d'estimation de l'angle de lacet trop importante, le robot pourrait perdre la cible de vue (du fait du non alignement avec elle) et ainsi perdre tout moyen d'estimer son attitude et de se localiser.

Cette expérience se veut donc rassurante concernant ce point, car on voit que même avec une erreur d'estimation plutôt importante ($> 10^\circ$), le robot parvient toujours à maintenir son équilibre au dessus de la cible. Cela peut s'expliquer par une étude du champ de vision de l'œil. En effet, l'équation (V.45) montre le lien entre l'erreur rétinienne mesurée et l'angle $\bar{\epsilon}_{r,\psi}$ correspondant. Le champ de vision minimal (pire cas parmi les quatre yeux) étant de $FOV_{MIN} = 18.3^\circ$, on peut estimer que l'angle maximal

mesurable $\bar{\epsilon}_{r_\psi}$ par l'œil est de :

$$\bar{\psi} = \bar{\epsilon}_\psi = \frac{1}{2} \tan^{-1} \left(\frac{D \cdot \tan \left(\frac{FOV_{\text{MIN}}}{2} \right)}{r_c} \right) \quad (\text{V.63})$$

Dans le cas précédent, le robot volait à une altitude de 1.3 m, et le contrôle d'attitude semble pouvoir maintenir l'angle de lacet dans une enveloppe de $\pm 5^\circ$ autour de 0. Ce qui laisse penser que l'erreur d'estimation maximale pourrait être de 23.7° , sans pour autant risquer de perdre de vue la cible. En effet, on voit ici que, malgré un mauvais alignement de plus de 10° du robot par rapport à la cible, cette dernière reste dans le champ de vision du robot, qui réduit peu à peu cette erreur de cap.

5 Application à la poursuite d'un Rover

5.1 Poursuite du Rover basée vision

Scénario : Maintenant que l'estimateur complet (sans accéléromètres) a été validé, nous pouvons utiliser le robot pour poursuivre un robot rover équipé d'une cible. Le robot, initialement placé sur le rover immobile, décolle grâce au contrôleur utilisant les données Vicon. Puis à $t = 106\text{s}$, on bascule sur le contrôleur basé sur l'EKF, afin d'effectuer un vol stationnaire au dessus du Rover. Puis, à partir de $t = 135\text{s}$, le Rover est piloté manuellement pour effectuer des translations selon l'axe Y . Dans un premier temps, les translations sont effectuées avec des arrêts et des changements de sens doux (voir figure V.18c $t = 145\text{s}$ et $t = 162\text{s}$). Puis, on effectue ensuite des arrêts et des changements de sens brusques (à $t = 175\text{s}$, $t = 182\text{s}$ et $t = 190\text{s}$). Puis le Rover s'arrête et le robot reste en vol stationnaire à son aplomb.

Résultats et discussions : On remarque donc que, après avoir basculé sur le contrôleur utilisant les estimés de l'EKF, le robot effectue un écart de plus de 60 cm selon X et de 90 cm selon Y . Puis il revient se positionner au dessus de la cible, et se maintient dans un rayon de ± 10 cm, selon les axes X et Y , mais avec un biais de plus de 5 cm sur l'axe X , et de 11 cm sur l'axe Y . Les biais en position s'expliquent par la calibration du "point 0" des servomoteurs. En effet, lors de l'assemblage initial de l'œil et du robot, l'axe des servomoteurs avait été collé au système de cardans, de façon à avoir un regard parfaitement orthogonal au corps du robot à leur position 0 (consigne d'angle à 0°). Mais lors des différents vols de mise aux points, des crash ainsi que des atterrissages brutaux ont décollé les axes des servomoteurs. Ces derniers ont ensuite été recollés dans une position légèrement différente. Cela

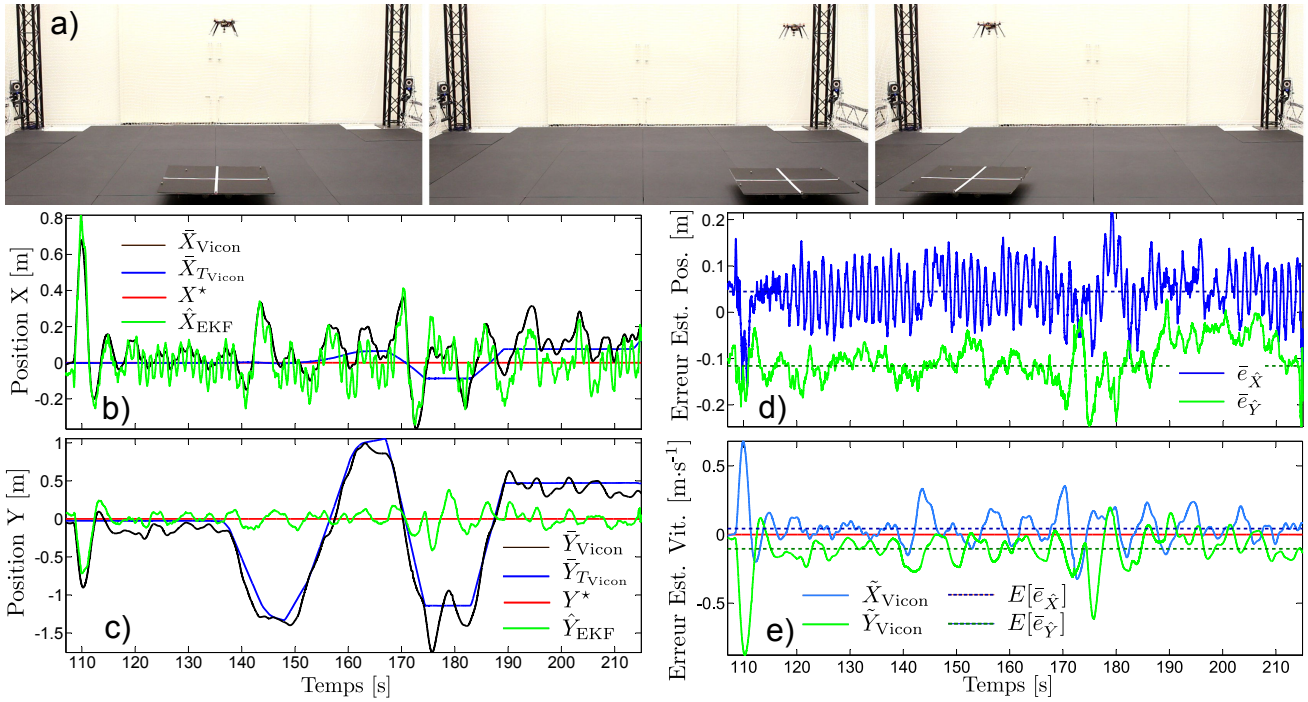


FIGURE V.18 – Poursuite d'un rover non coopératif dont la position n'est pas transmise au robot. a) Séries de photographies illustrant le quadricoptère en train de poursuivre le rover. b et c) Évolution des positions du réel du rover (bleu), de la position du robot (noir), de la consigne de position relative (rouge) et de la position relative estimée par le robot aérien (vert). d) L'erreur d'estimation de position sur les axes X (bleu continu) et Y (vert continu). En pointillés bleus, l'erreur moyenne d'estimation selon l'axe X et en pointillés verts, l'erreur moyenne d'estimation selon Y . e) L'erreur de poursuite de la cible selon l'axe X en bleu clair continu, et l'erreur de poursuite de la cible selon l'axe Y en vert clair continu. Les erreurs de poursuite sont centrées autour des erreurs moyennes d'estimations de position selon X et Y , respectivement tracées en pointillés bleus et pointillés verts.

introduit donc un biais sur la position estimée des servomoteurs :

$$\begin{cases} \hat{\theta}_{er_\phi}(p) = \frac{z_{servo}p + 1}{\frac{1}{\omega_{servo}^2}p^2 + \frac{2\eta}{\omega_{servo}p+1}} \cdot (\theta_{er_\phi}^*(p) + b_{\theta_{er_\phi}}) \\ \hat{\theta}_{er_\theta}(p) = \frac{z_{servo}p + 1}{\frac{1}{\omega_{servo}^2}p^2 + \frac{2\eta}{\omega_{servo}p+1}} \cdot (\theta_{er_\theta}^*(p) + b_{\theta_{er_\theta}}) \end{cases} \quad (V.64)$$

Ce biais entraîne donc une erreur sur l'estimation de position, qui dépend de l'altitude à laquelle le robot évolue et de la fonction tangente (ce biais est clairement visible dans les colonnes \tilde{X} et \tilde{Y} de la ligne *Moyenne* du tableau V.4).

Ce biais pourrait être facilement identifié, en effectuant un vol stationnaire à une hauteur connue. Cela n'a pas été réalisé, car il aurait été nécessaire de le ré-identifier à chaque fois que l'œil était endommagé, ce qui aurait été fastidieux. Et ce biais ne remet de toute façon pas en cause les résultats qui sont exposés ici.

Ensuite, durant la phase où le Rover se déplace de façon "douce", on remarque que le quadrirotor réussit à le poursuivre avec une précision de positionnement de ± 30 cm selon l'axe X et ± 15 cm selon l'axe Y (voir V.18e entre $t = 135$ s et $t = 170$ s). Lors de l'arrêt brutal du Rover à $t = 174$ s, le robot dépasse la position de ce dernier de plus de 60 cm. Cela est principalement dû aux limitations physiques du quadrirotor. En effet, alors que le rover peut s'arrêter instantanément en bloquant ses roues, le quadrirotor doit adopter un angle de roulis important, pour générer une forte décélération pour revenir ensuite seulement autour de la position demandée. Autrement dit, alors que le rover est commandé en vitesse, le quadrirotor est commandé en accélération. On voit bien qu'il s'agit de cette limitation physique car, lors de cette manœuvre, l'erreur d'estimation est seulement de l'ordre de 15 cm. Il s'agit bien d'une limitation physique et non d'une mauvaise estimation des mouvements de la cible. C'est lors de ces manœuvres que le découplage physique de l'œil prend tout son sens. En effet, sans ce découplage, les limitations physiques imposées au quadrirotor ne lui auraient pas permis de maintenir la cible dans son champ visuel. Or ici, l'œil poursuit librement la cible tout en permettant de continuer à bien estimer la position relative du Rover par rapport au quadrirotor, afin de laisser à ce dernier le temps d'effectuer sa manœuvre de demi tour.

		$\tilde{\phi}$ [°]	$\tilde{\theta}$ [°]	$\tilde{\psi}$ [°]	\tilde{X} [cm]	\tilde{Y} [cm]	\tilde{V}_X [cm · s ⁻¹]	\tilde{V}_Y [cm · s ⁻¹]
Moyenne	$E[\cdot]$	1.05	0.08	3.71	4.31	-10.35	0.35	-3.95
Écart Type	$\sigma(\cdot)$	1.47	0.63	2.85	5.47	4.49	14.41	8.80
Min	$\min(\cdot)$	-3.19	-2.44	-1.90	-16.22	-24.72	-48.02	-45.14
Max	$\max(\cdot)$	4.71	2.50	15.16	21.48	2.81	37.91	25.12

TABLE V.4 – Précisions obtenues pour les différentes estimations lors de la poursuite d'un Rover.

Pour conclure sur cette poursuite, le tableau V.4 présente les précisions obtenues pour les différentes estimations. On observe que, par rapport à l'expérience précédente, les précisions sont un peu moins bonnes. Ceci s'explique par le mouvement de la cible qui n'est pas décrit dans le modèle implémenté dans l'EKF (ce qui est normal car on ne connaît à priori, ni le modèle dynamique, ni les déplacements de la cible.) Ne bénéficiant pas de termes de prédictions pour les déplacements de la cible, les estimations sont forcément un peu moins bonnes que précédemment (cas d'une cible immobile, où les mouvements étaient générés "volontairement" par le quadrirotor).

5.2 Atterrissage sur cible mouvante

Scénario : Maintenant que nous avons montré les capacités de poursuite du quadrirotor, nous allons réaliser un atterrissage sur un rover en mouvement. Pour cela, on effectue comme précédemment, un décollage en utilisant le contrôleur basé sur les mesure Vicon. Puis on bascule sur le contrôleur utilisant les estimations de l'EKF afin d'effectuer un vol stationnaire au dessus du rover. Puis le rover est mis

en mouvement, induisant une poursuite de la part du robot. Puis, on demande au quadrirotor d'aller à une altitude de 0 m, tout en se maintenant à l'aplomb de la cible. Cela entraîne donc naturellement un atterrissage automatique. Ici, la descente avait été divisée en deux phases successives par sécurité.

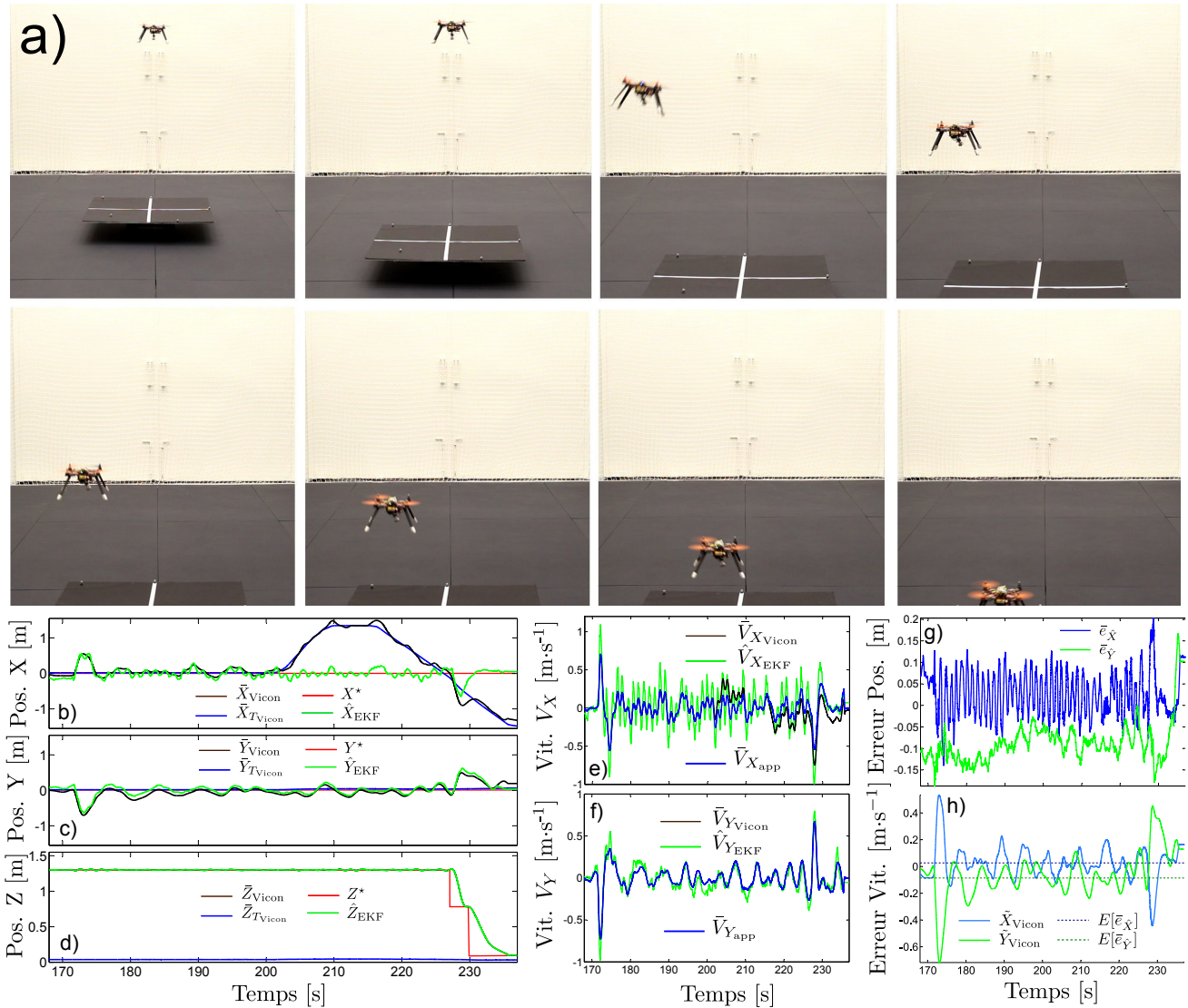


FIGURE V.19 – a) Séquence de photographies montrant l'atterrissage automatique du robot sur un rover en mouvement. b-c-d) Évolution des positions du réelles du rover (bleu), de la position du robot (noir), de la consigne de position relative (rouge) et de la position relative estimée par le robot (vert). e-f) Vitesses réelles du robot $\bar{V}_{X_{Vicon}}$ et $\bar{V}_{Y_{Vicon}}$ mesurées par le Vicon en noir. En vert, les vitesses estimées par l'EKF, qui correspondent aux vitesses apparentes réelles $\bar{V}_{X_{app}} = \bar{V}_{X_{Vicon}} - V_{X_{TVicon}}$ et $\bar{V}_{Y_{app}} = \bar{V}_{Y_{Vicon}} - V_{Y_{TVicon}}$ mesurées par le Vicon. g) Les erreurs d'estimations de positions en bleu pour l'axe X et en vert pour l'axe Y. h) Les erreurs de poursuites de la cible en bleu pour l'axe X et en vert pour l'axe Y. En pointillés bleu et vert, les erreurs moyennes d'estimations de positions sur les axes X et Y respectivement.

a) STATIONNAIRE		$\tilde{\phi}$	$\tilde{\theta}$	$\tilde{\psi}$	\tilde{X}	\tilde{Y}	\tilde{V}_X	\tilde{V}_Y
		[°]	[°]	[°]	[cm]	[cm]	[cm · s ⁻¹]	[cm · s ⁻¹]
Moyenne	$E[\cdot]$	0.11	-0.86	-0.59	2.79	-9.88	-1.85	0.82
Écart Type	$\sigma(\cdot)$	1.18	0.55	0.58	4.86	3.32	14.80	6.42
Min	$\min(\cdot)$	-2.27	-4.25	-2.45	-9.11	-15.74	-36.94	-15.64
Max	$\max(\cdot)$	2.57	0.43	0.52	12.97	-2.67	31.94	16.20
b) POURSUITE + ATTERRISSAGE		$\tilde{\phi}$	$\tilde{\theta}$	$\tilde{\psi}$	\tilde{X}	\tilde{Y}	\tilde{V}_X	\tilde{V}_Y
		[°]	[°]	[°]	[cm]	[cm]	[cm · s ⁻¹]	[cm · s ⁻¹]
Moyenne	$E[\cdot]$	-0.45	-0.75	2.96	2.07	-8.23	-2.19	1.83
Écart Type	$\sigma(\cdot)$	1.02	0.45	0.87	5.10	4.02	14.34	5.47
Min	$\min(\cdot)$	-2.94	-3.01	-5.5	-7.70	-17.88	-33.82	-18.65
Max	$\max(\cdot)$	2.73	0.66	-0.60	20.30	16.13	46.55	15.35

TABLE V.5 – a) Précision obtenues pour les différentes estimations lors de la phase de vol stationnaire. b) Précisions obtenues pour les différentes estimations lors de la phase de poursuite et d’atterrissage.

Résultats et discussions : On observe donc sur la figure V.19a une séquence de photographies mettant en évidence l’atterrissage réussi du quadrirotor sur le rover en mouvement (environ 20 cm · s⁻¹). Sur les courbes des figures V.19b-d, on observe que le quadrirotor poursuit avec succès la cible durant ses déplacements. Durant la phase de descente, on voit que le robot s’écarte un peu de la trajectoire attendue (il se décale sur le côté) mais maintient tout de même son regard sur la cible. Cela lui permet donc, de tout même atterrir sur le rover, à 16.5 cm du centre de la cible sur l’axe X et 12.9 cm sur l’axe Y.

Dans le tableau V.5, on présente les précisions d’estimation obtenues pour la phase de vol stationnaire et la phase de poursuite et atterrissage. On observe qu’en terme d’écart type sur les erreurs d’estimation, les performances sont meilleures pour la phase de poursuite que pour la phase de vol stationnaire. Cela peut s’expliquer par deux phénomènes. Le premier est que la phase de poursuite arrivant plus tard, les biais des gyromètres sont mieux estimés. Le deuxième phénomène peut être tout simplement dû à l’observabilité qui est moins bonne pour des vitesses s’approchant de 0. Lors de la poursuite, les vitesses sont un peu plus importantes, ce qui fournit probablement une information plus riche permettant une meilleure estimation.

6 Discussions finales et perspectives de travail

Dans ce chapitre, j’ai donc montré qu’il était possible de stabiliser entièrement un quadrirotor au dessus d’une cible de type croix à l’aide de l’œil développé dans le chapitre IV. J’ai donc proposé une méthode incrémentale, permettant de reconstruire petit à petit l’ensemble des états du robot, à l’aide de différents filtres de Kalman étendus, afin de valider progressivement l’estimation et l’asservissement de

l'ensemble des degrés de liberté du robot. Une première expérience a permis de démontrer lors d'un vol, l'efficacité des réflexes de stabilisation visuelle, ainsi que les capacités de poursuite qu'offre le système oculomoteur. La conception d'un premier EKF a ensuite permis de montrer que l'orientation de l'œil, ainsi verrouillé sur la cible, permettait de remonter à la position et la vitesse relative du quadrirotor, par rapport à une cible. Dans un second temps, un deuxième filtre de Kalman a permis d'estimer le roulis et le tangage du robot, et ce, en se passant des mesures des accéléromètres. Puis, j'ai montré qu'il était possible d'extraire de la mesure de lacet fournie par l'œil, l'orientation relative de la cible, par rapport à celle du robot. Une attention particulière a alors été portée sur cette mesure de lacet, afin d'expliquer la dégradation de la précision de localisation en lacet par rapport à celle obtenue pour chacun des yeux. Puis, j'ai montré que l'on pouvait, malgré tout, extraire une informations "basse fréquence" sur l'orientation de la cible, qui est suffisante pour asservir le cap du robot sur celui de la cible. Enfin, un dernier EKF a été développé, permettant de reconstituer les 15 états du robot. Pour cela, seuls les mesures des 3 gyromètres (dont on estime les biais), les mesures des 3 orientations relatives de la cible et une mesure de l'altitude du robot sont nécessaires. Finalement, j'ai montré comment ce quadrirotor et son système de vision pouvaient être utilisés pour poursuivre un véhicule non coopératif. De même, il a été montré que l'on pouvait faire atterrir le quadrirotor sur un rover en mouvement, équipé d'une cible.

Le but de ce chapitre était donc de démontrer, qu'il était possible de stabiliser entièrement un quadrirotor, sans utiliser les méthodes classiques qui se basent sur des centrales inertielles complètes. J'ai donc montré, que le regard pouvait être utilisé de façon efficace, à l'instar des accéléromètres, pour estimer les biais des gyromètres. De plus, l'utilisation du regard comme "inclinomètre" présente un avantage certain par rapport aux accéléromètres. En effet, en cas d'accélération latérales fortes, le vecteur de gravité se voit modifié, ce qui engendre un biais dans l'estimation. De même, pour certains mouvements, comme des trajectoires circulaires rapides, les mesures des accéléromètres se voient influencées par la force centrifuge. Le regard, lui, est insensible aux accélérations latérales ou à ces autres forces mesurées par les accéléromètres. De même, les mouvements de rotations pures du robot étant compensés par les réflexes, l'orientation du regard permet de mesurer les translations sans ambiguïté (autrement dit, les mouvements mesurés par l'œil et l'orientation du regard sont liés uniquement à des translations).

Les travaux futurs pourraient être nombreux :

- Durant tout le chapitre, nous avons conservé des contrôleurs en attitude et positions qui avaient été réglés pour le quadrirotor sans son système de vision, en adaptant uniquement les paramètres de masse et de matrice d'inertie. Ce choix avait été fait de façon à pouvoir mieux comparer les résultats entre eux. Une première amélioration consisterait donc à affiner le réglages des contrôleurs d'attitude et de positions, afin d'atténuer les oscillations parasites présentes en vol stationnaire. De même, des techniques de contrôle adaptatif robuste pourraient être testées, comme par exemple

le contrôle adaptatif rapide de type L1.

- Nous avons montré que ces oscillations étaient également dues à la précision réduite qu’offre l’orientation du regard (principalement dû au frottements et à la résolution de l’asservissement des servomoteurs.) Le concept de l’œil ayant été validé, il serait intéressant de créer une deuxième version du mécanisme de découplage. En effet, les servomoteurs avaient été choisis pour leur simplicité d’utilisation, mais ils présentent des limitations structurelles importantes (précision limitée, instabilités pour certaines fréquences d’excitation, fragilité de l’étage de réduction, etc.) Une deuxième version pourrait donc coupler des moteurs brushless Gimbal avec des capteurs de positions angulaires (capteurs magnétiques, codeurs, etc.) afin d’obtenir un asservissement du regard plus précis. Cela permettrait donc, de réduire les oscillations du regard autour d’une position d’équilibre et d’améliorer grandement les estimations de positions et de vitesses.
- Il resterait également à mettre en œuvre les méthodes de guidage par le regard qui avait été introduites dans la fin du chapitre II. En effet, l’œil présentant une très bonne dynamique ($800^\circ \cdot s^{-1}$), il devrait être possible de générer des saccades, permettant au robot de naviguer d’un cible à une autre.
- Il serait intéressant d’étudier les capacités de poursuite du robot sur des trajectoires plus complexes. Le rover pourrait par exemple, décrire un cercle ou un carré, ce qui permettrait de combiner simultanément des translations à des mouvements de lacet. Dans ces cas là, il serait sans doute intéressant d’ajouter un découplage de l’œil sur l’axe de lacet, afin de rendre la poursuite plus robuste aux changements d’orientation de la cible.

Les stratégies présentées ici sont fortement dépendantes de la connaissance de l’altitude. Ici, cette mesure était fournie par le Vicon, mais pour garantir l’autonomie du quadrirotor, il serait intéressant d’ajouter un capteur d’altitude, tel qu’un baromètre, ou encore un télémètre infrarouge par exemple. Une autre solution consisterait à extraire l’information de hauteur du capteur visuel directement. Plusieurs solutions sont alors envisageables :

- En effet, il a été montré que la sortie issue du traitement d’une paire dépend de l’angle sous tendu de la cible et de la position angulaire de la cible (voir figure IV.21 du chapitre IV). Pour une valeur de sortie de capteurs, il y a donc une infinité de couple (position angulaire, angle sous-tendu) possible. Dans ces travaux, nous nous sommes placées à une altitude connue, et avons réalisé une calibration afin de lever l’ambiguïté lié à l’angle sous tendu. Mais en utilisant l’information des paires voisines, on obtient 2 mesures indépendantes, qui permettent de déterminer à la fois l’angle sous tendu et la position angulaire de la cible. Dans le cas où la largeur de la cible est connue, on peut donc remonter à partir de l’angle sous-tendu à une information de distance à la cible (et donc de hauteur dans notre cas). Ces recherches font actuellement l’objet d’une thèse au sein de l’ISM à Marseille (thèse de Fabien Colonnier).

- Une deuxième solution, consisterait à appliquer un louvoiement d'amplitude connue avec l'œil. L'idée est de toujours poursuivre la cible du regard, mais d'ajouter des saccades de l'œil autour de la position de la cible. Ainsi, du fait que le centre de rotation de l'œil ne se trouve pas au niveau du centre optique du capteur, on crée une base binoculaire permettant d'estimer la distance de la cible. Le principe est expliqué par la figure V.20.

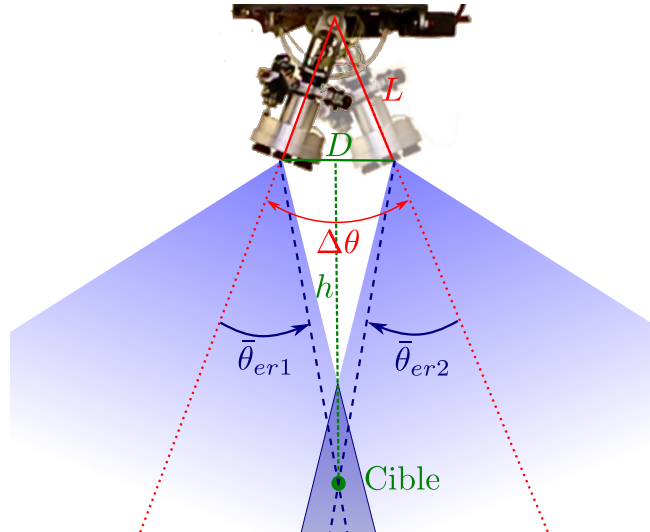


FIGURE V.20 – Principe d'estimation de la distance par rapport à la cible par louvoiement. La distance L étant connue, si on applique un louvoiement, d'amplitude $\Delta\theta$ connu, et que l'on suppose la cible immobile entre les instants séparant les mesures $\bar{\theta}_{er1}$ et $\bar{\theta}_{er2}$. On pourrait, par triangulation, remonter à l'information de distance h par rapport à la cible.

On peut également se poser la question de la complexité de l'EKF et sa plausibilité au sens biologique. Il semble peu probable que la mouche utilise les informations visuelles pour estimer l'ensemble de ses états (position, vitesse, attitude, etc.) Il semblerait plus vraisemblable que des boucles de rétroaction permettent d'asservir directement l'angle entre le cou et le corps, l'asservissement en attitude et en position par rapport à la cible serait alors une conséquence de cet asservissement du regard. Ce genre de contrôleurs pourrait être obtenus en adaptant par exemple les stratégies présentées dans [Rifaï *et al.*, 2013].

Chapitre VI

Conclusion

Sommaire

1	Travail accompli	226
2	Discussions	228
3	Travaux futurs possibles	229
4	Perspectives	229

1 Travail accompli

Durant cette thèse, j'ai donc posé les bases d'une stratégie de guidage visuel, basée sur le regard, qui permet de remonter simultanément à l'attitude et à la position d'un micro robot aérien, sous réserve que l'altitude soit connue. Le but principal de cette thèse était de faire une preuve de concept, concernant la possibilité de stabiliser un robot aérien sans accéléromètre, ni magnétomètre, à l'instar des insectes comme la mouche.

Afin d'atteindre cet objectif, nous nous sommes d'abord intéressé, dans le chapitre II, à la stabilisation d'un simple birotor, afin de démontrer l'intérêt d'introduire un découplage entre la "tête" et le "corps" du robot. Nous avons alors montré, que ce découplage, en plus de présenter des avantages évidents vis-à-vis de la perte de vue de la cible, était sans doute un point clé permettant aux insectes, comme la mouche, de voler sans accéléromètre. Nous avons ensuite énoncé une nouvelle technique de navigation possible, découlant directement de ce système : le robot peut se déplacer de cible en cible, en réalisant de rapides saccades avec son œil, pour changer de cible. Dans ce cas-là, tout ce que fait le robot découle de son regard ; c'est pour cela que l'on appelle ces stratégies : stratégies de guidage par le regard. Puis, toujours dans le cadre des travaux plaçant la stabilisation au cœur des réflexions, nous avons montré que le découplage du regard pouvait être un élément clé pour réaliser des tâches d'odométrie visuelle. Nous avons alors réussi, expérimentalement, à stabiliser un birotor au-dessus d'un panneau texturé. Dans ce cas, le découplage mécanique servait à rendre le système de vision sensible uniquement aux mouvements de translation du robot, étant donné que tous les mouvements liés à l'attitude, étaient annulés par le découplage actif.

Ensuite, dans le chapitre III, j'ai présenté en détail la conception d'une nouvelle plateforme quadrirotor open-source, basé sur Linux. La particularité de cette plateforme étant, qu'elle est entièrement programmable et pilotable, depuis l'environnement Matlab/Simulink. Ce choix découlant de la volonté de n'avoir plus qu'un environnement de développement pour : la simulation, la validation PIL et l'application finale. J'ai alors présenté brièvement la toolbox RT-MaG, qui permet de générer, depuis un modèle Simulink, une application temps réel dédiée à un environnement Linux embarqué. J'ai également présenté en détail, les choix de conception pour la partie logicielle et matérielle, ainsi que les différents modes de secours mis en place pour sécuriser la plateforme, en cas de défaillances matérielles. J'ai ensuite présenté en détail, le choix des divers composants, et rappelé l'importance capitale d'asservir la vitesse de rotation des hélices (ce qui n'est pas encore systématiquement fait dans de nombreux quadrirotors disponibles aujourd'hui.) Puis, j'ai rappelé les principaux phénomènes à prendre en compte lors de la modélisation d'un quadrirotor. J'ai alors pris soin de présenter deux types modèles, un pour la simulation, qui regroupe l'ensemble des phénomènes physiques les plus importants, et un pour le contrôle qui ne regroupe que les plus importants. Ensuite, j'ai comparé deux stratégies de stabilisation du quadrirotor, dans une arène de vol équipée d'un système Vicon. Puis, j'ai sélectionné la meilleure,

pour créer un mode de *secours* automatique, permettant de reprendre la main en cas de défaillance, afin de sécuriser l'expérimentation de nouveaux contrôleurs.

Dans le chapitre IV, j'ai décrit en détail la conception d'un système occulo-moteur bio-inspiré complet, permettant de localiser une cible selon 3 orientations, et de la poursuivre selon 2 directions. Cette œil minimaliste est au final capable de localiser une cible avec une précision inférieure à 0.1° , et ce, à une fréquence allant jusqu'à 1 kHz. Il offre également des capacités de poursuite de cible allant jusqu'à plus de $200^\circ \cdot \text{s}^{-1}$. Le tout avec seulement 24 pixels.

Enfin, dans le chapitre V, j'ai présenté le système complet, à savoir le quadrirotor du chapitre III équipé de l'œil du chapitre IV. J'ai alors montré, étape par étape, comment on pouvait, grâce à l'œil et l'orientation du regard, stabiliser le quadrirotor sans utiliser ni accéléromètre, ni magnétomètre. Au final le robot est capable de décoller, atterrir et même poursuivre efficacement une cible mobile sur la seule base de son œil, d'un gyromètre 3 axes et d'un capteur d'altitude. Nous avons même montré, que le robot était capable d'atterrir sur une cible en mouvement.

Les contributions de cette thèse sont donc multiples et on distinguera les contributions techniques et scientifiques.

Les deux principales contributions techniques de cette thèse sont donc :

- le développement d'une nouvelle plateforme quadrirotor bas coût entièrement open-source et open-hardware.
- une nouvelle toolbox Matlab/Simulink open-source permettant la génération automatique d'application temps réelle pour des cibles Linux embarquées. Cet outil permet de grandement réduire le temps de développement, et conserver le même environnement de développement (Simulink) pour valider l'ensemble des algorithmes de contrôle en simulation, en Processor-In-the-Loop et générer automatiquement l'application temps réel Linux correspondante pour la cible finale.

D'un point de vue scientifique, les contributions de cette thèse sont :

- une extension des travaux précédents sur les capteurs bio-inspirés reproduisant la vision de la mouche et dotés d'hyperacuité :
 - une nouvelle procédure de calibration robuste, permettant une linéarisation parfaite des sorties de chaque paire de pixels, pour une cible donnée. Ceci permet d'agrandir la plage de localisation d'une paire de photo-récepteurs, d'améliorer la qualité de la localisation et ouvre la possibilité de fusion multi-paire,
 - le développement d'un algorithme de fusions multi-paires permettant d'augmenter le champ visuel d'un œil, équipé de plusieurs paires de photo-récepteurs, tout en garantissant la linéarité du capteur. De plus, une nouvelle méthode de sélection de la meilleure paire est proposée (fonction ZSL), ce qui permet également de détecter l'absence de cible,

- le développement complet d'un nouveau système occulo-moteur, permettant la poursuite de cible avec une grande précision et une grande rapidité selon 2 axes de poursuite. Ce système permet, grâce à une combinaison originale de quatre yeux élémentaires et d'une fusion appropriée, d'étendre la localisation de cible du cas 1-D (une dimension) au cas 2-D (localisation de la cible selon deux directions) et de donner une mesure de son orientation (angle de lacet). Le capteur est ainsi capable de localiser précisément la position angulaire d'une cible selon 3 directions : azimut, élévation et orientation de la cible.
- le développement de nouvelles stratégies d'estimation et de stabilisation de micro drone, basées sur l'orientation du regard. L'originalité de ces stratégies étant de ne pas utiliser d'accéléromètres pour estimer l'attitude de l'engin, mais de déduire ces états de l'orientation du regard. A ma connaissance, c'est la première fois qu'un quadrirotor est stabilisé entièrement en position et attitude sans aide d'accéléromètre,
- la première application de robotique volante permettant la poursuite de cibles mouvantes et l'atterrissage sur des cibles en déplacement, n'utilisant comme capteurs que la vision et des gyromètres. On a montré que le découplage tête-corps peut être utilisé, au-delà de la poursuite de cible (comme c'est le but la plupart du temps), pour estimer l'ensemble des états du drone, tout en lui permettant la poursuite de la cible.

2 Discussions

Après lecture de ce manuscrit on peut se poser légitimement la question : "ces stratégies sont-elles biologiquement plausibles?" Le but de la thèse n'était pas d'apporter des réponses formelles, sur comment certains insectes font pour se stabiliser sans accéléromètre. Mais plutôt, "est-il possible de stabiliser un micro-drone avec seulement des capteurs semblables à ceux de l'insecte?" Cette preuve de concept étant faite, il serait intéressant de rechercher des formes de commandes différentes, probablement privées de la partie estimation d'états, qui seraient à même de stabiliser le même système de façon biologiquement crédible.

De même, les stratégies présentées ici nécessitent de connaître la hauteur de vol par rapport à la cible. Cette hauteur pourrait être mesurée comme classiquement dans les applications de robotique aérienne, c'est à dire avec un télémètre, un baromètre, des capteurs à ultra-son, etc. Mais, bien que l'on ne sache pas vraiment comment cette fonction est réalisé chez l'insecte (bien que rien ne prouve qu'il estime sa hauteur de vol, bien au contraire), cette information pourrait également être extraite de différentes manières à partir du capteur visuel, comme expliqué en fin du chapitre [V](#).

3 Travaux futurs possibles

Les travaux de cette thèse apportent encore beaucoup de questions, et de nombreux travaux dans la suite directe, ou non, pourraient encore être menés.

Il resterait bien sûr à :

- Valider expérimentalement la navigation point par point, découlant de saccades de l’œil, comme décrit dans le chapitre II,
- Tester la poursuite de cible sur des trajectoires plus compliquées, incorporant des changements de cap et de position simultanément,
- Rendre plus solide le système occulo-moteur. Actuellement, le système est actionné par des servomoteurs, qui sont sensibles à la casse (étage de réduction fragile) en cas de crash. Il serait intéressant de construire un système Gimbal miniaturisé basé sur des moteurs brushless Gimbal, qui sont beaucoup plus résistants (pas d’étage de réduction, prise direct sur l’axe du moteur), afin de rendre le système plus robuste. De plus cela permettrait de grandement améliorer la précision de l’orientation de la tourelle et donc de faire disparaître les oscillations indésirables, qui découlaient de ce manque de précision lié aux servomoteurs.
- De même, il serait intéressant d’ajouter des capteurs magnétiques ou encodeurs, pour lire la position réelle de la tourelle, et ne pas l’estimer à partir des commandes envoyées aux servomoteurs. Cela permettrait d’améliorer encore la précision de positionnement du système.

De même, il serait intéressant d’ajouter un découplage selon l’axe de lacet pour la vision, afin de permettre plus tard d’améliorer les performances de poursuite de cible et de mesure de l’angle de lacet.

4 Perspectives

Dans un cadre différent que la robotique bio-inspirée, on pourrait envisager de combiner cette information d’orientation du regard, avec celle des systèmes classiques, basés sur les IMU 9 axes. En effet, cela permettrait d’apporter de la redondance pour l’estimation d’attitude. De plus, l’estimation d’attitude liée au regard, ne souffre pas de distorsions liées aux fortes accélérations latérales, aux forces centrifuges, etc. (qui modifient l’orientation du vecteur de gravité mesuré par les accéléromètres). Cela pourrait donc permettre d’améliorer l’estimation d’attitude lors de certaines phases de vol.

Cela pourrait également permettre d’estimer localement (autour d’une cible donnée), la position du micro-drone. Ceci pourrait permettre d’accroître la précision d’un système de positionnement classique (le GPS par exemple), aux alentours d’une cible connue. Cela pourrait s’avérer particulièrement intéressant, pour des missions nécessitant une précision de positionnement accrue, dans une zone limitée.

Ce système pourrait même être utilisé pour permettre, en cas de perte de signal GPS, de stabiliser le drone malgré tout en se fixant sur un amère visuel, dont la dernière position GPS serait connue (la

position GPS peut être déduite de l'orientation de l'œil et de la position GPS du drone avant la perte GPS). Ainsi, il serait possible de continuer la mission, en se référant à la position de ce dernier amère connu.

Enfin une autre possibilité, est l'appontage automatique de véhicule volant sur des navires. En effet, nous avons montré que ce système était suffisamment fiable pour permettre à un quadrirotor d'atterrir automatiquement sur un robot rover en mouvement. Il serait donc possible d'étendre ce concept à l'atterrissage sur des navires en présence de houle.

Annexe A

Precision of the VICON localization system

1 Introduction

The Vicon's localization system is one of the most used motion tracking system. However, it is difficult to find information about the accuracy and the precision of this system, because these two parameters strongly depend of the cameras' position and configuration. For example, [Windolf *et al.*, 2008] assessed that the accuracy of such a system was about $0.063mm$ in a most favorable configuration with a precision of about $0.015mm$. [Nippon Engineering College in Tokyo, 2002] compared different localization systems and measured an accuracy of $0.24mm$.

In order to analyze the results obtained with various airborne and ground robots in our flying arena, it was necessary to assess the precision of the Vicon's localization system.

First, it is proposed to define the difference between *precision* and *accuracy*. The detailed definition of these two terms available at [Grubbs, 1973] or [Joint Committee for Guides in Metrology, 2012] and can be summarized as follows :

The *accuracy* of a measurement system is the "closeness of agreement between a measured quantity value and a true quantity value of a measurand." (from [Joint Committee for Guides in Metrology, 2012])

The *precision* of a measurement system is "the closeness of agreement between indications or measured quantity values obtained by replicate measurements on the same or similar objects under specified conditions." (from [Joint Committee for Guides in Metrology, 2012])

According to these definitions, assessing the *accuracy* of the Vicon System may be quite a difficult task because it requires using another accurate measurement system. For example, even if one used a simple accurate measuring tape for this purpose, many precautions and complex procedures would be

necessary because the measurements would have to be carried out in a large 3-D space; whereas the *precision* of the Vicon localization system can be assessed much more easily. It was therefore proposed in this study to determine the *precision* of the Vicon system used in our flying arena rather than the *accuracy* of this system.

2 Procedure and conditions

A given object was placed at various points in the flying arena for 30 seconds, and the data obtained with the Vicon system during this time were recorded. The structure holding the camera was motorized and could be moved up and down to adapt the "working volume" of the arena as required. All the results presented here were obtained with the maximum working volume ($6m \times 6mm$). The mean position measured, the standard deviation of the measurements, and the maximum deviation of each position in the arena were then calculated.

The object used in these tests was the "Wand" which is also a calibration tool provided by *Vicon Motion Systems Ltd* and endowed with IR reflectors. This object is shown in the figure A.1 in various position during the procedure.

3 The metrics

The precision was assessed here using two different metrics, the **standard deviation** and the **maximum deviation**. The values of these parameters were determined at each of the Wand positions tested, and then globally. Complete volume and overall values were obtained using the following procedure : P series of measurements were performed for δt -seconds each with the Vicon system at a frequency f_s . Each position is denoted p_k , where $k \in \mathbb{N}$ and $1 \leq k \leq P$. At each position p_k , we have N measurements denoted $x_{i,k}$ (where $N = \delta t \cdot f_s$), giving a vector with the 3 coordinates $x_{i,k} = (X_{i,k} \ Y_{i,k} \ Z_{i,k})^T$. At each position p_k , the mean value $E(p_k)$, the standard deviation $\sigma(p_k)$ and the maximum deviation measured $\epsilon_{max}(p_k)$, are defined as follows :

$$E(p_k) = \frac{1}{N} \sum_{i=1}^N x_{i,k} \quad (\text{A.1})$$

$$\sigma(p_k) = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_{i,k} - E(p_k))^2} \quad (\text{A.2})$$

$$\epsilon_{max}(p_k) = \max (|x_{1,k} - E(p_k)|, |x_{2,k} - E(p_k)|, \dots, |x_{N,k} - E(p_k)|) \quad (\text{A.3})$$

By extension, the overall standard deviation, denoted $E(\sigma)$, can be defined as the mean standard deviation of all the positions. The overall maximum deviation, denoted ϵ_{MAX} , can also be defined,

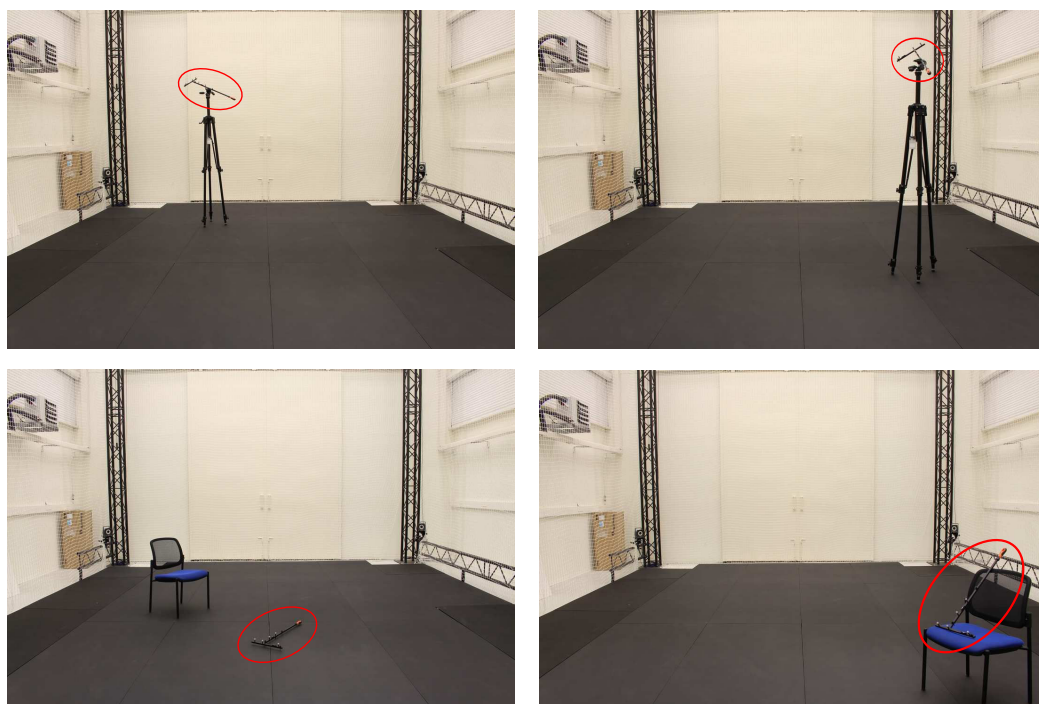


FIGURE A.1 – In order to assess the precision of the Vicon system, a given object (the "Wand" used to calibrate the system) was placed in various positions. Each of the "Wand" positions measured by the motion capture Vicon's system was recorded for 30 seconds. The "Wand" was placed in 9 different positions, each of which was repeated at 4 different heights, making 36 measurements in all, covering practically the whole volume of the flying arena. The Wand (surrounded by a red circle) is shown here at various points in the arena and at various heights.

corresponding to the largest value of all the maximum deviations :

$$E(\sigma) = \frac{1}{P} \sum_{k=1}^P \sigma(p_k) \quad (\text{A.4})$$

$$\epsilon_{MAX} = \max(\epsilon_{max}(p_1), \epsilon_{max}(p_2), \dots, \epsilon_{max}(p_P)) \quad (\text{A.5})$$

4 Results

The above procedure was applied to 36 different positions (9 positions at 4 different heights, covering a large part of the whole volume of the flying arena). The data obtained at each position were recorded for 30 seconds at a frequency of 500Hz. The arena was equipped with 17 T-40s cameras arranged as shown in the figure A.2. Before the experiment started, the calibration was repeated until an "image error" of less than 0.35 pixels was reached with each camera (in the Tracker 2.0 program), i.e., until the position of a marker's centroid could be measured with a precision greater than 0.35 pixels in the case of each camera (see figure A.2).

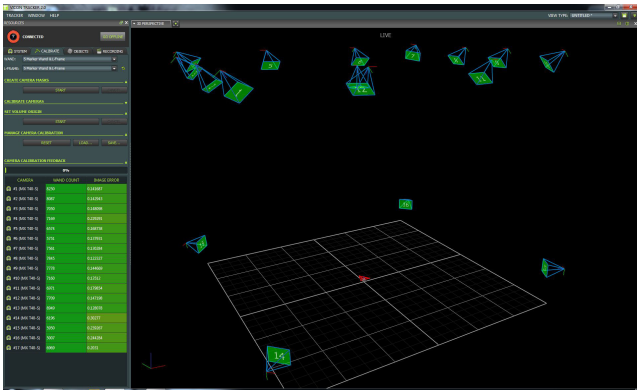


FIGURE A.2 – Screenshot of the Tracker software just before the experiment. The precision was assessed by the Tracker system in terms of the "image error", which is the estimated camera's positioning error in pixels. The use of multiple cameras makes it possible to achieve a sub-pixel resolution. Before all the experiments conducted in the arena, we checked that the calibration yielded an "image error" of less than 0.35 pixels.

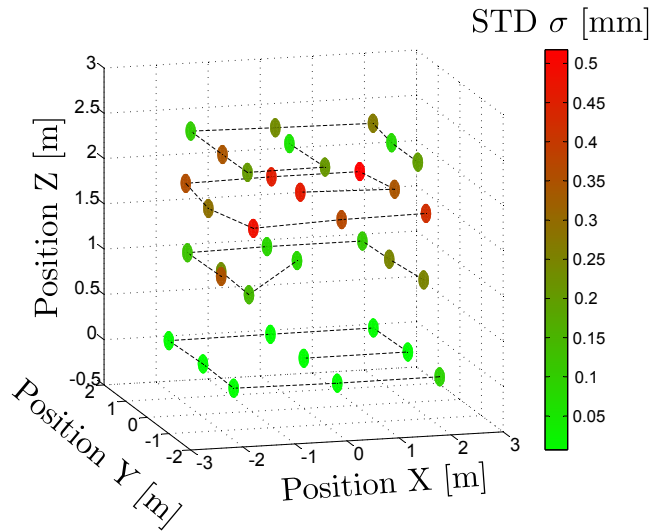


FIGURE A.3 – Diagram of the precision depending on the position of the object. The standard deviation given here is that of the "worst" of the X, Y or Z axes.

The figure A.4a gives the Wand's position measured versus time in the 36 positions. The time series of each position is delimited using two 'x' markers. The figure A.4b is a zoom showing one of the time series, with the corresponding standard deviation (blue shaded area), the mean position (black line) and the maximum deviation (red circle).

The table A.1 gives the standard deviations and maximum deviations measured at 6 different positions (chosen at random among the 36 positions tested). The values of these two parameters in the other positions were also computed, but have not been presented here for the sake of clarity. The precision varied because the number of cameras able to detect the object varied, depending on the object's position p_k . This situation is also illustrated in the figure A.3, where colored spheres indicate the precision at a given position. The lowest precision was observed at a height of about 1.8 m because the four cameras located at the bottom of the arena could not detect the object, which was also relatively far from the camera located at the top of the arena.

The table A.2 gives the minimum standard deviation (the greatest precision), the maximum standard deviation (the lowest precision) and the mean standard deviation (the average precision with the volume tested) obtained on each axis using the data recorded on the 36 different positions. These values reflect the global standard deviation, the global maximum standard deviation and global maximum deviation. It can be seen from this table that the maximum standard deviation was about $0.46mm$ on the X-axis, $0.52mm$ on the Y axis and $0.19mm$ on the Z-axis. The mean standard deviation measured was about $0.14mm$ on the X-axis, $0.15mm$ on the Y-axis and $0.07mm$ on the Z-axis. In both cases, it

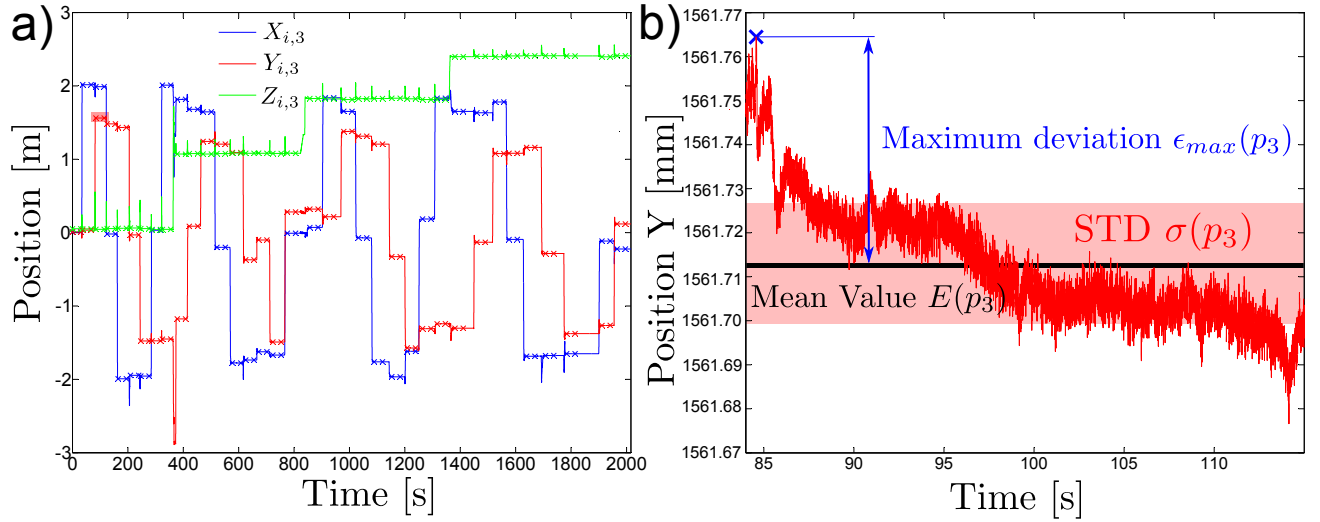


FIGURE A.4 – a) A recording of the 36 different positions delimited by 'x' markers on which the metrics were computed. The positions on the axes X, Y and Z are plotted in blue, red and green respectively. Peaks between two different positions in the recordings were due to the manual displacement of the Wand. b) Zoom corresponding to the measurements recorded at the position p_3 (corresponding to the red shaded area in figure a). The red line gives the position measured by the Vicon system with time, and the red shaded area corresponds to the standard deviation $\sigma(p_3)$. The blue crosses correspond to the maximum deviation $\epsilon_{max}(p_3)$ measured with respect to the mean position $E(p_3)$, which is plotted in the form of a black line.

a)	Standard deviation $\sigma(p_k)$ [mm]					
	p_{34}	p_{17}	p_{29}	p_5	p_{15}	p_{33}
X	0.1098	0.0970	0.0569	0.0148	0.1916	0.3442
Y	0.2063	0.1457	0.0816	0.0105	0.3536	0.0251
Z	0.1037	0.0580	0.0708	0.0052	0.0992	0.0988
b)	Maximum measured deviation $\epsilon_{max}(p_k)$ [mm]					
	p_1	p_2	p_3	p_4	p_5	p_6
X	0.2791	0.2554	0.1267	0.0755	0.5252	0.8023
Y	0.5453	0.3382	0.1727	0.0548	0.8859	0.0452
Z	0.2395	0.1313	0.1759	0.0231	0.3424	0.2236

TABLE A.1 – The standard deviation and maximum deviation measured at 6 of the 36 positions tested. a) The standard deviation (in millimetres) on each of the X, Y and Z axes. b) The maximum deviation measured with respect to the mean value (in millimetres) on each of the X, Y and Z axes.

can be reasonably assumed that the precision was greater than $1mm$, i.e., that the variations in the measurements (the precision) were in the sub-millimetric range.

	Min STD $\min(\sigma(p_k))$ [mm]	Max STD $\max(\sigma(p_k))$ [mm]	Mean STD $E(\sigma)$ [mm]	Max deviation ϵ_{MAX} [mm]
X	0.0025	0.46209	0.1375	1.1187
Y	0.0033	0.5173	0.1492	1.3057
Z	0.0042	0.1889	0.0714	0.3653

TABLE A.2 – The minimum standard deviation observed in the arena was found to be about $0.0025mm$, which was probably greatly underestimated; this value cannot be valid because it is well below the *accuracy* ("trueness") of the Vicon system. The maximum standard deviation observed in the arena was about $0.52mm$, which was highly accurate. The mean standard deviation observed can be said to be less than $0.2mm$. The maximum deviation measured was less than $1.5mm$ ($1.31mm$), which means that the position of the object was given by the Vicon system with a precision of $\pm 0f1.5mm$ in the worst case.

4.1 Discussion

Based on the results presented above, it can be concluded that the greatest positioning error observed here (in terms of maximum deviation measured with respect to the mean value) amounted to less than $1.5mm$. i.e., the lowest precision of the measurements obtained was $\pm 1.5mm$. A parameter more commonly used to assess the precision is the standard deviation. The mean precision of the Vicon system observed in the flying arena in its largest configuration was as small as $0.2mm$ in terms of the mean standard deviation and $0.6mm$ in terms of the greatest standard deviation (the worst case). In both cases, it can be assumed that the precision of the system tested was in the sub-millimetric range.

These results obviously depended on the positions of the cameras, the quality of the calibration and several other parameters. For example, all the cameras were supported by a motorized metallic structure which made it possible to adapt the "working" space. Since the metallic structure was relatively large, its position necessarily depended on the current temperature of the arena. To deal with this problem, the temperature in the arena was regulated during the experiments, and calibrations were performed at least twice a day (or more frequently if the precision was affected). In addition, the required level of precision could presumably be achieved only if some experimental conditions were met and the calibration was performed properly by repeating it until a maximum error of less than 0.35 pixels was reached in the *Tracker* program (see figure A.2). In short, a satisfactory level of precision can be achieved with this set-up only if the following conditions are met :

- The temperature must not change during the experiment ($\pm 1^\circ$),
- The calibration must have a precision of at least 0.35 pixels in the worst case (in that of the least well calibrated camera),
- The ambient light must be kept constant, especially in the infra-red band : the awnings/roller blinds of the arena were kept down during the present experiments to prevent the sunlight from disturbing the measurements.

As the levels of precision given in this paper were obtained with the largest configuration of the arena, it can be reasonably assumed that better (or at least equivalent) performances would be obtained with smaller configurations. In conclusion, the precision of our Vicon system, composed by 17 cameras (T40S), as a means of localizing objects in our flying arena can be said to be in the sub-millimetre range ($0.6mm$ in the case of the largest standard deviation).

Annexe B

Influences du système Vicon et des Néons sur l'œil

On propose ici de quantifier l'influence du système de capture de mouvement Vicon et de l'éclairage artificiel des néons qui sont tout deux des systèmes pulsés. Ces deux systèmes vont donc venir moduler le signal des photodiodes en plus de la vibration active de l'œil. Il est donc important de vérifier que leur influence reste négligeable d'un point de vue de la mesure finale. On s'intéresse donc aux trois cas suivants :

- sans Néons et sans Vicon,
- avec Néons et sans Vicon,
- avec Néons et avec Vicon,

Pour un éclairage naturel (fenêtres ouvertes et une ambiance lumineuse naturelle d'environ 500 lx), le bruit sur les signaux photodiodes est de l'ordre de ± 50 et de $\pm 0.2^\circ$ sur la mesure finale (figure [B.1b](#)). En présence des néons seuls, on observe que le niveau de bruit photodiodes reste presque inchangé (± 75), ce qui est dû au filtrage sélectif. On ne retrouve alors presque pas de composante fréquentielle liée aux néons dans le spectre des signaux photodiodes (figure [B.1a](#)). Le fait que le bruit sur la mesure de position soit moins important ($\pm 0.1^\circ$) s'explique par le fait que l'apport de bruit lié au 100 Hz est négligeable devant l'apport de luminosité (500 lx de plus que pour une lumière naturelle). En effet, plus de luminosité implique une intensité de contraste plus importante et donc un meilleur rapport signal sur bruit (amplitude de l'enveloppe du signal sur l'amplitude de bruit). Ainsi, la luminosité étant multipliée par 2, et le bruit photodiode multiplié par à peine 1.5, on obtient un RSB 1.8 fois supérieur.

En revanche, on observe que la présence du Vicon est beaucoup plus contraignante. Le niveau de bruit atteint ± 350 sur les signaux photodiodes et présente des modulations d'amplitude "très" basse fréquence (2 Hz) se trouvant dans la même plage fréquentielle que les signaux "utiles" (voir figure haute [B.1d](#)).

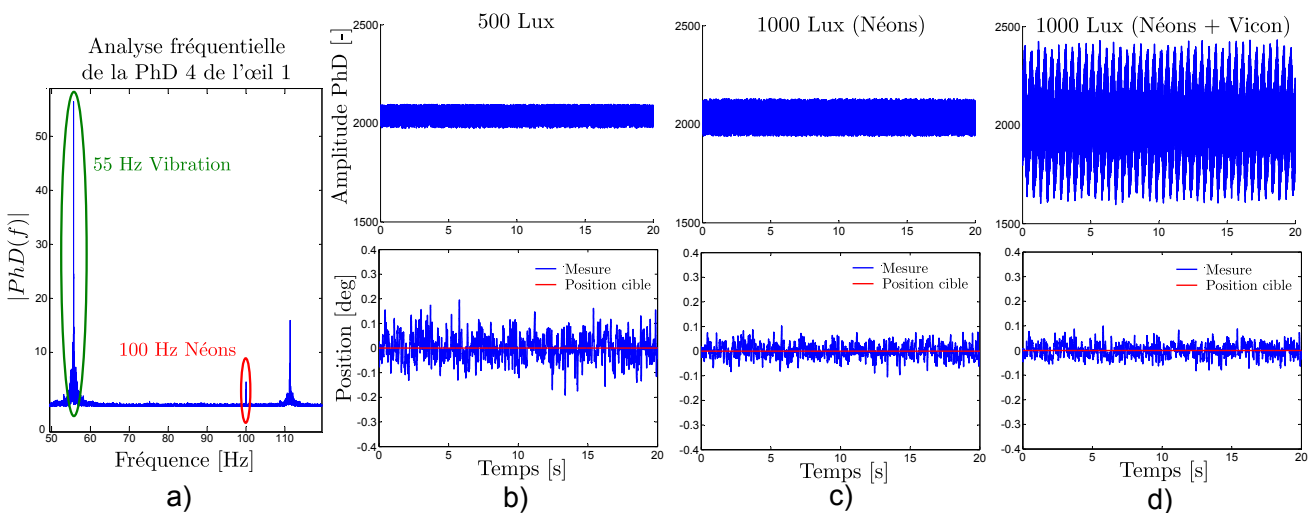


FIGURE B.1 – a) Analyse fréquentielle des signaux photodiodes en présence de néons. b) En haut, niveau de bruit ambiant pour une photodiode ne voyant "rien". En bas, niveau de bruit sur la mesure finale fusionnée pour une ambiance lumineuse naturelle de 500 lx. c) En haut, niveau de bruit ambiant pour une photodiode ne voyant "rien". En bas, niveau de bruit sur la mesure finale fusionnée pour une ambiance lumineuse artificielle aux néons de 1000 lx. d) En haut, niveau de bruit ambiant pour une photodiode ne voyant "rien". En bas, niveau de bruit sur la mesure finale fusionnée pour une ambiance lumineuse artificielle aux néons de 1000 lx et la présence de Vicon.

Le phénomène de bruit basse fréquence observé en présence du système Vicon s'explique par la légère différence existant entre l'horloge du système Vicon et l'horloge de l'œil. En effet, le système Vicon est constitué de 17 caméras sensibles à l'infrarouge et d'une couronne de LED infrarouge (appelée *strobe*), dont l'intensité est modulée à 500 Hz. Cette modulation d'intensité caractéristique est utilisée pour repérer et localiser les objets munis de marqueurs dans l'arène, mais dans notre cas, elle est également captée par l'œil qui est fortement sensible au domaine infrarouge. La numérisation des signaux photodiodes est effectuée ici à 1 kHz, mais le cadencement de cette "tâche" dépend de l'horloge de l'œil qui est légèrement différente de celle du système Vicon (ces deux horloges étant indépendantes). L'échantillonnage "glisse" donc le long du signal de modulation du Vicon, comme le montre les figures B.2c-d, entraînant alors une modulation d'amplitude "parasite", dont la fréquence dépend du décalage des horloges. Dans notre cas, l'étude de la fréquence du signal "parasite", ainsi que de son enveloppe, nous permettent de déterminer que la modulation (pulsation) du Vicon est de forme triangulaire avec un rapport cyclique¹ de 0.73 et une horloge différant de 0.2% de celle du système d'échantillonnage de l'œil.

En général, une façon d'éviter ce genre de "couplage" entre le système d'échantillonnage et les bruits périodiques environnants, est d'échantillonner "bien plus rapidement" que les fréquences parasites

1. Le rapport cyclique pour un signal triangulaire est défini comme le ratio de la durée de "montée" du signal sur la durée de "descente".

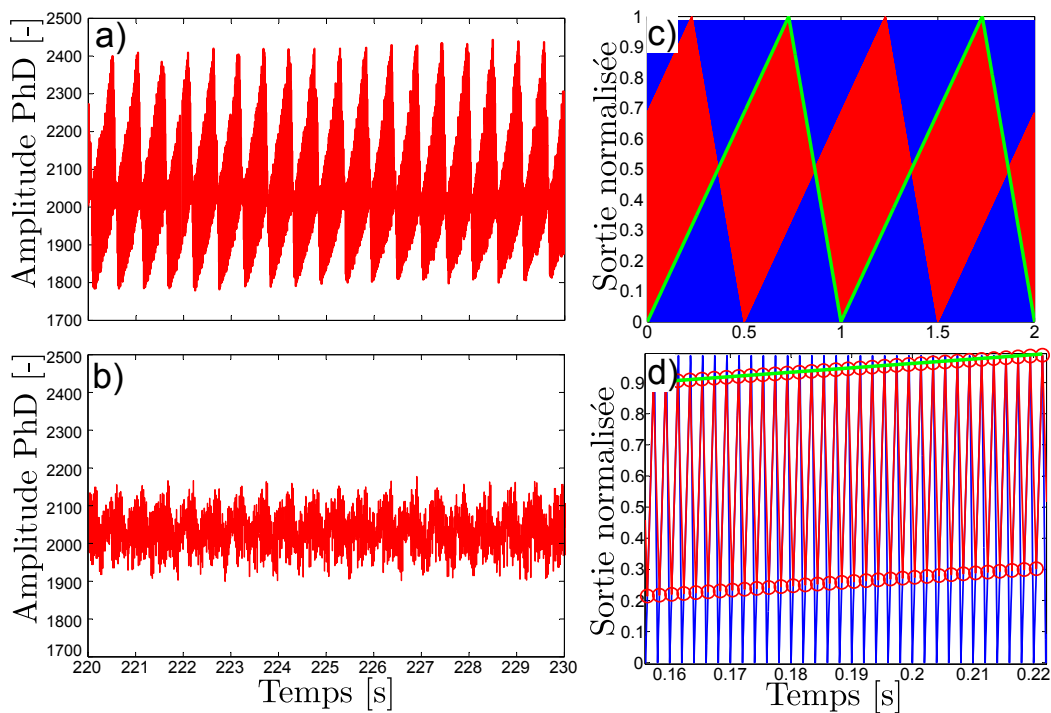


FIGURE B.2 – a) Exemple de réponse d'une des 6 photodiodes de l'œil lorsque le système Vicon est activé. Le système Vicon fait donc apparaître un bruit d'amplitude crête à crête d'environ 600 (sur un total de 4096) soit environ 14% du signal utile. Ce bruit présente une enveloppe "triangulaire" de fréquence faible d'environ 2 Hz. b) Les mêmes signaux photodiodes mais avec les rotors actifs (en rotation). Le bruit provoqué par le Vicon est fortement atténué par les vibrations dues à la rotation des hélices. c) Simulation des signaux photodiodes avec une fréquence d'impulsion du système Vicon de 500 Hz et un échantillonnage des signaux PhD effectué à 998 Hz (on échantillonne à 1 kHz sur l'œil dont l'horloge diffère de 0.2% de celle du système Vicon, ce qui donne 998 Hz dans le repère fréquentiel du Vicon.) En bleu le signal continu représentant la modulation analogique des impulsions réalisées par le système Vicon. En rouge ce même signal une fois échantillonné à une fréquence de 998 Hz par les ADC des photodiodes. En vert l'enveloppe du bruit résultant du décalage des horloges du Vicon et du système de numérisation. d) Zoom sur 60 ms, mettant en évidence le décalage progressif lié à la différence d'horloge donnant lieu au bruit basse fréquence. En bleu, le signal analogique simulé des impulsions du Vicon, en rouge le signal échantillonné par l'œil où chaque cercle correspond à un échantillon. En vert, l'enveloppe visible également sur la figure c).

présentes (en pratique un rapport 5 ou 10 est souvent suffisant). Ici, cela impliquerait d'échantillonner le signal à 5 kHz, ce qui est très élevé. Ceci peut être évité grâce au traitement appliqué aux signaux photodiodes qui permet de fortement réduire l'influence de ce bruit (voir la partie 3.1). En effet, le filtre sélectif permet d'éliminer une bonne partie de ce bruit. De plus, on remarque que le bruit, issu du système Vicon mesuré par chacune des photodiodes, est identique et synchrone. Or, la sortie du capteur est obtenue en effectuant la différence sur la somme de l'enveloppe des signaux de deux photodiodes voisines. Ainsi, lors de l'opération de différence, l'influence de ce bruit disparaît naturellement (au bruit électronique près). De plus, en observant la figure B.2b, on remarque que lorsque les hélices du

robot sont en rotation, les vibrations "parasites" qui sont générées, atténuent encore un peu plus le bruit lié au Vicon. En effet, l'amplitude des modulations "parasites" est fortement réduite sur la figure [B.2b](#) (hélices en rotation) par rapport à la figure [B.2a](#) (hélices arrêtées). Enfin, la fusion "multi-œil" permet encore de limiter une peu cette influence, car la mesure de position selon un axe est obtenue en faisant la moyenne des sorties des deux yeux liés cet axe. Cela explique donc que, malgré le fait que ce bruit semble d'amplitude très importante sur les signaux photodiodes bruts, il n'ai quasiment aucune influence sur la mesure de position finale, comme le prouve les figures [B.1a](#).

Cela prouve donc l'efficacité et la robustesse du traitement appliqué, permettant d'être très robuste aussi bien à la luminance ambiante qu'aux bruits parasites pouvant découler d'un système de lumière pulsée.

Bibliographie

- [Abdelkrim *et al.*, 2008] Abdelkrim, N., Aouf, N., Tsourdos, A. et White, B. (2008). Robust nonlinear filtering for INS/GPS UAV localization. *In Control and Automation, 2008 16th Mediterranean Conference on*, pages 695–702. (Cité en page 8.)
- [Ahrens *et al.*, 2009] Ahrens, S., Levine, D., Andrews, G. et How, J. (2009). Vision-based guidance and control of a hovering vehicle in unknown, GPS-denied environments. *In Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 2643–2648. (Cité en page 8.)
- [Arkin, 1998] Arkin, R. (1998). *Behavior-based robotics*. MIT press. (Cité en page 12.)
- [Ascending Technologies GmbH, 2011] Ascending Technologies GmbH (2011). Hummingbird and Pelican multicopters. (Cité en page 97.)
- [Ballard, 1991] Ballard, D. H. (1991). Animate vision. *Artificial Intelligence*, 48(1):57–86. (Cité en page 9.)
- [Barton, 2012] Barton, J. D. (2012). Fundamentals of Small Unmanned Aircraft Flight. *Johns Hopkins APL Technical Digest*, 31(2). (Cité en pages 8, 18 et 174.)
- [Bazin *et al.*, 2008] Bazin, J.-C., Kweon, I., Demonceaux, C. et Vasseur, P. (2008). UAV Attitude Estimation by Vanishing Points in Catadioptric Images. *In Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 2743–2749. (Cité en page 173.)
- [Boeddeker et Hemmi, 2009] Boeddeker, N. et Hemmi, J. M. (2009). Visual gaze control during peering flight manoeuvres in honeybees. *Proceedings of the Royal Society B : Biological Sciences*, pages –. (Cité en page 13.)
- [Boeddeker *et al.*, 2003] Boeddeker, N., Kern, R. et Egelhaaf, M. (2003). Chasing a dummy target : smooth pursuit and velocity control in male blowflies. *In Proc. R. Soc. Lond.B 270*, pages 393–399. (Cité en page 13.)
- [Bouabdallah, 2007] Bouabdallah, S. (2007). *Design and control of quadrotors with application to autonomous flying*. Thèse de doctorat, École Polytechnique federale de Lausanne. (Cité en page 58.)
- [Bouabdallah *et al.*, 2004] Bouabdallah, S., Murrieri, P. et Siegwart, R. (2004). Design and control of an indoor micro quadrotor. *In IEEE International Conference on Robotics and Automation, ICRA '04*, volume 5, pages 4393–4398 Vol.5. (Cité en page 58.)

- [Bošnjak *et al.*, 2012] Bošnjak, M., Matko, D. et Blažič, S. (2012). Quadrocopter Hovering Using Position-estimation Information from Inertial Sensors and a High-delay Video System. *J. Intell. Robot. Syst.*, 67(1):43–60. (Cité en page 8.)
- [Bristeau *et al.*, 2009] Bristeau, P.-J., Martin, P., Salaün, E. et Petit, N. (2009). The Role of Propeller Aerodynamics in the Model of a Quadrotor UAV. In *European Control Conference, ECC '09*, pages 683–688, Budapest. (Cité en page 70.)
- [Campbell et Wheeler, 2006] Campbell, M. et Wheeler, M. (2006). A Vision Based Geolocation Tracking System for UAV's. In *Guidance, Navigation, and Control and Co-located Conferences*, pages –. American Institute of Aeronautics and Astronautics. (Cité en page 181.)
- [Campbell et Whitacre, 2007] Campbell, M. et Whitacre, W. (2007). Cooperative Tracking Using Vision Measurements on SeaScan UAVs. *Control Systems Technology, IEEE Transactions on*, 15(4):613–626. (Cité en page 181.)
- [Carpenter, 1988] Carpenter, R. (1988). *Movements of the eye*. Pion London. (Cité en page 25.)
- [Castellanos *et al.*, 2005a] Castellanos, J., Lesecq, S., Marchand, N. et Delamare, J. (2005a). A low-cost air data attitude heading reference system for the tourism airplane applications. In *Sensors, IEEE*, pages 4 pp.–. (Cité en page 18.)
- [Castellanos *et al.*, 2005b] Castellanos, J., Lesecq, S., Marchand, N. et Delamare, J. (2005b). A Low-cost Air Data Attitude Heading Reference System for the Tourism Airplane Applications. In *Sensors, 2005 IEEE*, pages 4 pp.–. (Cité en page 173.)
- [Chiang *et al.*, 2002] Chiang, Y., Wang, L., Chang, F. et Peng, H. (2002). Constrained filtering method for attitude determination using GPS and gyro. *IEE Proceedings - Radar, Sonar and Navigation*, 149:258–264. (Cité en page 174.)
- [Collett et Land, 1975] Collett, T. S. et Land, M. F. (1975). Visual control of flight behaviour in the hoverfly *Syrpitta pipiens* L. *J. Comp. Physiol.*, 99(1):1–66. (Cité en page 13.)
- [Colonnier *et al.*, 2015a] Colonnier, F., Manecy, A., Juston, R., Mallot, H., Leitel, R., Floreano, D. et Viollet, S. (2015a). A small-scale hyperacute compound eye featuring active eye tremor : application to visual stabilization, target tracking, and short-range odometry. *Bioinspiration & Biomimetics*, 10(2):026002. (Cité en pages 4, 32, 33, 34, 184 et 185.)
- [Colonnier *et al.*, 2015b] Colonnier, F., Manecy, A., Juston, R. et Viollet, S. (2015b). Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye. In Wilson, S. P., Verschure, P. F., Mura, A. et Prescott, T. J., éditeurs : *Biomimetic and Biohybrid Systems*, volume 9222 de *Lecture Notes in Computer Science*, pages 153–163. Springer International Publishing. (Cité en page 3.)
- [Colonnier *et al.*, 2015c] Colonnier, F., Manecy, A., Juston, R. et Viollet, S. (2015c). Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye. In *4th inter-*

- national conference on biomimetics and biohybrid systems, Living Machines*, La Pedrera Barcelona, Spain. (Cité en page 4.)
- [Crassidis *et al.*, 2007] Crassidis, J. L., Markley, F. L. et Cheng, Y. (2007). Survey of Nonlinear Attitude Estimation Methods. *Journal of Guidance, Control, and Dynamics*, 30(1):12–28. (Cité en page 173.)
- [Cronin *et al.*, 2014] Cronin, T. W., Johnsen, S., Marshall, N. J. et Warrant, E. J. (2014). *Visual Ecology*. Princeton University Press. (Cité en page 104.)
- [de Croon *et al.*, 2009] de Croon, G., de Clercq, K., Ruijsink, R., Remes, B. et de Wagter, C. (2009). Design, aerodynamics, and vision-based control of the DelFly. *International Journal of Micro Air Vehicles*, 1(2):71–97. (Cité en page 100.)
- [Demonceaux *et al.*, 2006] Demonceaux, C., Vasseur, P. et Pégard, C. (2006). Omnidirectional vision on UAV for attitude computation. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2842–2847. (Cité en page 182.)
- [Dobrokhodov *et al.*, 2006] Dobrokhodov, V., Kammer, I., Jones, K. et Ghabcheloo, R. (2006). Vision-based tracking and motion estimation for moving targets using small UAVs. In *American Control Conference, 2006*, pages 6 pp.–. (Cité en page 181.)
- [Dusha *et al.*, 2007] Dusha, D., Boles, W. W. et Walker, R. (2007). Fixed-Wing Attitude Estimation Using Computer Vision Based Horizon Detection. In *12th Australian International Aerospace Congress*, pages 1–19, Melbourne Australia. (Cité en page 181.)
- [Ettinger *et al.*, 2003] Ettinger, S. M., Nechyba, M. C., Ifju, P. G. et Waszak, M. (2003). Vision-guided flight stability and control for micro air vehicles. *Advanced Robotics*, 17(7):617–640. (Cité en page 182.)
- [Expert et Ruffier, 2012] Expert, F. et Ruffier, F. (2012). Controlling Docking, Altitude and Speed in a Circular High-Roofed Tunnel thanks to the Optic Flow. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1125–1132. (Cité en page 100.)
- [Expert et Ruffier, 2015] Expert, F. et Ruffier, F. (2015). Flying over uneven moving terrain based on optic-flow cues without any need for reference frames or accelerometers. *Bioinspiration & Biomimetics*, 10(2):026003. (Cité en page 100.)
- [Fahle, 1991] Fahle, M. (1991). Psychophysical measurement of eye drifts and tremor by dichoptic or monocular vernier acuity. *Vision Research*, 31(2):209 – 222. (Cité en page 107.)
- [Ficheux et Bénard, 2012] Ficheux, P. et Bénard, E. (2012). *Linux embarqué, 4^{ème} édition*. Eyrolles. (Cité en page 83.)
- [Floreano *et al.*, 2013] Floreano, D., Pericet-Camara, R., Viollet, S., Ruffier, F., Brückner, A., Leitel, R., Buss, W., Menouni, M., Expert, F., Juston, R., Dobrzynski, M. K., L’Eplattenier, G., Recktenwald, F., Mallot, H. A. et Franceschini, N. (2013). Miniature curved artificial compound eyes. *Proceedings of the National Academy of Sciences*, 110(23):9267–9272. (Cité en pages 32 et 100.)

- [Franceschini, 1999] Franceschini, N. (1999). *Cerveaux et Machines*, chapitre De la mouche au biorobot : reconstruire pour mieux comprendre, pages 247–270. V. Bloch, Hermès Science Publications. (Cité en page 106.)
- [Franceschini et Chagneux, 1997] Franceschini, N. et Chagneux, R. (1997). Repetitive scanning in the fly compound eye. *In Göttingen Neurobiology Report*, volume 2. Thieme. (Cité en pages 107 et 108.)
- [Franceschini et Kirschfeld, 1971] Franceschini, N. et Kirschfeld, K. (1971). Etude optique in vivo des éléments photorécepteurs dans l’œil composé de *Drosophila*. *Biological Cybernetics*, 8(1):1–13. (Cité en pages 104 et 106.)
- [Geurten *et al.*, 2010] Geurten, B. R. H., Kern, R., Braun, E. et Egelhaaf, M. (2010). A syntax of hoverfly flight prototypes. *J. Exp. Biol*, 213(14):2461–2475. (Cité en page 13.)
- [Ghadiok *et al.*, 2012] Ghadiok, V., Goldin, J. et Ren, W. (2012). On the design and development of attitude stabilization, vision-based navigation, and aerial gripping for a low-cost quadrotor. *Auton. Robot.*, 33:41–68. 10.1007/s10514-012-9286-z. (Cité en page 16.)
- [Grubbs, 1973] Grubbs, F. E. (1973). Errors of Measurement, Precision, Accuracy and the Statistical Comparison of Measuring Instruments. *Technometrics*, 15(1):53–66. (Cité en page 231.)
- [Gurdan *et al.*, 2007] Gurdan, D., Stumpf, J., Achtelik, M., Doth, K.-M., Hirzinger, G. et Rus, D. (2007). Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz. *In Robotics and Automation, 2007 IEEE International Conference on*, pages 361–366. (Cité en pages 8, 9 et 174.)
- [Hamel et Mahony, 2006] Hamel, T. et Mahony, R. (2006). Attitude estimation on $SO(3)$ based on direct inertial measurements. *In Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2170–2175. (Cité en page 173.)
- [Hamel *et al.*, 2002] Hamel, T., Mahony, R., Lozano, R. et Ostrowski, J. (2002). Dynamic modelling and configuration stabilization for an X4-flyer. *15th IFAC World Congress*, 15:846–846. (Cité en pages 58 et 63.)
- [Hateren et Schilstra, 1999] Hateren, J. et Schilstra, C. (1999). Blowfly flight and optic flow. II. Head movements during flight. *J. Exp. Biol*, 202(11):1491. (Cité en page 13.)
- [Heisenberg, 1971] Heisenberg, M. (1971). Separation of Receptor and Lamina Potentials in the Electretinogram of Normal and Mutant *Drosophila*. *Journal of Experimental Biology*, 55:85–100. (Cité en page 107.)
- [Hengstenberg, 1972] Hengstenberg, R. (1972). Eye Movements in the Housefly *Musca domestica*. *In* Wehner, R., éditeur : *Information Processing in the Visual Systems of Arthropods*, pages 93–96. Springer Berlin Heidelberg. (Cité en pages 107 et 108.)
- [Hengstenberg, 1988] Hengstenberg, R. (1988). Mechanosensory control of compensatory head roll during flight in the blowfly *Calliphora erythrocephala* Meig. *Journal of Comparative Physiology A*, 163(2):151–165. (Cité en pages 13 et 182.)

- [Herissé *et al.*, 2012] Herissé, B., Hamel, T., Mahony, R. et Russotto, F.-X. (2012). Landing a VTOL Unmanned Aerial Vehicle on a Moving Platform Using Optical Flow. *Robotics, IEEE Transactions on*, 28(1):77–89. (Cité en page 8.)
- [Hoffmann *et al.*, 2010] Hoffmann, F., Goddemeier, N. et Bertram, T. (2010). Attitude estimation and control of a quadcopter. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 1072–1077. (Cité en page 173.)
- [Horridge, 1977] Horridge, G. A. (1977). The compound eye of insects. *Scientific American*, 237:108–120. (Cité en pages 104 et 105.)
- [Huang *et al.*, 2009] Huang, H., Hoffmann, G., Waslander, S. et Tomlin, C. (2009). Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering. In *IEEE International Conference on Robotics and Automation, ICRA '09*, pages 3277–3282. (Cité en pages 58, 61 et 68.)
- [Jakobsen et Johnson, 2005] Jakobsen, O. et Johnson, E. (2005). Control Architecture for a UAV-Mounted Pan/Tilt/Roll Camera Gimbal. In *Infotech@Aerospace Conferences*, pages –. American Institute of Aeronautics and Astronautics. (Cité en page 181.)
- [Johnson, 1994] Johnson, W. (1994). *Helicopter Theory*. Dover Books on Aeronautical Engineering Series. Dover Publications. (Cité en page 61.)
- [Joint Committee for Guides in Metrology, 2012] Joint Committee for Guides in Metrology (2012). *International Vocabulary of Metrology - Basic and General Concepts and Associated Terms*, volume JCGM 200 :2012. VIM 3rd édition. (Cité en page 231.)
- [Juston, 2013] Juston, R. (2013). *De l'œil élémentaire à l'œil composé artificiel : application à la stabilisation visuelle en vol stationnaire*. Thèse de doctorat, Université d'Aix-Marseille. (Cité en pages 103 et 130.)
- [Juston *et al.*, 2013] Juston, R., Kerhuel, L., Franceschini, N. et Viollet, S. (2013). Hyperacute Edge and Bar Detection in a Bioinspired Optical Position Sensing Device. *Mechatronics, IEEE/ASME Transactions on*, PP(99):1–10. (Cité en pages 9, 10 et 13.)
- [Juston *et al.*, 2014] Juston, R., Kerhuel, L., Franceschini, N. et Viollet, S. (2014). Hyperacute Edge and Bar Detection in a Bioinspired Optical Position Sensing Device. *Mechatronics, IEEE/ASME Transactions on*, 19(3):1025–1034. (Cité en pages 103, 129 et 130.)
- [Juston et Viollet, 2012] Juston, R. et Viollet, S. (2012). A miniature bio-inspired position sensing device for the control of micro-aerial robots. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1118–1124. (Cité en page 10.)
- [Juusola et French, 1997] Juusola, M. et French, A. S. (1997). Visual Acuity for Moving Objects in First- and Second-Order Neurons of the Fly Compound Eye. *Journal of Neurophysiology*, 77(3):1487–1495. (Cité en page 107.)

- [Kaps et Schmid, 1996] Kaps, F. et Schmid, A. (1996). Mechanism and possible behavioural relevance of retinal movements in the ctenid spider *Cupiennius salei*. *Journal of Experimental Biology*, 199(Pt 11):2451–2458. (Cité en page 107.)
- [Kendoul *et al.*, 2009] Kendoul, F., Fantoni, I. et Nonami, K. (2009). Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles. *Robot. Auton. Syst.*, 57(6-7):591–602. (Cité en page 8.)
- [Kendoul *et al.*, 2010] Kendoul, F., Fantoni, I. et Nonami, K. (2010). Guidance and nonlinear control system for autonomous flight of minirotorcraft unmanned aerial vehicles. *J. Field. Robot.*, 27(3):311–334. (Cité en page 9.)
- [Kerhuel, 2009] Kerhuel, L. (2009). *Capteurs optiques minimalistes et réflexes oculomoteurs biomimétiques. Applications à la robotique aérienne*. Thèse de doctorat, Information, Systèmes, Structures, Université Montpellier II. (Cité en pages 103, 129, 131, 132 et 141.)
- [Kerhuel *et al.*, 2010] Kerhuel, L., Viollet, S. et Franceschini, N. (2010). Steering by Gazing : An Efficient Biomimetic Control Strategy for Visually Guided Micro Aerial Vehicles. *Robotics, IEEE Transactions on*, 26(2):307–319. (Cité en pages 10, 16 et 143.)
- [Kerhuel *et al.*, 2012] Kerhuel, L., Viollet, S. et Franceschini, N. (2012). The VODKA Sensor : A Bio-Inspired Hyperacute Optical Position Sensing Device. *Sensors Journal, IEEE*, 12(2):315–324. (Cité en pages 9, 10, 13, 103, 108, 109, 121, 129, 130, 132 et 133.)
- [Kim *et al.*, 2010] Kim, J., Kang, M.-S. et Park, S. (2010). Accurate Modeling and Robust Hovering Control for a Quad-rotor VTOL Aircraft. *Journal of Intelligent & Robotic Systems*, 57:9–26. 10.1007/s10846-009-9369-z. (Cité en pages 8, 13 et 174.)
- [Kingston et Beard, 2004] Kingston, D. et Beard, R. (2004). Real-Time Attitude and Position Estimation for Small UAVs Using Low-Cost Sensors. *In AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*. American Institute of Aeronautics and Astronautics. (Cité en page 174.)
- [Kirschfeld et Franceschini, 1969] Kirschfeld, K. et Franceschini, N. (1969). Ein Mechanismus zur Steuerung des Lichtflusses in den Rhabdomeren des Komplexauges von *Musca*. *Kybernetik*, 6:13–22. (Cité en page 106.)
- [Ko *et al.*, 2010] Ko, H., Poletti, M. et Rucci, M. (2010). Microsaccades precisely relocate gaze in a high visual acuity task. *Nature Neuroscience*, 13(12):1549–1553. (Cité en page 107.)
- [Kuiper et Leutscher-Hazelhoff, 1965] Kuiper, J. et Leutscher-Hazelhoff, J. (1965). High-precision repetitive firing in the insect optic lobe and a hypothesis for its function in object location. *Nature*, 206:1158–1160. (Cité en page 107.)
- [Land, 1982] Land, M. (1982). Scanning Eye Movements in a Heteropod Mollusc. *Journal of Experimental Biology*, 96(1):427–430. (Cité en page 107.)

- [Land et Collett, 1997] Land, M. et Collett, T. (1997). *From living eyes to seeing machines*, chapitre A survey of active vision in invertebrates, pages 16–36. Oxford University Press Oxford. (Cité en pages 104 et 106.)
- [Land, 1969] Land, M. F. (1969). Movements of the Retinae of Jumping Spiders (Salticidae : Dendryphantinae) in Response to Visual Stimuli. *Journal of Experimental Biology*, 51(2):471–493. (Cité en page 107.)
- [Land et Nilsson, 2012] Land, M. F. et Nilsson, D.-E. (2012). *Animal eyes*. Oxford Animal Biology Series. Oxford University Press, Oxford. (Cité en page 104.)
- [Lee et al., 2010] Lee, T., Leoky, M. et McClamroch, N. (2010). Geometric Tracking Control of a Quadrotor UAV on SE(3). In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 5420–5425. (Cité en pages 87 et 89.)
- [Lightsey et Madsen, 2003] Lightsey, E. G. et Madsen, J. (2003). Three-Axis Attitude Determination Using Global Positioning System Signal Strength Measurements. *Journal of Guidance, Control, and Dynamics*, 26(2):304–310. (Cité en page 174.)
- [Lim et al., 2012] Lim, H., Park, J., Lee, D. et Kim, H. J. (2012). Build Your Own Quadrotor : Open-Source Projects on Unmanned Aerial Vehicles. *IEEE Robot. Automat. Mag.*, 19(3):33–45. (Cité en page 8.)
- [Liske, 1977] Liske, E. (1977). The influence of head position on the flight behaviour of the fly. *Calliphora erythrocephala*. *Journal of Insect Physiology*, 23(3):375 – 379. (Cité en page 182.)
- [Lo et Pak, 1981] Lo, M. et Pak, W. L. (1981). Light-induced pigment granule migration in the reticular cells of *Drosophila melanogaster*. Comparison of wild type with ERG-defective mutants. *The Journal of general physiology*, 77(2):155–175. (Cité en page 106.)
- [Ma et al., 2013] Ma, K. Y., Chirarattananon, P., Fuller, S. B. et Wood, R. J. (2013). Controlled Flight of a Biologically Inspired, Insect-Scale Robot. *Science*, 340(6132):603–607. (Cité en page 100.)
- [Mafrica et al., 2015] Mafrica, S., Godiot, S., Menouni, M., Boyron, M., Expert, F., Juston, R., Marchand, N., Ruffier, F. et Viollet, S. (2015). A bio-inspired analog silicon retina with Michaelis-Menten auto-adaptive pixels sensitive to small and large changes in light. *Opt. Express*, 23(5):5614–5635. (Cité en page 100.)
- [Mahony et al., 2011] Mahony, R., Euston, M., Kim, J., Coote, P. et Hamel, T. (2011). A non-linear observer for attitude estimation of a fixed-wing unmanned aerial vehicle without GPS measurements. *Transactions of the Institute of Measurement and Control*, 33 (6):699–717. (Cité en page 18.)
- [Mahony et al., 2005] Mahony, R., Hamel, T. et Pflimlin, J.-M. (2005). Complementary filter design on the special orthogonal group SO(3). In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pages 1477–1484. (Cité en page 173.)

- [Mahony *et al.*, 2008] Mahony, R., Hamel, T. et Pfimlin, J.-M. (2008). Nonlinear Complementary Filters on the Special Orthogonal Group. *Automatic Control, IEEE Transactions on*, 53(5):1203–1218. (Cité en pages 16, 18, 86 et 173.)
- [Mahony *et al.*, 2012] Mahony, R., Kumar, V. et Corke, P. (2012). Multirotor Aerial Vehicles : Modeling, Estimation, and Control of Quadrotor. *Robotics Automation Magazine, IEEE*, 19(3):20–32. (Cité en pages 8, 58, 65, 68, 86 et 176.)
- [Manecy, 2014] Manecy, A. (2014). RT-MaG Project, <http://www.gipsa-lab.fr/projet/RT-MaG/>. (Cité en pages 46 et 82.)
- [Manecy *et al.*, 2013a] Manecy, A., Juston, R., Marchand, N. et Viollet, S. (2013a). Decoupling the Eye : A Key toward a Robust Hovering for Sighted Aerial Robots. In *Advances in Aerospace Guidance, N. et Control*, éditeurs : *Chu, Qiping and Mulder, Bob and Choukroun, Daniel and Kampen, Erik-Jan and Visser, Coen and Looye, Gertjan*, pages 317–336. Springer Berlin Heidelberg. (Cité en page 4.)
- [Manecy *et al.*, 2013b] Manecy, A., Juston, R., Marchand, N. et Viollet, S. (2013b). Decoupling the Eye : a Key toward a Robust Hovering for Sighted Aerial Robots. In *EuroGNC 2013, 2nd CEAS Specialist Conference on Guidance, Navigation & Control*, Delft, Netherland. (Cité en pages 4, 20, 22, 28 et 143.)
- [Manecy *et al.*, 2015] Manecy, A., Marchand, N. et Ruffier, Franck and, V. S. (2015). X4-MaG : a Low-Cost Open-Source Micro-Quadrotor and its Linux-based Controller. *International Journal of Micro Aerial Vehicles*, 7(2):89–110. (Cité en pages 4, 98 et 99.)
- [Manecy *et al.*, 2014a] Manecy, A., Marchand, N. et Viollet, S. (2014a). Hovering by Gazing : a Novel Strategy for Implementing Saccadic Flight-Based Navigation in GPS-Denied Environments. *International Journal of Advanced Robotic Systems*, 11(66): . (Cité en page 4.)
- [Manecy *et al.*, 2014b] Manecy, A., Marchand, N. et Viollet, S. (2014b). RT-MaG : An Open-Source SIMULINK Toolbox for Linux-Based Real-Time Robotic Applications. In *Robotics and Biomimetics (ROBIO), 2014 IEEE International Conference on*, pages 173–180. (Cité en pages 4 et 81.)
- [Manecy *et al.*, 2012] Manecy, A., Viollet, S. et Marchand, N. (2012). Bio-inspired hovering control for an aerial robot equipped with a decoupled eye and a rate gyro. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 1110–1117. (Cité en pages 5, 20, 22 et 29.)
- [Marins *et al.*, 2001] Marins, J., Yun, X., Bachmann, E., McGhee, R. et Zyda, M. (2001). An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 4, pages 2003–2011 vol.4. (Cité en page 173.)

- [Martinez-Conde *et al.*, 2013] Martinez-Conde, S., Otero-Millan, J. et Macknik, S. L. (2013). The impact of microsaccades on vision : towards a unified theory of saccadic function. *Nature Reviews Neuroscience*, 14(2):83–96. (Cité en page 107.)
- [Meier *et al.*, 2011] Meier, L., Tanskanen, P., Fraundorfer, F. et Pollefeys, M. (2011). PIXHAWK : A System for Autonomous Flight using Onboard Computer Vision. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2992–2997. (Cité en page 16.)
- [Mellinger et Kumar, 2011] Mellinger, D. et Kumar, V. (2011). Minimum Snap Trajectory Generation and Control for Quadrotors. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2520–2525. (Cité en page 89.)
- [Mellinger *et al.*, 2012] Mellinger, D., Michael, N. et Kumar, V. (2012). Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors. *The International Journal of Robotics Research*, 31(5):664–674. (Cité en page 97.)
- [Michael *et al.*, 2010] Michael, N., Mellinger, D., Lindsey, Q. et Kumar, V. (2010). The GRASP Multiple Micro-UAV Testbed. *Robotics Automation Magazine, IEEE*, 17(3):56–65. (Cité en pages 90 et 97.)
- [Mondragón *et al.*, 2010a] Mondragón, I. F., Campoy, P., Martinez, C. et Olivares-Méndez, M. A. (2010a). Omnidirectional vision applied to Unmanned Aerial Vehicles (UAVs) attitude and heading estimation. *Robotics and Autonomous Systems*, 58(6):809 – 819. Omnidirectional Robot Vision. (Cité en pages 173 et 182.)
- [Mondragón *et al.*, 2010b] Mondragón, I. F., Olivares-Méndez, M. A., Campoy, P., Martínez, C. et Mejias, L. (2010b). Unmanned aerial vehicles UAVs attitude, height, motion estimation and control using visual systems. *Autonomous Robots*, 29:17–34. (Cité en pages 9, 173 et 182.)
- [Mori *et al.*, 2007a] Mori, R., Hirata, K. et Kinoshita, T. (2007a). Vision-based guidance control of a small-scale unmanned helicopter. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2648 –2653. (Cité en page 8.)
- [Mori *et al.*, 2007b] Mori, R., Hirata, K. et Kinoshita, T. (2007b). Vision-based guidance control of a small-scale unmanned helicopter. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2648 –2653. (Cité en page 174.)
- [Munguia et Manecy, 2012] Munguia, R. et Manecy, A. (2012). State estimation for a bio-inspired hovering robot equipped with an angular sensor. In *Electrical Engineering, Computing Science and Automatic Control (CCE), 2012 9th International Conference on*, pages 1–6. (Cité en page 5.)
- [Nalbach, 1993] Nalbach, G. (1993). The halteres of the blowfly *Calliphora*. *J. Comp. Physiol.*, 173:293–300. (Cité en page 13.)
- [Nalbach et Hengstenberg, 1994] Nalbach, G. et Hengstenberg, R. (1994). The halteres of the blowfly *Calliphora*. *J. Comp. Physiol.*, 175:695–708. (Cité en page 13.)

- [Nemra et Aouf, 2010] Nemra, A. et Aouf, N. (2010). Robust INS/GPS Sensor Fusion for UAV Localization Using SDRE Nonlinear Filtering. *Sensors Journal, IEEE*, 10(4):789–798. (Cité en pages 8 et 174.)
- [Nice, 2004] Nice, E. B. (2004). *Design of a four rotor hovering vehicle*. Thèse de doctorat, Cornell University. (Cité en page 58.)
- [Nippon Engineering College in Tokyo, 2002] Nippon Engineering College in Tokyo (2002). Results :Basic measurement accuracy, and processing time. (Cité en page 231.)
- [Niven, 2006] Niven, J. E. (2006). Visual motion : homing in on small target detectors. *J. Curr. Biol.*, 16(8):R292–R294. (Cité en page 13.)
- [Nordström *et al.*, 2006] Nordström, K., Barnett, P. D. et O’Carroll, D. C. (2006). Insect detection of small targets moving in visual clutter. *PLOS Biology*, 4(3):e54. (Cité en page 13.)
- [Ozawa et Chaumette, 2011] Ozawa, R. et Chaumette, F. (2011). Dynamic Visual Servoing with Image Moments for a Quadrotor Using a Virtual Spring Approach. *In Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5670–5676. (Cité en page 9.)
- [Peng *et al.*, 2000] Peng, H., Chiang, Y.-T., Chang, F. R. et Wang, L.-S. (2000). Maximum-likelihood-based filtering for attitude determination via GPS carrier phase. *In Position Location and Navigation Symposium, IEEE 2000*, pages 480–487. (Cité en page 174.)
- [Pounds *et al.*, 2006] Pounds, P., Mahony, R. et Corke, P. (2006). Modelling and control of a quadrotor robot. *In Australasian Conference on Robotics and Automation 2006*, Auckland, New Zealand. Australian Robotics and Automation Association Inc. (Cité en page 58.)
- [Pounds *et al.*, 2010] Pounds, P., Mahony, R. et Corke, P. (2010). Modelling and control of a large quadrotor robot. *Control Engineering Practice*, 18(7):691 – 699. Special Issue on Aerial Robotics. (Cité en page 58.)
- [Preuss et Hengstenberg, 1992] Preuss, T. et Hengstenberg, R. (1992). Structure and kinematics of the prosternal organs and their influence on head position in the blowfly *Calliphora erythrocephala* Meig. *J. Comp. Physiol.*, 171(4):483–493. (Cité en page 13.)
- [Pringle, 1948] Pringle, J. W. S. (1948). The Gyroscopic Mechanism of the Halteres of Diptera. *Philosophical Transactions of the Royal Society of London B : Biological Sciences*, 233(602):347–384. (Cité en page 183.)
- [Prouty, 2003] Prouty, R. (2003). *Helicopter Performance, Stability, and Control*. Krieger. (Cité en pages 61 et 62.)
- [Quanser, 2009] Quanser (2009). QUARC® 2.0, Soft Real-Time Control Software for Windows. (Cité en page 46.)

- [Quigley *et al.*, 2005] Quigley, M., Goodrich, M., Griffiths, S., Eldredge, A. et Beard, R. (2005). Target Acquisition, Localization, and Surveillance Using a Fixed-Wing Mini-UAV and Gimbaled Camera. *In Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2600–2605. (Cité en page 181.)
- [Rafi *et al.*, 2006] Rafi, F., Khan, S., Shafiq, K. et Shah, M. (2006). Autonomous target following by unmanned aerial vehicles. *Proc. SPIE*, 6230:623010–623010–8. (Cité en page 181.)
- [Rifaï *et al.*, 2013] Rifaï, H., Guerrero-Castellanos, J.-F., Marchand, N. et Poulin-Vittrant, G. (2013). Biomimetic-based output feedback for attitude stabilization of a flapping-wing micro aerial vehicle. *Robotica*, 31:955–968. (Cité en page 224.)
- [Rondon *et al.*, 2010] Rondon, E., Garcia-Carrillo, L.-R. et Fantoni, I. (2010). Vision-based altitude, position and speed regulation of a quadrotor rotorcraft. *In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 628–633. (Cité en pages 8 et 13.)
- [Roubieu *et al.*, 2014] Roubieu, F. L., Serres, J. R., Colonnier, F., Franceschini, N., Viollet, S. et Ruffier, F. (2014). A biomimetic vision-based hovercraft accounts for bees’ complex behaviour in various corridors. *Bioinspiration & biomimetics*, 9(3):036003. (Cité en page 100.)
- [Salaün et Martin, 2010] Salaün, E. et Martin, P. (2010). Design and implementation of a low-cost observer-based attitude and heading reference system. *Control Engineering Practice*, 18(7):712–722. (Cité en pages 16 et 18.)
- [Sandeman, 1978] Sandeman, D. C. (1978). Eye-scanning during walking in the crab *Leptograpsus variegatus*. *Journal of Comparative Physiology*, 124:249–257. 10.1007/BF00657056. (Cité en page 107.)
- [Satoh *et al.*, 2014] Satoh, A. K., Li, B. X., Xia, H. et Ready, D. F. (2014). Calcium-Activated Myosin V Closes the Drosophila Pupil. *Current Biology*, 18(13):951–955. (Cité en page 106.)
- [Shabayek *et al.*, 2012] Shabayek, A. E. R., Demonceaux, C., Morel, O. et Fofi, D. (2012). Vision Based UAV Attitude Estimation : Progress and Insights. *Journal of Intelligent & Robotic Systems*, 65(1-4):295–308. (Cité en pages 173 et 174.)
- [Snyder, 1979] Snyder, A. (1979). *Handbook of Sensory Physiology*, volume 7 / 6 / 6 A de *Handbook of Sensory Physiology*, chapitre Physics of Vision in Compound Eyes, pages 225–313. Springer Verlag Berlin Heidelberg. (Cité en pages 104 et 106.)
- [Spica *et al.*, 2013] Spica, R., Robuffo Giordano, P., Ryll, M., Bülthoff, H. H. et Franchi, A. (2013). An Open-Source Hardware/Software Architecture for Quadrotor UAVs. *In 2nd Workshop on Research, Education and Development of Unmanned Aerial System*, Compiègne, France. (Cité en page 76.)
- [Stavenga, 2003a] Stavenga, D. (2003a). Angular and spectral sensitivity of fly photoreceptors. I. Integrated facet lens and rhabdomere optics. *Journal of Comparative Physiology A*, 189(1):1–17. (Cité en page 104.)

- [Stavenga, 2003b] Stavenga, D. (2003b). Angular and spectral sensitivity of fly photoreceptors. II. Dependence on facet lens F-number and rhabdomere type in *Drosophila*. *Journal of Comparative Physiology A*, 189(3):189–202. (Cité en page 104.)
- [Strausfeld *et al.*, 1987] Strausfeld, N., Seyan, H. et Milde, J. (1987). The Neck Motor System of the Fly *Calliphora erythrocephala*. 1. Muscles and Motor Neurons. *J. Comp. Physiol., A* 160:205–224. (Cité en page 13.)
- [Tayebi *et al.*, 2007] Tayebi, A., McGilvray, S., Roberts, A. et Moallem, M. (2007). Attitude Estimation and Stabilization of a Rigid Body Using Low-cost Sensors. *Algorithms*, 2:5. (Cité en page 173.)
- [Taylor et Krapp, 2007] Taylor, G. K. et Krapp, H. G. (2007). Sensory Systems and Flight Stability : What do Insects Measure and Why? In Casas, J. et Simpson, S., éditeurs : *Insect Mechanics and Control*, volume 34 de *Advances in Insect Physiology*, pages 231 – 316. Academic Press. (Cité en page 13.)
- [Thurrowgood *et al.*, 2009] Thurrowgood, S., Soccol, D., Moore, R., Bland, D. et Srinivasan, M. (2009). A Vision Based System for Attitude Estimation of UAVs. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 5725–5730. (Cité en page 173.)
- [Van Hataren et Schilstra, 1999] Van Hataren, J. H. et Schilstra, C. (1999). Blowfly Flight and Optic Flow II. Head Movements During Flight. *Journal of Experimental Biology*, 202:1491–1500. (Cité en page 182.)
- [Viollet, 2014] Viollet, S. (2014). Vibrating Makes for Better Seeing : from the Fly’s Micro Eye Movements to Hyperacute Visual Sensors. *Frontiers in Bioengineering and Biotechnology*, 2(9). (Cité en pages 103 et 107.)
- [Viollet et Franceschini, 2001] Viollet, S. et Franceschini, N. (2001). Super-accurate visual control of an aerial minirobot. *Proc. of Autonomous Minirobots for Research and Edutainment AMIRE, U. Rückert, J. Sitté and U. Witkowski, Eds., Paderborn, Germany*, pages 215–224. (Cité en page 143.)
- [Viollet et Franceschini, 2005] Viollet, S. et Franceschini, N. (2005). A high speed gaze control system based on the Vestibulo-Ocular Reflex. *Robot. Auton. Syst*, 50:147–161. (Cité en page 10.)
- [Viollet et Franceschini, 2010] Viollet, S. et Franceschini, N. (2010). A Hyperacute Optical Position Sensor Based on Biomimetic Retinal Micro-Scanning. *Sensors and Actuators A : Physical*, 160(1-2):60 – 68. (Cité en page 129.)
- [Viollet *et al.*, 2014] Viollet, S., Godiot, S., Leitel, R., Buss, W., Breugnon, P., Menouni, M., Juston, R., Expert, F., Colonnier, F., L’Eplattenier, G., Brückner, A., Kraze, F., Mallot, H., Franceschini, N., Pericet-Camara, R., Ruffier, F. et Floreano, D. (2014). Hardware Architecture and Cutting-Edge Assembly Process of a Tiny Curved Compound Eye. *Sensors*, 14(11):21702–21721. (Cité en page 100.)

- [Viollet et Zeil, 2013] Viollet, S. et Zeil, J. (2013). Feed-forward and visual feed-back control of head roll orientation in wasps (*Polistes humilis*, Vespidae, Hymenoptera). *J. Exp. Biol.* (Cité en page 13.)
- [Wang *et al.*, 2005] Wang, I., Dobrokhodov, V., Kaminer, I. et Jones, K. (2005). On Vision-Based Target Tracking and Range Estimation for Small UAVs. In *Guidance, Navigation, and Control and Co-located Conferences*, pages –. American Institute of Aeronautics and Astronautics. (Cité en page 181.)
- [Webb, 2000] Webb, B. (2000). What does robotics offer animal behaviour? *Animal Behaviour*, 60(5): 545 – 558. (Cité en page 12.)
- [Weiss *et al.*, 2011] Weiss, S., Scaramuzza, D. et Siegwart, R. (2011). Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments. *J. Field. Robot.*, 28(6):854–874. (Cité en page 16.)
- [Wendel *et al.*, 2006] Wendel, J., Meister, O., Schlaile, C. et Gert, F. T. (2006). An Integrated GPS/MEMS-IMU Navigation System for an Autonomous Helicopter. *Aerospace Science and Technology*, 10(6):527 – 533. (Cité en pages 8 et 174.)
- [Wenzel *et al.*, 2009] Wenzel, K. E., Rosset, P. et Zell, A. (2009). Low-Cost Visual Tracking of a Landing Place and Hovering Flight Control with a Microcontroller. In *Selected papers from the 2nd international Symposium on UAV*, pages 297–311. (Cité en pages 8 et 13.)
- [Westheimer, 1981] Westheimer, G. (1981). Visual Hyperacuity. In Autrum, H., Perl, E., Schmidt, R. et Ottoson, D., éditeurs : *Progress in Sensory Physiology*, volume 1 de *Progress in Sensory Physiology*, pages 1–30. Springer Berlin Heidelberg. (Cité en pages 10 et 107.)
- [Westheimer, 2009] Westheimer, G. (2009). Hyperacuity. *Encyclopedia of Neuroscience*, 5:45–50. (Cité en page 107.)
- [Wiederman *et al.*, 2008] Wiederman, S. D., Shoemaker, P. A. et O’Carroll, D. (2008). A Model for the Detection of Moving Targets in Visual Clutter Inspired by Insect Physiology. *PLOS. ONE.*, 3(7):e2784. (Cité en page 13.)
- [Wijngaard, 2010] Wijngaard, W. (2010). Accuracy of insect position control as revealed by hovering male *Eristalis nemorum*. In Bruin, J., éditeur : *Proceedings of the Netherlands Entomological Society Meeting*, volume Volume 21, page 75. (Cité en pages 13 et 17.)
- [Windolf *et al.*, 2008] Windolf, M., Götzen, N. et Morlock, M. (2008). Systematic accuracy and precision analysis of video motion capturing systems-exemplified on the Vicon-460 system. *Journal of Biomechanics*, 41(12):2776 – 2780. (Cité en page 231.)
- [Wireshark Foundation, 2006] Wireshark Foundation (2006). Wireshark : Network protocol analyzer for Unix and Windows. (Cité en page 46.)

- [Yang *et al.*, 2013a] Yang, S., Scherer, S. A. et Zell, A. (2013a). An Onboard Monocular Vision System for Autonomous Takeoff, Hovering and Landing of a Micro Aerial Vehicle. *J. Intell. Robot. Syst.*, 69(1-4):499–515. (Cité en page [8](#).)
- [Yang *et al.*, 2013b] Yang, S., Scherer, S. A. et Zell, A. (2013b). An Onboard Monocular Vision System for Autonomous Takeoff, Hovering and Landing of a Micro Aerial Vehicle. *Journal of Intelligent & Robotic Systems*, 69(1-4):499–515. (Cité en page [174](#).)
- [Zhang *et al.*, 2009a] Zhang, T., Ye, K., Achtelik, M., Kuhnlenz, K. et Buss, M. (2009a). Autonomous hovering of a vision/IMU guided quadrotor. In *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pages 2870–2875. (Cité en page [8](#).)
- [Zhang *et al.*, 2009b] Zhang, T., Ye, K., Achtelik, M., Kuhnlenz, K. et Buss, M. (2009b). Autonomous Hovering of a Vision/IMU Guided Quadrotor. In *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pages 2870–2875. (Cité en page [174](#).)

Résumé — Les insectes sont capables de prouesses remarquables lorsqu'il s'agit d'éviter des obstacles, voler en environnement perturbé ou poursuivre une cible. Cela laisse penser que leurs capacités de traitement, aussi minimalistes soient-elles, sont parfaitement optimisées pour le vol. A cela s'ajoute des mécanismes raffinés, comme la stabilisation de la vision par rapport au corps, permettant d'améliorer encore plus leurs capacités de vol.

Ces travaux de thèse présentent l'élaboration d'un micro drone de type quadrirotor, qui ressemble fortement à un insecte sur le plan perceptif (vibration rétinienne) et reprend des points structurels clés, tels que le découplage mécanique entre le corps et le système visuel. La conception du quadrirotor (de type open-source), son pilotage automatique et son système occulo-moteur sont minutieusement détaillés. Des traitements adaptés permettent, malgré un très faible nombre de pixels (24 pixels seulement), de poursuivre finement du regard une cible en mouvement. A partir de là, nous avons élaboré des stratégies basées sur le pilotage par le regard, pour stabiliser le robot en vol stationnaire, à l'aplomb d'une cible et asservir sa position ; et ce, en se passant d'une partie des capteurs habituellement utilisés en aéronautique tels que les magnétomètres et les accéléromètres. Le quadrirotor décolle, se déplace et atterrit de façon autonome en utilisant seulement ses gyromètres, son système visuel original mimant l'œil d'un insecte et une mesure de son altitude. Toutes les expérimentations ont été validées dans une arène de vol, équipée de caméras VICON.

Enfin, nous décrivons une nouvelle toolbox qui permet d'exécuter en temps réel des modèles Matlab/Simulink sur des calculateurs Linux embarqués de façon complètement automatisée (<http://www.gipsa-lab.fr/projet/RT-MaG/>). Cette solution permet d'écrire les modèles, de les simuler, d'élaborer des lois de contrôle pour enfin, piloter en temps réel, le robot sous l'environnement Simulink. Cela réduit considérablement le "time-to-flight" et offre une grande flexibilité (possibilité de superviser l'ensemble des données de vol, de modifier en temps réel les paramètres des contrôleurs, etc.)

Mots clés : quadrirotor, vol stationnaire, capteur optique, estimation d'attitude, bio-inspiration, stabilisation du regard, hyperacuité, insecte, vision, réflexe vestibulo-oculaire, fixation visuelle, poursuite fine, modélisation, commande linéaire, commande non linéaire, identification de paramètres, observation d'état.

Abstract — Insects, like hoverflies are able of outstanding performances to avoid obstacles, reject disturbances and hover or track a target with great accuracy. These means that fast sensory motor reflexes are at work, even if they are minimalist, they are perfectly optimized for the flapping flight at insect scale. Additional refined mechanisms, like gaze stabilization relative to the body, allow to increase their flight capacity.

In this PhD thesis, we present the design of a quadrotor, which is highly similar to an insect in terms of perception (visual system) and implements a bio-inspired gaze control system through the mechanical decoupling between the body and the visual system. The design of the quadrotor (open-source), its pilot and its decoupled eye are thoroughly detailed. New visual processing algorithms make it possible to faithfully track a moving target, in spite of a very limited number of pixels (only 24 pixels). Using this efficient gaze stabilization, we developed new strategies to stabilize the robot above a target and finely control its position relative to the target. These new strategies do not need classical aeronautic sensors like accelerometers and magnetometers. As a result, the quadrotor is able to take off, move and land automatically using only its embedded rate-gyros, its insect-like eye, and an altitude measurement. All these experiments were validated in a flying arena equipped with a VICON system.

Finally, we describe a new toolbox, called RT-MaG toolbox, which generate automatically a real-time standalone application for Linux systems from a Matlab/Simulink model (<http://www.gipsa-lab.fr/projet/RT-MaG/>). These make it possible to simulate, design control laws and monitor the robot's flight in real-time using only Matlab/Simulink. As a result, the "time-to-flight" is considerably reduced and the final application is highly reconfigurable (real-time monitoring, parameter tuning, etc.)

Keywords : quadrotor, hovering flight, optical sensor, attitude estimation, bio-inspiration, gaze stabilization, hyperacuity, insect, vision, vestibulo-ocular reflex, visual fixation, accurate tracking, modeling, linear control, nonlinear control, parameter identification, state observer.
