

N° d'ordre : 3666

THÈSE

présentée devant

l'Université de Rennes 1

pour obtenir

le grade de DOCTEUR DE L'UNIVERSITÉ DE RENNES 1
mention TRAITEMENT DU SIGNAL ET TÉLÉCOMMUNICATIONS

par

Odile Bourquardez

Équipe d'accueil : IRISA – Équipe-projet Lagadic

École Doctorale : Matisse

Composante universitaire : IFSIC

Titre de la thèse :

**Commande d'engins volants
par asservissement visuel**

Soutenue le 11 février 2008 devant la commission d'examen

M. :	Patrick	BOUTHEMY	Président
MM. :	José Raul	AZINHEIRA	Rapporteurs
	Tarek	HAMEL	
MM. :	Jacques	GANGLOFF	Examineurs
	Bruno	PATIN	
M. :	François	CHAUMETTE	Directeur de thèse

Remerciements

Mes remerciements vont tout d'abord aux membres du jury. Merci à Patrick Bouthemy, directeur du Centre de Recherche INRIA Rennes Bretagne Atlantique, qui m'a fait l'honneur de présider le jury. Merci aussi à Tarek Hamel, professeur à l'Université de Nice-Sophia-Antipolis, et José Raul Azinheira, professeur à l'Instituto Superior Técnico à Lisbonne, d'avoir bien voulu accepter la charge de rapporteurs. Je tiens également à remercier Jacques Gangloff, professeur à l'Université Louis Pasteur de Strasbourg, d'avoir bien voulu participer à mon jury de thèse. Ce sont également ses enseignements qui m'ont initialement donné goût à l'asservissement visuel, et m'ont ensuite menée sur la voie de la thèse de doctorat. Merci à Bruno Patin, ingénieur R&D, chez Dassault Aviation à Saint Cloud, pour l'intérêt et l'attention qu'il a porté à mon travail, depuis le début.

Je remercie François Chaumette, directeur de recherche à l'INRIA Rennes Bretagne Atlantique, qui m'a proposé ce sujet et a dirigé ma thèse. Il a su m'écouter, et guider mes choix par ses conseils éclairés. Je tiens également à le remercier pour ses qualités humaines. Sa disponibilité et son optimisme m'ont souvent aidée à reprendre confiance en moi face aux difficultés.

Merci à la DGA, qui a financé mon travail, ainsi qu'à François Chaumette et à Bruno Patin, initiateurs du projet.

Ma thèse m'a permis de m'initier au domaine de l'aéronautique : merci à tous ceux qui m'y ont aidée. En particulier, merci à Jean-Pierre Belmont, de Dassault Aviation Saint Cloud qui a bien voulu consacrer un peu de son temps à répondre à mes questions de néophyte concernant le modèle avion. Merci également aux personnes de Dassault Aviation Istres, qui m'ont fourni le simulateur : Jean-Pierre Tomaszower, Erica Palmyre, et Guillaume Rimbault.

Ma thèse m'a également donné l'opportunité de collaborer avec des chercheurs de différents laboratoires. Merci à Tarek Hamel, Robert Mahony, Nicolas Guenard, Laurent Eck et Vincent Moreau, avec qui j'ai travaillé dans le cadre du projet Robvolint. Un merci plus particulier à Nicolas Guenard, dont le dynamisme a permis de rendre fructueuses les phases d'expérimentation au CEA. Je n'oublie pas non plus le PICS, ses initiateurs, et surtout Robert Mahony qui m'a accueillie à l'ANU de Canberra pendant

un mois. J'ai également apprécié travailler avec lui pour ses conseils, sa rigueur et ses explications toujours claires.

À l'IRISA, nous avons la chance de pouvoir vraiment nous concentrer sur notre sujet de recherche : merci à tous ceux qui, par leur travail, nous rendent la vie plus facile au quotidien. En particulier, je tiens à remercier les assistantes Stéphanie et Céline, ainsi que Anne et Émilie, de la bibliothèque, sans oublier Pierrette, Laurence et leurs joyeux sourires à la cafétéria.

Merci à ceux qui ont accepté de passer du temps à relire et corriger certaines parties de ce manuscrit : Muriel, Andrea et Roméo. Bien sûr, un merci spécial à Nicolas, qui m'a encouragée à lui rapporter chaque jour un peu de lecture.

Enfin, je remercie tous les membres de l'équipe Lagadic, chacun contribuant à l'ambiance chaleureuse qui y règne. En premier lieu merci à ceux dont j'ai partagé le bureau : Cédric, David, et surtout Muriel. Attention, discrétion, sympathie, ... : il était agréable de travailler à vos côtés. Roméo, merci pour les services rendus : je savais pouvoir compter sur toi. Merci aussi à tous les autres : Fabien, Christophe, Alexandre, Éric, Nicolas, Anthony, Andrew, Sinisa, Hai, Albert, Wael, Mikaël, Laid, Andrea, Fabien, Claire, Rafik, Mohammed, Fabien, Anthony, Céline, Amaury, Xiang, ... et ceux que j'ai peut-être oubliés ici. Votre aide et vos conseils m'ont souvent été précieux. Merci également pour les discussions pendant les pauses — tant sur des sujets scientifiques que très variés — ces moments d'échange et de convivialité ont été très enrichissants. Merci aussi aux non francophones d'entre vous qui m'ont permis d'améliorer un peu mon niveau d'anglais.

Je remercie ma famille, qui m'a encouragée et soutenue tout au long de ma thèse.

Merci à toi, Nicolas, d'être près de moi, tout simplement.

Table des matières

Table des matières	3
Notations	7
Unités	9
Glossaire	11
Introduction	13
1 Contexte de l'étude	17
1.1 L'asservissement visuel	17
1.1.1 Principes de base	17
1.1.1.1 Formation des images	18
1.1.1.2 Asservissement visuel	21
1.1.2 Différents types d'asservissement visuel	24
1.1.2.1 Asservissement visuel 3D	24
1.1.2.2 Asservissement visuel 2D	26
1.1.2.3 Autres types d'asservissements visuels	27
1.2 La commande par vision d'engins volants	28
1.2.1 Divers engins volants	28
1.2.1.1 Hélicoptères	28
1.2.1.2 Dirigeables	34
1.2.1.3 Avions	36
1.2.1.4 Drones bio-inspirés	38
1.2.2 Différents types d'informations visuelles	39
1.2.2.1 Informations visuelles 3D	39
1.2.2.2 Informations visuelles 2D	41
1.2.3 Diverses lois de commandes utilisées	41
1.2.3.1 Stratégie de commande	42
1.2.3.2 Type de contrôleur	43
1.3 Conclusion	45

2	Asservissement visuel d'un quadrirotor	47
2.1	Objectif et problématique	47
2.2	Simulations : conditions expérimentales	49
2.3	Utilisation de la projection perspective	52
2.4	Utilisation de la projection sphérique	57
2.4.1	Modélisation des informations visuelles	57
2.4.2	Commande proportionnelle	61
2.4.3	Linéarisation à la position désirée	62
2.4.4	Commande partitionnée	64
2.4.5	Changement d'échelle de l'information visuelle	67
2.4.5.1	Commande stable	72
2.4.5.2	Commande classique avec approximation de la matrice d'interaction	73
2.4.5.3	Commande proportionnelle	73
2.4.6	Commande stable avec information visuelle modifiée	74
2.5	Analyse des propriétés des lois de commande étudiées	78
2.5.1	Propriétés des lois de commande	78
2.5.2	Comparaison des trois meilleures lois de commande	81
2.6	Résultats expérimentaux sur un prototype d'hélicoptère miniature	82
2.6.1	Conditions expérimentales	82
2.6.1.1	Prototype utilisé	82
2.6.1.2	Protocole expérimental	84
2.6.2	Résultats avec les moments de l'image perspective	84
2.6.3	Résultats avec le centre de gravité sphérique	85
2.6.3.1	Linéarisation à la position désirée	85
2.6.3.2	Commande partitionnée	87
2.6.3.3	Commande proportionnelle avec changement d'échelle	87
2.6.3.4	Commande stable avec nouveau changement d'échelle	90
2.6.4	Analyse de la sensibilité au bruit	91
2.7	Conclusion	91
3	Atterrissage d'un avion par asservissement visuel	95
3.1	Objectif et problématique	95
3.1.1	Définitions et notations	96
3.1.2	Fonctionnement d'un avion	102
3.1.3	Commande	107
3.1.4	Comportement d'un avion pendant l'atterrissage	107
3.1.4.1	Approche	109
3.1.4.2	Arrondi	109
3.1.4.3	Toucher de la piste et roulage	111
3.1.4.4	Rôle des volets	111
3.1.5	Atterrissage par asservissement visuel	112
3.2	Modélisation de l'avion	112
3.2.1	Modèle non-linéaire	113

3.2.2	Modèle linéarisé et découplage	114
3.2.3	Passage au repère de la caméra	120
3.3	Modélisation d'informations visuelles	123
3.3.1	Objectif	123
3.3.2	Informations visuelles utilisées	123
3.4	Commande	131
3.4.1	Lois de commande	131
3.4.1.1	Commande latérale	131
3.4.1.2	Commande longitudinale.	137
3.4.2	Génération de trajectoires	138
3.4.2.1	Trajectoire latérale	138
3.4.2.2	Trajectoire longitudinale	140
3.4.3	Lois de commande pour le suivi de trajectoires	143
3.5	Résultats de simulation	143
3.5.1	Simulateur utilisé	144
3.5.2	Résultats	147
3.5.2.1	Commande longitudinale	147
3.5.2.2	Commande latérale	163
3.5.2.3	Commande complète avec suivi de trajectoire	182
3.6	Conclusion	207
Conclusions et perspectives		209
	Asservissement visuel d'un quadrirotor	209
	Contributions	209
	Perspectives	211
	Atterrissage d'un avion par asservissement visuel	211
	Contributions	211
	Perspectives	212
	Conclusion	213
A Calculs relatifs à la modélisation des informations visuelles utilisées pour la commande de l'hélicoptère		217
A.1	Calcul de la dérivée de la fonction $\mathcal{L} = \frac{1}{2}(\delta_{11} ^2 + \lambda^2 \delta_{12}^2)$	217
A.2	Calcul de la dérivée de l'information visuelle $\mathbf{f} = \mathbf{F}(\mathbf{q})\mathbf{q}_n$	218
A.3	Calcul de $G(\mathbf{q})$	220
B Passage du repère géocentrique au repère lié à la piste d'atterrissage		223
C Représentation des rotations		227
D Paramètres du modèle d'avion		231
D.1	Valeurs numériques pour le modèle d'avion	231
D.2	Valeurs numériques pendant la phase d'approche	232
E Analyse de la loi de commande latérale		233

F	Calculs des paramètres des trajectoires	239
F.1	Calcul des coefficients du polynôme $Y^*(t)$	239
F.2	Calcul des coefficients du polynôme $Z^*(X)$	241
	Bibliographie	245
	Table des figures	259
	Résumé	266
	Abstract	266

Notations

Généralités

- les scalaires sont en général notés en minuscules
- les vecteurs sont en général notés en minuscules grasses
- les matrices sont en général notées en majuscules grasses

Géométrie

- \mathcal{F}_o : repère inertiel fixe, de référence
- \mathcal{F}_c : repère lié à la caméra
- ${}^i\mathbf{R}_j$: matrice de rotation représentant l'orientation du repère \mathcal{F}_j dans le repère \mathcal{F}_i
- ${}^i\mathbf{t}_j$: vecteur de translation représentant la position du repère \mathcal{F}_j dans le repère \mathcal{F}_i
- ${}^i\mathbf{M}_j = \begin{bmatrix} {}^i\mathbf{R}_j & {}^i\mathbf{t}_j \\ 0 & 1 \end{bmatrix}$: matrice homogène de changement de repère (passage du repère \mathcal{F}_j au repère \mathcal{F}_i)

Asservissement visuel

- $\mathbf{v} = (v_x, v_y, v_z)$: vecteur vitesse en translation
- $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$: vecteur vitesse en rotation
- $\mathbf{v} = (\mathbf{v}, \boldsymbol{\omega})$: torseur cinématique
- \mathbf{s} : informations visuelles
- \mathbf{L}_s : matrice d'interaction associée à \mathbf{s}
- l'exposant * indique qu'il s'agit de la valeur calculée à la position désirée

Vol de l'avion

- α : incidence
- β : dérapage
- γ : pente
- ϕ : roulis

- θ : tangage
- ψ : lacet
- \mathcal{F}_a : repère lié à l'avion
- \mathcal{F}_f : repère inertiel fixe
- $\mathbf{v}_a = (v_{a_x}, v_{a_y}, v_{a_z})$: vecteur vitesse de translation de l'avion, exprimé dans le repère lié à l'avion \mathcal{F}_a
- $v_a = |\mathbf{v}_a|$: module du vecteur vitesse de translation de l'avion
- $\boldsymbol{\omega}_a = (\omega_{a_x}, \omega_{a_y}, \omega_{a_z})$: vecteur vitesse de rotation de l'avion, exprimé dans le repère lié à l'avion \mathcal{F}_a
- m : masse de l'avion
- \mathbf{I} : matrice d'inertie de l'avion
- $\mathbf{g} = (0, 0, g)$: vecteur accélération de la gravité, exprimé dans \mathcal{F}_F
- $F_{\{A,W,T,L,D\}}$: force aérodynamique (A), force due à la gravité (W), force de propulsion (T), portance (L), traînée (D)
- $M_{\{A,T\}}$: moment aérodynamique (A), moment dû à la propulsion (T)
- $\delta_{\{l,r,m,T\}}$: commande de la gouverne de gauchissement (l), direction (r), profondeur (m), et de la poussée des moteurs (T)

Outils mathématiques

- \mathbf{I}_k : matrice identité de rang k
- $\mathbf{0}_{m \times n}$: matrice nulle de dimensions $m \times n$
- \cong : égal à une constante multiplicative près
- \simeq : environ égal
- \lesssim, \gtrsim : inférieur, ou supérieur à environ
- \times : opérateur de produit vectoriel
- $[\mathbf{w}]_{\times}$: matrice de pré-produit vectoriel associé au vecteur $\mathbf{w} = (w_1, w_2, w_3)$ (c'est-à-dire telle que $[\mathbf{w}]_{\times} \mathbf{u} = \mathbf{w} \times \mathbf{u}$ pour tout vecteur \mathbf{u}) : $[\mathbf{w}]_{\times} = \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix}$
- $|\mathbf{u}|$: module du vecteur \mathbf{u}
- $|u|$: valeur absolue de u
- $\cos x$: cosinus de x , $\cos x$
- $c^p x$: cosinus de x , à la puissance p , $(\cos x)^p$
- $\sin x$: sinus de x , $\sin x$
- $s^p x$: sinus de x , à la puissance p , $(\sin x)^p$
- j : le nombre imaginaire tel que $j^2 = -1$

Unités

Dans cette thèse, nous utilisons essentiellement les unités du système international :

- distances en mètres (m),
- vitesses en mètres par seconde (m/s),
- accélérations en mètres par seconde au carrée (m/s²),
- angles en radians (rad), à l'exception des courbes de résultat, où l'on représente les angles en degrés (deg), mais cela est alors indiqué,
- vitesses angulaires en radians par seconde (rad/s).

Cependant, en aéronautique, il est habituel d'utiliser d'autres unités :

- distances verticales en pieds (ft)
- distances horizontales en mille marin, ou nautique (NM),
- vitesse verticale en pied par minute (ft/min),
- vitesse horizontale en nœud (kt), ou mille nautique par heure.

La correspondance entre ces deux systèmes d'unité est la suivante :

- 1 ft = 0.3048 m,
- 1 NM = 1852 m,
- 100 ft/min = 0.508 m/s,
- 1 kt = 0.514 m/s.

Glossaire

- ANU : Université Nationale Australienne (*the Australian National University*)
- AFCS : système de commande de vol automatique (*Automatic Flight Control System*)
- CEA/LIST : Commissariat à l'Énergie Atomique, Laboratoire d'Intégration des Systèmes et des Technologies
- ddl(s) : Degré(s) De Liberté
- caméra CCD : caméra dont le capteur photographique est de type CCD (*Charge-Coupled Device*, détecteurs à couplage de charge)
- caméra CMOS : caméra dont le capteur photographique est de type CMOS (*Complementary Metal Oxide Semi-conductor*)
- DSP : processeur de signal numérique (*Digital Signal Processor*)
- EPFL : École Polytechnique Fédérale de Lausanne
- FBW : commande de vol électrique (*Fly-By-Wire*)
- GAS : stabilité asymptotique globale (*Global Asymptotic Stability*)
- GPS : géopositionnement par satellite (*Global Positioning System*)
- HF : Haute Fréquence
- IBVS : asservissement visuel 2D (*Image-Based Visual Servoing*)
- I3S : laboratoire Informatique, Signaux et Systèmes
- ILS : système d'atterrissage aux instruments (*Instrument Landing System*)
- IMU : centrale inertielle (*Inertial Measurement Unit*)
- IRISA : Institut de Recherche en Informatique et Systèmes Aléatoires
- IST : *Instituto Superior Técnico*
- LTI : système linéaire et invariant dans le temps (*Linear Time Invariant*)
- LQR : commande optimale, appelée aussi commande linéaire quadratique LQR (*Linear Quadratic Regulator*)
- OpenCV : bibliothèque gratuite d'analyse d'images et de vision par ordinateur (*Open Computer Vision Library*)
- PBVS : asservissement visuel 3D (*Pose-Based Visual Servoing*)
- PC : ordinateur personnel (*Personal Computer*)
- PICS : Programme International de Collaboration Scientifique
- Robea : ROBotique et Entités Artificielles
- Robvolint : ROBot VOLant d'INTérieur
- UAV : drone, ou engin volant sans pilote (*Unmanned Aerial Vehicle*)
- ViSP : bibliothèque gratuite dédiée à l'asservissement visuel (*Visual Servoing*)

Platform)

- VTOL : engin volant capable de décoller et d'atterrir verticalement (*Vertical Take Off and Landing*)

Introduction

Concevoir des robots autonomes et intelligents fait appel à de nombreux domaines d'expertise. Matériellement, on peut dire qu'un robot est essentiellement constitué de pièces mécaniques et de cartes électroniques. Des capteurs donnent au robot des informations sur son état interne et sur l'environnement dans lequel il évolue, et des actionneurs lui permettent d'effectuer différentes tâches. Le robot est également équipé d'un système logiciel plus ou moins évolué, tenant lieu d'"intelligence". Le rôle de la partie logicielle consiste à utiliser judicieusement les mesures des capteurs afin de commander les actionneurs et réaliser ainsi les tâches attendues. Pour mettre au point de tels logiciels, de vastes champs d'expertise sont à exploiter, tels que les techniques de traitement de signal, d'automatique et de modélisation, les problématiques d'interface homme-machine, etc.

Dans cette thèse, nous nous intéressons aux techniques d'asservissement visuel, qui consistent à mettre au point des lois de commande à partir de données issues d'un capteur de vision. L'asservissement visuel a tout d'abord été mis en œuvre pour des robots manipulateurs [Hutchinson 96], qui sont les robots les plus classiquement utilisés, notamment dans l'industrie. L'application de l'asservissement visuel à de nombreux domaines de la robotique a ensuite été envisagée. Citons par exemple les robots médicaux, pour l'aide au chirurgien [Gangloff 04] ou la réalisation automatique d'échographies [Krupa 07], les applications de micro-robotique [Ferreira 01], les robots bio-inspirés (conçus de façon à imiter le comportement du monde animal) [Ruffier 04, Zufferey 06], les robots d'aide aux personnes handicapées [Dune 07], les robots mobiles à roues [Royer 04] — modes de transports du futur ou robots de plus petite taille —, les robots à pattes, et notamment les humanoïdes [Mansard 07], les robots sous-marins [Dunbabin 04], et enfin les robots volants — domaine auquel on s'intéresse ici. Notons que le capteur de vision utilisé n'est pas toujours une caméra classique, mais peut être une sonde échographique [Krupa 07], une caméra à une dimension [Zufferey 06], une caméra omnidirectionnelle [Hrabar 03, Demonceaux 07], etc.

Actuellement, de nombreux projets de recherche sont dédiés à la mise au point de robots volants autonomes [Ava], [Col], [Csi], [Flo], [Mit], [Com], etc. L'ajout d'une caméra embarquée à ces engins permet d'améliorer leur autonomie et d'élargir leur champ d'application, en leur fournissant des informations sur l'espace dans lequel ils évoluent, leur permettant ainsi de s'adapter à un environnement changeant. La mise en œuvre d'une loi de commande d'asservissement visuel nécessite une phase préalable de traitement des images, afin d'extraire des images fournies par la caméra les informations

utiles à la réalisation de la tâche considérée. Par exemple, si la tâche à réaliser consiste à atterrir, il sera nécessaire de détecter la piste d'atterrissage dans les images. À partir des données de vision, une loi de commande d'asservissement visuel peut alors être appliquée. Pour notre travail, nous nous sommes limités à la conception des lois de commande, en considérant que la phase de traitement d'image était réalisée.

Notons que les lois de commande d'asservissement visuel dépendent à la fois du type de cible considéré (qui peut être, selon l'application envisagée une cible "fabriquée", une piste d'atterrissage, la façade d'un bâtiment, une ligne haute tension, une scène inconnue, etc.), et du type de robot considéré (robot manipulateur à six degrés de liberté, robot à roues, hélicoptère, avion, etc.). Notre travail porte sur l'utilisation de l'asservissement visuel dans le cadre de deux applications : l'une concerne le positionnement et la stabilisation d'un hélicoptère à quatre hélices (appelé quadrirotor) par rapport à une cible connue, et l'autre consiste à réaliser l'atterrissage d'un avion par asservissement visuel. Le quadrirotor fait partie de la famille des engins volants à voilure tournante, il est capable d'effectuer des vols stationnaires. L'avion est pour sa part un engin à ailes fixes, qui doit se déplacer pour voler. Dans les deux cas, les schémas d'asservissement visuel classiques existant pour des robots de type bras manipulateur ne sont pas adaptés. En effet, les effets de couplage entre les degrés de liberté doivent être pris en compte lors de la commande par asservissement visuel d'engins volants, puisque le système possède moins d'entrées de commande que de degrés de liberté (système sous actionné). Nous avons alors proposé des schémas d'asservissement visuel s'appuyant sur des informations visuelles extraites des scènes observées, et adaptés aux spécificités de chacun des engins volants considérés.

Notre travail a été réalisé au sein du projet Lagadic, à l'IRISA — Institut de Recherche en Informatique et Systèmes Aléatoires — à Rennes, dont les principaux axes de recherche concernent la vision robotique et l'asservissement visuel. Une part de notre travail s'inscrit dans le cadre du projet national ROBVOLINT (ROBot Volant d'INTérieur), issu d'une collaboration avec les laboratoires suivants : l'I3S — Informatique Signaux et Systèmes de Sophia Antipolis —, le CEA/LIST — Commissariat à l'Énergie Atomique, Laboratoire d'Intégration des Systèmes et des Technologies, à Fontenay aux Roses — et l'IRCCyN — Institut de Recherche en Communications et en Cybernétique de Nantes —, puis également avec l'ANU — *Australian National University* — à Canberra dans le cadre d'un PICS (Programme International de Collaboration Scientifique). L'objectif du projet était de concevoir un drone de type quadrirotor et sa commande, en environnement intérieur, afin d'explorer de manière autonome un immeuble ou un tunnel par exemple. Nous avons participé à la mise au point d'une partie de ce projet consacrée à l'asservissement visuel du quadrirotor par rapport à une cible connue. La deuxième partie de notre travail s'inscrit dans le cadre d'une collaboration avec la DGA (Délégation Générale de l'Armement) et l'industriel Dassault Aviation. Ouvert à de nouvelles solutions permettant aux avions sans pilote d'acquérir plus d'autonomie, Dassault Aviation s'est intéressé à l'application des techniques d'asservissement visuel dans ce contexte. Notre travail a porté sur l'atterrissage automatique d'un avion par asservissement visuel.

Organisation du manuscrit. Nous présentons dans le chapitre 1 le contexte de notre travail. Nous décrivons les différentes techniques d’asservissement visuel, et présentons un état de l’art de la commande d’engins volants par asservissement visuel. Nous nous intéressons en particulier au choix des informations visuelles, aux stratégies de commande et aux types de lois de commande utilisés.

Le chapitre 2 décrit la partie de notre travail concernant l’asservissement visuel d’un quadrirotor. Après avoir décrit les spécificités de l’engin considéré et le problème posé, nous proposons différentes lois de commande permettant au quadrirotor de se positionner au-dessus d’une cible visuelle connue. Des informations visuelles issues de la projection perspective de la cible sont utilisées dans un premier temps, puis des informations visuelles issues de la projection sphérique. Nous proposons ensuite une série de schémas de commande permettant d’améliorer les propriétés et les performances du système bouclé. Finalement, nous montrons les résultats expérimentaux obtenus sur un prototype de quadrirotor. Les meilleures lois de commande ont été implémentées sur le quadrirotor : les résultats sont détaillés et analysés.

L’atterrissage d’un avion par asservissement visuel fait l’objet du chapitre 3. Nous décrivons dans un premier temps le fonctionnement de l’avion, en particulier pendant la phase d’atterrissage. L’objectif consiste ensuite à considérer une caméra embarquée sur l’avion et à concevoir une loi de commande agissant au niveau des entrées pilote et reposant sur des informations visuelles. Cette loi de commande doit permettre à l’avion de rejoindre la trajectoire nominale d’approche, la suivre, réaliser la manœuvre d’arrondi et se poser sur une piste. Nous présentons ensuite le modèle d’avion considéré, mis sous forme linéaire et découplant les mouvement latéraux et longitudinaux. Compte tenu du modèle d’avion et de la scène observée (la piste d’atterrissage), nous proposons alors des informations visuelles puis des lois de commande adaptées au problème posé. Afin d’améliorer le comportement de l’asservissement visuel en faisant en sorte que les consignes ne soient pas trop éloignées des informations visuelles courantes, nous proposons ensuite une méthode de planification de la trajectoire, tenant compte du modèle de l’avion. Nous présentons finalement des résultats de simulation reposant sur un modèle réaliste d’avion.

Le dernier chapitre présente les conclusions de notre travail et propose quelques perspectives.

Chapitre 1

Contexte de l'étude

Dans cette partie, nous présentons tout d'abord un état de l'art de l'asservissement visuel. Ensuite, nous nous focalisons sur l'application de cette approche au domaine particulier des engins volants.

1.1 L'asservissement visuel

Rendre les robots autonomes nécessite de les munir de capteurs, leur fournissant des informations sur leur état et leur environnement. L'utilisation de caméras s'avère extrêmement utile pour réaliser des tâches de localisation, navigation, exploration, notamment dans des contextes où l'environnement est peu ou pas connu. L'asservissement visuel, technique de commande reposant sur l'utilisation de caméra comme capteur, fait l'objet de travaux de recherches depuis une vingtaine d'années. Notons que pour se familiariser avec les concepts de l'asservissement visuel, plusieurs tutoriaux abordent à la fois les notions de base et les concepts plus évolués de ce domaine [Hutchinson 96, Corke 96, Eur, Chaumette 06, Chaumette 07].

1.1.1 Principes de base

Différentes configurations peuvent être envisagées pour commander un robot par asservissement visuel. Déjà, la caméra peut être embarquée sur le robot (on parlera alors de système *eye in hand*) ou déportée (on parlera alors de système *eye to hand*) [Hutchinson 96]. Par ailleurs, une caméra embarquée peut être utilisée pour observer une cible visuelle particulière : soit pour se placer à une position désirée par rapport à elle, soit pour la suivre. La cible considérée peut donc être fixe ou mobile. On peut finalement envisager des systèmes complexes avec plusieurs caméras, plusieurs robots, etc. Nous nous plaçons ici dans le cadre d'un robot équipé d'une caméra embarquée et d'une cible visuelle fixe (cas le plus fréquent en pratique).

Par la suite, après avoir rappelé les équations usuelles modélisant la géométrie d'une caméra, nous montrons comment les informations visuelles construites à partir des mesures extraites de l'image peuvent être mises à profit pour commander des robots.

1.1.1.1 Formation des images

Dans cette partie, nous nous intéressons à la modélisation du capteur de vision utilisé : la caméra. En effet, pour pouvoir tirer parti au mieux des mesures fournies par un capteur, il est nécessaire de modéliser le mécanisme d'acquisition des données afin d'établir une relation entre celles-ci et le monde réel. Autrement dit, dans le cas de la caméra, nous cherchons connaître la relation entre l'image obtenue et la scène observée. Notons que nous nous contentons ici de présenter les aspects de modélisation géométrique. Cependant, dans d'autres contextes que le nôtre reposant sur des techniques d'asservissements visuels rapides, il est également nécessaire de tenir compte de la dynamique de la caméra [Corke 96, Gangloff 00].

Le modèle de caméra le plus couramment utilisé est le modèle sténopé, reposant sur la projection perspective. Il modélise assez fidèlement le comportement d'une caméra perspective "classique". Dans les cas moins courants de caméras grand angle, caméras omnidirectionnelles, sondes échographiques, etc., il conviendrait bien entendu d'utiliser des modèles différents.

Le modèle sténopé permet de lier les coordonnées d'un point de l'espace 3D à ses coordonnées en pixels dans l'image, selon le modèle de projection illustré sur la figure 1.1. Il peut se décomposer en trois étapes. Il s'agit tout d'abord d'exprimer les coordonnées du point 3D considéré dans le repère lié à la caméra (à l'aide des paramètres extrinsèques de la caméra). Ensuite, le modèle de projection perspective permet d'exprimer les coordonnées du point image. Enfin, on obtient les coordonnées pixelliques du point image à l'aide des paramètres intrinsèques de la caméra. Notons que nous n'aborderons ici que les concepts de base de la vision par ordinateur. Le lecteur intéressé trouvera de plus amples détails dans des ouvrages traitant de ce sujet, tel que [Faugeras 93] par exemple.

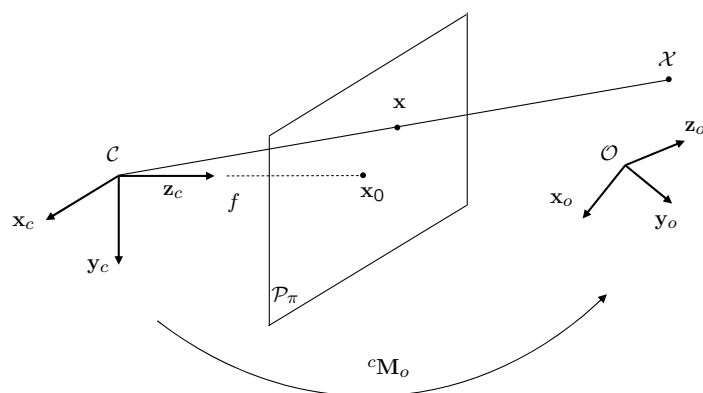


FIG. 1.1 – Modélisation de la caméra.

Changement de repère. Notons $\mathcal{F}_c = (\mathcal{C}, \mathbf{x}_c, \mathbf{y}_c, \mathbf{z}_c)$ le repère de la caméra, dont l'origine se situe au centre de projection \mathcal{C} . Soit \mathcal{X} un point de l'espace 3D de coordonnées homogènes $\mathbf{P}_o = (X_o, Y_o, Z_o, 1)$ dans un repère fixe $\mathcal{F}_o = (\mathcal{O}, \mathbf{x}_o, \mathbf{y}_o, \mathbf{z}_o)$ lié à la scène. La première étape, de changement de repère, consiste à exprimer les coordonnées du point \mathcal{X} dans le repère \mathcal{F}_c (voir figure 1.1).

Pour cela, la connaissance de l'orientation et de la position de la caméra par rapport au repère de l'objet est indispensable. Ces paramètres, au nombre de six (trois positions et trois orientations) sont appelés paramètres extrinsèques de la caméra. On note ${}^c\mathbf{R}_o$ et ${}^c\mathbf{t}_o$ les matrices de rotation et le vecteur de translation (respectivement) permettant de passer du repère \mathcal{F}_o au repère \mathcal{F}_c . On note

$${}^c\mathbf{M}_o = \begin{bmatrix} {}^c\mathbf{R}_o & {}^c\mathbf{t}_o \\ 0 & 1 \end{bmatrix}$$

la matrice homogène de changement de repère de \mathcal{F}_o vers \mathcal{F}_c . Notons $\mathbf{P}_c = (X_c, Y_c, Z_c, 1)$ les coordonnées homogènes du point \mathcal{X} exprimées dans le repère de la caméra. On a :

$$\mathbf{P}_c = {}^c\mathbf{M}_o \mathbf{P}_o. \quad (1.1)$$

Projection perspective. L'étape suivante consiste à projeter le point \mathcal{X} sur le plan image de la caméra (voir figure 1.1).

Le plan défini par les axes $(\mathcal{C}, \mathbf{x}_c)$ et $(\mathcal{C}, \mathbf{y}_c)$ est parallèle au plan image \mathcal{P}_π . La distance minimale du centre de projection \mathcal{C} au plan image est notée f et appelée distance focale. L'axe $(\mathcal{C}, \mathbf{z}_c)$ coupe le plan image \mathcal{P}_π en un point \mathbf{x}_0 appelé point principal, de coordonnées $(0, 0, f)$ dans \mathcal{F}_c (voir figure 1.1).

En l'absence de phénomènes de distorsion, ce point \mathcal{X} se projette sur le plan image \mathcal{P}_π en un point \mathbf{x} de coordonnées $\mathbf{p} = (x_c, y_c, z_c)$ dans \mathcal{F}_c telles que (voir figure 1.1) :

$$x_c = f \frac{X_c}{Z_c} \quad (1.2)$$

$$y_c = f \frac{Y_c}{Z_c} \quad (1.3)$$

$$z_c = f \quad (1.4)$$

Puisque tous les points du plan image \mathcal{P}_π ont pour coordonnée selon l'axe \mathbf{z}_c la distance focale f , il est plus commode de travailler avec des coordonnées normalisées, en ramenant la dernière composante à 1. On considère donc les coordonnées normalisées $\bar{\mathbf{p}} = (x, y, 1)$ telles que

$$\bar{\mathbf{p}} = \frac{1}{f} \mathbf{p}. \quad (1.5)$$

Notons que l'opération de projection s'écrit alors sous la forme matricielle suivante :

$$Z_c \bar{\mathbf{p}} = \mathbf{A} \mathbf{P}_c \quad (1.6)$$

avec

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (1.7)$$

Passage aux coordonnées pixelliques. La dernière étape consiste maintenant à exprimer les coordonnées en pixels. Il faut alors tenir compte de la position du point principal sur le plan image (rarement exactement au centre, du fait que la matrice CCD n'est pas forcément centrée sur l'axe optique), ainsi que de la taille des pixels (qui n'est pas forcément la même selon chaque axe) (voir figure 1.2).

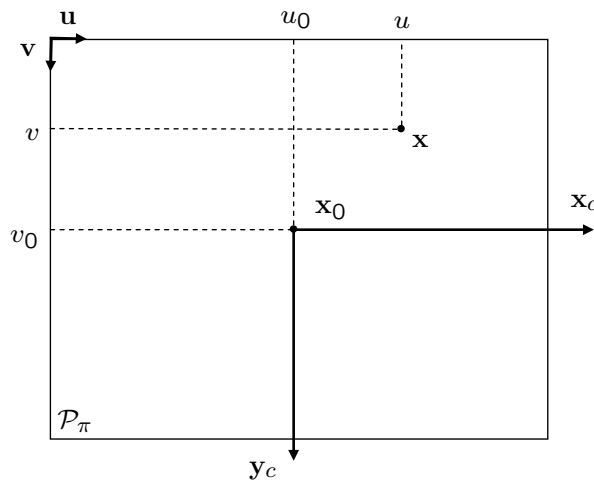


FIG. 1.2 – Schéma du plan image de la caméra.

Les coordonnées pixelliques du point \mathbf{x} , notées u et v , s'expriment alors de la façon suivante :

$$u = u_0 + \frac{x_c}{l_x}, \quad v = v_0 + \frac{y_c}{l_y}, \quad (1.8)$$

où (u_0, v_0) sont les coordonnées pixelliques du point principal, et l_x, l_y la taille des pixels respectivement selon les directions \mathbf{x}_c et \mathbf{y}_c .

Notons que l'on peut à nouveau exprimer cette relation sous forme matricielle :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}' \mathbf{p} \quad (1.9)$$

avec

$$\mathbf{K}' = \begin{bmatrix} 1/l_x & 0 & u_0/f \\ 0 & 1/l_y & v_0/f \\ 0 & 0 & 1/f \end{bmatrix} \quad (1.10)$$

En passant aux coordonnées normalisées $\bar{\mathbf{p}}$ (1.5), on voit que l'on peut "faire entrer" le paramètre f dans la matrice \mathbf{K}' . Cela permet de rassembler tous les paramètres intrinsèques de la caméra au sein d'une même matrice notée usuellement \mathbf{K} . On obtient alors :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}\bar{\mathbf{p}} \quad \text{avec} \quad \mathbf{K} = \begin{bmatrix} f/l_x & 0 & u_0 \\ 0 & f/l_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (1.11)$$

1.1.1.2 Asservissement visuel

Après avoir décrit le capteur utilisé, nous allons voir à présent comment exploiter des données image dans la boucle de commande d'un système robotique.

Type de consigne. Du point de vue de la consigne, on distingue deux grandes familles d'asservissement visuel : les asservissements visuels directs et les asservissements visuels indirects [Weiss 87, Malis 02]. Nous en présentons les principes généraux ici, et nous les illustrerons plus précisément dans le cas des engins volants à la section 1.2.3.

Dans le cas des asservissements visuels directs, les données issues de la caméra sont utilisées dans la partie bas-niveau de la commande. Le correcteur de l'asservissement visuel communique directement avec les actionneurs du robot, les consignes peuvent être des forces, couples, etc. L'asservissement en position du robot (bas niveau) est donc réalisé à partir des informations visuelles. Une telle architecture a été utilisée conjointement avec des caméras rapides, à fréquence d'acquisition élevée (de l'ordre de 500 Hz) dans le cadre des asservissements rapides [Gangloff 00]. Par exemple dans le domaine médical, des applications d'asservissement visuel pour du suivi de cœur battant ont ainsi pu être réalisées [Ginhoux 04].

Dans le cas des asservissements visuels indirects, le correcteur de l'asservissement visuel transmet des consignes de vitesse à un correcteur bas-niveau. Celui-ci se charge de communiquer avec les actionneurs du robot de manière à atteindre la consigne demandée. Dans ce type de commande, la partie bas-niveau est cadencée à une fréquence plus élevée que la boucle de commande de vision (classiquement cadencée à 25 Hz). Ce type de stratégie ne permet pas de tenir compte de phénomènes dynamiques de fréquence élevés dans la boucle d'asservissement visuel. Il a cependant l'avantage d'être plus facilement transposable d'un robot à un autre, puisqu'il repose sur les contrôleurs bas niveau, bien souvent fournis par les fabricants. Ce type de stratégie est le plus utilisé, c'est celui que nous allons considérer ici (voir figure 1.3).

Principe. L'objectif de l'asservissement visuel consiste à commander un système robotique en exploitant des informations visuelles issues des images acquises par une caméra. Cette commande est effectuée en représentant la tâche à réaliser (positionnement, suivi de cible, etc.) par des primitives visuelles extraites des images. Le principe consiste alors à minimiser l'erreur $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$, où \mathbf{s} est un vecteur contenant les k valeurs courantes des informations visuelles choisies, et \mathbf{s}^* leurs valeurs désirées (nous présentons plus loin les différents choix possibles pour \mathbf{s}).

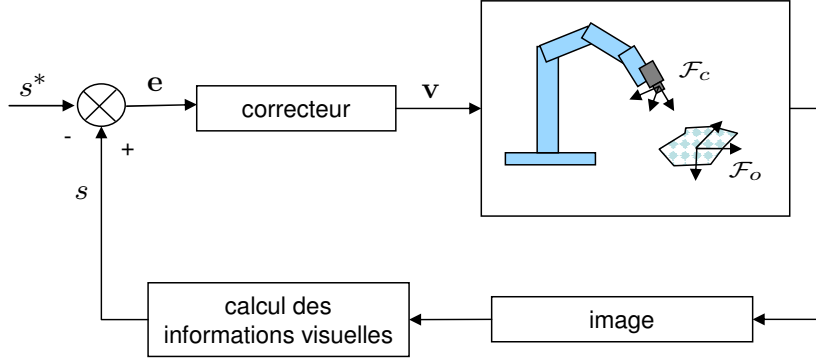


FIG. 1.3 – Schéma bloc de l'asservissement visuel cinématique avec caméra embarquée sur un robot manipulateur.

La matrice d'interaction associée à \mathbf{s} , notée \mathbf{L}_s (de taille $k \times 6$) permet de relier les variations de \mathbf{s} et les mouvements de la caméra par rapport à la scène observée. On a

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v} \quad (1.12)$$

où $\mathbf{v} = (\mathbf{v}, \boldsymbol{\omega})$ est le torseur cinématique de la caméra : il contient les trois vitesses de translation et les trois vitesses de rotation de la caméra par rapport à la cible fixe, exprimées dans le repère de la caméra \mathcal{F}_c . Dans le cas d'une cible fixe, $\dot{\mathbf{s}}^* = 0$ et on obtient donc

$$\dot{\mathbf{e}} = \mathbf{L}_s \mathbf{v}. \quad (1.13)$$

Loi de commande. La loi de commande la plus simple consiste alors à contraindre chaque composante de l'erreur \mathbf{e} à décroître exponentiellement : $\dot{\mathbf{e}} = -\lambda \mathbf{e}$ (où λ est une constante positive). En considérant \mathbf{v} comme entrée de commande, on obtient alors :

$$\mathbf{v} = -\lambda \mathbf{L}_s^+ \mathbf{e} \quad (1.14)$$

ou \mathbf{L}_s^+ est la pseudo-inverse de \mathbf{L}_s , de taille $6 \times k$.

Les valeurs de la matrice d'interaction (et de sa pseudo inverse) ne sont en pratique pas connues précisément à chaque itération. On utilise donc souvent une estimation de \mathbf{L}_s^+ , notée $\widehat{\mathbf{L}}_s^+$:

$$\mathbf{v} = -\lambda \widehat{\mathbf{L}}_s^+ \mathbf{e}. \quad (1.15)$$

Si on remplace alors \mathbf{v} par son expression dans (1.13), on obtient :

$$\dot{\mathbf{e}} = -\lambda \mathbf{L}_s \widehat{\mathbf{L}}_s^+ \mathbf{e}. \quad (1.16)$$

On voit alors que l'on a une décroissance exponentielle de chaque composante de l'erreur \mathbf{e} seulement si $\mathbf{L}_s \widehat{\mathbf{L}}_s^+ = \mathbf{I}_6$.

Notons que si $k = 6$ et que \mathbf{L}_s est inversible, on a alors

$$\mathbf{v} = -\lambda \widehat{\mathbf{L}}_s^{-1} \mathbf{e}, \quad (1.17)$$

et donc une parfaite décroissance exponentielle de l'erreur si $\widehat{\mathbf{L}}_s = \mathbf{L}_s$:

$$\dot{\mathbf{e}} = -\lambda \mathbf{e}. \quad (1.18)$$

Stabilité. Les fonctions de Lyapunov permettent de démontrer la stabilité d'un système dynamique. On peut les considérer comme des fonctions d'énergie généralisées. Démontrer la stabilité asymptotique globale (notée GAS pour *Global Asymptotic Stability*) d'un système revient alors à montrer que pour une fonction d'énergie donnée, l'énergie est toujours dissipée, sauf au point d'équilibre. Ainsi, toutes les trajectoires du système vont converger asymptotiquement vers ce point d'équilibre.

On considère ici la fonction de Lyapunov $\mathcal{L} = \frac{1}{2}|\mathbf{e}|^2$, et on calcule sa dérivée. On obtient :

$$\dot{\mathcal{L}} = \mathbf{e}^\top \dot{\mathbf{e}} \quad (1.19)$$

$$= \mathbf{e}^\top \mathbf{L}_s \mathbf{v}. \quad (1.20)$$

Avec la loi de commande (1.14), on obtient alors :

$$\dot{\mathcal{L}} = -\lambda \mathbf{e}^\top \mathbf{L}_s \widehat{\mathbf{L}}_s^+ \mathbf{e}. \quad (1.21)$$

Montrer que $\mathbf{L}_s \widehat{\mathbf{L}}_s^+ > 0$ permettra alors d'en déduire que le système est GAS.

Notons que si $k = 6$, que \mathbf{L}_s est inversible, et que $\widehat{\mathbf{L}}_s = \mathbf{L}_s$, le système bouclé est GAS, puisqu'on obtient

$$\dot{\mathcal{L}} = -\lambda \mathbf{e}^\top \mathbf{L}_s \widehat{\mathbf{L}}_s^{-1} \mathbf{e} = -\lambda \mathbf{e}^\top \mathbf{e} < 0. \quad (1.22)$$

Dans le cas où $k > 6$, la condition $\mathbf{L}_s \widehat{\mathbf{L}}_s^+ > 0$ ne peut pas être satisfaite, puisque la matrice $\widehat{\mathbf{L}}_s^+$ est de dimension $k \times k$, et de rang 6 au plus. Le système possède alors plusieurs minima locaux, et l'on s'intéresse alors à montrer que le système est localement asymptotiquement stable [Chaumette 06].

En pratique. Au lieu de recalculer $\widehat{\mathbf{L}}_s^+$ à chaque itération (ce qui est parfois impossible si l'on ne fait pas de reconstruction 3D), on se contente souvent d'utiliser la matrice d'interaction calculée à la position désirée, notée $\widehat{\mathbf{L}}_{s^*}$. On a alors la loi de commande

$$\mathbf{v} = -\lambda \widehat{\mathbf{L}}_{s^*}^+ \mathbf{e}. \quad (1.23)$$

Une autre idée, proposée dans [Malis 04], et reposant sur un développement en série de Taylor au second ordre, consiste à utiliser pour loi de commande

$$\mathbf{v} = -\lambda \left(\frac{\widehat{\mathbf{L}}_s + \widehat{\mathbf{L}}_{s^*}}{2} \right)^+ \mathbf{e}. \quad (1.24)$$

1.1.2 Différents types d'asservissement visuel

Nous allons voir maintenant que les informations visuelles \mathbf{s} peuvent être de différents types. Soit ces données sont directement exprimées dans le plan image de la caméra (on parlera alors d'asservissement visuel 2D, ou IBVS pour *Image Based Visual Servoing*), soit les images sont exploitées pour reconstruire la pose (position et orientation) de l'engin (on parlera alors d'asservissement visuel 3D, ou PBVS pour *Pose Based Visual Servoing*). Nous donnons les caractéristiques des matrices d'interaction et les propriétés du système obtenu dans les deux cas. Enfin, d'autres types d'asservissement visuel existent. On peut envisager un asservissement visuel hybride, reposant sur des informations 3D mais avec une reconstruction partielle de la pose, un asservissement visuel reposant sur le flot optique, des stratégies de partitionnement, de basculement entre plusieurs lois de commande, etc. Nous en dresserons un bref aperçu.

1.1.2.1 Asservissement visuel 3D

Considérer la caméra comme un capteur de pose est sans doute l'approche la plus simple. La décision d'ajouter une caméra à un robot découle souvent des limitations des autres capteurs utilisés : biais induits par les capteurs de type odométrique lors de l'intégration des positions successives [Chenavier 92], nécessité de réagir à un environnement inconnu [CoronadoVergara 05], données GPS peu fiables dans des espaces confinés [Royer 04], etc. Les données de vision sont alors utilisées pour fournir une estimation de la pose de l'engin. Cette pose estimée peut alors être exploitée directement [Weiss 87], ou apporter sa contribution à un algorithme intégrant différentes mesures au sein d'un processus de filtrage [Chenavier 92, Wilson 96]. La loi de commande pourra alors rester la même, que la pose soit estimée à partir de la caméra ou de tout autre capteur.

Estimation de la pose. La première étape pour réaliser un asservissement visuel 3D est l'estimation de la pose. Pour ce faire, utiliser une seule caméra dans un environnement inconnu ne suffit pas (incertitude sur la profondeur). Deux stratégies peuvent donc être envisagées :

- considérer que la cible observée est connue, ou
- utiliser deux caméras observant la même scène, et mettre en correspondance les images obtenues.

Nous ne détaillons pas ici les techniques de calcul de pose. On pourra par exemple se référer à [Tsai 87, Lowe 87, DeMenthon 95, Marchand 02] pour le calcul de pose avec une seule caméra, et à [Faugeras 93, Hartley 00] pour les techniques de stéréovision. Notons seulement que la connaissance précise des paramètres de calibration des caméras est primordiale pour une estimation fiable de la pose.

Matrice d'interaction. Selon le choix que l'on fait pour \mathbf{s} , la matrice d'interaction obtenue peut avoir de très intéressantes propriétés [Chaumette 04].

Soit ${}^o\mathbf{t}_c$ la position de la caméra dans un repère fixe \mathcal{F}_o . Soit $\theta\mathbf{u}$ la représentation de la rotation ${}^c\mathbf{R}_c$ entre le repère \mathcal{F}_c et le repère \mathcal{F}_c^* (\mathcal{F}_c^* est le repère lié à la caméra

lorsqu'elle a atteint sa position désirée). On définit

$$\mathbf{s} = ({}^o\mathbf{t}_c, \theta\mathbf{u}). \quad (1.25)$$

La matrice d'interaction associée est alors de la forme

$$\mathbf{L}_s = \begin{pmatrix} {}^o\mathbf{R}_c & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{L}_{\theta\mathbf{u}} \end{pmatrix} \quad (1.26)$$

où $\mathbf{L}_{\theta\mathbf{u}}$ est donné dans [Malis 99] :

$$\mathbf{L}_{\theta\mathbf{u}} = \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_{\times} + \left(1 - \frac{\text{sinc}(\theta)}{\text{sinc}^2(\frac{\theta}{2})}\right)[\mathbf{u}]_{\times}^2. \quad (1.27)$$

La matrice d'interaction est bloc-diagonale : les degrés de liberté de translation sont découplés des degrés de liberté de rotation. Ainsi, les mouvements de translation pourront être commandés en minimisant l'erreur de position, et les mouvements de rotation en minimisant l'erreur d'orientation.

Notons que la pose peut être vue de différentes manières. Choisissons par exemple

$$\mathbf{s} = ({}^c\mathbf{t}_o, \theta\mathbf{u}), \quad (1.28)$$

avec ${}^c\mathbf{t}_o$ la position de la cible observée, exprimée dans le repère lié à la caméra \mathcal{F}_c . La relation classique de cinématique nous donne ${}^c\dot{\mathbf{t}}_o = -\mathbf{v} - \boldsymbol{\omega} \times {}^c\mathbf{t}_o$, et l'on obtient donc :

$$\mathbf{L}_s = \begin{pmatrix} -\mathbf{I}_3 & [{}^c\mathbf{t}_o]_{\times} \\ \mathbf{0}_3 & \mathbf{L}_{\theta\mathbf{u}} \end{pmatrix}. \quad (1.29)$$

On voit cette fois que l'on n'a plus de découplage entre translations et rotations. La forme triangulaire supérieure reste cependant assez intéressante, le découplage est partiel.

Intérêts et inconvénients. Le choix d'informations visuelles \mathbf{s} représentant la pose permet d'obtenir des matrices d'interaction creuses. Avec (1.27), on a un découplage total entre les translations et les rotations. Dans ce cas, on peut séparer le problème en deux et s'intéresser aux translations indépendamment des rotations. De plus, une loi de commande assurant une bonne trajectoire des composantes de \mathbf{s} permet d'assurer également une bonne trajectoire de la caméra dans l'espace 3D, puisque \mathbf{s} représente directement la pose. Estimer la pose à partir des données de vision permet aussi d'intégrer simplement ces mesures dans des schémas de commande existants, sans concevoir de loi de commande spécifique au cas de l'utilisation de la caméra.

Cependant, cette technique présente deux inconvénients majeurs. D'une part, la phase préalable d'estimation de pose n'est pas triviale, elle nécessite une très bonne calibration des caméras, et pose des problèmes de mise en correspondance, de précision, etc. D'autre part, selon le choix que l'on a fait pour \mathbf{s} , on peut soit assurer que la caméra se déplace en ligne droite dans l'espace 3D (avec (1.25)) — mais alors rien n'assure que la cible reste dans le champ de vision de la caméra — soit assurer que l'origine de la cible observée décrit une ligne droite dans le plan image (avec (1.28)) — mais alors la trajectoire 3D de la caméra n'est plus une ligne droite [Chaumette 07]. Ainsi, selon l'application que l'on souhaite réaliser, on préférera l'une ou l'autre approche.

1.1.2.2 Asservissement visuel 2D

En considérant des informations visuelles 2D, exprimées dans le plan image, on peut s'affranchir de l'étape de calcul de pose, et assurer une bonne trajectoire de la cible dans le plan image.

Le choix le plus simple consiste à utiliser des points de la cible considérée. Pour un point de l'image \mathbf{x} de coordonnées $\mathbf{p} = (x, y)$ dans le repère \mathcal{F}_c , image d'un point \mathcal{X} de la cible, de coordonnées (X, Y, Z) dans le repère lié à la caméra \mathcal{F}_c , la matrice d'interaction associée au vecteur d'informations visuelles $\mathbf{s} = (x, y)$ a la forme suivante [Hutchinson 96] :

$$\mathbf{L}_{\mathbf{p}} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix}. \quad (1.30)$$

Supposons à présent que la cible considérée est un carré. On construit le vecteur d'informations visuelles \mathbf{s} en "empilant" les coordonnées des quatre sommets : $\mathbf{s} = (x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$. La matrice d'interaction correspondante a la forme suivante :

$$\mathbf{L}_{\mathbf{s}} = \begin{bmatrix} \mathbf{L}_{\mathbf{p}_1} \\ \mathbf{L}_{\mathbf{p}_2} \\ \mathbf{L}_{\mathbf{p}_3} \\ \mathbf{L}_{\mathbf{p}_4} \end{bmatrix}. \quad (1.31)$$

S'il n'y a pas de minima locaux et que $\widehat{\mathbf{L}}_{\mathbf{s}}^+ = \mathbf{L}_{\mathbf{s}}^+$, la loi de commande

$$\mathbf{v} = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^+ (\mathbf{s} - \mathbf{s}^*) \quad (1.32)$$

assure que les informations visuelles \mathbf{s} convergent exponentiellement vers leurs valeurs désirées \mathbf{s}^* — si cette trajectoire est réalisable en pratique, comme nous allons le voir par la suite. Chacun des points se déplace alors dans le plan image en ligne droite en direction de sa position désirée. Si la position initiale de la caméra est suffisamment proche de la position désirée — de manière à éviter les minima locaux — cela correspond à la convergence de la caméra vers sa position désirée [Chaumette 98]. Cependant, ce sont les points de l'image qui ont une trajectoire déterminée, et non pas la caméra. Ainsi, il se peut que des mouvements inattendus de la caméra soient générés : par exemple, une rotation selon l'axe optique s'accompagne de mouvements de translation selon l'axe optique (problème d'avance-retrait). Le cas le plus extrême est le problème bien connu de la rotation à 180 degrés (voir figure 1.4) : selon le choix des informations visuelles, leur déplacement rectiligne dans l'image peut conduire à rejeter la caméra à l'infini [Chaumette 98].

Afin d'obtenir un bon comportement de la caméra, il est nécessaire de choisir convenablement les informations visuelles utilisées. On peut penser à utiliser les coordonnées cylindriques des points de la cible dans l'image, le centre de gravité de la cible, les coordonnées de droites (coordonnées classiques (ρ, θ) , coordonnées de Plücker [Andreff 00, Mahony 05a]), les coordonnées du point de fuite et de la ligne d'horizon [Rives 04, Ettinger 02], les moments d'ordre 1 de la cible projetée sur une sphère

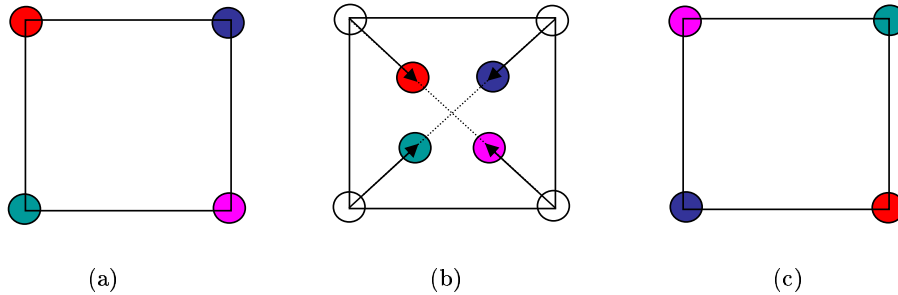


FIG. 1.4 – Illustration du problème d'avance-retrait : (a) : position initiale de l'image du carré. (c) : position désirée de l'image du carré (correspondant à une rotation de la caméra de 180 degrés). (b) : position intermédiaire lorsque les quatre sommets se déplacent en ligne droite (selon les flèches) : les quatre points convergent au centre, ce qui correspond à reculer la caméra, et ce jusqu'à l'infini.

[Hamel 02], etc. Choisir des informations visuelles aussi représentatives que possible de la position 3D de la caméra par rapport à la cible permet, à travers le comportement des informations visuelles, d'assurer un bon comportement de la caméra. Par exemple, l'exploitation des moments a permis de concevoir des informations visuelles particulièrement intéressantes [Tahri 05]. Elles présentent en effet des propriétés de découplage : matrice d'interaction creuse, ce qui fait qu'une information visuelle n'est liée qu'à un degré de liberté. Les informations visuelles évoluent alors de la même manière que les composantes de la pose de la caméra dans l'espace 3D. Ainsi, en choisissant bien les informations visuelles images, il est possible d'obtenir un très bon comportement dans l'espace 3D, bien que la commande contraigne uniquement les informations visuelles 2D. En plus du choix judicieux des informations visuelles, on peut également envisager des schémas de commande partitionnés, où les composantes posant potentiellement problème sont traitées séparément [Corke 01].

1.1.2.3 Autres types d'asservissements visuels

Outre les asservissements visuels 3D et 2D, d'autres variantes de l'asservissement visuel existent : l'asservissement visuel 2D 1/2, qui prend en compte des informations visuelles 2D et 3D [Malis 99, Deguchi 98, Morel 99, Cervera 03], et l'asservissement visuel $d2D/dt$, qui considère le mouvement dans l'image (*optic flow*). Ainsi, l'*optic flow* a été utilisé dans [Crétual 01], notamment pour des applications de navigation dans un couloir, pour éviter les collisions [Hrabar 05]. Par ailleurs, des stratégies de *switching* entre asservissement visuel 2D et 3D ont également été développées [Deng 05, Gans 07] en vue d'éviter des singularités et minima locaux, ou pour obtenir des propriétés de stabilité.

1.2 La commande par vision d'engins volants

Rendre les engins volants utilisables par des opérateurs non expérimentés demande une plus grande automatisation. Outre la centrale inertielle (appelée aussi IMU pour *Inertial Measurement Unit*), quasi incontournable pour permettre à l'engin de naviguer correctement, le GPS (*Global Positioning System*) s'avère un capteur fort utile lorsque l'on souhaite positionner l'engin par rapport à son environnement. Cependant, d'une part, le GPS ne fournit qu'une position absolue : par exemple dans le cas où une procédure d'atterrissage d'urgence est nécessaire, le GPS est inutile pour déterminer la zone adéquate la plus proche [Fitzgerald 05, Mejias 06a]. D'autre part les mesures obtenues par GPS ne sont pas toujours fiables notamment lorsque l'engin évolue dans des espaces confinés, tels que l'environnement urbain, ou l'intérieur de bâtiments, à des fins d'observation par exemple [Green 04, Hrabar 05]. Il apparaît alors clairement que l'emploi de caméra(s) embarquée(s) s'avère très utile voire incontournable pour réaliser ces différentes tâches de manière automatique, sans information a priori.

Par ailleurs, les phases de décollage et d'atterrissage d'un avion, qui s'avèrent particulièrement critiques, nécessitent un travail important de l'opérateur. L'ILS (*Instrument Landing System*) est un système couramment utilisé pour réaliser un atterrissage automatique. D'une part, la mise en place de caméra(s) embarquée(s) pourrait être mise à profit afin d'augmenter la fiabilité des systèmes existants, en apportant des informations supplémentaires au système. D'autre part, l'ILS n'est utilisable qu'en présence des installations nécessaires au sol. Là-encore, l'utilisation de caméra(s) embarquée(s) pourrait permettre de s'affranchir de cette contrainte en réalisant un système complètement autonome.

1.2.1 Divers engins volants

Les travaux de commande par vision d'engins volants ont porté sur un large panel d'engins. Nous présentons ici les différents types d'engins volants considérés, ainsi que quelques projets ayant utilisé un capteur de vision dans le but d'accroître l'autonomie de tels systèmes.

1.2.1.1 Hélicoptères

De nombreuses équipes de recherche se sont intéressées aux hélicoptères miniatures. Ces engins possèdent la propriété de décoller et atterrir verticalement (abrégiée usuellement VTOL, pour *Vertical Take Off and Landing*), et d'effectuer des vols stationnaires, ce qui leur permet d'évoluer dans des espaces limités, et d'être capables de rester en vol au-dessus d'une zone précise [Saripalli 03]. Le plus souvent, ces engins de petite taille sont téléopérés, mais la commande manuelle nécessite alors l'intervention d'un opérateur humain expérimenté [Guenard 06b]. Rendre de tels engins capables d'évoluer de manière autonome permettrait de s'affranchir de ce besoin, et ainsi d'élargir leurs domaines d'application. En particulier, leur autonomie et leur petite taille rendrait possible l'exploration de zones potentiellement dangereuses ou difficiles d'accès pour un opérateur humain [Hamel 06]. Outre les applications militaires, un large panel d'application civiles

et industrielles peut être envisagé [Car, Com]. Ainsi, de petits hélicoptères autonomes pourraient être déployés utilement pour des missions de surveillance et de reconnaissance tels que la détection de feux de forêt [Ollero 05], la supervision du réseau routier [Doherty 00], l'aide aux secours pour la recherche de victimes, l'inspection de ponts, de barrages ou de bâtiments difficilement accessibles [Caballero 05, LeBras 06, Hamel 06], l'inspection des lignes électriques [Mejias 06a], le traitement des cultures, la cartographie d'un terrain [Ollero 05], etc. Afin de mener à bien ces missions, il faut que l'hélicoptère soit capable de décoller, se diriger vers la zone voulue, éviter les obstacles éventuels, localiser un objet d'intérêt, se positionner par rapport à cet objet et éventuellement le suivre, enregistrer ou transmettre les images acquises, revenir à la station sol, localiser un endroit où atterrir en toute sécurité, et finalement atterrir. Pour répondre à ces besoins, il s'agit d'une part de mettre au point des lois de commande bas-niveau facilitant la commande manuelle de l'engin afin qu'un opérateur novice puisse l'utiliser sans risque [Hamel 06]. D'autre part, des lois de commande haut-niveau pourront remplacer la commande manuelle afin de permettre un comportement complètement autonome [Saripalli 03] par exemple dans le cas de l'atterrissage d'urgence, de l'évitement d'obstacle, du suivi précis de cible par vision, etc.

Nous présentons bien sûr dans cette partie les projets utilisant des hélicoptères miniatures de type "classique", avec un rotor principal à axe vertical et un rotor de queue. D'autres engins, que l'on peut également qualifier d'hélicoptères au sens où ce sont des « aéronefs à voilure tournante dont le ou les rotors procurent à eux seuls la propulsion et la sustentation pendant toutes les phases du vol » [Wikihelico], font également l'objet de travaux de recherche. Ces engins, au comportement similaire aux hélicoptères, permettent de répondre à des contraintes supplémentaires en terme de sécurité, de masse de l'engin, de mise en œuvre plus facile etc. Nous décrirons donc les travaux portant sur un hélicoptère à hélice carénée, d'autres concernant un hélicoptère à hélices coaxiales, et enfin le cas des hélicoptères à quatre hélices, appelés quadrirotors.

Hélicoptères "classiques". De nombreux projets ont cherché à intégrer des capteurs de vision sur des modèles réduits d'hélicoptères de type "classique" (équipés d'un rotor principal à axe vertical et d'un rotor de queue, voir figure 1.7 par exemple).

Le projet AVATAR (*Autonomous Vehicle Aerial Tracking And Reconnaissance*) [Ava], mené à l'université de Californie du sud (*Robotic Embedded Systems Laboratory, University of Southern California*) à Los Angeles aux États Unis, s'est intéressé à l'atterrissage sur une cible détectée par la caméra embarquée (cible fixe [Saripalli 03], ou en mouvement [Saripalli 07]). Le déplacement de l'engin dans un "couloir" a également été considéré, afin de rester à une distance suffisante des parois latérales, tout en évitant les obstacles [Hrabar 05]. Des travaux ont été menés à l'université polytechnique de Madrid (*Universidad Politécnica*) en Espagne, dans le but de placer l'engin face à une fenêtre [Mejias 06b] (voir figure 1.5). Enfin, l'utilisation d'une caméra omnidirectionnelle a été envisagée pour le positionnement de l'engin par rapport à des cibles visuelles [Hrabar 03].

L'atterrissage automatique d'hélicoptère a également été considéré par le projet BEAR (*BERkeley AeRobot*) de l'université de Berkeley [Bea, Shakernia 99, Sharp 01].



FIG. 1.5 – L'hélicoptère du projet AVATAR, *University of Southern California*, Los Angeles, États Unis.

Une équipe du CSIRO (*Commonwealth Scientific and Industrial Research Organisation*) à Brisbane en Australie, développe un mini hélicoptère, dans le cadre de missions civiles et en premier lieu l'inspection des lignes électriques [Csi] (voir figure 1.6).



FIG. 1.6 – L'hélicoptère du *CSIRO*, *ICT Center*, Brisbane, Australie. (Envergure : 2.1 m).

Le projet COLIBRI, mené à l'université polytechnique de Madrid en Espagne, se place également dans le contexte de l'inspection de lignes électriques par un petit hélicoptère, et s'intéresse en particulier à la détection d'une zone dégagée adaptée à un atterrissage forcé. L'environnement est alors inconnu, les lignes électriques constituant de plus des obstacles particulièrement dangereux pour l'hélicoptère [Col, Mejias 06a].

Le projet de véhicule volant autonome du WITAS (*Wallenberg Laboratory for Information Technology and Autonomous Systems, Unmanned Aerial Vehicle Project*) à l'université de Linköping en Suède, s'est principalement intéressé à la mise au point d'un hélicoptère autonome dédié à la surveillance du réseau routier [Wit, Doherty 00] (voir figure 1.7). On peut également citer l'hélicoptère HUMMINGBIRD de l'université de Stanford, utilisé pour du suivi de véhicule au sol [Hum, Rock 98].

Les recherches menées à l'université de Carnegie Mellon [Car, Amidi 99] ont pour but d'utiliser un capteur de vision pour diverses tâches telles que la stabilisation en vol, le suivi de trajectoire, l'atterrissage et la reconnaissance d'objets d'intérêt.



FIG. 1.7 – L'hélicoptère du projet WITAS à l'université de Linköping en Suède.

Utiliser une caméra pour remplacer le GPS lorsqu'il s'avère inopérant [Wu 05] (institut de technologie de Géorgie, *Georgia tech*), ou pour pallier les éventuelles pannes survenues aux autres capteurs embarqués [DeWagter 05] (université de technologie de Delft, Hollande) a également été proposé. [DeWagter 05] s'intéresse également aux tâches de détection de cible visuelle, construction d'une carte du terrain, et atterrissage par vision.

Le projet HELINSPEC de l'université de Séville (Espagne) [Caballero 05], a pour objet l'inspection de bâtiments et structures difficiles d'accès à l'aide d'un petit hélicoptère équipé d'une caméra.

Citons également le projet européen COMETS [Com, Ollero 04, Ollero 05], qui considère des applications de détection et localisation de feux de forêt, ainsi que la cartographie de terrain, en incluant également des techniques de coopération entre des hélicoptères et un dirigeable etc.

Enfin, l'un des objectifs du projet européen PEGASE (*helicoPter and aEronef navigation Airborne System Experimentations*) [Pega], consiste à proposer de nouvelles solutions pour l'atterrissage et le décollage automatique des hélicoptères, à partir de données de vision.

Hélicoptères à hélice carénée. Des contraintes d'encombrement et de sécurité ont conduit au développement d'engins à hélice carénée, où les pales sont confinées à l'intérieur de la structure mécanique de l'engin. Le drone HoverEye (voir figure 1.8), développé par la société Bertin Technologies (Montigny-le-Bretonneux, France), en est un exemple. Cet engin, équipé de deux rotors contrarotatifs sur le même axe et de gouvernes dans le flux d'air des rotors, est capable de décoller et atterrir verticalement. Parmi les atouts du HoverEye, on peut citer la sécurité, la rapidité de mise en œuvre, et la stabilité en vol stationnaire. Ainsi, le rôle de l'HoverEye est de transmettre à un expert les images recueillies, en étant en quelque sorte "l'œil déporté de l'expert", pour l'inspection de zones à risques ou d'ouvrages d'art par exemples. Les problématiques de mise au point de modes autonomes, et concernant la nécessité de résister au vent ont été étudiées

[Pfimlin 06]. La stabilisation par vision du HoverEye devant une cible a également été réalisée [LeBras 06].



FIG. 1.8 – Le drone HoverEye de Bertin Technologies. (Hauteur 70 cm, diamètre 50 cm, propulsion électrique, 3 kg) [Pfimlin 06].

Hélicoptères à hélices coaxiales. L'hélicoptère à hélices coaxiales contra rotatives (voir figure 1.9) présente des avantages en terme de stabilité. À l'université Tohoku, Sendai, au Japon, un système est développé pour commander un tel hélicoptère à partir de deux caméras placées au sol [Yoshihata 07].



FIG. 1.9 – L'hélicoptère à hélices coaxiales XRB Lama de Hirobo.

L'université de Bangor, en Écosse a développé un projet autour de la surveillance des lignes électriques par un robot téléopéré de ce type (voir figure 1.10). L'équipe s'est intéressée à l'utilisation de la vision pour l'évitement d'obstacle et la planification de trajectoire [Ban, Williams 01].

Quadrirotors. Les quadrirotors, engins à quatre rotors (nommés aussi *X4-flyer*) ont des propriétés similaires aux hélicoptères au sens où ils sont capables de décoller et atter-



FIG. 1.10 – L'hélicoptère à hélices coaxiales de l'université de Bangor.

rir verticalement, et de réaliser des vols stationnaires. Leurs champs d'applications sont donc très similaires. Comme leur nom l'indique, les *X4-flyers* sont constitués de quatre moteurs contrarotatifs placés à chaque extrémité d'une croix (en forme de X) entraînant quatre hélices et permettant à l'ensemble de voler (voir figure 1.11). Contrairement au cas de l'hélicoptère, les hélices de ce genre d'engin sont à pas fixe (l'orientation de ses pales est constante). C'est un de ses intérêts. En effet, pour changer de direction, l'hélicoptère modifie l'inclinaison de ses pales, via un dispositif mécanique complexe. Par contre, les hélices du quadrirotor sont fixes, et c'est la différence de vitesse de rotation des différents moteurs qui permet à l'engin de se déplacer. Le quadrirotor a également l'avantage d'avoir une charge utile plus importante que l'hélicoptère, et une très bonne manœuvrabilité [Altuğ 05].



FIG. 1.11 – Le quadrirotor du projet ROBVOLINT. (Envergure : 50 cm).

Le projet français ROBVOLINT (Robot Volant d'INTérieur) s'est intéressé au quadrirotor en environnement d'intérieur. Le scénario envisagé consistait à explorer un tunnel ou un immeuble. Pour cela, un prototype de quadrirotor a été développé, et diverses fonctionnalités ont été mises au point, dans le but de faciliter son maniement et de le rendre partiellement autonome [Rob, Hamel 06].

On peut mentionner les travaux réalisés à l'université de Cambridge [Kemp 05] qui traitent de l'estimation de la pose pour un quadrirotor, ou encore le MIT (*Massachusetts Institute of Technology, Aerospace Controls Laboratory*), qui propose d'utiliser une

cible à effet de Moiré pour réaliser le calcul de pose [Tournier 06]. L'objectif à long terme du MIT est de proposer un système de plusieurs véhicules (volants et au sol) capables d'évoluer de manière autonome pour des missions longues. Outre la réalisation de la mission en elle-même, des aspects de détection automatique de pannes, d'interface utilisateur, de gestion de plusieurs engins, etc. sont entre autres étudiés.

Une équipe de l'université Cornell (*Cornell University*) à Ithaca, dans l'État de New York, s'intéresse à l'utilisation de plusieurs quadrirotors pour former un réseau d'antennes et transmettre ainsi plus efficacement les informations recueillies par les drones [Breheny 03]. Pour ce faire, ils se sont également intéressés à l'utilisation de caméra comme capteur déporté pour le positionnement des engins [Earl 04].

Citons également les travaux réalisés au laboratoire GRASP (*General Robotics Automation Sensing and Perception*) à l'université de Pennsylvanie [Altug 05], et les études menées au laboratoire Heudiasyc de l'université de Technologie de Compiègne [Romero 06], dont l'objet est d'étudier la commande des quadrirotors, en intégrant des données issues de la vision.

1.2.1.2 Dirigeables

Tout comme les hélicoptères et les quadrirotors, les dirigeables (aussi nommés LTA pour *Lighter Than Air*) sont capables d'atterrir et décoller verticalement, et d'évoluer en vol stationnaire. En outre, en comparaison avec les hélicoptères, ils sont moins bruyants, provoquent moins de turbulences, se caractérisent par une consommation énergétique plus faible, une charge utile plus importante, et une plus grande endurance. Notons cependant qu'ils sont moins facilement manœuvrables que les hélicoptères [Elfes 98]. Les dirigeables ont donc des atouts pour des applications de transport de charges particulièrement lourdes, et dans le cadre de missions de longue durée. Leur utilisation pour l'exploration planétaire, ou en tant que plates-formes stratosphériques est à l'étude. Des missions de surveillance et d'observation de longue durée sont également envisageables. Enfin, les dirigeables sont parfois utilisés à des fins publicitaires [Lacroix 05].

Le LAAS (Laboratoire d'Architecture et d'Analyse des Systèmes) à Toulouse travaille au développement d'un dirigeable. L'idée est de mettre au point une plate-forme expérimentale polyvalente et générique (voir figure 1.12). L'engin est équipé d'un système de stéréo-vision. Des solutions de navigation autonome et de cartographie de terrain inconnu (SLAM : *Simultaneous Localisation And Mapping*), reposant sur un filtre de Kalman étendu ont été mises au point [Hygounen 04].

Le projet international AURORA (*Autonomous Unmanned Remote Monitoring Robotic Airship*) implique des laboratoires français (l'INRIA de Sophia Antipolis), brésilien (*Robotics and Computer Vision Laboratory-LRV/ITI* à Campinas) et portugais (l'IST — *Instituto Superior Técnico* — de Lisbonne). Il a également pour but de développer un dirigeable doté de grandes capacités d'autonomie durant toutes les phases classiques de vol (décollage, vol stationnaire, vol de croisière et atterrissage). L'étude et la surveillance de l'environnement, du climat, de la biodiversité sont les principales applications envisagées. Le dirigeable semble en effet tout indiqué pour mener à bien ces missions : elles ne requièrent pas de déplacements rapides, et le dirigeable a l'avantage de ne per-



FIG. 1.12 – Le dirigeable Karma du LAAS (8 m de long, 15 m^3 , 1.90 m de diamètre max.).

turber l'environnement que très peu, ce qui améliore la fiabilité des mesures obtenues [Elfes 98]. Des travaux utilisant des données de la caméra embarquée ont été réalisés pour l'atterrissage [Azinheira 02] et le suivi de route [Silveira 02, Rives 04].

Le projet international NARVAL (*Navigation of Autonomous Robots via Active Environmental Perception*) a pour objet l'utilisation de la vision pour la commande de robots sous-marins [Nar]. Dans [Zwaan 00], l'objectif est de suivre par vision une station d'amarrage. Afin de tester et mettre au point les développements réalisés, des expérimentations sont réalisées à l'IST de Lisbonne sur un petit dirigeable d'intérieur (voir figure 1.13) dont le comportement dynamique est assez proche de celui d'un robot sous-marin [Zwaan 00]. Ce dirigeable a l'avantage d'être plus facile à mettre en œuvre que le robot sous-marin.



FIG. 1.13 – Le dirigeable du projet NARVAL.

Enfin, le laboratoire GRASP de l'université de Pennsylvanie s'est intéressé à l'asservissement visuel d'un petit dirigeable d'intérieur, pour valider des développements théoriques de commande par vision pour les engins volants [Zhang 99]. L'objectif était de positionner le dirigeable au-dessus d'une cible visuelle connue.

1.2.1.3 Avions

Nous nous intéressons à présent aux avions — « aéronefs plus lourd que l'air, entraînés par un organe moteur (dans le cas d'un engin sans moteur, on parle de planeur), dont la sustentation en vol est obtenue principalement par des réactions aérodynamiques sur des surfaces qui restent fixes dans des conditions données de vol » [Wikavion].

Pour un avion, le vol stationnaire est impossible, puisqu'il doit se déplacer afin de se maintenir en vol. De plus, les phases de décollage et atterrissage nécessitent l'installation d'une piste. L'espace et les infrastructures nécessaires à leur déploiement sont donc plus importants que pour les engins à décollage et atterrissage vertical. Par rapport aux hélicoptères par exemple, cela représente une contrainte, mais les petits avions ont l'avantage de consommer moins d'énergie que les hélicoptères, et donc de pouvoir effectuer des vols plus longs, à moindre coût [Elfes 98]. Les petits avions semblent donc bien adaptés pour certaines applications, telles que la surveillance du réseau routier par exemple.

Avions miniatures. À Berkeley, le C3UV (*Center for Collaborative Control of Unmanned Vehicles*) de l'université de Californie travaille sur des petits avions (voir figure 1.14). L'objectif est de rendre autonome, autant que possible, une flotte de petits avions. Dans ce but, un engin a été équipé d'une caméra, et une application de suivi de route a été étudiée [Frew 04].



FIG. 1.14 – Une petit avion (2.80 mètres d'envergure) du C3UV à Berkeley.

Une étude a été menée en partenariat entre INRIA Sophia-Antipolis et l'IST de Lisbonne, pour l'asservissement visuel d'un petit avion autonome. Une comparaison de plusieurs lois de commande d'asservissement visuel pour l'atterrissage automatique d'un petit avion a été réalisée en simulation [Rives 02].

Un projet de petit avion — sur lequel une caméra devait être embarquée — a également été initié au LAAS à Toulouse (voir figure 1.15).

À l'Université de Floride (Gainesville, États-Unis), le laboratoire MAV (*Micro Air Vehicle Laboratory*) conçoit des drones miniatures (dimensions de l'ordre de 15 cm). Une stratégie de commande par vision de ces engins a été proposée dans [Ettinger 02].



FIG. 1.15 – Le petit avion Lhassa du LAAS. (3 mètres d'envergure, 6 kg)

Des études sont également menées au laboratoire de vision par ordinateur d'Orlando (*University of Central Florida*, États-Unis), pour la navigation d'un avion à partir de données visuelles. L'objectif est d'automatiser le suivi d'une cible mobile grâce à une caméra embarquée [Rafi 06, Flo].

Dans le même ordre d'idée, les chercheurs de *Brigham Young University*, à Provo (Utah, États-Unis) s'intéressent à la mise au point d'une caméra embarquée sur un support *pan-tilt* (c'est-à-dire avec deux degrés de libertés de rotation), et à la génération de trajectoire pour suivre une cible avec un petit avion [Quigley 05b] (voir figure 1.16). L'objectif est d'alléger le travail de l'opérateur, en automatisant les tâches de décollage, d'atterrissage, et des opérations de recherche par vision [Quigley 05a].

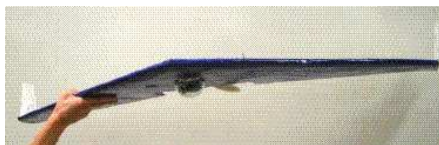


FIG. 1.16 – Le petit avion de *Brigham Young University* (1 mètre d'envergure).

Avions de taille standard. En ce qui concerne l'utilisation de caméras sur des avions de taille standard (non modèles réduits), des travaux portent sur le traitement d'image visant à détecter la piste d'atterrissage [Soni 94, Huertas 90, Han 01]. Des aspects de commande sont évoqués dans [Valasek 02, Fravolini 03], pour des applications de ravitaillement en vol automatique. Des systèmes de vision utilisant de la lumière structurée ont également été conçus pour permettre une meilleure localisation d'un avion par rapport à l'autre lors des phases de ravitaillement [Junkins 99, Valasek 02]. Notons également les travaux décrits dans [Chatterji 96], pour l'estimation de la position et de l'orientation de l'avion par rapport à une piste d'atterrissage. Le but est ici d'améliorer les conditions d'atterrissages de nuit, en utilisant les indicateurs lumineux de la piste pour estimer la position de l'engin. Enfin, le projet européen PEGASE [Pega], s'intéresse à l'automatisation des tâches de décollage et d'atterrissage des avions, à partir de données de vision.

Planeur. Des expérimentations de commande par vision sur un planeur ont été réalisées à l'institut de technologie de Géorgie (voir figure 1.17). Le planeur n'ayant pas de système de propulsion, ce choix permet de réduire les problèmes liés à la maintenance de l'appareil. L'unique capteur embarqué sur le planeur est une caméra. L'objectif est d'établir une loi de commande à partir des données de vision, afin de permettre au planeur de passer par une fenêtre ouverte [Proctor 04].

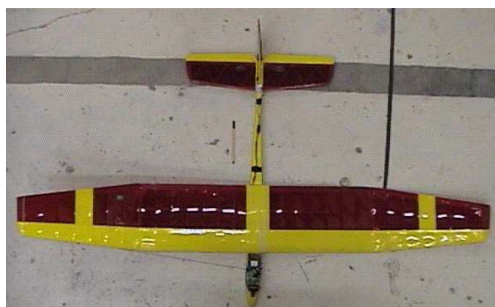


FIG. 1.17 – Le planeur de l'institut de technologie de Géorgie (2 mètres d'envergure) [Proctor 04].

1.2.1.4 Drones bio-inspirés

Afin de concevoir des lois de commande adaptées aux engins volants, une approche consiste à prendre en compte les contraintes supplémentaires liées au fait que l'engin vole, au sein de lois de commandes existant pour d'autres systèmes. Une autre approche, dite "bio-inspirée", consiste à tirer profit de l'observation de la nature, en particulier des insectes, pour ensuite concevoir des systèmes de vision et des lois de commande similaires.

Une équipe du LIS (*Laboratory of Intelligent Systems*, à l'EPFL — École Polytechnique Fédérale de Lausanne — en Suisse) développe un drone d'intérieur ultra-léger (10 grammes, voir figure 1.18). Mise à part la commande de l'altitude, qui est radio-commandée, il est complètement autonome. Il régule sa vitesse en se basant sur les données fournies par un anémomètre conçu spécialement. Il est également équipé de gyroscopes et de petites caméras CMOS à une dimension (environ 80 pixels par caméra) et à grand champ de vue (de l'ordre de 120 degrés). L'une d'elle est orientée vers l'avant et permet à l'engin d'éviter automatiquement les obstacles [Zufferey 06].

L'Université Drexel de Philadelphie propose des systèmes d'évitement d'obstacle et d'atterrissage à partir d'une caméra embarquée sur un micro-drone semblable à celui de l'EPFL [Green 04]. L'application envisagée est du type exploration, reconnaissance, recherche pour le sauvetage, dans des environnements contraints tels que des caves, des forêts, des tunnels, etc.

À l'Université de Sydney, des travaux portant sur la réalisation d'un capteur de vision de petite taille, à faible consommation, grand champ de vue, haute résolution et faible distorsion sont réalisés. L'objectif est d'en doter un petit engin volant, dans le

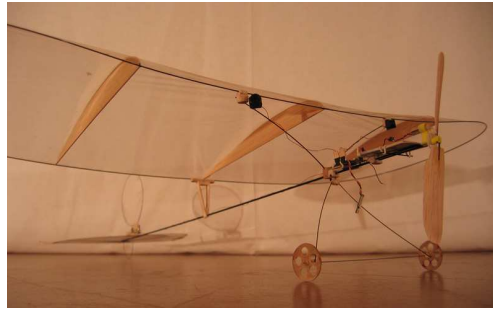


FIG. 1.18 – Le micro-drone ultra léger (30 centimètres pour seulement 10 grammes) de l'EPFL.

cadre d'une approche bio-inspirée [Tisse 05].

Au laboratoire Mouvement et perception (CNRS/Université de la Méditerranée) à Marseille, des travaux sont également menés dans le domaine de la robotique bio-inspirée. L'objectif est de reproduire sur des micro-robots le fonctionnement des insectes, en particulier en ce qui concerne leurs capteurs de vision et les lois de commande qui en découlent [Ruffier 04, Serres 06].

1.2.2 Différents types d'informations visuelles

Dans les paragraphes précédents, nous avons proposé un aperçu des différents engins volants pour lesquels une commande par vision a été mise en œuvre et des applications possibles. Nous allons voir maintenant comment les deux types d'informations visuelles (2D et 3D, présentées à la section 1.1.2) sont utilisées dans le cadre d'asservissement visuel d'engins volants.

1.2.2.1 Informations visuelles 3D

La technique la plus utilisée consiste à se servir de caméra(s) pour estimer la pose (position et orientation) de l'engin par rapport à une cible visuelle observée [Shakernia 99]. Dans ce cas, la vision remplace ou complète les informations issues du GPS. Le contrôleur n'a pas besoin d'être conçu spécialement pour convenir au capteur utilisé, que la pose soit estimée à partir du GPS ou de caméra(s). On est dans le cadre de l'asservissement visuel 3D (voir section 1.1.2.1) Afin de calculer la pose à partir de données de vision, deux stratégies peuvent être envisagées. Soit on utilise plusieurs caméras, et la mise en correspondance des images vues par chaque caméra permet d'estimer la pose de la caméra par rapport à l'objet observé. Soit on ne considère qu'une seule caméra, mais alors les informations de l'image sont insuffisantes pour définir complètement la pose, et il est nécessaire de tenir compte d'information(s) supplémentaire(s) (par exemple des données d'autres capteurs ou une connaissance a priori sur la cible visuelle).

Plusieurs caméras. Parmi les travaux ayant utilisé plusieurs caméras pour estimer la pose d'un engin volant, on peut citer [Altuğ 05], qui considère une caméra embarquée

sur un quadrirotor, et une caméra au sol de type PTZ (*pan/tilt/zoom* : la caméra a un zoom réglable, et est montée sur une tourelle commandable en roulis et en tangage).

Plus classiquement, une paire de caméras stéréoscopique est embarquée sur l'engin volant [Amidi 99, Rock 98, Buskey 03, Wang 05]. Dans [Amidi 99], après avoir mis en correspondance les deux images obtenues, le mouvement 3D d'un hélicoptère est estimé à partir du mouvement détecté dans les images, combiné à des mesures fournies par un capteur de rotation. Cet "odomètre visuel" permet ainsi d'estimer la pose relative de l'hélicoptère par rapport à une cible visuelle. [Rock 98] propose d'utiliser une méthode classique de triangulation pour obtenir la pose d'un hélicoptère. [Buskey 03] limite l'utilisation de la paire stéréoscopique à l'estimation de l'altitude, et préfère combiner des informations visuelles à des données inertielles pour estimer les autres paramètres de la pose. Dans [Hrabar 05], l'estimation de la pose obtenue à partir d'une paire stéréoscopique est utilisée pour la détection et l'évitement d'obstacles.

Une autre approche utilisant deux caméras permet de naviguer dans un couloir. Pour cela, on utilise le flot optique mesuré dans deux images : l'une obtenue avec une caméra orientée vers la gauche et l'autre vers la droite [Hrabar 05, Serres 06]. Ces techniques sont similaires au comportement des abeilles, qui peuvent voler dans un couloir en équilibrant le flot optique des images des deux murs [Srinivasan 96]. Intuitivement, le principe est que, pendant que l'on avance dans un couloir, si l'on s'approche de l'un des murs, le flot optique mesuré dans l'image de ce mur va augmenter, et celle du mur opposé diminuer. Équilibrer les deux permet de rester à équidistance des deux murs.

Une seule caméra. Bien souvent, la caméra n'est pas le seul capteur utilisé. On n'estime alors à partir de la vision que certains paramètres de pose, ce qui est faisable avec une seule caméra. [Demonceaux 07] propose par exemple d'estimer l'attitude (roulis et tangage) d'un engin volant dans un environnement urbain, avec une caméra omnidirectionnelle. La position et l'orientation partielles d'un hélicoptère par rapport à un *helipad* (cible visuelle en forme de H matérialisant la zone sur laquelle les hélicoptères atterrissent) sont déduits des moments de l'image de l'*helipad* dans [Saripalli 03]. Dans [Proctor 04], une caméra est utilisée pour estimer la position d'un planeur par rapport à une fenêtre.

Pour améliorer les estimations issues de la vision, et/ou les compléter pour obtenir une estimation complète de l'état de l'engin, les informations visuelles peuvent être combinées à d'autres mesures, au sein d'un processus de filtrage. Par exemple, l'estimation de la position latérale et de l'orientation d'un petit avion par rapport à l'axe d'une route est obtenue en utilisant conjointement la caméra embarquée et la centrale inertielle dans [Frew 04]. Dans [Furst 99, Wu 05], un filtre de Kalman étendu permet d'estimer la pose à partir de données visuelles, de mesures issues du GPS, et des capteurs inertiels.

Considérer que le modèle de la cible observée est connu permet également d'estimer la pose d'un engin volant avec une seule caméra embarquée [Sharp 01, Romero 06]. En supposant que la largeur de la piste est connue, on peut estimer la pose partielle d'un avion par rapport à une piste d'atterrissage [Rives 02].

Enfin, une approche bio-inspirée reposant sur l'utilisation du flot optique permet de commander l'altitude d'un engin équipé d'une caméra orientée vers le sol [Ruffier 04].

Le flot optique mesuré dans une caméra embarquée orientée "vers l'avant" de l'engin permet également d'estimer le temps à collision (*time to contact*), soit la durée avant impact, et ainsi d'éviter les collisions [Zufferey 06].

1.2.2.2 Informations visuelles 2D

L'utilisation d'informations issues directement du plan image permet de s'affranchir de l'étape de reconstruction 3D. Le contrôleur de l'asservissement visuel doit alors tenir compte de la matrice d'interaction, qui fait le lien entre les informations visuelles et le comportement 3D de l'engin volant. On est dans le cadre de l'asservissement visuel 2D (voir section 1.1.2.2). Le choix des informations visuelles est alors guidé d'une part par les caractéristiques de la scène observée, et d'autre part par les propriétés que l'on souhaite donner à la matrice d'interaction.

Par exemple, dans [Hrabar 03], l'objectif est de placer la caméra au centre de gravité de plusieurs cibles visuelles identiques. Une caméra omnidirectionnelle est utilisée, et la surface de la projection de chaque cible dans le plan image est utilisée dans la loi de commande. [Zwaan 00] utilise directement la position du centre de gravité et la surface d'une cible visuelle dans le plan image. L'objectif est de centrer un dirigeable équipé d'une caméra orientée vers le sol au-dessus de la cible considérée, et de commander son altitude. Dans [Zhang 99], un dirigeable est asservi sur une balle. Les coordonnées du centre de la balle et de son rayon dans le plan image sont utilisées dans la loi de commande. Un dirigeable est également asservi dans [Azinheira 02]. La cible considérée est un cercle tracé sur le sol, et un point placé sur la verticale passant au centre du cercle. Des paramètres extraits de la projection de cette cible sur le plan image sont utilisés pour construire l'erreur que la loi de commande doit minimiser.

Citons enfin divers travaux ayant considéré des structures linéaires (routes, pistes d'atterrissage, etc.), modélisées par des droites parallèles, comme cible visuelle. [Rives 02] considère trois droites, et les coordonnées polaires de leurs projections sur le plan image (conjointement à d'autres mesures) pour l'atterrissage d'un avion. Dans [Rives 04], seules deux droites sont utilisées pour permettre à un dirigeable de suivre une structure linéaire. Les informations visuelles sont obtenues en combinant les paramètres des droites, et en utilisant le point de fuite et l'orientation de la ligne d'horizon, dans le but d'obtenir une matrice d'interaction découplée. Dans [Mahony 05a], l'information visuelle est construite spécifiquement pour être adaptée à la dynamique des engins volants, à partir des coordonnées de Plücker binormalisées des droites.

1.2.3 Diverses lois de commandes utilisées

Il existe de très nombreuses manières de concevoir la commande par vision des robots volants. En effet, d'une part les données de vision peuvent être exploitées sur les organes bas-niveau et/ou haut-niveau du système selon la stratégie de commande adoptée : asservissement visuel direct ou indirect. D'autre part, les lois de commande utilisées dans le contrôleur de l'asservissement visuel sont de types variés.

1.2.3.1 Stratégie de commande

Dans la plupart des cas, deux boucles de commande sont imbriquées (voir figure 1.19). Une boucle "haut-niveau" fournit des consignes à la partie "bas-niveau". La boucle "haut-niveau" est celle qui permet à l'engin d'évoluer de manière autonome, en remplaçant l'opérateur humain. Dans la majorité des cas, c'est ici que les données de vision sont exploitées [Rock 98, Saripalli 03, Frew 04, Mejias 06a]. La partie "bas-niveau" se charge de faire évoluer l'engin de manière à atteindre les valeurs de consignes reçues de la boucle "haut-niveau". Même si elle intègre parfois des données de vision, elle repose pour sa part essentiellement sur des capteurs de type centrale inertielle, GPS, etc. Par

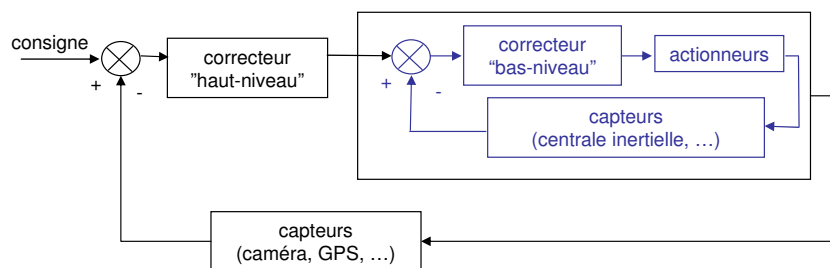


FIG. 1.19 – Schéma-bloc de la stratégie de commande avec deux boucles imbriquées. En bleu la boucle de commande "bas-niveau" gérant la navigation de l'engin volant et la commande de ses actionneurs (gouvernes, moteurs, etc.). En noir la boucle de commande "haut-niveau" dans laquelle intervient le plus souvent la caméra embarquée.

exemple, dans [Frew 04], les données de la caméra embarquée et des capteurs inertiels sont exploitées par le correcteur pour calculer des consignes sous forme de vitesses de rotation, afin que l'avion suive l'axe de la route. Une boucle de commande "bas-niveau" se charge alors de calculer la position des gouvernes permettant d'atteindre les vitesses de rotation demandées. Cette approche permet de dissocier la partie tâche à accomplir (par exemple suivre la route) de la partie navigation (par exemple la commande de la position des gouvernes pour atteindre un angle de cap particulier). Il existe également des variantes, comme par exemple l'architecture de commande bas niveau *behaviour-based* du projet AVATAR [Saripalli 03]. Dans cette stratégie de commande, plusieurs boucles de commande assurent chacune le bon déroulement d'une partie du comportement bas niveau de l'hélicoptère (maintien de l'altitude, maintien du cap, etc.). La commande haut niveau se charge pour sa part de définir quelle altitude, quel cap, etc. l'engin doit avoir pour accomplir la tâche désirée.

Certains travaux [Zhang 99, Proctor 04, LeBras 06, Guenard 07] proposent cependant d'utiliser la caméra au niveau bas, en liant directement les données de vision à la dynamique de l'engin. La caméra embarquée est alors exploitée dans le cadre de l'aide à la navigation. Les sorties du correcteur qui utilisent les données de vision sont des commandes "bas-niveau" (par exemple des valeurs de force, de couple, les angles de rotation des gouvernes, etc.).

1.2.3.2 Type de contrôleur

À partir de informations visuelles choisies, différents types de contrôleurs peuvent être mis en œuvre afin d'accomplir la tâche envisagée. Des techniques de commande floue [Huang 05], de commande linéaire (PID, commande optimale, etc.), ou non linéaires (techniques de backstepping, fonctions de Lyapunov, etc.) sont utilisées.

Commande de type PID. Pour concevoir la loi de commande, on a bien souvent recours à un modèle simplifié de l'engin. En effet, s'affranchir des non-linéarités et des couplages permet d'appliquer des lois de commandes simples.

La commande la plus simple est le correcteur proportionnel : la sortie du correcteur est alors proportionnelle à l'écart entre la valeur fournie par les capteurs et la consigne. L'ajout des termes dérivé et intégral, pour obtenir ainsi le correcteur PID (Proportionnel, Intégral, Dérivé) permet d'améliorer les performances de cette commande (notamment en ce qui concerne le temps de réponse et l'erreur statique). Ce correcteur se décline également sous la forme du correcteur PD (Proportionnel Dérivé) et du correcteur PI (Proportionnel Intégral).

Supposons que le système considéré ait m entrées de commandes $u_1, u_2, \dots, u_n, \dots, u_m$, et que l'on souhaite commander m sorties $y_1, y_2, \dots, y_n, \dots, y_m$. Le modèle du système considéré est linéaire et découplé (une entrée de commande fait évoluer une information visuelle) :

$$\begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \\ \dots \\ y_m \end{pmatrix} = \begin{pmatrix} k_1 & 0 & \dots & 0 & \dots & 0 \\ 0 & k_2 & \dots & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & \dots & 0 \\ 0 & 0 & \dots & k_n & \dots & 0 \\ 0 & 0 & \dots & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & \dots & k_m \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ u_n \\ \dots \\ u_m \end{pmatrix},$$

Le correcteur PID s'applique alors sur chaque composante y_n à commander :

$$u_n = k_{I_n} \int (y_n^* - y_n) dt + k_{P_n} (y_n^* - y_n) + k_{D_n} \frac{d}{dt} (y_n^* - y_n)$$

(voir figure 1.20). Notons que les gains intégral, proportionnel et dérivé k_{I_n} , k_{P_n} , et k_{D_n} sont à déterminer (le plus souvent manuellement) pour chacune des m composantes à commander.

Ainsi, dans [Ettinger 02], les angles de roulis et de tangage d'un micro-drone sont supposés découplés. Ils sont stabilisés via une loi de commande — basée vision — de type PD.

Dans [Frew 04], l'objectif est de suivre une route. Les capteurs (inertiels et caméra) permettent d'obtenir une estimation de l'erreur latérale par rapport à l'axe de la route. Cette erreur est alors utilisée par un correcteur PID, qui calcule les vitesses de rotation nécessaires à l'annulation de cette erreur.

Une loi de commande de type proportionnelle est également mise en œuvre dans [Proctor 04] pour commander les gouvernes d'un planeur. Trois correcteurs PD sont

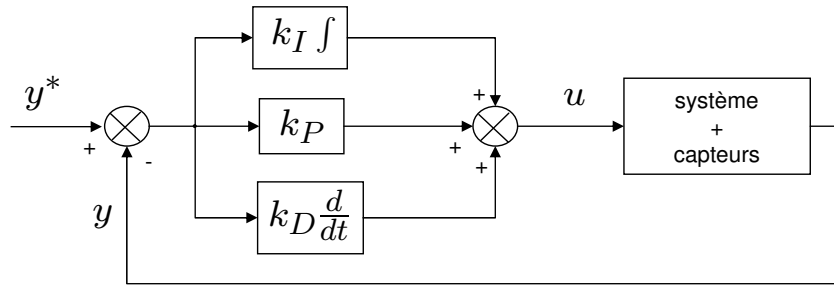


FIG. 1.20 – Schéma-bloc de la commande PID.

implémentés dans [Zwaan 00] pour commander les positions latérales, longitudinales et verticales d'un dirigeable à partir de trois informations visuelles. [Mejias 06b] propose également deux correcteurs PD et un PID pour commander l'orientation d'un hélicoptère.

Commande optimale. Lorsque le modèle de l'engin, liant les entrées de commande considérées et les informations visuelles utilisées est donné sous forme d'une représentation d'état linéaire continue, la conception d'une commande de type LQR (*Linear Quadratic Regulator*) est bien adaptée.

Supposons que le modèle du système considéré ait la forme suivante :

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{x}\end{aligned}$$

où \mathbf{u} est le vecteur des entrées de commande, \mathbf{y} les sorties à commander, \mathbf{x} un vecteur d'état du système, et \mathbf{A} , \mathbf{B} , et \mathbf{C} respectivement les matrices de système, d'entrée et de sortie.

La conception d'un correcteur de type LQR consiste alors à calculer la matrice optimale \mathbf{K} telle que la loi de commande $\mathbf{u} = -\mathbf{K}\mathbf{x}$ minimise la fonction quadratique $J(\mathbf{u}) = \int (\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{u}^\top \mathbf{R}\mathbf{u})dt$ (où les matrices \mathbf{Q} et \mathbf{R} sont définies manuellement).

La loi de commande obtenue est alors de type retour d'état, puisqu'elle prend en compte les valeurs de l'état \mathbf{x} et non celles de la sortie \mathbf{y} . Cependant, seules les valeurs de sortie sont effectivement connues car fournies par les capteurs. Ainsi, dans [Rives 02, Silveira 02, Rives 04], le correcteur utilisé est en fait $\mathbf{u} = \mathbf{K}\mathbf{C}^+(\mathbf{y} - \mathbf{y}^*)$ (voir figure 1.21). Notons que des variantes, ajoutant des termes intégral [Rives 04] et/ou dérivé [Azinheira 02, Silveira 02] à la commande, sont également employées.

Commande non linéaire. Des lois de commandes reposant sur la technique du backstepping et utilisant des données de vision ont été conçues pour des quadrirotors [Hamel 02, Altuğ 05]. Dans [Hamel 02], la synthèse s'appuie sur un modèle dynamique de l'engin à structure pyramidale, et la stabilisation par étape de chaque étage du

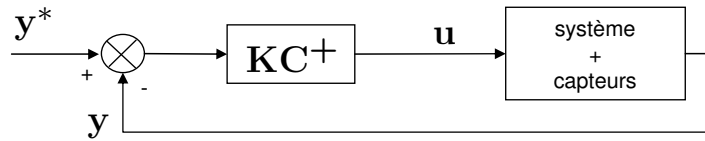


FIG. 1.21 – Schéma-bloc de la commande LQR avec retour de sortie.

ystème. À chaque étape du processus de backstepping, une commande virtuelle est générée et une fonction de Lyapunov est proposée, afin d'assurer la convergence du système. Les lois de commande conçues ont par la suite été expérimentées avec succès [Guenard 07].

1.3 Conclusion

Dans ce chapitre, nous avons présenté les concepts de base sur lesquels s'appuie notre travail. Dans un premier temps, nous avons présenté le modèle de projection perspective et les techniques les plus classiques utilisées en asservissement visuel. En particulier, on a distingué l'asservissement visuel basé image, intégrant les informations visuelles de type 2D directement dans la loi de commande, de l'asservissement visuel 3D qui considère une étape d'estimation de la pose de la caméra par rapport à la scène observée. Dans un second temps, nous avons présenté divers travaux ayant porté sur l'utilisation d'une loi de commande par asservissement visuel pour commander des engins volants. Les engins volants considérés sont de divers types (hélicoptères, avions, dirigeables, drones bio-inspirés), et correspondent à un vaste panel d'applications. Diverses stratégies de commande ont été utilisées. Les données de vision et les lois de commande associées peuvent être de différents types, et intervenir à différents niveaux selon l'agencement des boucles de commande.

Dans notre travail, nous nous sommes intéressés à deux types d'engins volants : un hélicoptère de type quadrirotor et un avion. Dans les deux cas, l'objectif est de concevoir des lois de commande d'asservissement visuel à partir de données de vision obtenues d'une caméra embarquée. Dans le premier cas, on a cherché à positionner le quadrirotor au-dessus d'une cible visuelle connue. Dans le deuxième cas, la scène observée est une piste d'atterrissage, et on a réalisé l'atterrissage de l'avion avec une loi de commande d'asservissement visuel. Nous avons cherché à utiliser autant que possible des informations visuelles de type 2D, afin de ne pas avoir à considérer le calcul de pose. Par ailleurs, nous avons considéré qu'une boucle de commande interne permet la navigation et la commande bas-niveau des engins. Nos lois de commande visuelles interviennent à un second niveau en fournissant des consignes aux boucles de commande les plus internes. Ces travaux sont présentés en détail dans les chapitres suivants.

Chapitre 2

Asservissement visuel d'un quadrirotor

Dans ce chapitre, nous nous intéressons à commander par vision la position des robots volants capables d'évoluer en vol stationnaire. Nous proposons pour cela une série de schémas de commande d'asservissement visuel. Ce travail a été réalisé pour une part dans le cadre du projet Robea Rovolint, puis avec le soutien d'un PICS. Ainsi, il a été l'objet d'une collaboration avec les laboratoires suivants : l'I3S à Sophia Antipolis et l'ANU à Canberra, essentiellement pour les aspects théoriques, et le CEA/LIST de Fontenay aux Roses en particulier pour la mise en œuvre expérimentale sur le prototype de quadrirotor.

2.1 Objectif et problématique

L'objectif est d'asservir la position d'un engin volant en vol stationnaire au-dessus d'une cible visuelle. Pour cela, on considère une caméra embarquée qui observe le sol. Notons que l'on s'intéresse ici uniquement à la partie asservissement visuel proprement dite, c'est-à-dire que l'on ne traite pas du problème de détection de la cible dans les images. On se place donc à une position initiale où la cible est visible.

Par ailleurs, nous nous intéressons à l'asservissement visuel 2D, c'est-à-dire que l'on ne souhaite pas estimer la pose de l'engin, mais exprimer les consignes directement à partir de données image.

Afin de commander la position de l'engin (trois translations : altitude et positions latérales et longitudinales), nous souhaitons mettre au point une loi de commande d'asservissement visuel cinématique. En d'autres termes, notre loi de commande va fournir des consignes de vitesse de translation à une boucle de commande interne qui se charge d'exécuter ces consignes (voir figure 2.1). On se place donc dans le cadre d'une stratégie de commande à deux niveaux, comme c'est le cas dans la majorité des travaux traitant de la commande par vision d'engins volants [Rock 98, Saripalli 03, Frew 04, LeBras 06].

Bien que notre loi de commande ne fournisse que des consignes de vitesses de trans-

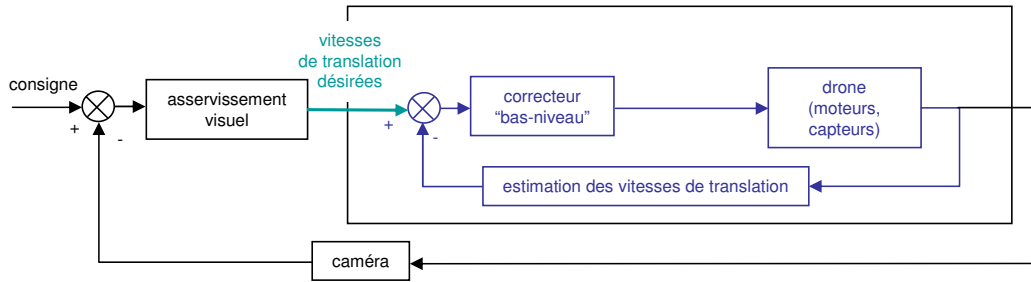


FIG. 2.1 – Schéma bloc de la stratégie de commande utilisée pour l'asservissement visuel de la position d'un drone. Notons que le fonctionnement de la boucle de commande "bas-niveau" sera plus détaillé dans la partie 2.6.1.1 présentant les expérimentations réalisées.

lation, la dynamique des engins volants à vol stationnaire fait que des mouvements de rotation sont également générés. Par exemple, dans le cas du quadrirotor, les mouvements de translation latéraux et longitudinaux sont obtenus en inclinant la force de poussée générée par les moteurs, et donc en inclinant le quadrirotor lui-même (voir figure 2.2). Ainsi, on voit bien que lorsque notre commande d'asservissement visuel va demander des vitesses de translation latérales et longitudinales, l'engin va s'incliner. Mais si l'engin s'incline, la caméra va s'incliner aussi, et l'image courante sera donc modifiée sans que le mouvement de translation demandé ne soit encore réalisé. Ainsi, il est nécessaire d'utiliser un schéma d'asservissement visuel qui soit aussi découplé que possible des mouvements de rotation de l'engin.

Dans les parties suivantes, nous proposons donc une série de lois de commande d'asservissement visuel tentant de répondre à ces exigences. Deux approches sont proposées. La première repose sur les propriétés de découplage des moments de la projection perspective d'un objet, et met en œuvre les techniques classiques d'asservissement visuel. La seconde — que l'on décline en plusieurs lois de commande — exploite des informations visuelles issues d'une projection sphérique, et tire profit de leurs propriétés cinématiques afin de mieux tenir compte des contraintes liées à la dynamique de l'engin. Les lois de commande proposées ont tout d'abord été implémentées en simulation, afin de mettre en évidence leurs atouts et leurs faiblesses. Les propriétés auxquelles nous nous intéressons sont les comportements transitoire et asymptotique (stabilité, convergence de toutes les composantes à la même vitesse), la relation entre l'espace image et l'espace 3D, et la propriété de passivité. Les trois schémas de commande les plus intéressants ont également été implémentés sur un quadrirotor.

Dans la partie 2.3, nous présentons la loi de commande reposant sur la projection perspective. Nous considérons ensuite la projection sphérique dans la partie 2.4. Une série de lois de commande est alors proposée. Leurs propriétés sont montrées et illustrées par des simulations. La partie 2.5 propose ensuite une analyse comparée de toutes les lois de commande proposées. Enfin, les résultats expérimentaux sur un quadrirotor, avec les meilleures lois de commande, sont présentés dans la partie 2.6.

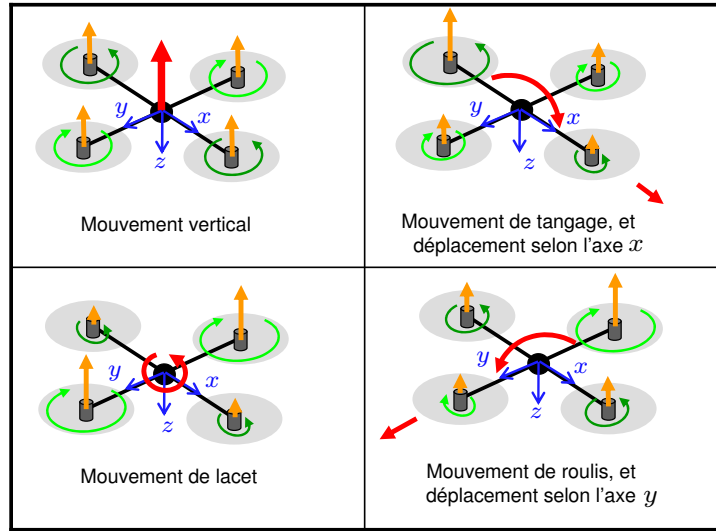


FIG. 2.2 – Schéma de fonctionnement d'un quadrirotor : la variation de la vitesse de rotation des moteurs (flèches vertes), modifie l'amplitude de la force de poussée développée par chaque moteur (flèches jaunes), et permet de modifier l'altitude, l'angle de lacet, et d'effectuer des mouvements de roulis et tangage, qui induisent des déplacements latéraux et longitudinaux (flèches rouges).

2.2 Simulations : conditions expérimentales

Comme nous venons de l'évoquer, nous proposons dans ce chapitre une série de lois de commande de translation reposant sur différents types d'informations visuelles. Pour illustrer les performances de l'ensemble de ces lois de commande, nous nous appuyons sur des résultats de simulation. Afin d'améliorer la lisibilité, nous présentons ces résultats au fur et à mesure, pour chaque loi de commande. Nous décrivons tout d'abord les conditions expérimentales pour lesquelles ces résultats ont été obtenus.

Nous utilisons une cible visuelle constituée de quatre points blancs sur fond noir formant un rectangle (dimensions : 14 cm de longueur pour 10 cm de largeur). Les positions initiales et désirées sont les mêmes pour toutes les expériences. La figure 2.3 montre les images réelles correspondantes de la cible visuelle. La position (en mètres) et l'orientation (en degrés) initiales du repère lié à la caméra \mathcal{F}_c dans le repère fixe lié à la cible \mathcal{F}_o (voir figure 2.4) sont telles que

$$\begin{aligned} {}^o\mathbf{t}_c &= (X_i, Y_i, Z_i) = (-0.21, 0.096, -0.87) \\ \theta\mathbf{u} &= (\theta\mathbf{u}_{x_i}, \theta\mathbf{u}_{y_i}, \theta\mathbf{u}_{z_i}) = (1.05, 20.68, 3.76) \end{aligned} \quad (2.1)$$

où ${}^o\mathbf{t}_c$ est le vecteur de translation et où $\theta\mathbf{u}$ est la représentation de la rotation ${}^o\mathbf{R}_c$ entre le repère \mathcal{F}_o et le repère \mathcal{F}_c . L'objectif est que la caméra atteigne la position désirée

(représentée par le repère \mathcal{F}_c^*) telle que :

$$\begin{aligned} {}^o\mathbf{t}_c^* &= (X^*, Y^*, Z^*) = (0, 0, -0.5) \\ \theta\mathbf{u}^* &= (\theta\mathbf{u}_x^*, \theta\mathbf{u}_y^*, \theta\mathbf{u}_z^*) = (0, 0, 0). \end{aligned} \quad (2.2)$$

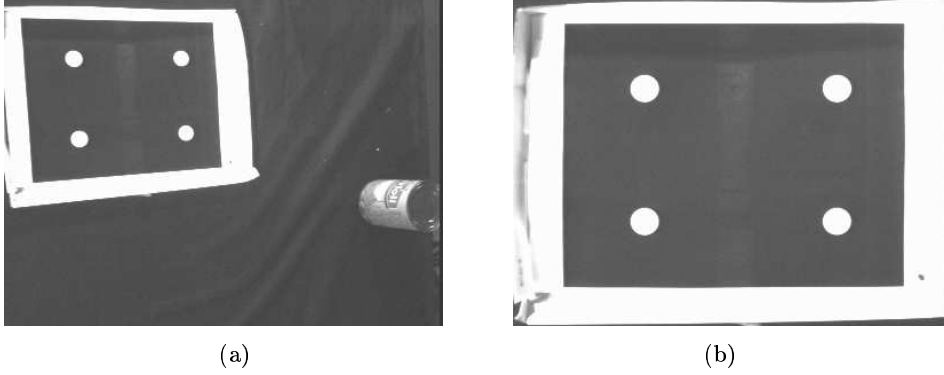


FIG. 2.3 – Images initiale (a) et finale (b) correspondant aux simulations.

On considère une caméra pouvant se déplacer selon 6 ddls que l'on commande en vitesses, exprimées dans le repère qui lui est lié. Nous avons implémenté nos algorithmes en C/C++ sous Linux. La bibliothèque ViSP-2 (*Visual Servoing Platform*) [Marchand 05, Visp] est utilisée pour les fonctions de base de l'asservissement visuel : projection des points sur le plan image, calcul de matrices d'interactions, et pour déplacer la caméra.

Pour commander les ddls de translation, chacune des lois de commande proposée dans la suite a été implémentée successivement. En ce qui concerne les ddls de rotation, nous avons utilisé un asservissement visuel 3D (voir chapitre 1, section 1.1.2.1). Puisque l'on est en simulation, la pose est connue a priori sans calcul de pose. En particulier, on peut disposer de la représentation $\theta\mathbf{u}$ de la rotation ${}^{c^*}\mathbf{R}_c$ entre le repère \mathcal{F}_c et le repère \mathcal{F}_c^* . On a

$$\frac{d\theta\mathbf{u}}{dt} = \mathbf{L}_{\theta\mathbf{u}}\boldsymbol{\omega} \quad (2.3)$$

où l'expression de $\mathbf{L}_{\theta\mathbf{u}}$ est donnée au chapitre 1, équation (1.27). Notons que la matrice d'interaction $\mathbf{L}_{\theta\mathbf{u}}$ est telle que $\mathbf{L}_{\theta\mathbf{u}}^{-1}\theta\mathbf{u} = \theta\mathbf{u}$ [Malis 98]. La loi de commande permettant une décroissance exponentielle de $\theta\mathbf{u}$ est alors la suivante :

$$\boldsymbol{\omega} = -k_\omega \widehat{\mathbf{L}_{\theta\mathbf{u}}^{-1}}\theta\mathbf{u} = -k_\omega\theta\mathbf{u}. \quad (2.4)$$

On a représenté sur la figure 2.5 l'évolution de l'asservissement des rotations — identique pour toutes les simulations. Notons qu'on a utilisé un gain $k_\omega = 0.1$. Comme on s'y attendait, on observe bien une décroissance exponentielle des composantes de la rotation.

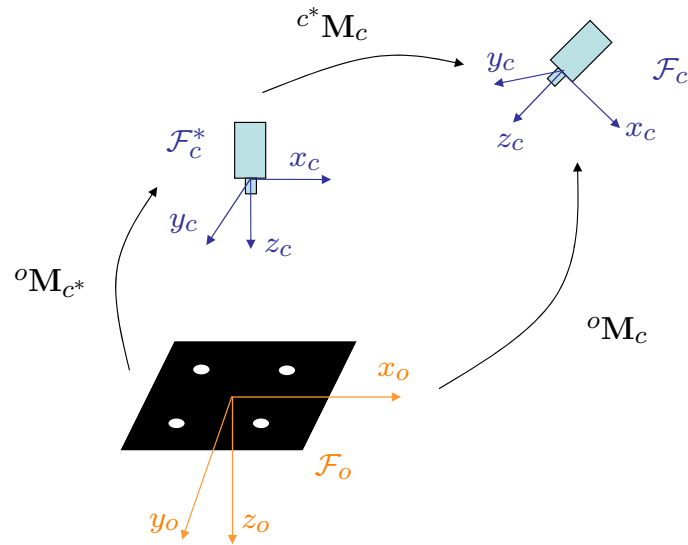


FIG. 2.4 – Repères liés à la cible (\mathcal{F}_o), et à la caméra, à la position initiale (\mathcal{F}_c) et à la position désirée (\mathcal{F}_c^*), et matrices homogènes iM_j représentant la pose du repère \mathcal{F}_j dans le repère \mathcal{F}_i .

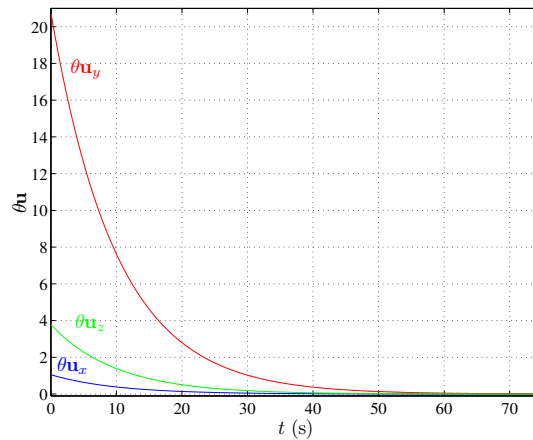


FIG. 2.5 – Asservissement des rotations : évolution au cours du temps de l'erreur sur l'orientation de la caméra $\theta_{\mathbf{u}} = (\theta_{\mathbf{u}_x}, \theta_{\mathbf{u}_y}, \theta_{\mathbf{u}_z})$.

Afin de mieux comparer les différentes lois de commande de translation, on a également pris un gain identique, de 0.1 pour chaque loi de commande. Cependant, nous allons utiliser différentes informations visuelles, dont les ordres de grandeur ne sont pas toujours similaires. De plus, certaines lois de commande sont mal conditionnées et génèrent des pics de vitesse importants au départ. Afin de rester dans un cadre réaliste où la réalisation de telles consignes serait impossible, nous avons choisi de saturer la commande en vitesse de translation de la façon suivante : lorsque $|\mathbf{v}| > 0.25$ m/s, on utilise en fait \mathbf{v}_{sat} tel que $\mathbf{v}_{sat} = 0.25 \frac{\mathbf{v}}{|\mathbf{v}|}$. Ainsi, même si les vitesses sont très grandes, cela ne pose pas de problème en pratique. De plus, le comportement relatif des composantes de \mathbf{v} et donc des positions 3D de la caméra n'est pas modifié par la saturation.

Pour chaque expérience, nous représentons l'erreur sur les positions successives de la caméra dans l'espace 3D ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^* = (X, Y, Z - Z^*)$, exprimée dans le repère fixe \mathcal{F}_o . Notons que l'on représente $Z - Z^*$, pour plus de lisibilité des résultats, Z^* étant égal à -0.5 m. En ce qui concerne l'espace image, nous représenterons dans chaque cas les composantes de l'erreur visuelle que l'on souhaite voir converger exponentiellement vers 0.

2.3 Utilisation de la projection perspective

Une première approche consiste à considérer des informations visuelles issues des moments de la projection perspective d'un objet, associées à un changement d'échelle bien choisi. En effet, comme nous le verrons, le principal avantage des informations visuelles choisies réside dans leurs propriétés de découplage [Tahri 05], propriété particulièrement utile dans le cas qui nous intéresse pour commander séparément les degrés de liberté (ddl).

Notons \mathcal{F}_o le repère fixe de référence, et \mathcal{F}_c le repère lié à la caméra embarquée. Dans toute la suite, on se place dans le cadre d'une cible fixe, et le repère fixe de référence pourra donc être le repère lié à la cible. Notons $\bar{\mathbf{p}} = (x, y, 1)$ les coordonnées d'un point de l'image sur le plan image \mathcal{P}_π (cf. section 1.1.1.1). Les moments d'ordre $i + j$ d'une région \mathcal{P}_p de l'image s'écrivent :

$$m_{ij} = \iint_{\bar{\mathbf{p}} \in \mathcal{P}_p} x^i y^j dx dy. \quad (2.5)$$

La surface a et les coordonnées (x_g, y_g) du centre de gravité d'une zone de l'image sont définies à partir des moments d'ordre zéro et un : $a = m_{00}$, $x_g = \frac{m_{10}}{a}$ et $y_g = \frac{m_{01}}{a}$. Afin d'obtenir une relation quasi linéaire et découplée entre l'espace image et l'espace 3D, [Tahri 05] propose le vecteur d'informations visuelles $\mathbf{s} = (x_n, y_n, a_n)$ défini tel que

$$a_n = Z^* \sqrt{\frac{a^*}{a}}, \quad x_n = a_n x_g, \quad y_n = a_n y_g$$

où a^* représente la surface désirée et Z^* la profondeur désirée entre la caméra et la cible.

Les matrices d'interaction (voir section 1.1.1.2) \mathbf{L}_{x_n} , \mathbf{L}_{y_n} , \mathbf{L}_{a_n} , respectivement associées à x_n , y_n et a_n sont alors obtenues à partir des matrices d'interaction associées à x_g , y_g , a (établies dans [Chaumette 04]), en utilisant le fait que $\dot{a}_n = -\frac{a_n}{2a}\dot{a}$, $\dot{x}_n = \dot{a}_n x_g + a_n \dot{x}_g$ et $\dot{y}_n = \dot{a}_n y_g + a_n \dot{y}_g$. On a :

$$\begin{aligned}\mathbf{L}_{x_n} &= \begin{bmatrix} \frac{x_n A}{2} - \frac{a_n}{Z} & \frac{x_n B}{2} & -\frac{x_n}{2} \left(\frac{3}{Z} - C \right) + x_{nv_z} \\ a_n \epsilon_{11} & -a_n (1 + \epsilon_{12}) & y_n \end{bmatrix} \\ \mathbf{L}_{y_n} &= \begin{bmatrix} \frac{y_n A}{2} & \frac{y_n B}{2} - \frac{a_n}{Z} & -\frac{y_n}{2} \left(\frac{3}{Z} - C \right) + y_{nv_z} \\ a_n (1 + \epsilon_{21}) & -a_n \epsilon_{11} & -x_n \end{bmatrix} \\ \mathbf{L}_{a_n} &= \begin{bmatrix} \frac{a_n A}{2} & \frac{a_n B}{2} & -\frac{a_n}{2} \left(\frac{3}{Z} - C \right) \\ -3y_n/2 & 3x_n/2 & 0 \end{bmatrix}\end{aligned}$$

où $\frac{1}{Z} = Ax_g + By_g + C$, A , B et C étant les paramètres représentant la configuration du plan de l'objet. On a par ailleurs :

$$\begin{aligned}x_{nv_z} &= x_n/Z + 4a_n(An_{20} + Bn_{11}) \\ y_{nv_z} &= y_n/Z + 4a_n(An_{11} + Bn_{02}) \\ \epsilon_{11} &= 4n_{11} - x_g y_g / 2 \\ \epsilon_{12} &= 4n_{20} - x_g^2 / 2 \\ \epsilon_{21} &= 4n_{02} - y_g^2 / 2\end{aligned}$$

où n_{20} , n_{02} et n_{11} sont les moments centrés normalisés d'ordre deux :

$$n_{ij} = \mu_{ij}/a \quad \text{avec} \quad \begin{cases} \mu_{20} = m_{20} - ax_g^2 \\ \mu_{02} = m_{02} - ay_g^2 \\ \mu_{11} = m_{11} - ax_g y_g \end{cases}$$

En notant

$$\mathbf{s} = (x_n, y_n, a_n) \tag{2.6}$$

le vecteur d'informations visuelles, on peut écrire la relation liant la variation de \mathbf{s} aux vitesses de translation et de rotation de la caméra, exprimées dans le repère de la caméra (respectivement \mathbf{v} et $\boldsymbol{\omega}$) sous la forme

$$\dot{\mathbf{s}} = \mathbf{L}_s \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = \mathbf{L}_v \mathbf{v} + \mathbf{L}_\omega \boldsymbol{\omega} \tag{2.7}$$

avec

$$\mathbf{L}_v = \begin{bmatrix} \frac{x_n A}{2} - \frac{a_n}{Z} & \frac{x_n B}{2} & -\frac{x_n}{2} \left(\frac{3}{Z} - C \right) + x_{nv_z} \\ \frac{y_n A}{2} & \frac{y_n B}{2} - \frac{a_n}{Z} & -\frac{y_n}{2} \left(\frac{3}{Z} - C \right) + y_{nv_z} \\ \frac{a_n A}{2} & \frac{a_n B}{2} & -\frac{a_n}{2} \left(\frac{3}{Z} - C \right) \end{bmatrix} \tag{2.8}$$

et

$$\mathbf{L}_\omega = \begin{bmatrix} a_n \epsilon_{11} & -a_n(1 + \epsilon_{12}) & y_n \\ a_n(1 + \epsilon_{21}) & -a_n \epsilon_{11} & -x_n \\ -3y_n/2 & 3x_n/2 & 0 \end{bmatrix}. \quad (2.9)$$

Lorsque le plan image de la caméra \mathcal{P}_π est parallèle au plan de la cible observée, on a $A = B = 0$, et $Z^* \sqrt{a^*} = Z \sqrt{a}$ et donc $a_n/Z = 1$. On déduit aussitôt de (2.8) l'expression de \mathbf{L}_v lorsque le plan image reste parallèle au plan de la cible [Tahri 05] :

$$\mathbf{L}_v^{\parallel} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \quad (2.10)$$

Les bonnes propriétés de découplage et de linéarité de \mathbf{L}_v^{\parallel} seront utilisées par la suite.

Notons $\mathbf{s}^* \in \mathcal{F}_c$ le vecteur d'informations visuelles désirées, et

$$\mathbf{e} = \mathbf{s} - \mathbf{s}^* \quad (2.11)$$

le vecteur d'erreur. Notons que l'on choisit \mathbf{s}^* constant. On a donc

$$\dot{\mathbf{e}} = \dot{\mathbf{s}}. \quad (2.12)$$

Classiquement en asservissement visuel, on cherche à faire décroître exponentiellement l'erreur \mathbf{e} . Pour cela, on impose que $\dot{\mathbf{e}}$ soit de la forme $\dot{\mathbf{e}} = -\lambda \mathbf{e}$, avec λ une constante positive. Ainsi, en utilisant \mathbf{e} pour commander les ddls de translation, on obtient la loi de commande suivante :

$$\mathbf{v} = -(\widehat{\mathbf{L}}_v)^{-1}(\lambda \mathbf{e} + \widehat{\mathbf{L}}_\omega \boldsymbol{\omega}), \quad \lambda > 0. \quad (2.13)$$

En utilisant \mathbf{L}_v^{\parallel} pour $\widehat{\mathbf{L}}_v$ et \mathbf{L}_ω pour $\widehat{\mathbf{L}}_\omega$, on obtient :

$$\begin{aligned} \mathbf{v} &= \lambda \mathbf{e} + \mathbf{L}_\omega \boldsymbol{\omega} \\ &= \lambda(\mathbf{s} - \mathbf{s}^*) + \mathbf{L}_\omega \boldsymbol{\omega}, \quad \lambda > 0. \end{aligned} \quad (2.14)$$

Pour l'application que nous avons considérée ici, cette hypothèse est acceptable. En effet, la caméra est montée sous le quadrirotor, et est orientée verticalement vers le sol. Les angles entre le plan image et le plan de la cible restent donc de l'ordre de quelques degrés.

Considérons à présent la fonction de Lyapunov \mathcal{L} définie par

$$\mathcal{L} = \frac{1}{2} |\mathbf{e}|^2. \quad (2.15)$$

Sa dérivée par rapport au temps s'écrit $\dot{\mathcal{L}} = \mathbf{e}^\top \dot{\mathbf{e}}$. Avec (2.7) et (2.12), on a alors

$$\dot{\mathcal{L}} = \mathbf{e}^\top (\mathbf{L}_v \mathbf{v} + \mathbf{L}_\omega \boldsymbol{\omega}). \quad (2.16)$$

En remplaçant \mathbf{v} par son expression, donnée par la loi de commande (2.14), on obtient :

$$\dot{\mathcal{L}} = \mathbf{e}^\top (\lambda \mathbf{L}_v \mathbf{e} + \mathbf{L}_v \mathbf{L}_\omega \boldsymbol{\omega} + \mathbf{L}_\omega \boldsymbol{\omega}). \quad (2.17)$$

Pour pouvoir appliquer le théorème de Lyapunov et montrer que le système bouclé par la loi de commande (2.14) est globalement asymptotiquement stable (GAS), il faudrait montrer que $\dot{\mathcal{L}}$ est définie négative, ce qui, au vu de l'expression (2.17), n'est pas évident. Notons que l'on aurait $\dot{\mathcal{L}} < 0$ si l'hypothèse de parallélisme entre plan image et plan de la cible était toujours parfaitement respectée (car en remplaçant \mathbf{L}_v par $-\mathbf{I}_3$ dans (2.17) on obtient $\dot{\mathcal{L}} = -\lambda \mathbf{e}^\top \mathbf{e}$). Cependant, cette hypothèse reste une approximation et on n'a pas montré que $\dot{\mathcal{L}}$ est définie négative dans le cas général, on ne peut donc pas conclure que le système est GAS.

Dans la simulation, la position de la caméra désirée étant centrée par rapport à la cible, à 0.5 m de haut (voir paragraphe 2.2), les informations visuelles désirées sont $x_n^* = y_n^* = 0$, $a_n^* = Z^* = 0.5$, et on a $a^* = 0.056$. La loi de commande (2.14) donne de très bons résultats (voir figure 2.6). On observe une décroissance quasi exponentielle des composantes de l'erreur visuelle (voir figure 2.6.a), et également un très bon comportement dans l'espace 3D (voir figure 2.6.b). En fait, le lien entre l'espace image et l'espace 3D étant quasi linéaire et découplé ($\mathbf{L}_v \simeq -\mathbf{I}_3$), contraindre l'erreur à décroître exponentiellement dans l'image est alors quasi équivalent à faire décroître exponentiellement l'erreur de position. Cela conduit la caméra à avoir un comportement très satisfaisant dans l'espace 3D, bien que la commande repose sur des informations image. Bien que le plan image et le plan de la cible ne soient pas parallèles dans la simulation (voir figure 2.5), on obtient quand même de très bons résultats dans le cadre de cette hypothèse.

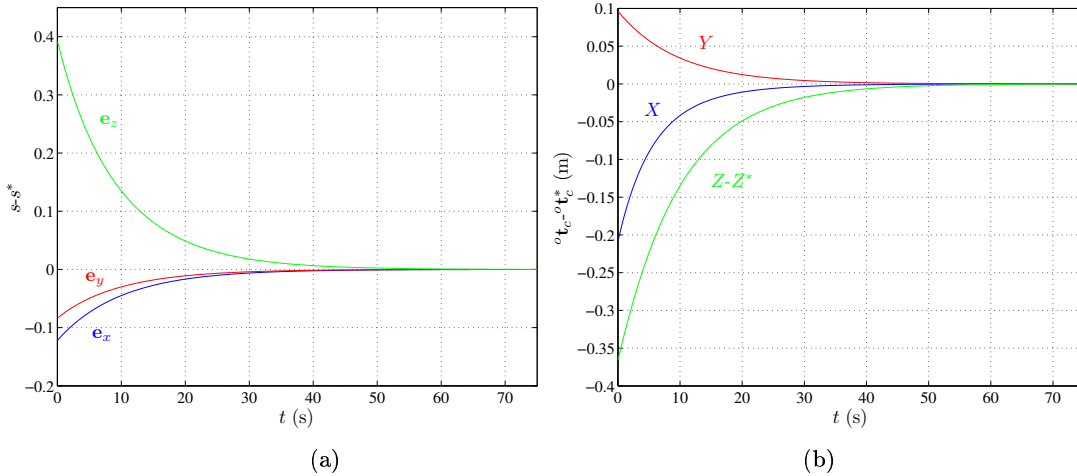


FIG. 2.6 – Commande $\mathbf{v} = \lambda(\mathbf{s} - \mathbf{s}^*) + \mathbf{L}_\omega \boldsymbol{\omega}$: évolution au cours du temps de l'erreur sur les informations visuelles $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

Une dernière simplification consiste à considérer que les mouvements de rotation de l'engin sont lents et de faible amplitude. Cela nous conduit à négliger le terme $\mathbf{L}_\omega \boldsymbol{\omega}$

devant $\lambda \mathbf{e}$ dans (2.14). En pratique, cela permet d'éviter d'avoir à estimer les vitesses de rotation $\boldsymbol{\omega}$. On obtient la loi de commande suivante :

$$\begin{aligned} \mathbf{v} &= \lambda \mathbf{e} \\ &= \lambda(\mathbf{s} - \mathbf{s}^*), \quad \lambda > 0. \end{aligned} \quad (2.18)$$

Considérons à nouveau la fonction de Lyapunov \mathcal{L} (2.15) et sa dérivée par rapport au temps $\dot{\mathcal{L}}$ (2.16). En remplaçant \mathbf{v} par son expression (2.18) dans (2.16), on obtient :

$$\dot{\mathcal{L}} = \mathbf{e}^\top (\mathbf{L}_v \lambda \mathbf{e} + \mathbf{L}_\omega \boldsymbol{\omega}). \quad (2.19)$$

Comme précédemment, rien ne prouve que le système est GAS : il faudrait pour cela montrer que $\dot{\mathcal{L}}$ est toujours définie négative. Notons qu'en pratique dans les simulations (et ensuite lors des expérimentations, voir section 2.6.2) on n'a cependant pas rencontré de zone instable.

On voit sur la figure 2.7 que les mouvements de rotation — qui ne sont plus compensés dans la loi de commande comme précédemment —, perturbent la convergence de l'erreur visuelle et de la position 3D. La convergence est moins rapide que précédemment (on a pourtant utilisé le même gain $\lambda = 0.1$), puisqu'il faut "attendre" que les rotations soient nulles pour que les translations puissent terminer de converger (comparer les figures 2.6 et 2.7).

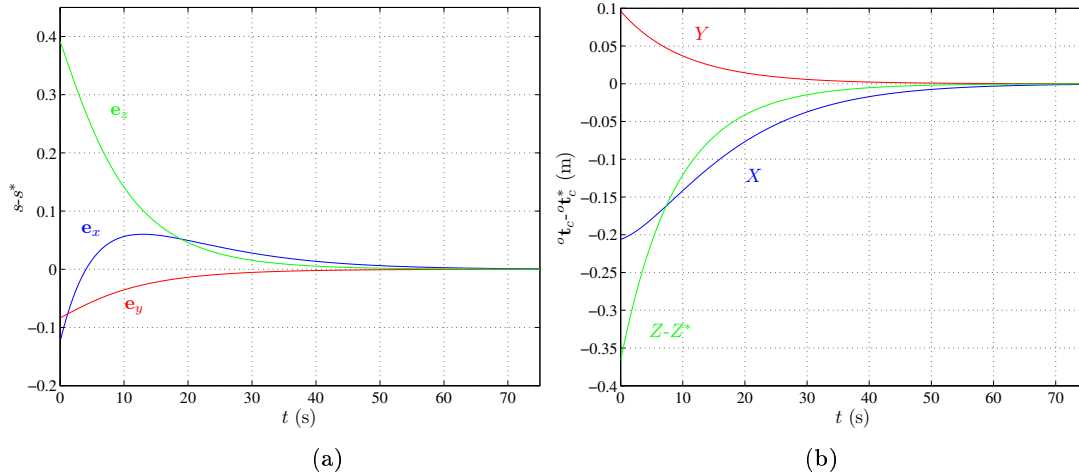


FIG. 2.7 – Commande $\mathbf{v} = \lambda(\mathbf{s} - \mathbf{s}^*)$: évolution au cours du temps de l'erreur sur les informations visuelles $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

L'avantage de la loi de commande (2.18) réside dans sa simplicité de mise en œuvre, puisqu'elle repose exclusivement sur les informations visuelles image \mathbf{s} , et ne requiert pas de reconstruction de la pose ni d'estimation des vitesses de rotation. Nous l'avons mise en œuvre sur le prototype de quadrirotor (voir section 2.6.2), qui respecte assez bien les approximations considérées ici : plan image de la caméra parallèle à la cible observée, et mouvements de rotations lisses et lents. Cependant, cette approche n'est

pas adaptée à la dynamique de tous les engins, et peut donc manquer de robustesse lors de manœuvres brusques. Pour cette raison, nous proposons dans la suite des schémas de commande de translation plus appropriés, qui prennent effectivement en compte la dynamique des robots volants considérés, sans négliger les mouvements de rotation.

2.4 Utilisation de la projection sphérique

Dans cette partie, nous considérons la projection des points du plan de l'image perspective sur une sphère virtuelle. Nous allons tout d'abord voir comment l'utilisation du centre de gravité de l'image sphérique permet d'obtenir des informations visuelles dotées de la propriété de passivité. Ensuite, nous proposons et analysons une série de schémas de commande d'asservissement visuel reposant sur ces informations visuelles. La conception de ces schémas de commande repose sur l'utilisation d'une fonction de Lyapunov, comme proposé dans [Hamel 02]. Nous montrons en quoi les lois de commande simples de type proportionnel [Hamel 02], ou basées sur une linéarisation du système à la position désirée [Mahony 02b], ne sont pas satisfaisantes en pratique. Nous proposons donc ensuite de nouveaux schémas de commande, de manière à corriger les problèmes rencontrés, en améliorant les performances transitoires et asymptotiques du système.

2.4.1 Modélisation des informations visuelles

Projection sphérique. Pour simplifier, on se place dans le cas d'une caméra dont la distance focale est unitaire. Cela n'est aucunement restrictif, puisque l'on peut toujours se ramener à cette situation en utilisant les coordonnées normalisées des points image (voir section 1.1.1.1).

Soit \mathbf{x}_s un point de la sphère virtuelle \mathcal{S}^2 , correspondant au point \mathbf{x} du plan image \mathcal{P}_π , et donc au \mathcal{X} de l'espace 3D (voir figure 2.8). Notons $\mathbf{P}_c = (X_c, Y_c, Z_c)$ les coordonnées du point \mathcal{X} , $\mathbf{p} = (x, y, 1)$ les coordonnées du point \mathbf{x} , et $\mathbf{p}_s = (x_s, y_s, z_s)$ les coordonnées du point \mathbf{x}_s , dans le repère de la caméra $\mathcal{F}_c = (\mathcal{C}, \mathbf{x}_c, \mathbf{y}_c, \mathbf{z}_c)$. De simples considérations géométriques permettent d'écrire la relation entre les coordonnées d'un point sur le plan image perspective et ses coordonnées une fois projeté sur le plan image sphérique :

$$\mathbf{p}_s = \frac{\mathbf{p}}{|\mathbf{p}|} \quad (2.20)$$

où $|\mathbf{p}|$ est la norme du vecteur \mathbf{p} . Cette relation nous permet, pour chaque point de l'image perspective fournie par la caméra, de calculer les coordonnées du point correspondant sur la sphère virtuelle.

Les moments du premier ordre d'une région \mathcal{S} de la sphère \mathcal{S}^2 s'écrivent :

$$\mathbf{q} = \int_{\mathbf{p}_s \in \mathcal{S}} \mathbf{p}_s d\mathbf{p}_s.$$

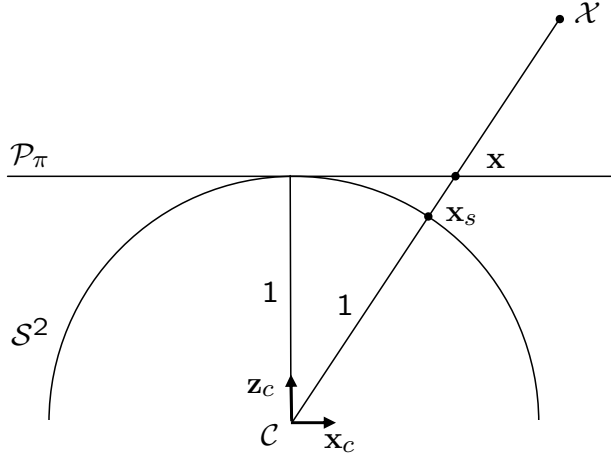


FIG. 2.8 – Schéma de la projection perspective \mathbf{x} et sphérique \mathbf{x}_s d'un point 3D \mathcal{X} .

Il est courant d'utiliser des cibles constituées de points plutôt que de formes continues. Ainsi, dans le cas discret de n points $\{\mathbf{x}_s^i\}$, $i \in [1, \dots, n]$ de coordonnées \mathbf{p}_s^i dans \mathcal{F}_c (correspondant aux n points 3D $\{\mathcal{X}^i\}$ d'une cible), les moments du premier ordre (qui sont aussi les coordonnées du centre de gravité non normées) des points de l'image sphérique s'écrivent sous la forme d'une somme discrète :

$$\mathbf{q} = \sum_{i=1}^n \mathbf{p}_s^i.$$

Notons que \mathbf{q} est un vecteur de dimension trois. Comme on peut le voir de manière intuitive sur la figure 2.9, la composante de \mathbf{q} selon l'axe \mathbf{z}_c est fonction de la variation de la profondeur entre la caméra et la cible observée. Cette relation, non linéaire, sera exploitée par la suite.

Cinématique des informations visuelles. Il est démontré dans [Hamel 02] que la variation du vecteur d'informations visuelles \mathbf{q} est reliée aux vitesses de la caméra de la façon suivante :

$$\dot{\mathbf{q}} = -[\boldsymbol{\omega}]_{\times} \mathbf{q} - \mathbf{Q} \mathbf{v}, \quad (2.21)$$

avec

$$\mathbf{Q} = \sum_{i=1}^n \frac{\boldsymbol{\pi}_{\mathbf{p}_s^i}}{|\mathbf{P}_c^i|} \quad (2.22)$$

et $\boldsymbol{\pi}_{\mathbf{p}_s^i} = (\mathbf{I}_3 - \mathbf{p}_s^i \mathbf{p}_s^{i\top})$, et où comme précédemment, \mathbf{v} et $\boldsymbol{\omega}$ sont respectivement les vitesses de translation et de rotation de la caméra, exprimées dans le repère de la caméra \mathcal{F}_c .

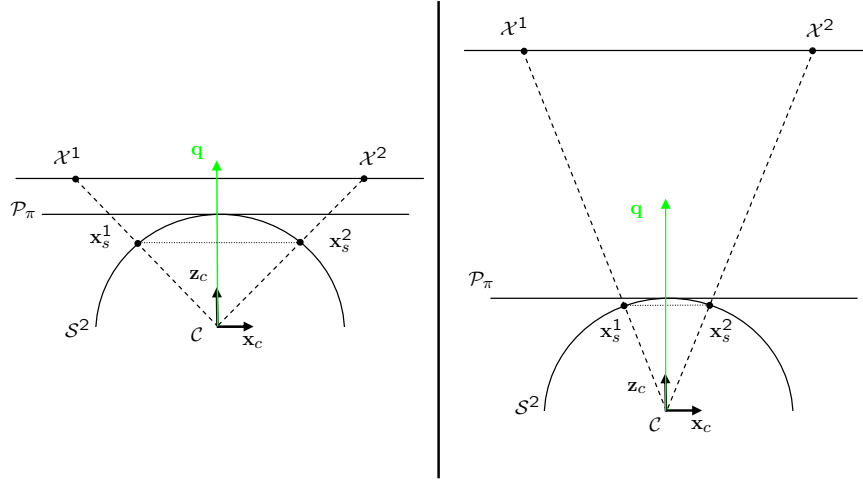


FIG. 2.9 – Schéma de la projection d'une cible constituée de deux points \mathcal{X}^1 et \mathcal{X}^2 sur la sphère virtuelle. \mathbf{q} est le vecteur des coordonnées du centre de gravité des points de l'image sphérique. On se place dans le cas simple où la cible est symétrique, centrée et parallèle au plan image. Le vecteur \mathbf{q} n'a alors qu'une composante selon l'axe \mathbf{z}_c . À gauche, lorsque la cible est très proche de la caméra, $|\mathbf{q}|$ est petit, alors qu'à droite, lorsque la cible est loin de la caméra, $|\mathbf{q}|$ est grand. La sphère étant de rayon unitaire, on voit que $|\mathbf{q}|$ tend vers 2 lorsque la cible tend à s'éloigner à l'infini.

On peut remarquer que la matrice \mathbf{Q} n'est pas connue à chaque instant, puisqu'elle est fonction de $|\mathbf{P}_c^i|$, et donc de la position 3D des points considérés. Cependant, dans [Hamel 02], il est démontré que la matrice \mathbf{Q} est définie positive, du moment que l'on considère au moins deux points différents de l'espace image.

Habituellement, les informations visuelles désirées que l'on considère en asservissement visuel sont exprimées dans le repère lié à la caméra. La particularité du choix fait dans [Hamel 02] est d'exprimer la direction du vecteur de consignes à atteindre, \mathbf{b} , dans le repère inertiel fixe \mathcal{F}_o . Ainsi, à chaque itération on se ramènera au repère de la caméra en calculant \mathbf{q}^* de la façon suivante :

$$\mathbf{q}^* = {}^o\mathbf{R}_c^\top \mathbf{b} \quad (2.23)$$

où ${}^o\mathbf{R}_c$ représente l'orientation de la caméra dans le repère fixe \mathcal{F}_o . Notons que cette matrice de rotation est donc supposée connue. Cette hypothèse est relativement courante dans le cas de la commande d'engins volants, où l'on peut tirer profit des capteurs inertiels embarqués de manière à estimer l'orientation de l'engin [Mahony 05b]. En pratique cependant, on cherche souvent à positionner l'engin volant au-dessus de la cible, et on choisit le repère fixe de manière à ce que son axe \mathbf{z}_o soit vertical et orienté comme la gravité terrestre. Le vecteur \mathbf{b} ne possède alors qu'une composante selon l'axe \mathbf{z}_o : il est de la forme $\mathbf{b} = (0, 0, b)$. On voit immédiatement qu'il n'est dans ce cas pas nécessaire de déterminer entièrement la matrice de rotation ${}^o\mathbf{R}_c^\top$. Seule la dernière ligne de ${}^o\mathbf{R}_c$ est utile pour calculer \mathbf{q}^* avec (2.23). Or, par définition, cette ligne correspond à l'orientation de l'axe vertical \mathbf{z}_o dans le repère lié à la caméra \mathcal{F}_c . Il suffit donc, à l'aide

des capteurs embarqués, de déterminer à chaque itération la direction de \mathbf{z}_o (qui est la direction de la gravité) dans le repère caméra.

Finalement, l'erreur considérée est définie comme suit :

$$\boldsymbol{\delta} = \mathbf{q} - \mathbf{q}^*. \quad (2.24)$$

Comme le vecteur $\mathbf{b} \in \mathcal{F}_o$ est fixe dans le repère de référence, la dynamique de $\mathbf{q}^* \in \mathcal{F}_c$ est contrainte par celle de la caméra, et on a :

$$\dot{\mathbf{q}}^* = -[\boldsymbol{\omega}]_{\times} \mathbf{q}^*. \quad (2.25)$$

On déduit alors de (2.21), (2.24) et (2.25) la dynamique de l'erreur :

$$\dot{\boldsymbol{\delta}} = -[\boldsymbol{\omega}]_{\times} \boldsymbol{\delta} - \mathbf{Q}\mathbf{v}. \quad (2.26)$$

Notons que cette relation peut s'écrire sous la forme d'une matrice d'interaction, en utilisant le fait que $-[\boldsymbol{\omega}]_{\times} \boldsymbol{\delta} = [\boldsymbol{\delta}]_{\times} \boldsymbol{\omega}$ (propriété des matrices de produit vectoriel) :

$$\dot{\boldsymbol{\delta}} = \begin{bmatrix} -\mathbf{Q} & [\boldsymbol{\delta}]_{\times} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix}. \quad (2.27)$$

Propriété des informations visuelles : la passivité. Pour commencer, notons que $\boldsymbol{\delta}$ est fonction à la fois des mouvements de rotation et des mouvements de translation de la caméra (2.26). Le découplage n'est donc pas une caractéristique de $\boldsymbol{\delta}$. Cependant, c'est l'exploitation de ses propriétés de passivité qui va nous permettre de concevoir des schémas de commande découplés.

Considérons la norme du vecteur d'erreur $\boldsymbol{\delta}$, et calculons sa dérivée. On obtient, en utilisant (2.26) :

$$\begin{aligned} |\dot{\boldsymbol{\delta}}| &= \frac{1}{|\boldsymbol{\delta}|} \boldsymbol{\delta}^{\top} \dot{\boldsymbol{\delta}} \\ &= \frac{1}{|\boldsymbol{\delta}|} (\boldsymbol{\delta}^{\top} [\boldsymbol{\delta}]_{\times} \boldsymbol{\omega} - \boldsymbol{\delta}^{\top} \mathbf{Q}\mathbf{v}). \end{aligned} \quad (2.28)$$

Une propriété classique des matrices antisymétriques est que $\boldsymbol{\delta}^{\top} [\boldsymbol{\delta}]_{\times} = \mathbf{0}$. On obtient donc :

$$|\dot{\boldsymbol{\delta}}| = -\frac{\boldsymbol{\delta}^{\top} \mathbf{Q}\mathbf{v}}{|\boldsymbol{\delta}|}. \quad (2.29)$$

On en déduit que $|\boldsymbol{\delta}|$, la norme de l'erreur visuelle dépend seulement de la position de la caméra par rapport à la cible.

Intéressons-nous à présent à l'expression de l'erreur dans le repère fixe : ${}^o\boldsymbol{\delta} = {}^o\mathbf{R}_c \boldsymbol{\delta}$. Une propriété des matrices de rotation donne cette fois que ${}^o\dot{\mathbf{R}}_c = {}^o\mathbf{R}_c [\boldsymbol{\omega}]_{\times}$. En utilisant cette propriété et (2.26), on obtient pour la dérivée de ${}^o\boldsymbol{\delta}$:

$$\begin{aligned} {}^o\dot{\boldsymbol{\delta}} &= {}^o\dot{\mathbf{R}}_c \boldsymbol{\delta} + {}^o\mathbf{R}_c \dot{\boldsymbol{\delta}} \\ &= {}^o\mathbf{R}_c [\boldsymbol{\omega}]_{\times} \boldsymbol{\delta} + {}^o\mathbf{R}_c (-[\boldsymbol{\omega}]_{\times} \boldsymbol{\delta} - \mathbf{Q}\mathbf{v}) \\ &= -{}^o\mathbf{R}_c \mathbf{Q}\mathbf{v}. \end{aligned} \quad (2.30)$$

On en déduit que ${}^o\boldsymbol{\delta}$ dépend également uniquement de la position, et qu'il n'est donc pas modifié par les mouvements de rotation de la caméra.

Remarques concernant les conditions expérimentales. Notons que pour les simulations réalisées (et il en sera de même pour les expérimentations sur le quadrirotor présentées à la section 2.6.2), le vecteur désiré \mathbf{q}^* est choisi de manière à ce que, à la convergence de l’asservissement, la caméra se trouve à la verticale de la cible, centrée, et à une hauteur de 0.5 mètres (voir paragraphe 2.2). L’orientation désirée est donc telle que ${}^o\mathbf{R}_{c^*} = \mathbf{I}_3$, et on calcule à partir de l’image désirée $\mathbf{q}^* = (0, 0, 3.94)$. On prend alors pour \mathbf{b} , l’information visuelle désirée dans le repère fixe : $\mathbf{b} = (0, 0, 3.94)$. On voit que l’on se trouve dans le cas évoqué précédemment où \mathbf{b} n’a qu’une composante selon l’axe \mathbf{z}_0 , et où seule l’estimation de l’orientation de l’axe vertical dans le repère lié à la caméra est nécessaire pour calculer \mathbf{q}^* à chaque itération avec (2.23).

Notons également que pour les simulations suivantes, utilisant les informations visuelles sphériques, outre les positions 3D de la caméra, nous représentons l’évolution au cours du temps de l’erreur visuelle dans le repère fixe ${}^o\boldsymbol{\delta} = {}^o\mathbf{R}_c\boldsymbol{\delta}$ — dont l’évolution n’est pas perturbée par les mouvements de rotation de la caméra (2.30).

2.4.2 Commande proportionnelle

Nous considérons tout d’abord la loi de commande la plus simple reposant sur les informations visuelles sphériques présentées : un correcteur proportionnel [Hamel 02, Mahony 02b].

Soit \mathcal{L} une fonction de Lyapunov définie par

$$\mathcal{L} = \frac{1}{2}|\boldsymbol{\delta}|^2. \quad (2.31)$$

Notons que $\mathcal{L} > 0$ pour tout $\boldsymbol{\delta}$, et que la fonction \mathcal{L} peut être vue comme l’énergie de $\boldsymbol{\delta}$. En utilisant (2.29), et en dérivant \mathcal{L} on obtient de manière immédiate :

$$\dot{\mathcal{L}} = \boldsymbol{\delta}^\top \dot{\boldsymbol{\delta}} \quad (2.32)$$

$$= -\boldsymbol{\delta}^\top \mathbf{Q}\mathbf{v}. \quad (2.33)$$

Choisir \mathbf{v} tel que

$$\mathbf{v} = k_\delta \boldsymbol{\delta}, \quad k_\delta > 0 \quad (2.34)$$

permet donc d’obtenir

$$\dot{\mathcal{L}} = -k_\delta \boldsymbol{\delta}^\top \mathbf{Q}\boldsymbol{\delta}. \quad (2.35)$$

La matrice \mathbf{Q} (définie par (2.22)) dépend de la position de la cible par rapport à la caméra, et n’est donc pas connue a priori. Cependant, comme mentionné dans 2.4.1, \mathbf{Q} a la particularité d’être définie positive. On en déduit donc que $\dot{\mathcal{L}}$ est négative pour tout $\boldsymbol{\delta}$ non nul. Le théorème de Lyapunov assure alors que le système est GAS : en d’autres termes, la loi de commande proposée permet de déplacer la caméra de manière à stabiliser l’erreur visuelle $\boldsymbol{\delta}$ à zéro, quelle que soit la position initiale de la caméra.

On voit bien à travers cette première loi de commande simple tout l’intérêt de la propriété de passivité de $\boldsymbol{\delta}$: on peut concevoir une loi de commande (de vitesses de translation) qui garantit la convergence de l’erreur visuelle, et ce quels que soient les mouvements de rotation de l’engin.

Notons cependant qu'aucun système utilisant la vision ne peut être dit GAS au sens strict, puisque la cible visuelle peut être occultée, sortir du champ de vision, etc. Ici et dans toute la suite, on utilisera donc le terme GAS lorsque la loi de commande considérée assure que le système est GAS dans le cadre de bonnes conditions d'utilisation.

En injectant la loi de commande (2.34) dans (2.30), on obtient :

$$\begin{aligned} {}^o\dot{\boldsymbol{\delta}} &= -k_{\delta} {}^o\mathbf{R}_c \mathbf{Q} \boldsymbol{\delta} \\ &= -k_{\delta} {}^o\mathbf{R}_c \mathbf{Q} {}^o\mathbf{R}_c^{\top} {}^o\boldsymbol{\delta}. \end{aligned} \quad (2.36)$$

On voit que ${}^o\boldsymbol{\delta}$ décroît exponentiellement, mais que sa décroissance est affectée par la matrice \mathbf{Q} . Or, il se trouve que \mathbf{Q} est en général mal conditionnée [Hamel 02] : $\lambda_{min}(\mathbf{Q}) \ll \lambda_{max}(\mathbf{Q})$, où $\lambda_{min}(\mathbf{Q})$ et $\lambda_{max}(\mathbf{Q})$ sont respectivement la plus petite et la plus grande des valeurs propres de \mathbf{Q} . Plus précisément, la troisième valeur propre de \mathbf{Q} est beaucoup plus petite que les autres. On le voit par exemple dans le cas de \mathbf{Q}^* , la valeur de la matrice \mathbf{Q} à la position désirée :

$$\mathbf{Q}^* \simeq \text{diag}(7.73, 7.81, 0.226). \quad (2.37)$$

Il s'ensuit que la convergence de ${}^o\boldsymbol{\delta}_z$, affectée par $\lambda_{min}(\mathbf{Q})$, est beaucoup plus lente que la convergence de ${}^o\boldsymbol{\delta}_x$ et ${}^o\boldsymbol{\delta}_y$. Bien que le système converge asymptotiquement, son comportement transitoire n'est donc pas satisfaisant.

Notons au passage qu'il n'est pas surprenant de constater un comportement différent entre l'axe optique \mathbf{z}_c de la caméra et les deux axes du plan image \mathbf{x}_c et \mathbf{y}_c . L'asservissement visuel 2D selon l'axe optique \mathbf{z}_c a d'ailleurs fait l'objet de travaux spécifiques, comme dans [Mahony 02a] où le comportement 3D du système selon l'axe \mathbf{z}_c est analysé pour différents choix d'informations visuelles, ou dans [Corke 01] où les composantes de vitesse selon l'axe \mathbf{z}_c sont traitées séparément des autres composantes.

La figure 2.10 présente les résultats expérimentaux obtenus en simulation avec la loi de commande (2.34). On peut voir que la convergence de ${}^o\boldsymbol{\delta}_z$ est beaucoup plus lente que la convergence des autres composantes ${}^o\boldsymbol{\delta}_x$ et ${}^o\boldsymbol{\delta}_y$. Il en est de même dans l'espace 3D, où la caméra atteint la position désirée en X et Y en quelques itérations, alors qu'il faut attendre longtemps pour que l'altitude Z parvienne à la valeur demandée.

La loi de commande proportionnelle permet d'obtenir de bonnes propriétés théoriques (en particulier système GAS). Cependant, le comportement transitoire obtenu, aussi bien pour les positions 3D que pour les informations visuelles, n'est pas satisfaisant.

2.4.3 Linéarisation à la position désirée

Afin d'uniformiser le comportement des trois composantes de l'information visuelle et de la position, on peut ensuite penser à utiliser l'inverse de la matrice d'interaction. C'est d'ailleurs ce qui est fait habituellement en asservissement visuel 2D [Chaumette 06].

En effet, si on choisit pour loi de commande

$$\mathbf{v} = k_{\mathbf{Q}} \mathbf{Q}^{-1} \boldsymbol{\delta}, \quad k_{\mathbf{Q}} > 0, \quad (2.38)$$

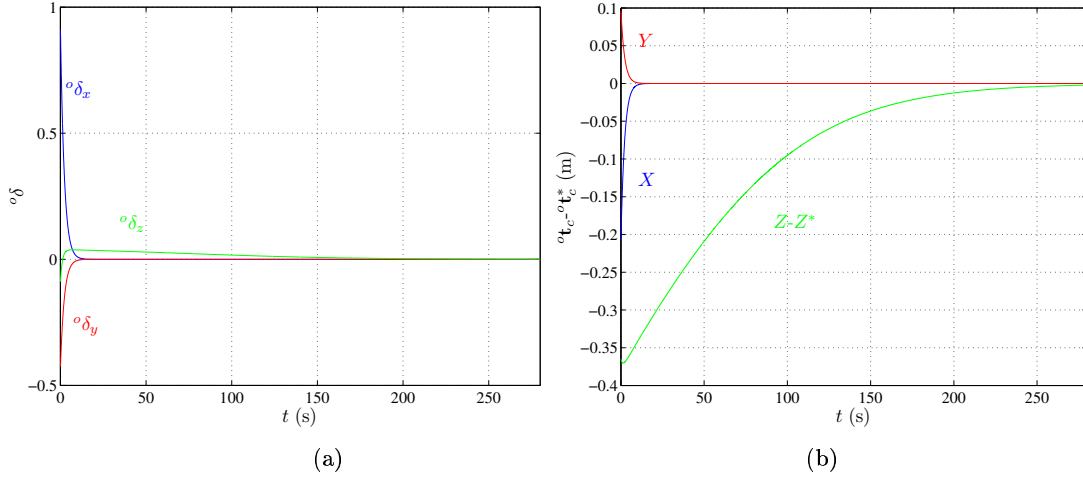


FIG. 2.10 – Commande proportionnelle $\boldsymbol{v} = k_\delta \boldsymbol{\delta}$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\boldsymbol{\delta} = {}^o\mathbf{R}_c \boldsymbol{\delta}$ (a), et de l'erreur sur la position de la caméra ${}^o \mathbf{t}_c - {}^o \mathbf{t}_c^*$ (b).

on obtient alors pour la dérivée de la fonction de Lyapunov (définie comme précédemment en (2.31)) :

$$\begin{aligned} \dot{\mathcal{L}} &= -k_Q \boldsymbol{\delta}^\top \mathbf{Q} \mathbf{Q}^{-1} \boldsymbol{\delta} \\ &= -k_Q \boldsymbol{\delta}^\top \boldsymbol{\delta}. \end{aligned} \quad (2.39)$$

On a alors $\dot{\mathcal{L}} < 0$, et donc un système GAS. De plus, l'effet de la matrice \mathbf{Q} est exactement compensé par \mathbf{Q}^{-1} , ce qui assure une décroissance exponentielle de chaque composante de l'erreur visuelle à la même vitesse. On a en effet :

$$\begin{aligned} {}^o \dot{\boldsymbol{\delta}} &= -k_Q {}^o \mathbf{R}_c \mathbf{Q} \mathbf{Q}^{-1} \boldsymbol{\delta} \\ &= -k_Q {}^o \mathbf{R}_c {}^o \mathbf{R}_c^\top {}^o \boldsymbol{\delta} \\ &= -k_Q {}^o \boldsymbol{\delta}. \end{aligned} \quad (2.40)$$

Ce choix semble donc intéressant en théorie. On voit sur la figure 2.11.a que les trois composantes de l'information visuelle convergent exponentiellement de la même manière, comme attendu théoriquement. Par contre, le comportement n'est pas aussi bon dans l'espace 3D. En effet, les trois composantes de la position convergent à la même vitesse, mais on remarque un dépassement important sur l'axe vertical (voir figure 2.11.b).

De plus, comme nous l'avons déjà mentionné, la matrice \mathbf{Q}^{-1} dépend de la position 3D de la cible, qui est en pratique inconnue sans calcul de pose — cadre dans lequel on se place pour notre étude.

Comme on le fait souvent en asservissement visuel, une solution consiste à utiliser la matrice d'interaction désirée \mathbf{Q}^* au lieu de la matrice d'interaction courante \mathbf{Q} [Mahony 02b]. On obtient alors la loi de commande suivante :

$$\boldsymbol{v} = k_* \mathbf{Q}^{*-1} \boldsymbol{\delta}, \quad k_* > 0, \quad (2.41)$$

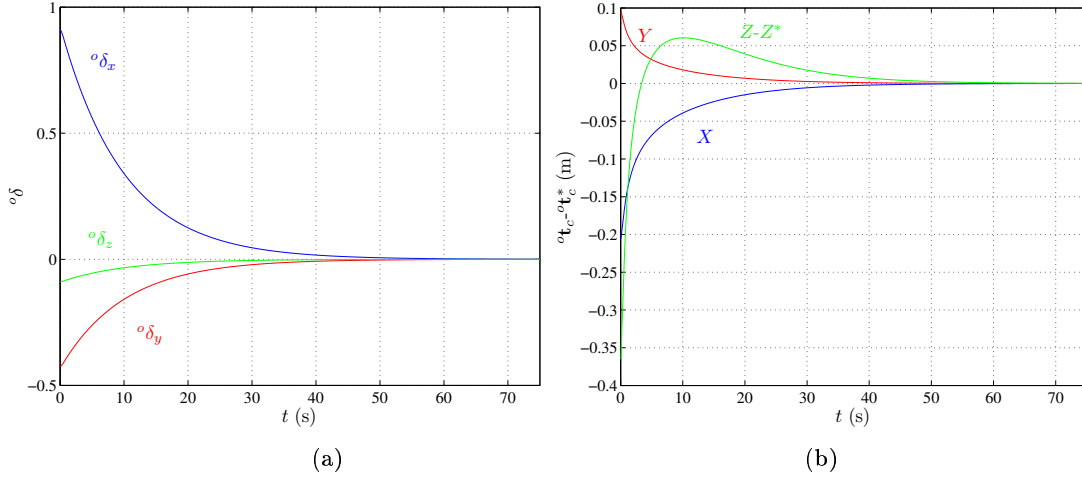


FIG. 2.11 – Commande classique $v = k_Q \mathbf{Q}^{-1} \delta$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\delta = {}^o\mathbf{R}_c \delta$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

et la dérivée de la fonction de Lyapunov devient

$$\dot{\mathcal{L}} = -k_* \delta^\top \mathbf{Q} \mathbf{Q}^{*-1} \delta. \quad (2.42)$$

On voit alors que $\dot{\mathcal{L}}$ est négative si $\mathbf{Q} \mathbf{Q}^{*-1}$ est définie positive. On ne l'a pas démontré pour tout l'espace, mais en pratique on n'a pas rencontré de zone instable, on a bien $\mathbf{Q} \mathbf{Q}^{*-1} > 0$ dans les simulations réalisées.

On peut voir sur la figure 2.12.a que la convergence des composantes de l'information visuelle se fait globalement à la même vitesse. Par contre, on remarque que initialement, les trois composantes ne se comportent pas de la même manière, contrairement aux résultats précédents (comparer 2.12.a et 2.11.a, pendant les 10 premières secondes). Concernant l'évolution dans l'espace 3D, les trois composantes de la position 3D de la caméra convergent en même temps. Cependant, on voit sur la figure 2.12.b que le comportement transitoire de l'altitude présente un dépassement, ce qui n'est pas souhaitable en pratique.

2.4.4 Commande partitionnée

Les problèmes rencontrés avec les deux lois de commande considérées dans les sections précédentes proviennent de la composante selon l'axe \mathbf{z}_c . Une solution consiste alors à isoler cette composante, selon un schéma d'asservissement visuel partitionné [Corke 01].

L'erreur sur les informations visuelles est séparée en deux parties δ_{11} et δ_{12} définies ainsi :

$$\delta_{11} = [\mathbf{q}_n^*]_\times \mathbf{q}, \quad \delta_{12} = \mathbf{q}_n^{*\top} \delta, \quad \text{avec } \mathbf{q}_n^* = \frac{\mathbf{q}^*}{|\mathbf{q}^*|}. \quad (2.43)$$

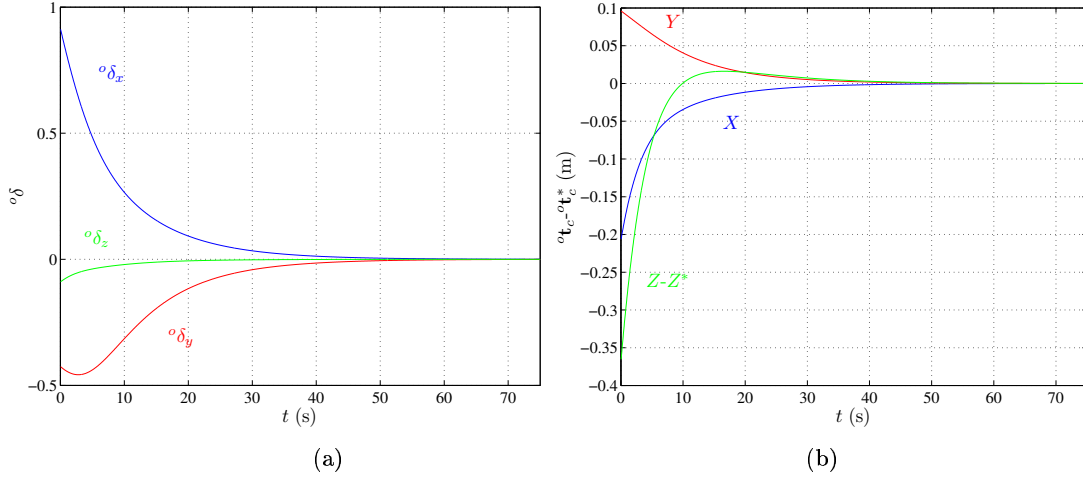


FIG. 2.12 – Commande reposant sur la linéarisation à la position désirée $\mathbf{v} = k_* \mathbf{Q}^{*-1} \delta$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\delta = {}^o\mathbf{R}_c \delta$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

δ_{11} est donc un vecteur de dimension 3, orthogonal à \mathbf{q}^* et à \mathbf{q} . δ_{12} est un scalaire qui représente la projection de δ sur l'axe de direction \mathbf{q}^* (voir figure 2.13).

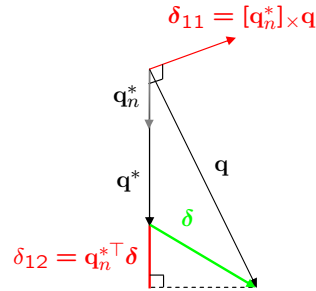


FIG. 2.13 – Schéma représentant $\delta_{11} = [\mathbf{q}_n^*] \times \mathbf{q}$ et $\delta_{12} = \mathbf{q}_n^{*\top} \delta$. Notons que δ_{11} est orthogonal à $\mathbf{q}_n^* \delta_{12}$.

Ensuite, au lieu de considérer comme précédemment l'erreur δ dans la loi de commande, on définit une nouvelle erreur $\delta_{\mathbf{A}}$ telle que :

$$\delta_{\mathbf{A}} = \delta_{11} + \lambda \mathbf{q}_n^* \delta_{12}, \quad (2.44)$$

où λ est un gain que l'on déterminera plus tard.

Notons que l'on a choisi $\delta_{\mathbf{A}}$ sous la forme d'une somme de deux termes :

- δ_{11} , qui, comme on l'a vu, est orthogonal à \mathbf{q}_n^* , et
- $\lambda \mathbf{q}_n^* \delta_{12}$, qui contient toute la composante de l'erreur selon la direction de \mathbf{q}_n^* .

L'objectif est de placer la caméra à la verticale de la cible. Lorsque l'on s'approche de la convergence, la direction de \mathbf{q}^* devient très proche de la direction de l'axe optique \mathbf{z}_c ,

ce qui permet donc d'ajuster la vitesse de convergence dans la direction de l'axe optique via le paramètre λ .

Soit $\mathcal{L} = \frac{1}{2}|\boldsymbol{\delta}_{\mathbf{A}}|^2$ une fonction de Lyapunov. Cette fonction peut s'écrire sous la forme suivante :

$$\mathcal{L} = \frac{1}{2}(|\boldsymbol{\delta}_{11}|^2 + \lambda^2 \delta_{12}^2) \quad (2.45)$$

puisque le terme croisé $[\boldsymbol{\delta}_{11}]_{\times} \mathbf{q}_n^*$ est nul, car par définition $\boldsymbol{\delta}_{11}$ est orthogonal à \mathbf{q}_n^* (2.43).

Pour calculer les dérivées de $\boldsymbol{\delta}_{11}$ et de δ_{12} , on utilise :

- la propriété des matrices de préproduit vectoriel $[\mathbf{u}]_{\times}[\mathbf{v}]_{\times} - [\mathbf{v}]_{\times}[\mathbf{u}]_{\times} = [[\mathbf{u}]_{\times}\mathbf{v}]_{\times}$, pour tout vecteur \mathbf{u} et \mathbf{v}
- la propriété des produits mixtes $\langle \mathbf{u}, \mathbf{v}, \mathbf{w} \rangle = \langle \mathbf{w}, \mathbf{u}, \mathbf{v} \rangle$ (le produit mixte reste inchangé par permutation circulaire des trois vecteurs), pour tout vecteur \mathbf{u}, \mathbf{v} et \mathbf{w} , et avec $\langle \mathbf{u}, \mathbf{v}, \mathbf{w} \rangle = \mathbf{u}^{\top}([\mathbf{v}]_{\times}\mathbf{w})$.

En rappelant (2.21), (2.25), (2.26), on peut alors calculer $\dot{\boldsymbol{\delta}}_{11}$ et $\dot{\delta}_{12}$:

$$\dot{\boldsymbol{\delta}}_{11} = -[\boldsymbol{\omega}]_{\times}\boldsymbol{\delta}_{11} - [\mathbf{q}_n^*]_{\times}\mathbf{Q}\mathbf{v}, \quad (2.46)$$

$$\dot{\delta}_{12} = -\mathbf{q}_n^{*\top}\mathbf{Q}\mathbf{v}, \quad (2.47)$$

et on en déduit l'expression de $\dot{\mathcal{L}}$ (voir calculs en annexe A.1) :

$$\dot{\mathcal{L}} = -\boldsymbol{\delta}_{\mathbf{A}}^{\top}\mathbf{A}(\mathbf{q}_n^*)\mathbf{Q}\mathbf{v}, \quad (2.48)$$

avec

$$\mathbf{A}(\mathbf{q}_n^*) = [\mathbf{q}_n^*]_{\times} + \lambda\mathbf{q}_n^*\mathbf{q}_n^{*\top}. \quad (2.49)$$

On choisit alors la loi de commande suivante :

$$\mathbf{v} = k_{\mathbf{A}}\mathbf{A}(\mathbf{q}_n^*)^{\top}\boldsymbol{\delta}_{\mathbf{A}}, \quad k_{\mathbf{A}} > 0. \quad (2.50)$$

Ainsi, en injectant l'expression de \mathbf{v} dans (2.48), on obtient :

$$\dot{\mathcal{L}} = -k_{\mathbf{A}}\boldsymbol{\delta}_{\mathbf{A}}^{\top}\mathbf{A}(\mathbf{q}_n^*)\mathbf{Q}\mathbf{A}(\mathbf{q}_n^*)^{\top}\boldsymbol{\delta}_{\mathbf{A}}. \quad (2.51)$$

Comme \mathbf{Q} est définie positive et que \mathbf{A} est une matrice non singulière, on en déduit que $\mathbf{A}(\mathbf{q}_n^*)\mathbf{Q}\mathbf{A}(\mathbf{q}_n^*)^{\top}$ est positive, et donc que le système bouclé est GAS, c'est-à-dire que $\boldsymbol{\delta}_{\mathbf{A}}$ converge vers 0. L'erreur visuelle $\boldsymbol{\delta}_{\mathbf{A}}$ étant composée des deux vecteurs orthogonaux $\boldsymbol{\delta}_{11}$ et $\mathbf{q}_n^*\delta_{12}$ (2.44), ceux-ci vont également converger vers 0. On en déduit alors que la direction de \mathbf{q} va converger vers la direction de \mathbf{q}^* (2.46), et que le module de l'erreur selon cette direction va converger vers 0 (2.47). Cela a pour conséquence directe que l'erreur initiale $\boldsymbol{\delta} = \mathbf{q} - \mathbf{q}^*$ converge vers 0.

Le gain λ est déterminé de manière à assurer un temps de convergence équivalent à chaque composante de $\boldsymbol{\delta}_{\mathbf{A}}$. Pour cela, on cherche à avoir $\mathbf{A}(\mathbf{q}_n^*)\mathbf{Q}\mathbf{A}(\mathbf{q}_n^*)^{\top} \cong \mathbf{I}_3$. Dans les expérimentations présentées, on a utilisé $\lambda = \sqrt{34}$, ce qui donne $\mathbf{A}(\mathbf{q}_n^*)\mathbf{Q}\mathbf{A}(\mathbf{q}_n^*)^{\top} \simeq \text{diag}(7.8, 7.7, 7.7)$.

La figure 2.14 présente les résultats obtenus avec cette loi de commande. En fait, le comportement transitoire de l'altitude est encore moins bon que précédemment : pour atteindre la position demandée, la distance entre la caméra et la cible doit diminuer, mais on constate sur la figure 2.14.b qu'elle commence par augmenter. Ce comportement inattendu s'explique par le fait que l'erreur δ est projetée selon la direction de \mathbf{q}^* pour amplifier la composante correspondante avec le gain λ . Cela est bien adapté asymptotiquement, lorsque l'axe de la caméra est proche de la position désirée. Comme illustré sur la figure 2.15.b, on a alors $\delta_{12} > 0$: le gain λ permet dans ce cas d'accélérer la convergence de la composante verticale. Cependant, initialement, δ_{12} est ici négatif (comme sur la figure 2.15.(a)) : λ amplifie donc un mouvement dans le mauvais sens durant cette phase.

Le partitionnement de l'erreur a permis d'établir une loi de commande assurant que le système est GAS. De plus, on a introduit un paramètre λ permettant aux trois composantes de l'erreur et de la position de converger à la même vitesse près de la position désirée. Cependant, la compensation utilisée pour ce faire n'est bien adaptée qu'asymptotiquement, c'est-à-dire lorsque la caméra est très proche de la position désirée.

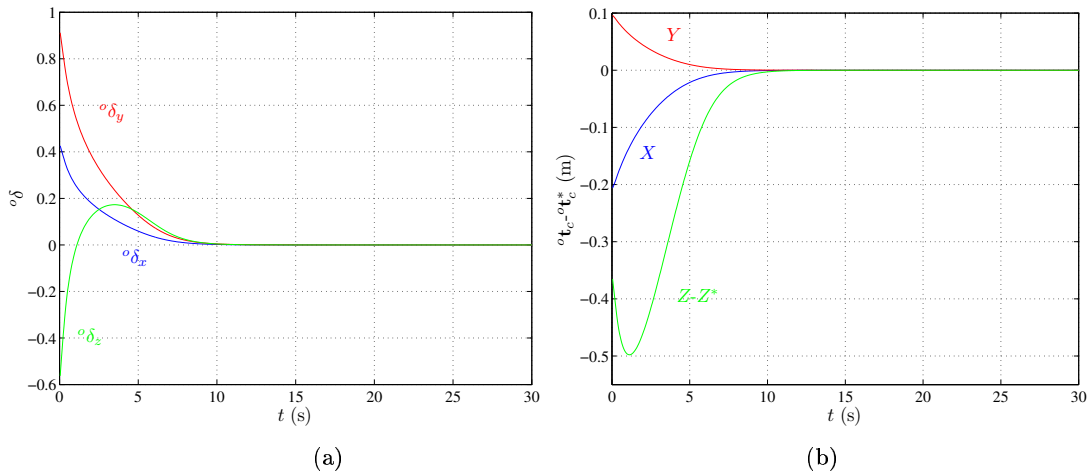


FIG. 2.14 – Commande partitionnée $\mathbf{v} = k_{\mathbf{A}} \mathbf{A}(\mathbf{q}_n^*)^\top \delta_{\mathbf{A}}$: évolution au cours du temps de l'erreur sur les informations visuelles $^\circ\delta_{\mathbf{A}} = ^\circ\mathbf{R}_c \delta_{\mathbf{A}}$ (a), et de l'erreur sur la position de la caméra $^\circ\mathbf{t}_c - ^\circ\mathbf{t}_c^*$ (b).

Notons que ce partitionnement a également été mis à profit dans [Guenard 07] dans le cadre d'un asservissement visuel dynamique du quadrirotor.

2.4.5 Changement d'échelle de l'information visuelle

Les lois de commande précédentes (linéarisation à la position désirée, section 2.4.3 et partitionnement de l'erreur, section 2.4.4) n'ont pas permis de résoudre complètement le problème rencontré au départ (convergence très lente de la composante selon l'axe \mathbf{z}_c avec la loi de commande proportionnelle, section 2.4.2). En effet, dans les deux cas, compenser asymptotiquement le temps de convergence améliore effectivement le

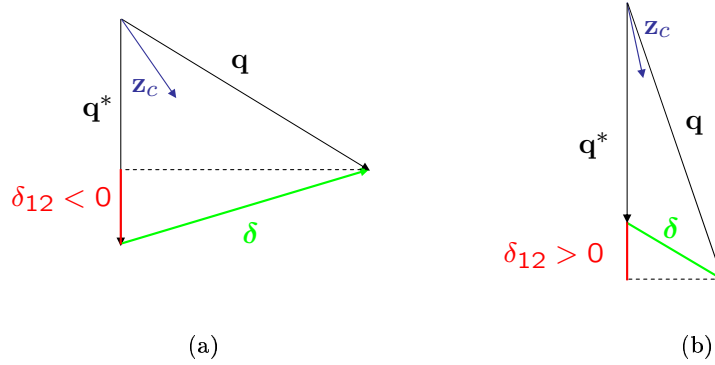


FIG. 2.15 – Schéma représentant δ_{12} initialement (a), et lorsqu'on est proche de la convergence (b), dans le cas de l'expérience considérée ici.

comportement asymptotique, mais produit des effets inopportuns (voire désastreux si un robot atteint ses butées, si le quadrirotor s'élève beaucoup et que la cible visuelle devient de ce fait trop petite dans l'image donc mal détectable, si le quadrirotor en vient à toucher le plafond...) sur le comportement transitoire.

Nous proposons donc d'utiliser une nouvelle information visuelle, conçue à partir de l'erreur δ , à laquelle nous appliquons un facteur de changement d'échelle. Le but est double : assurer la convergence de toutes les composantes de l'information visuelle à la même vitesse, et améliorer la relation entre l'espace image et l'espace 3D, tout au long de l'asservissement. Pour ce faire, on cherche des informations visuelles aussi proches que possible de la position 3D de la caméra, afin d'obtenir une matrice d'interaction proche de l'identité et de découpler les ddls [Tahri 05]. Une loi de commande 2D assurant un comportement satisfaisant des informations visuelles permettra ainsi d'obtenir de bonnes trajectoires pour la caméra dans l'espace 3D.

Comme nous l'avons vu, l'essentiel du problème réside dans le comportement du système selon l'axe optique de la caméra. Nous avons déjà évoqué le fait que la composante de \mathbf{q} selon l'axe \mathbf{z}_c est fonction de la variation de la profondeur Z entre la caméra et la cible observée (voir section 2.4.1 et figure 2.9). De simples considérations géométriques nous permettent d'obtenir une approximation de cette relation (voir figure 2.16) :

$$Z \simeq \frac{R|\mathbf{q}|}{\sqrt{n^2 - |\mathbf{q}|^2}} \quad (2.52)$$

où n est le nombre de points de la cible, et R le rayon de la cible. Par la suite, on notera

$$F(|\mathbf{q}|) = \frac{R|\mathbf{q}|}{\sqrt{n^2 - |\mathbf{q}|^2}}. \quad (2.53)$$

On voit sur la figure 2.17 qu'utiliser l'information visuelle \mathbf{q} pour commander les mouvements de profondeur devrait bien fonctionner lorsque la caméra est très proche

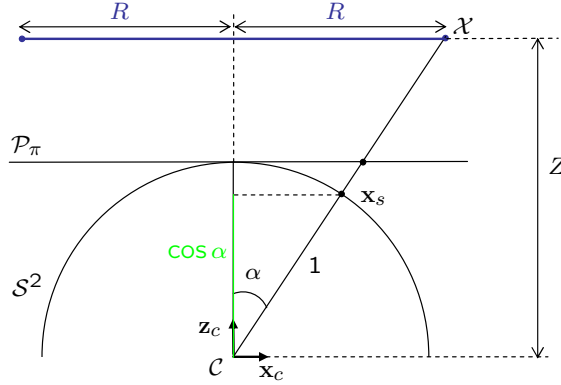


FIG. 2.16 – Projection d'un point \mathcal{X} de la cible de rayon R sur la sphère \mathcal{S}^2 . On se place dans le cas simple où la cible est symétrique, centrée et parallèle au plan image. On a $Z = R \frac{\cos \alpha}{\sin \alpha}$, d'où en élevant tout au carré $Z^2 = \frac{R^2 (\cos \alpha)^2}{1 - (\cos \alpha)^2}$. Or, chaque point \mathbf{x}_s apporte $\cos \alpha$ comme contribution à $|\mathbf{q}|$ sur l'axe \mathbf{z}_c . Donc pour n points, on a $|\mathbf{q}| = n \cos \alpha$, d'où on tire $\cos \alpha = \frac{|\mathbf{q}|}{n}$. En remplaçant $\cos \alpha$ par cette expression, on obtient la relation suivante : $Z = \frac{R|\mathbf{q}|}{\sqrt{n^2 - |\mathbf{q}|^2}}$.

de la cible. En effet, dans ce cas la relation entre l'information visuelle et la profondeur est quasi linéaire. Cependant, en pratique on ne travaille pas dans cette zone, mais dans la partie de la courbe où $|\mathbf{q}|$ varie très peu lorsque la profondeur varie beaucoup. On peut également se rendre compte de ce phénomène en constatant que finalement une très petite partie de la sphère est utilisée en pratique. Déjà, la dimension finie du plan image \mathcal{P}_π limite la surface utilisée sur \mathcal{S}^2 (voir figure 2.18). De plus, ce phénomène est encore plus important en pratique puisqu'on ne travaille jamais dans les conditions limites où la cible est tout au bord de l'image. Ainsi, en pratique l'information visuelle \mathbf{q} varie très peu quand la profondeur varie (voir figure 2.18).

Afin de tenir compte de ce problème inhérent à l'information visuelle \mathbf{q} , nous proposons une nouvelle information visuelle \mathbf{f} définie à partir de \mathbf{q} et du facteur d'échelle (2.53) :

$$\mathbf{f} = F(|\mathbf{q}|)\mathbf{q}_n \quad (2.54)$$

où \mathbf{q}_n est, comme dans la section précédente, le moment du premier ordre normalisé : $\mathbf{q}_n = \frac{\mathbf{q}}{|\mathbf{q}|}$. On peut remarquer que \mathbf{f} dépend du rayon de la cible R . En pratique ce paramètre peut être considéré comme un gain, et il suffit d'utiliser une valeur approximative de R . Les propriétés des lois de commande proposées par la suite ne sont pas affectées par R , du moment que l'on choisit $R > 0$, ce qui est forcément le cas en pratique.

À partir de l'information visuelle \mathbf{f} , on définit l'information visuelle désirée $\mathbf{f}^* = F(|\mathbf{q}^*|)\mathbf{q}_n^*$ puis l'erreur $\delta_{\mathbf{f}}$ comme suit :

$$\begin{aligned} \delta_{\mathbf{f}} &= \mathbf{f} - \mathbf{f}^* \\ &= F(|\mathbf{q}|)\mathbf{q}_n - F(|\mathbf{q}^*|)\mathbf{q}_n^*. \end{aligned} \quad (2.55)$$

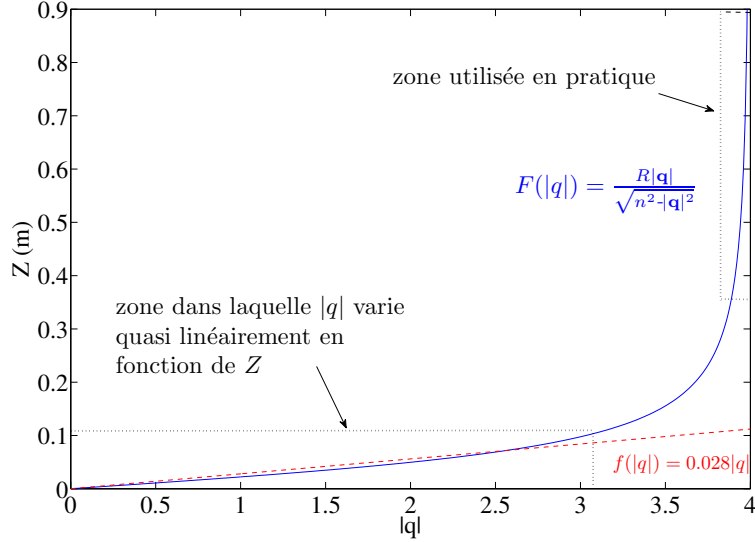


FIG. 2.17 – Représentation de la relation entre $|\mathbf{q}|$ et la profondeur Z (avec $R = 0.086$ et $n = 4$). On voit que pour $Z \lesssim 0.12$ m, la relation est quasi linéaire ($Z \simeq 0.028|\mathbf{q}|$). Pour $Z \gtrsim 0.12$ m $|\mathbf{q}|$ ne varie presque pas.

On cherche maintenant, comme dans les sections précédentes, à exprimer $\dot{\delta}_{\mathbf{f}}$ en fonction des vitesses de la caméra $\boldsymbol{\omega}$ et \mathbf{v} . Pour ce faire, commençons par exprimer la dérivée de \mathbf{f} . On a, en dérivant (2.54) :

$$\dot{\mathbf{f}} = \frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|} |\mathbf{q}| \dot{\mathbf{q}}_n + F(|\mathbf{q}|) \dot{\mathbf{q}}_n. \quad (2.56)$$

En utilisant (2.21), on obtient alors (les détails du calcul sont donnés dans l'annexe A.2) :

$$\dot{\mathbf{f}} = -[\boldsymbol{\omega}]_{\times} \mathbf{f} - \mathbf{M} \mathbf{Q} \mathbf{v}, \quad (2.57)$$

avec

$$\mathbf{M}(\mathbf{q}) = \frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|} \mathbf{q}_n \mathbf{q}_n^{\top} + \frac{F(|\mathbf{q}|)}{|\mathbf{q}|} (\mathbf{I}_3 - \mathbf{q}_n \mathbf{q}_n^{\top}). \quad (2.58)$$

De la même manière que $\dot{\mathbf{q}}^* = -[\boldsymbol{\omega}]_{\times} \mathbf{q}^*$, \mathbf{f}^* n'est fonction que de la rotation de la caméra : $\dot{\mathbf{f}}^* = -[\boldsymbol{\omega}]_{\times} \mathbf{f}^*$. On en déduit l'expression de $\dot{\delta}_{\mathbf{f}}$, avec (2.57) :

$$\dot{\delta}_{\mathbf{f}} = -[\boldsymbol{\omega}]_{\times} \delta_{\mathbf{f}} - \mathbf{M} \mathbf{Q} \mathbf{v}. \quad (2.59)$$

Notons que l'information visuelle $\delta_{\mathbf{f}}$ possède la propriété de passivité, tout comme δ . On voit en effet que ${}^o \delta_{\mathbf{f}} = {}^o \mathbf{R}_c \delta_{\mathbf{f}}$, est indépendant des mouvements de rotation de la caméra, puisque comme dans la partie 2.4.1, on obtient :

$${}^o \dot{\delta}_{\mathbf{f}} = -{}^o \mathbf{R}_c \mathbf{M} \mathbf{Q} \mathbf{v}. \quad (2.60)$$

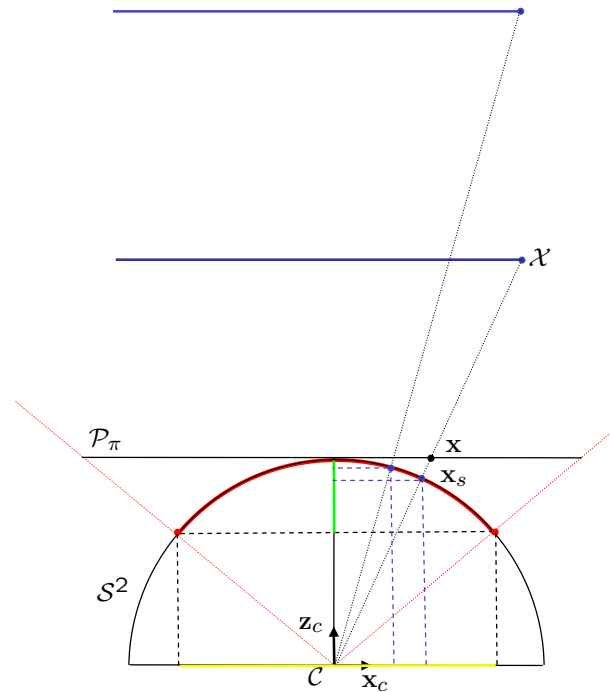


FIG. 2.18 – Schéma de la projection sphérique. En pointillées rouges sont représentés les rayons de projection limites, correspondant à une image aux extrémités du plan image \mathcal{P}_π . Ces rayons délimitent la partie de la sphère susceptible d'être utilisée (en rouge). On voit alors que pour un point de la sphère, sa coordonnée selon l'axe z_c (qui est sa contribution à la composante de \mathbf{q} sur l'axe z_c) ne varie que très peu en fonction de la distance du point : en vert est représentée l'amplitude des variations possibles de cette composante. On a également représenté en jaune les valeurs possibles pour la composante d'un point selon l'axe x_c : l'amplitude de variation est beaucoup plus importante que selon z_c . En bleu on a représenté deux positions "réalistes" de la cible (pour deux profondeurs différentes) : on voit que les composantes selon z_c de leurs projections sphériques sont très proches.

Considérons à présent la fonction de Lyapunov $\mathcal{L} = \frac{1}{2}|\boldsymbol{\delta}_f|^2$. En utilisant (2.59), on obtient immédiatement :

$$\dot{\mathcal{L}} = -\boldsymbol{\delta}_f^\top \mathbf{M} \mathbf{Q} \mathbf{v}. \quad (2.61)$$

En nous basant sur cette nouvelle information visuelle \mathbf{f} et sur $\boldsymbol{\delta}_f$, nous avons ensuite considéré trois lois de commande.

2.4.5.1 Commande stable

La première idée consiste, à la vue de (2.61), à choisir

$$\mathbf{v} = k_M \mathbf{M} \boldsymbol{\delta}_f, \quad k_M > 0, \quad (2.62)$$

de manière à assurer la stabilité du système. En effet, on obtient ainsi

$$\dot{\mathcal{L}} = -k_M \boldsymbol{\delta}_f^\top \mathbf{M}(\mathbf{q}) \mathbf{Q} \mathbf{M}(\mathbf{q}) \boldsymbol{\delta}_f. \quad (2.63)$$

$\mathbf{M}(\mathbf{q})$ étant une matrice symétrique non singulière (voir (2.58)) et $\mathbf{Q} > 0$, la théorie de Lyapunov nous permet comme précédemment de conclure que le système est GAS et que l'erreur $\boldsymbol{\delta}_f$ converge vers zéro. De plus, les temps de convergence sont donnés par les valeurs propres de $\mathbf{M}(\mathbf{q}) \mathbf{Q} \mathbf{M}(\mathbf{q})$.

Notons ${}^c \mathbf{t}_o$ la position de la cible par rapport à la caméra, exprimée dans le repère lié à la caméra \mathcal{F}_c . Les relations classiques de cinématique nous donnent alors ${}^c \dot{\mathbf{t}}_o = -[\boldsymbol{\omega}]_\times {}^c \mathbf{t}_o - \mathbf{v}$. L'information visuelle \mathbf{f} étant choisie de manière à être une approximation de la position de l'objet par rapport à la caméra, exprimée dans \mathcal{F}_c , on a $\mathbf{f} \simeq {}^c \mathbf{t}_o$. On en déduit alors

$$\dot{\mathbf{f}} \simeq -[\boldsymbol{\omega}]_\times \mathbf{f} - \mathbf{v}. \quad (2.64)$$

En identifiant les termes de (2.57), on voit alors que $\mathbf{M} \mathbf{Q} \simeq \mathbf{I}_3$, et donc

$$\mathbf{M} \simeq \mathbf{Q}^{-1}. \quad (2.65)$$

On en déduit que $\mathbf{M} \mathbf{Q} \mathbf{M} \simeq \mathbf{Q}^{-1}$. Comme on l'a déjà vu, la matrice \mathbf{Q} est mal conditionnée. Donc $\mathbf{Q}^{-1} \simeq \mathbf{M} \mathbf{Q} \mathbf{M}$ l'est aussi. La loi de commande proposée conduit donc à un comportement inverse de celui de la section 2.4.2 : $\lambda_{\min}(\mathbf{Q})$, la plus petite valeur propre de \mathbf{Q} , devient $\lambda_{\max}(\mathbf{Q}^{-1})$, la plus grande valeur propre de \mathbf{Q}^{-1} . Cette valeur propre affecte la composante selon l'axe \mathbf{z}_c , ce qui conduit à une convergence beaucoup plus rapide selon cette direction que selon les autres. Par exemple, dans la simulation, on a $\mathbf{Q}^{-1} \simeq \text{diag}(0.129, 0.128, 4.41)$.

On peut voir ce phénomène sur la figure 2.19 : dans l'espace image comme dans l'espace 3D, on se retrouve confronté à un problème de convergence similaire à celui rencontré initialement (section 2.4.2). On avait alors une convergence très lente sur l'axe \mathbf{z}_c . Ici, c'est la convergence selon les axes \mathbf{x}_c et \mathbf{y}_c qui est lente.

La loi de commande proposée en (2.62) permet d'assurer la stabilité du système. Du fait du choix de l'information visuelle, le comportement dans l'espace 3D est très proche

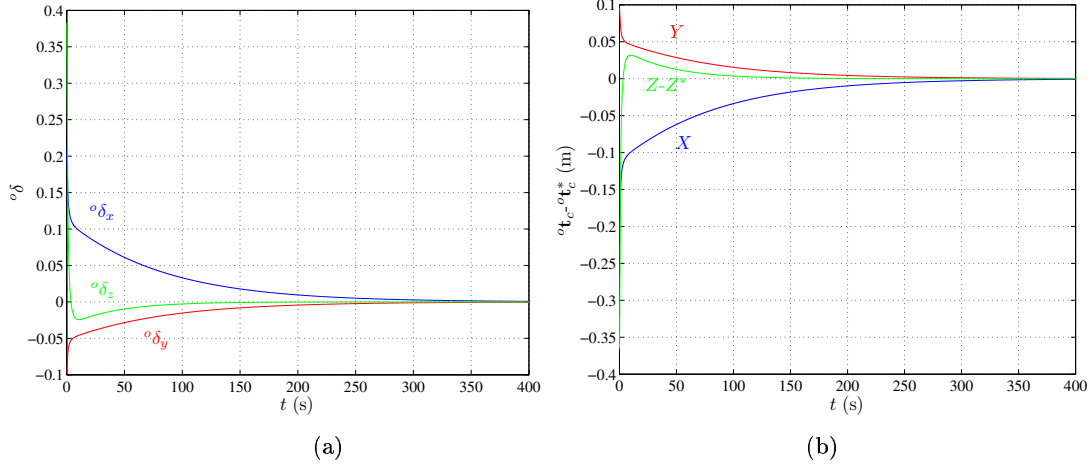


FIG. 2.19 – Commande stable $\mathbf{v} = k_M \mathbf{M} \delta_f$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\delta_f = {}^o\mathbf{R}_c \delta_f$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

du comportement dans l'espace image (relation quasi linéaire et découplée entre les informations visuelles et les positions 3D). Cependant, en pratique on retrouve le problème de temps de convergence rencontré précédemment, du fait du mauvais conditionnement de \mathbf{Q} .

2.4.5.2 Commande classique avec approximation de la matrice d'interaction

Comme nous l'avons vu à la section 2.4.3, la loi de commande $\mathbf{v} = k_Q \mathbf{Q}^{-1} \delta$, $k_Q > 0$ ne peut pas être utilisée en pratique, puisque \mathbf{Q} dépend de la position 3D des points de la cible. Ainsi, dans 2.4.3, nous avons utilisé l'approche classique qui consiste à considérer la valeur désirée de \mathbf{Q} .

Or, avec l'approximation (2.65), la matrice \mathbf{M} constitue une approximation de la matrice d'interaction \mathbf{Q}^{-1} . Nous proposons alors la loi de commande suivante :

$$\mathbf{v} = k_q \mathbf{M} \delta, \quad k_q > 0. \quad (2.66)$$

\mathbf{M} n'est pas constante, et on espère donc obtenir de meilleurs résultats qu'avec \mathbf{Q}^{*-1} (section 2.4.3). Cependant, comme on peut le voir sur la figure 2.20, le comportement dans l'espace 3D pose toujours problème : la phase transitoire présente un dépassement sur l'axe \mathbf{z}_c . L'approximation proposée ici semblait a priori meilleure que les solutions asymptotiques des sections 2.4.3 et 2.4.4, mais au final les performances du système ne sont pas réellement améliorées.

2.4.5.3 Commande proportionnelle

Puisque $\mathbf{f} \simeq {}^c\mathbf{t}_o$, une commande intuitive consiste à choisir tout simplement :

$$\mathbf{v} = k_f \delta_f, \quad k_f > 0, \quad (2.67)$$

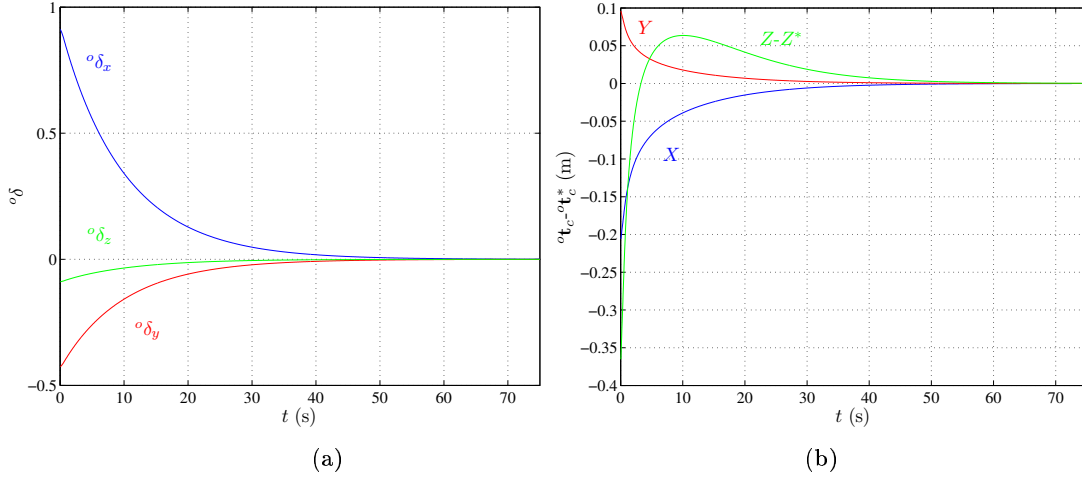


FIG. 2.20 – Commande classique avec approximation de la matrice d'interaction $\mathbf{v} = k_q \mathbf{M} \delta$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\delta_f = {}^o\mathbf{R}_c \delta_f$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

ce qui permet de s'approcher d'une commande dans l'espace 3D.

En injectant \mathbf{v} dans (2.61), on obtient alors pour la dérivée de la fonction de Lyapunov :

$$\dot{\mathcal{L}} = -k_f \delta_f^\top \mathbf{M} \mathbf{Q} \delta_f. \quad (2.68)$$

Comme nous venons de le voir, $\mathbf{M} \mathbf{Q} \simeq \mathbf{I}_3$, donc on espère obtenir des temps de convergence très proches pour chaque composante de l'erreur visuelle et de la position.

On voit sur la figure 2.21 que les résultats obtenus sont très satisfaisants. En effet, le choix de la loi de commande (2.67) permet que l'erreur sur les informations visuelles converge vers 0 de manière quasi exponentielle (voir figure 2.21.a), et à la même vitesse pour toutes les composantes. De plus, le choix de l'information visuelle \mathbf{f} implique que ce bon comportement se retrouve directement dans l'espace des positions 3D (voir figure 2.21.b). La position d'équilibre est atteinte en un temps raisonnable, puisque les trois composantes convergent à la même vitesse (on n'a plus besoin, comme précédemment, "d'attendre" que certaines composantes convergent).

Notons qu'un avantage certain à la loi de commande proposée (2.67) réside dans sa simplicité de mise en œuvre : ce n'est qu'un simple correcteur proportionnel sur l'erreur visuelle δ_f .

Bien que les résultats obtenus en pratique soient très satisfaisants, nous n'avons pas démontré la stabilité du système. En effet, rien ne dit que $\mathbf{M} \mathbf{Q} > 0$ dans tout l'espace de travail, donc la fonction de Lyapunov (2.68) ne nous permet pas de conclure.

2.4.6 Commande stable avec information visuelle modifiée

Dans cette section, on cherche à mettre au point un schéma de commande basé sur l'information visuelle sphérique \mathbf{q} , et possédant à la fois toutes les bonnes propriétés

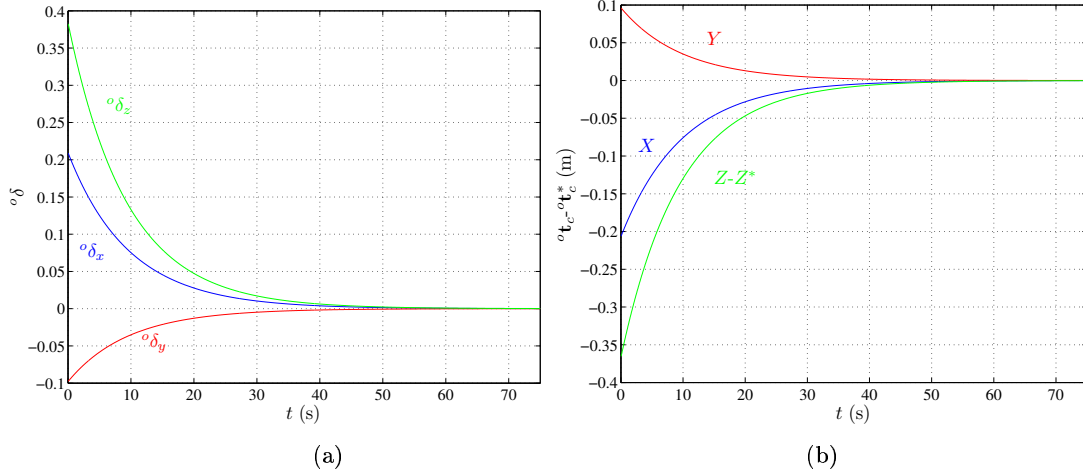


FIG. 2.21 – Commande proportionnelle $\mathbf{v} = k_f \delta_f$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\delta_f = {}^o\mathbf{R}_c \delta_f$ (a), et de l'erreur sur la position de la caméra ${}^o t_c - {}^o t_c^*$ (b).

vues précédemment :

- passivité,
- convergence à la même vitesse de toutes les composantes,
- bon comportement pendant la phase transitoire,
- lien "aussi linéaire que possible" entre l'espace image et l'espace 3D,
- système GAS.

Pour cela, on s'inspire du schéma de commande précédent, auquel il ne manque que la propriété de stabilité. On définit un nouveau facteur d'échelle $G(|\mathbf{q}|)$, et une nouvelle information visuelle \mathbf{g} telle que

$$\mathbf{g} = G(|\mathbf{q}|)\mathbf{q}_n \quad (2.69)$$

avec

$$\dot{\mathbf{g}} = -[\boldsymbol{\omega}]_{\times} \mathbf{g} - \mathbf{H}\mathbf{Q}\mathbf{v}. \quad (2.70)$$

$G(|\mathbf{q}|)$ va alors être choisi de manière à ce que la matrice \mathbf{H} permette d'obtenir la propriété de stabilité asymptotique.

De la même manière que pour la matrice \mathbf{M} (équation (2.58), calculs donnés en annexe A.2), on obtient la relation entre la matrice \mathbf{H} et l'information visuelle \mathbf{q} :

$$\mathbf{H}(\mathbf{q}) = \frac{\partial G(|\mathbf{q}|)}{\partial |\mathbf{q}|} \mathbf{q}_n \mathbf{q}_n^{\top} + \frac{G(|\mathbf{q}|)}{|\mathbf{q}|} (\mathbf{I}_3 - \mathbf{q}_n \mathbf{q}_n^{\top}). \quad (2.71)$$

On définit ensuite l'erreur sur les informations visuelles :

$$\begin{aligned} \delta_{\mathbf{g}} &= \mathbf{g} - \mathbf{g}^* \\ &= G(|\mathbf{q}|)\mathbf{q}_n - G(|\mathbf{q}^*|)\mathbf{q}_n^*. \end{aligned} \quad (2.72)$$

Avec (2.70), on obtient :

$$\dot{\boldsymbol{\delta}}_{\mathbf{g}} = -\boldsymbol{\omega} \times \boldsymbol{\delta}_{\mathbf{g}} - \mathbf{H}\mathbf{Q}\mathbf{v} \quad (2.73)$$

et on voit immédiatement (comme avec $\boldsymbol{\delta}_{\mathbf{f}}$, section 2.4.5) que l'erreur visuelle $\boldsymbol{\delta}_{\mathbf{g}}$ possède la propriété de passivité.

La première idée qui vient ensuite, afin d'assurer que le système soit GAS, est de choisir pour loi de commande $\mathbf{v} = k_{\mathbf{g}}\mathbf{H}\boldsymbol{\delta}_{\mathbf{g}}$, $k_{\mathbf{g}} > 0$. En effet, ce choix donnera pour la dérivée de la fonction de Lyapunov $\mathcal{L} = \frac{1}{2}|\boldsymbol{\delta}_{\mathbf{g}}|^2$: $\dot{\mathcal{L}} = -k_{\mathbf{g}}\boldsymbol{\delta}_{\mathbf{g}}^{\top}\mathbf{H}\mathbf{Q}\mathbf{H}\boldsymbol{\delta}_{\mathbf{g}}$. Étant donné que $\mathbf{Q} > 0$, et que \mathbf{H} est une matrice symétrique non singulière (voir (2.71)), la théorie de Lyapunov nous permettra une nouvelle fois de conclure à la stabilité asymptotique globale du système. L'erreur visuelle $\boldsymbol{\delta}_{\mathbf{g}}$ convergera exponentiellement vers 0, et le temps de convergence des différentes composantes sera donné par les valeurs propres de $\mathbf{H}\mathbf{Q}\mathbf{H}$. On voit alors que si l'on fait en sorte que $\mathbf{H}\mathbf{Q}\mathbf{H}$ soit bien conditionnée et aussi proche que possible de la matrice identité, les composantes de l'erreur vont toutes converger à la même vitesse. À la vue de (2.65), c'est-à-dire $\mathbf{M} \simeq \mathbf{Q}^{-1}$, un choix intuitif pour \mathbf{H} consiste à prendre $\mathbf{H} = \sqrt{\mathbf{M}}$, ce qui donnera $\mathbf{H}\mathbf{Q}\mathbf{H} \simeq \mathbf{I}_3$.

En pratique, les contraintes ainsi obtenues sur \mathbf{H} et $G(|\mathbf{q}|)$ sont très restrictives, et ce problème n'a pas de solution simple. On relâche donc les contraintes par l'introduction d'un paramètre $\alpha(|\mathbf{q}|)$ tel que

$$\mathbf{H}\mathbf{Q}\mathbf{H} \simeq \alpha(|\mathbf{q}|)^2\mathbf{I}_3 \quad (2.74)$$

et $\alpha(|\mathbf{q}^*|) = 1$. L'expression de $\alpha(|\mathbf{q}|)$ peut alors être obtenue en résolvant une équation différentielle ordinaire du premier ordre (les détails du calcul sont donnés dans l'annexe A.3).

Choisir la loi de commande

$$\mathbf{v} = k_{\mathbf{g}} \frac{\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \boldsymbol{\delta}_{\mathbf{g}}, \quad k_{\mathbf{g}} > 0, \quad (2.75)$$

donne alors pour la dérivée de la fonction de Lyapunov

$$\dot{\mathcal{L}} = -k_{\mathbf{g}}\boldsymbol{\delta}_{\mathbf{g}}^{\top} \frac{\mathbf{H}(\mathbf{q})\mathbf{Q}\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \boldsymbol{\delta}_{\mathbf{g}}, \quad (2.76)$$

et la théorie de Lyapunov garantit alors que le système est GAS, $\boldsymbol{\delta}_{\mathbf{g}}$ converge vers 0.

Les temps de convergence des composantes de $\boldsymbol{\delta}_{\mathbf{g}}$ sont donnés par les valeurs propres de $\frac{\mathbf{H}(\mathbf{q})\mathbf{Q}\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2}$. Or, nous avons conçu l'information visuelle $\boldsymbol{\delta}_{\mathbf{g}}$ de manière à avoir $\mathbf{H}(\mathbf{q})\mathbf{Q}\mathbf{H}(\mathbf{q}) \simeq \alpha(|\mathbf{q}|)^2\mathbf{I}_3$ et $\alpha(|\mathbf{q}^*|)^2 = 1$. Cela assure donc un bon comportement asymptotique des composantes de l'erreur visuelle.

La nouvelle information visuelle $\mathbf{g} = G(|\mathbf{q}|)\mathbf{q}_n$ est conçue de la même manière que la précédente $\mathbf{f} = F(|\mathbf{q}|)\mathbf{q}_n$: la direction est donnée par \mathbf{q}_n , et le module est donné par le facteur d'échelle $G(|\mathbf{q}|)$ et $F(|\mathbf{q}|)$ respectivement. La figure 2.22 montre que la pente de $G(|\mathbf{q}|)$ est moins raide que celle de $F(|\mathbf{q}|)$. Ainsi, $G(|\mathbf{q}|)$ devrait être moins sensible au bruit de mesure contenu dans l'information visuelle $|\mathbf{q}|$. Par contre, la relation entre

l'espace image et l'espace 3D n'est pas aussi linéaire avec l'information visuelle \mathbf{g} qu'avec \mathbf{f} . En effet, $F(|\mathbf{q}|)$ étant issu d'un calcul approximatif de la profondeur, les composantes de l'information visuelle $\mathbf{f} = F(|\mathbf{q}|)\mathbf{q}_n$ correspondent directement aux positions de la caméra dans l'espace 3D. Avec \mathbf{g} , on a $G(|\mathbf{q}|) = \alpha(|\mathbf{q}|)\sqrt{|\mathbf{q}|F(|\mathbf{q}|)}$, et la relation entre espace image et espace 3D est donc non linéaire (voir figure 2.22). Cela peut induire un comportement inadapté de la caméra dans l'espace 3D pendant la phase transitoire. En pratique, on n'a cependant pas observé de tels phénomènes.

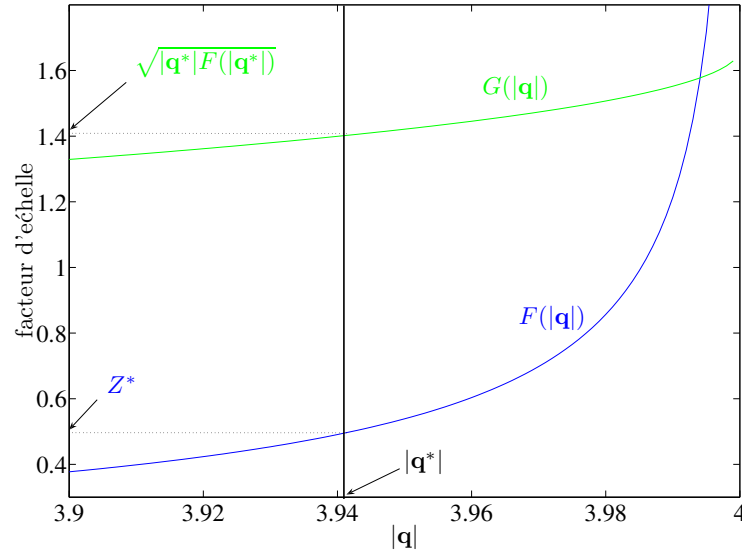


FIG. 2.22 – Courbes représentant les facteurs d'échelle $F(|\mathbf{q}|) = \frac{R|\mathbf{q}|}{\sqrt{n^2 - |\mathbf{q}|^2}}$ et $G(|\mathbf{q}|) = \alpha(|\mathbf{q}|)\sqrt{|\mathbf{q}|F(|\mathbf{q}|)}$ (avec $R = 0.086$, $n = 4$), dans la zone utilisée en pratique. Notons que $F(|\mathbf{q}^*|) = Z^*$ et $G(|\mathbf{q}^*|) = \sqrt{|\mathbf{q}^*|Z^*}$.

Les résultats obtenus en simulation sont très satisfaisants (voir figure 2.23). Le comportement dans l'espace image (voir figure 2.23.a), tout comme le comportement dans l'espace 3D (voir figure 2.23.b) est très bon, tant pendant la phase transitoire qu'asymptotiquement. Par rapport à la loi de commande précédente, on peut dire que les résultats pratiques sont aussi bons. Sur le plan théorique, assurer la stabilité asymptotique globale du système nous a conduit à perdre le lien quasi linéaire entre espace image et espace 3D.

Il est évident que l'implémentation de cette dernière loi de commande requiert plus d'étapes de calculs que le simple correcteur proportionnel de la section précédente. La figure 2.24 montre comment faire en pratique pour calculer la loi de commande à partir des informations visuelles \mathbf{q} et \mathbf{q}^* .

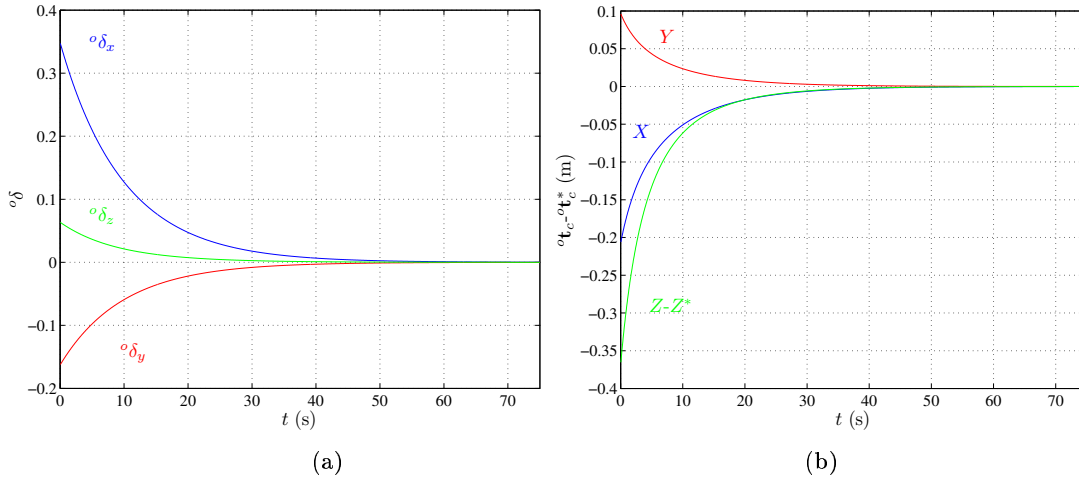


FIG. 2.23 – Commande stable $\mathbf{v} = k_{\mathbf{g}} \frac{\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \delta_{\mathbf{g}}$: évolution au cours du temps de l'erreur sur les informations visuelles ${}^o\delta_{\mathbf{g}} = {}^o\mathbf{R}_c \delta_{\mathbf{g}}$ (a), et de l'erreur sur la position de la caméra ${}^o\mathbf{t}_c - {}^o\mathbf{t}_c^*$ (b).

2.5 Analyse des propriétés des lois de commande étudiées

2.5.1 Propriétés des lois de commande

Dans les parties 2.3 et 2.4, nous avons présenté une série de lois de commande. Le tableau 2.1 dresse un récapitulatif des propriétés de ces lois de commande, en termes de stabilité, comportement transitoire, linéarité et passivité.

En pratique, le bon comportement transitoire (convergence de toutes les composantes de la position sans divergence ni dépassement transitoires, noté CT), et le fait que le temps de convergence asymptotique soit identique pour toutes les composantes de la position (noté BLES pour *Balanced Locally Exponentially Stable*), sont deux propriétés essentielles. Avec les lois de commande 2 et 5, toutes les composantes de la position ne convergent pas à la même vitesse, ce qui pose évidemment problème en pratique. Quant aux lois de commande 3, 4, et 6, toutes trois peuvent conduire à des comportements transitoires inattendus, inacceptables en pratique. Les trois lois de commande 1 (loi de commande utilisant les moments de la projection perspective), 7 (correcteur proportionnel, avec information visuelle sphérique et changement d'échelle) et 8 (loi de commande stable reposant sur une information visuelle sphérique modifiée) donnent pour leur part de bons résultats en pratique.

Une relation approximativement linéaire entre l'espace image et l'espace 3D (notée RAL) est également très utile. En effet, la loi de commande d'asservissement visuel 2D contraint le comportement des informations visuelles dans l'espace image. Lorsque le lien entre espace image et espace 3D est approximativement linéaire, un bon comportement dans l'espace image se traduira par un bon comportement dans l'espace 3D.

La stabilité asymptotique globale (GAS) est également une propriété théorique importante. L'utilisation de fonctions de Lyapunov nous a permis de montrer cette pro-

Pré-requis :

- R : rayon approximatif de la cible
- n : nombre de points de la cible
- \mathbf{b} : information visuelle désirée dans le repère fixe \mathcal{F}_o

À chaque itération, on réalise alors la série de calculs suivants :

- à partir des points de l'image, calcul de \mathbf{q} et $\mathbf{q}^* = {}^o\mathbf{R}_c^\top \mathbf{b}$
- calcul de $|\mathbf{q}|$, $\mathbf{q}_n = \frac{\mathbf{q}}{|\mathbf{q}|}$, $|\mathbf{q}^*|$, $\mathbf{q}_n^* = \frac{\mathbf{q}^*}{|\mathbf{q}^*|}$
- calcul de $F(|\mathbf{q}|)$ et $F(|\mathbf{q}^*|)$ avec

$$F(|\mathbf{q}|) = \frac{R|\mathbf{q}|}{\sqrt{n^2 - |\mathbf{q}|^2}}$$

- calcul de

$$\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|} = \frac{n^2 R}{(n^2 - |\mathbf{q}|^2)^{3/2}}$$

- calcul de $D(|\mathbf{q}|)$ et $D(|\mathbf{q}^*|)$ avec

$$D(|\mathbf{q}|) = n^2 - |\mathbf{q}|^2$$

- calcul de

$$C(|\mathbf{q}^*|) = \left(\frac{D(|\mathbf{q}^*|)^{\frac{1}{4}}}{|\mathbf{q}^*|} \sqrt{\frac{n - \sqrt{D(|\mathbf{q}^*|)}}{n + \sqrt{D(|\mathbf{q}^*|)}}} \right)^{-1}$$

- calcul de $\alpha(|\mathbf{q}|)$ et $\alpha(|\mathbf{q}^*|)$ avec

$$\alpha(|\mathbf{q}|) = C(|\mathbf{q}^*|) \frac{D(|\mathbf{q}|)^{1/4}}{|\mathbf{q}|} \sqrt{\frac{n - \sqrt{D(|\mathbf{q}|)}}{n + \sqrt{D(|\mathbf{q}|)}}}$$

- calcul de

$$\frac{\partial G(|\mathbf{q}|)}{\partial |\mathbf{q}|} = \alpha(|\mathbf{q}|) \sqrt{\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}}$$

- calcul de $G(|\mathbf{q}|)$ et $G(|\mathbf{q}^*|)$ avec

$$G(|\mathbf{q}|) = \alpha(|\mathbf{q}|) \sqrt{|\mathbf{q}| F(|\mathbf{q}|)}$$

- calcul de

$$\mathbf{H}(\mathbf{q}) = \frac{\partial G(|\mathbf{q}|)}{\partial |\mathbf{q}|} \mathbf{q}_n \mathbf{q}_n^\top + \frac{G(|\mathbf{q}|)}{|\mathbf{q}|} (\mathbf{I}_3 - \mathbf{q}_n \mathbf{q}_n^\top)$$

- calcul de \mathbf{g} et \mathbf{g}^* avec

$$\mathbf{g} = G(|\mathbf{q}|) \mathbf{q}_n \quad \text{et} \quad \mathbf{g}^* = G(|\mathbf{q}^*|) \mathbf{q}_n^*$$

- calcul de

$$\delta_{\mathbf{g}} = \mathbf{g} - \mathbf{g}^*$$

On a alors tous les éléments nécessaires au calcul de la loi de commande (k_g est un gain à choisir en fonction du système utilisé) :

$$\mathbf{v} = k_g \frac{\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \delta_{\mathbf{g}}$$

FIG. 2.24 – Implémentation de la loi de commande (2.75).

	Schéma de commande	Loi de commande	GAS	BLES	CT	RAL	P
1.	Moments de la projection perspective (partie 2.3)	$\mathbf{v} = \lambda \mathbf{e}$	×	✓	bon	✓	×
2.	Com. proportionnelle (partie 2.4.2)	$\mathbf{v} = k\boldsymbol{\delta}$	✓	×	mauvais	×	✓
3.	Linéarisation à la position désirée (partie 2.4.3)	$\mathbf{v} = k_* \mathbf{Q}^{*-1} \boldsymbol{\delta}$	×	✓	moyen	×	✓
4.	Com. partitionnée (partie 2.4.4)	$\mathbf{v} = k_{\mathbf{A}} \mathbf{A}(\mathbf{q}_0^*)^\top \boldsymbol{\delta}_{\mathbf{A}}$	✓	✓	moyen	×	✓
5.	Com. stable avec CE (partie 2.4.5.1)	$\mathbf{v} = k_{\mathbf{M}} \mathbf{M} \boldsymbol{\delta}_{\mathbf{f}}$	✓	×	mauvais	×	✓
6.	Approximation de la matrice d'interaction (partie 2.4.5.2)	$\mathbf{v} = k_{\mathbf{q}} \mathbf{M} \boldsymbol{\delta}$	×	✓	moyen	×	✓
7.	Com. proportionnelle avec CE (partie 2.4.5.3)	$\mathbf{v} = k_{\mathbf{f}} \boldsymbol{\delta}_{\mathbf{f}}$	×	✓	bon	✓	✓
8.	Com. stable avec information visuelle modifiée (partie 2.4.6)	$\mathbf{v} = k_{\mathbf{g}} \frac{\mathbf{H}(\mathbf{q})}{\alpha(\mathbf{q})^2} \boldsymbol{\delta}_{\mathbf{g}}$	✓	✓	bon	×	✓

TAB. 2.1 – Propriétés des différentes lois de commande présentées.

Définitions des acronymes et abréviations utilisés :

- Com. : commande,
 CE : changement d'échelle,
 GAS : *global asymptotic stability* (stabilité asymptotique globale),
 BLES : *balanced locally exponentially stable* (stabilité exponentielle locale équilibrée),
 CT : conditionnement de la phase transitoire,
 RAL : relation approximativement linéaire entre l'espace image et l'espace 3D,
 P : passivité.

priété pour plusieurs lois de commande. Enfin, comme nous l'avons déjà vu (section 2.1), être capable de commander les mouvements de translation indépendamment des manœuvres de rotation réalisées par l'engin est particulièrement intéressant, notamment dans le cas du quadrirotor (propriété de passivité, notée P).

2.5.2 Comparaison des trois meilleures lois de commande

La loi de commande reposant sur les moments de la projection perspective de l'objet (1) permet d'obtenir une relation linéaire entre les informations visuelles dans l'espace image et les positions de la caméra dans l'espace 3D, tant que la rotation entre l'image et le plan de la cible n'est pas trop importante. De plus, les résultats de commande cinématique utilisant ces informations visuelles montrent de bons comportements transitoires et une bonne convergence asymptotique, tant dans l'espace image que dans l'espace 3D [Tahri 05]. Dans notre cas, l'inconvénient de cette loi de commande réside dans l'hypothèse de faible rotation, qui empêche de montrer que le système est GAS. De plus, la propriété de passivité n'est pas vérifiée, comme avec les informations visuelles sphériques. Bien qu'elle fonctionne correctement en pratique en simulation dans le cadre de faibles rotations, cette loi de commande n'est pas vraiment adaptée à la dynamique du quadrirotor. Ses performances risquent donc d'être perturbées dans le cas de mouvements de rotations importants.

La loi de commande proportionnelle avec changement d'échelle de l'information visuelle (7) a des propriétés similaires à celle de la de commande 1. En effet, les informations visuelles sont dans les deux cas très proches de la position 3D. La commande, en contraignant la trajectoire des informations visuelles, assure alors également de bonnes trajectoires dans l'espace 3D. De plus, pour les deux lois de commande, le correcteur se limite à un gain proportionnel. Une fois les informations visuelles calculées, ce correcteur est donc très facilement mis en œuvre. L'avantage de la loi de commande 7 sur la précédente réside dans la propriété de passivité : le module du vecteur des informations visuelles utilisées est indépendant des mouvements de rotation de l'engin. Cette loi de commande devrait donc faire preuve d'une plus grande robustesse aux manœuvres agressives de l'engin volant. L'inconvénient des informations visuelles sphériques intervenant ici (en spécifiant la direction du vecteur considéré dans le repère fixe) vient de la nécessité de connaître à chaque itération l'attitude de l'engin, alors qu'avec la loi 1, seules des données issues de l'image sont nécessaires à chaque itération. Enfin, la stabilité asymptotique globale du système bouclé avec la loi de commande 7 n'est pas prouvée. Notons qu'au cours des expérimentations réalisées — en simulation et sur le quadrirotor — aucun cas d'instabilité n'a été rencontré.

La dernière loi de commande 8 repose sur la modification des informations visuelles utilisées dans 7, de manière à assurer la stabilité asymptotique globale du système bouclé. Son seul inconvénient est qu'avec cette modification, on perd la relation linéaire entre l'espace image et l'espace 3D, ce qui peut conduire à des comportements transitoires dégradés dans l'espace 3D.

On va donc tester les trois lois de commande 1, 7 et 8 sur le quadrirotor.

2.6 Résultats expérimentaux sur un prototype d'hélicoptère miniature

Dans cette partie, la plupart des lois de commande présentées précédemment sont expérimentées sur un quadrirotor. Les lois de commande considérées sont :

- la loi de commande classique utilisant les moments d'ordre zéro et un de l'image perspective (1, section 2.3),
- la loi de commande classique utilisant une linéarisation à la position désirée, mais avec des informations visuelles sphériques (3, section 2.4.3),
- la loi de commande partitionnée (4, section 2.4.4),
- le correcteur proportionnel utilisant les moments de l'image sphérique avec le changement d'échelle F (7, section 2.4.5.3),
- la loi de commande utilisant les moments de l'image sphérique avec le changement d'échelle G (8, section 2.4.6).

La tâche à réaliser consiste à stabiliser la quadrirotor à une position déterminée au dessus d'une cible.

2.6.1 Conditions expérimentales

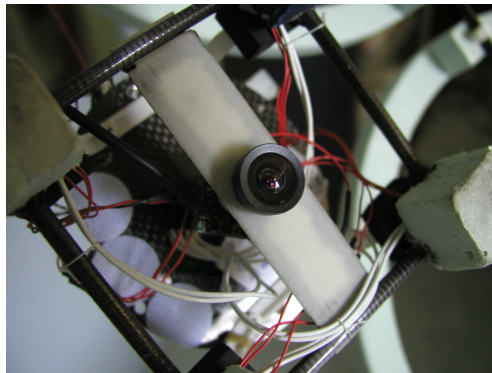
2.6.1.1 Prototype utilisé

Les expérimentations ont été menées sur le quadrirotor développé par l'équipe du CEA/LIST de Fontenay aux Roses (voir figure 2.25). Il est actionné par quatre moteurs électriques, et embarque différentes cartes électroniques : une carte dédiée à la commande des moteurs, une carte pour la centrale inertielle (IMU), et une carte DSP (*Digital Signal Processing*). La première carte s'occupe de réguler les vitesses de rotation des quatre moteurs. La carte centrale inertielle embarque 3 accéléromètres, 3 capteurs de vitesse angulaire, et 2 magnétomètres. Enfin, la troisième carte embarque un DSP. Outre les algorithmes de commande bas-niveau, il contient des algorithmes de filtrage complémentaire, pour estimer l'attitude de l'engin à partir des mesures des accéléromètres et des vitesses de rotation [Guenard 06a]. À partir des données de la centrale inertielle, les consignes à atteindre par chaque moteur sont calculées au sein du DSP de manière à stabiliser l'engin [Guenard 05, Guenard 06a]. Le quadrirotor est équipé d'une batterie Lithium-Polymer, et possède une autonomie en vol de l'ordre de 10 minutes. Il pèse environ 550 grammes. Les déplacements du quadrirotor peuvent être commandés manuellement via un poste de pilotage constitué d'un *joystick* et d'un PC (*Pentium 4*). Les consignes sont transmises au quadrirotor via une liaison sans fil. Les contraintes de taille et de poids ont conduit au choix d'une caméra VTQ-54 (figure 2.26.a), commercialisée par la société Opto Vision de Toulouse. Cette caméra couleur pèse environ 30 g, et mesure 26×22 mm. Elle est équipée d'une focale de 3.7 mm. De plus un micro émetteur HF vidéo permet de transmettre le signal vidéo au poste de pilotage. La cible utilisée est constituée de quatre ronds noirs délimitant un rectangle (30×40 cm) (voir figure 2.26.b).

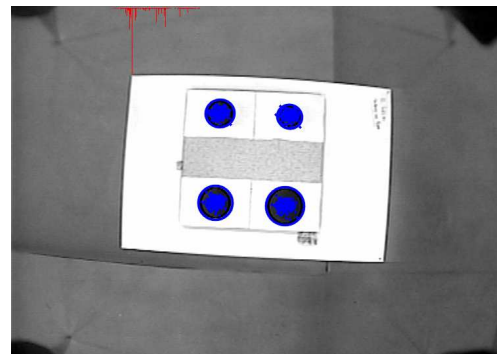
Les algorithmes sont implémentés en C/C++ sous *Windows* (*Visual.NET*) sur le



FIG. 2.25 – Le quadrirotor utilisé pour les expérimentations.



(a)



(b)

FIG. 2.26 – La caméra embarquée sur le quadrirotor (a), et la vue de la cible par la caméra embarquée (b).

poste de pilotage. Les bibliothèques de traitement d'image et d'asservissement visuel OpenCV (*Open Computer Vision Library*) [Ocv] et ViSP-2 [Marchand 05, Visp] sont utilisées pour le suivi des points et pour les fonctions de base de l'asservissement visuel. La boucle d'asservissement visuel est cadencée à 60 ms. La boucle de régulation bas-niveau implémentée sur le DSP, et qui assure la stabilité de l'engin est cadencée à 6 ms.

Les lois de commande d'asservissement visuel proposées étant de type cinématique (consigne sous forme de vitesses de translation), le contrôleur bas-niveau a besoin d'une estimation des vitesses de translations courantes. Celles-ci sont obtenues à partir d'une estimation de la pose de l'engin par rapport à la cible. Cette pose est obtenue grâce à un filtre particulière [Arulampalam 02] qui fusionne les données de la vision et les mesures de l'IMU. En l'absence d'autres moyens de mesure, nous considérons cette estimation de la pose comme la "vérité terrain" pour représenter le comportement de l'engin.

2.6.1.2 Protocole expérimental

Afin que la comparaison des différentes lois de commande proposées ait un sens, il est nécessaire que les positions initiales et désirées de l'engin soient identiques pour toutes les expérimentations. Pour cela, un algorithme classique utilisant l'estimation de la pose permet de stabiliser l'engin à la position initiale voulue. On démarre alors l'algorithme d'asservissement visuel 2D, et on enregistre les positions estimées de l'engin (afin de représenter par la suite le comportement du quadrirotor). L'objectif est alors d'atteindre la position désirée. La position initiale est définie par $X \simeq 0.7$ m, $Y \simeq -0.65$ m, $Z \simeq 2$ m, et la position désirée par $X \simeq 0$ m, $Y \simeq 0$ m, $Z \simeq 1.4$ m, ce qui correspond à placer le drone au dessus et au centre de la cible. Les vitesses de consigne sont saturées à 20 cm/s, afin que l'engin reste dans un domaine de vol quasi-stationnaire [Guenard 06b]. Notons qu'à la position désirée, on a $\mathbf{b} \simeq (0, 0, 3.96)$ et

$$\mathbf{Q}^* = \text{diag}(2.35, 2.36, 0.057). \quad (2.77)$$

Dans les parties suivantes, les lois de commande les plus intéressantes pour commander les translations du quadrirotor sont considérées successivement. Pour chaque expérimentation, on a reporté la position de la caméra par rapport à la cible et les vitesses de consigne issues de la loi de commande d'asservissement visuel. On représente également l'évolution de l'erreur sur les informations visuelles choisies, ainsi que le comportement des quatre points de la cible dans l'image.

2.6.2 Résultats avec les moments de l'image perspective

La première expérience réalisée repose sur l'utilisation des moments de l'image perspective, dans le cadre de l'hypothèse de faibles rotations entre la caméra et la cible (loi de commande 1). La loi de commande utilisée est donc très simple : $\mathbf{v} = \lambda(\mathbf{s} - \mathbf{s}^*)$, $\lambda > 0$ (voir section 2.3). Les résultats sont présentés sur la figure 2.27 : après environ 8 secondes, le quadrirotor atteint la position désirée, puis reste stabilisé à cette position

(voir figure 2.27.a). On peut voir également que les vitesses (sorties de la loi de commande, figure 2.27.b) et les erreurs sur les informations visuelles (figure 2.27.c) évoluent de la même façon (mis à part le fait que l'on sature les vitesses) : ceci est normal puisqu'elles ne sont liées que par un simple gain proportionnel. Elles convergent bien, mais on peut cependant remarquer que la composante selon l'axe Z ne converge pas exactement vers 0. Ce léger problème est dû à une faible erreur statique au niveau de la boucle de commande bas-niveau, qui ne satisfait pas exactement la consigne fournie par l'asservissement visuel. Cela n'enlève rien au fait que le quadrirotor est capable de se stabiliser à la position désirée grâce à la loi de commande cinématique proposée. Enfin, sur la figure 2.27.d, on a représenté les trajectoires des quatre points de la cible dans l'image. Globalement, les résultats obtenus sur le quadrirotor sont finalement très bons. De plus, la loi de commande est un simple correcteur proportionnel, et seules des informations image sont utilisées : l'avantage pratique de cette loi de commande réside dans sa facilité d'implémentation. Notons qu'avec l'engin que nous avons utilisé, les rotations sont faibles, ce qui permet de rester dans le cadre de l'hypothèse considérée lors de la conception de la loi de commande. Avec des engins dont la partie bas-niveau réagit différemment, les résultats seraient sûrement différents.

2.6.3 Résultats avec le centre de gravité sphérique

Parmi les lois de commandes proposées reposant sur les informations visuelles sphériques, seules certaines d'entre elles ont été expérimentées sur le drone. En effet celles ayant présenté un trop mauvais comportement transitoire en simulation (2 et 5) n'ont pas été implémentées sur le drone pour des raisons pratiques évidentes. La loi de commande 6, présentant les mêmes propriétés que 3 (voir table 2.1), n'a pas non plus été expérimentée : il n'y a aucune raison pour qu'elle donne de meilleurs résultats que la loi de commande 3 sur le drone puisque en simulation elle n'a pas donné satisfaction.

On a donc expérimenté sur le quadrirotor les lois de commande suivantes, reposant sur des informations visuelles sphériques :

- les deux lois de commandes "asymptotiques" avec l'information visuelle sphérique initiale : 3 et 4,
- les lois de commande avec changement d'échelle de l'information visuelle sphérique, dont on espère de bons résultats pratiques : 7, 8.

2.6.3.1 Linéarisation à la position désirée

Cette loi de commande repose sur l'approximation classique consistant à considérer la matrice d'interaction calculée à la position désirée au lieu de l'estimer à chaque itération : $\mathbf{v} = k_* \mathbf{Q}^{*-1} \boldsymbol{\delta}$, $k_* > 0$ (loi de commande 3, voir section 2.4.3). La figure 2.28 présente les résultats obtenus. Déjà, on voit que la composante selon Z de la position évolue initialement dans le sens opposé à celui que l'on souhaite (voir figure 2.28.a). Le quadrirotor se trouve initialement plus haut que la position désirée, et, au lieu d'engager immédiatement un mouvement vers le bas, il s'élève pendant la phase transitoire. L'altitude désirée est ensuite atteinte, mais cela s'est fait au prix d'un mouvement non

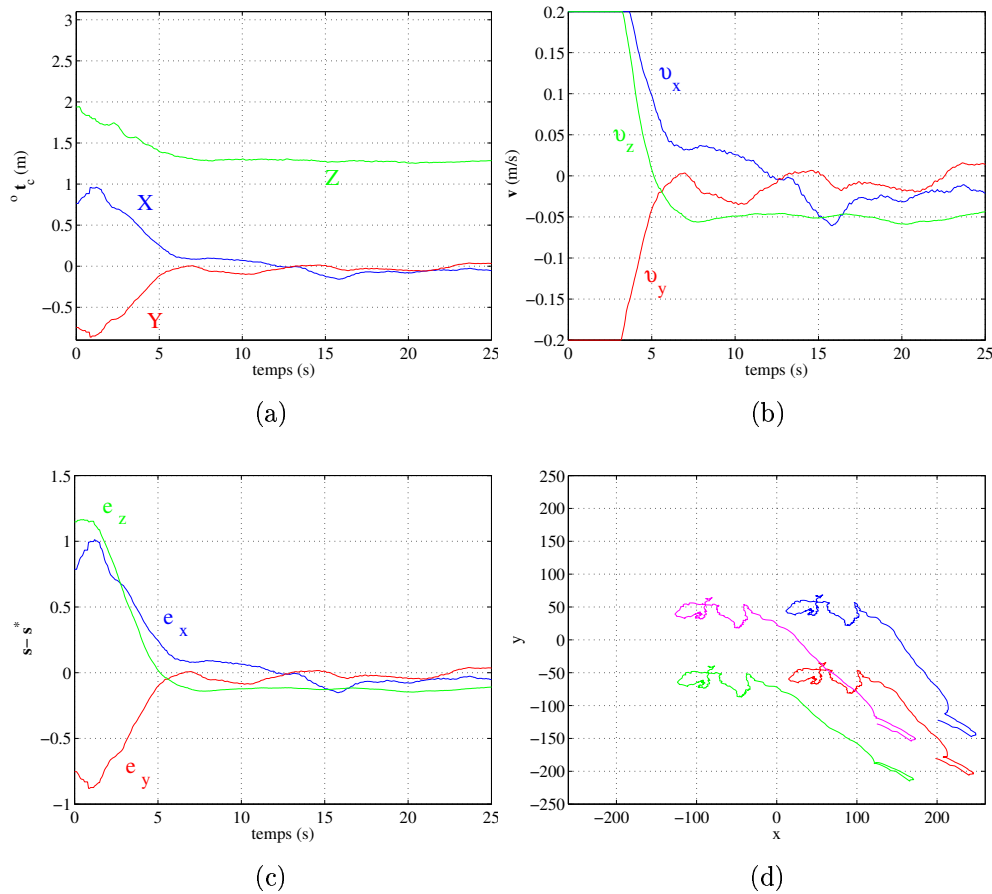


FIG. 2.27 - $v = 0.4(s - s^*)$: évolution au cours du temps (en secondes) de la position de la caméra dans le repère fixe (en mètres) (a), et de la consigne en vitesses v issue de la loi de commande d'asservissement visuel (en mètres par secondes) (b). L'évolution de l'erreur sur les informations visuelles est représentée en (c), et les trajectoires des quatre points de la cible dans le plan image sont tracées en (d).

souhaité qui aurait pu mettre en péril l'intégrité du système si celui-ci avait été amené à atteindre les limites de son espace d'évolution. On peut voir sur la figure 2.28.b que les vitesses de consigne calculées par la loi de commande d'asservissement visuel sont fortement perturbées, et ce même une fois que le drone a atteint la position désirée ($t > 25$ s). Notons que l'on ne retrouve pas ces perturbations au niveau des positions de l'engin (voir figure 2.28.a) : la boucle bas-niveau, qui reçoit les vitesses comme consignes, joue le rôle d'un filtre passe-bas, et l'engin ne réalise donc pas exactement les consignes de vitesse demandées. Les figures 2.28.c et 2.28.d montrent le comportement dans l'espace image : les composantes de l'erreur sur l'information visuelle convergent vers 0 (voir figure 2.28.c), et l'image de la cible sur le plan image va se placer au centre (voir figure 2.28.d). Comme on s'y attendait, cette loi de commande permet d'atteindre la position désirée, mais d'une manière peu précise (perturbations), au prix d'un comportement transitoire dangereux.

2.6.3.2 Commande partitionnée

La commande partitionnée (loi de commande 4) est du même type que la précédente dans le sens où toutes deux tentent de résoudre le problème de convergence lente selon l'axe Z rencontré au départ (section 2.4.2) en s'intéressant à la position désirée de l'engin. Ici, le partitionnement consiste à isoler cette composante, et la loi de commande proposée assure que le système est GAS (voir section 2.4.4). Dans l'expérience présentée ici, on a utilisé un gain $\lambda = 6.44$, ce qui donne $\mathbf{A}(\mathbf{q}_0^*)\mathbf{Q}^*\mathbf{A}(\mathbf{q}_0^*)^\top \cong 2.35\mathbf{I}_3$ (voir section 2.4.4). Comme on peut le voir sur les résultats pratiques obtenus (voir figure 2.29), cette propriété théorique n'est ici d'aucune aide pour améliorer les performances du système. Dans l'espace image, les composantes de l'erreur sur l'information visuelle convergent vers 0 à la même vitesse (voir figure 2.29.c) et l'image de la cible dans le plan image converge vers la position désirée (voir figure 2.29.d). Cependant, comme précédemment, le drone effectue un mouvement transitoire inapproprié (voir figure 2.29.a), et les vitesses sont relativement perturbées (voir figure 2.29.b).

2.6.3.3 Commande proportionnelle avec changement d'échelle

La loi de commande 7, de type proportionnelle en considérant un changement d'échelle sur les informations visuelles sphériques, $\mathbf{v} = k_f \delta_f$, $k_f > 0$, a donné de bons résultats en simulation (voir section 2.4.5.3). Le comportement du quadrirotor est également très satisfaisant (voir figure 2.30). L'engin est capable d'atteindre sa position désirée en 10 secondes environ, ce qui est beaucoup plus rapide qu'avec les deux commandes précédentes (temps de convergence de l'ordre de 30 s), et du même ordre de grandeur qu'avec les moments de l'image perspective (voir section 2.6.2, figure 2.27.a). De plus, on n'observe pas de phénomène transitoire gênant comme dans les deux cas précédents. Les vitesses et erreurs sur les informations visuelles sont proportionnelles et convergent vers 0 (voir figures 2.30.b et 2.30.c). On peut voir sur la figure 2.30.d le déplacement des points de la cible dans le plan image : ils rejoignent la position désirée, puis restent dans son voisinage. Cette loi de commande donne de très bons résultats

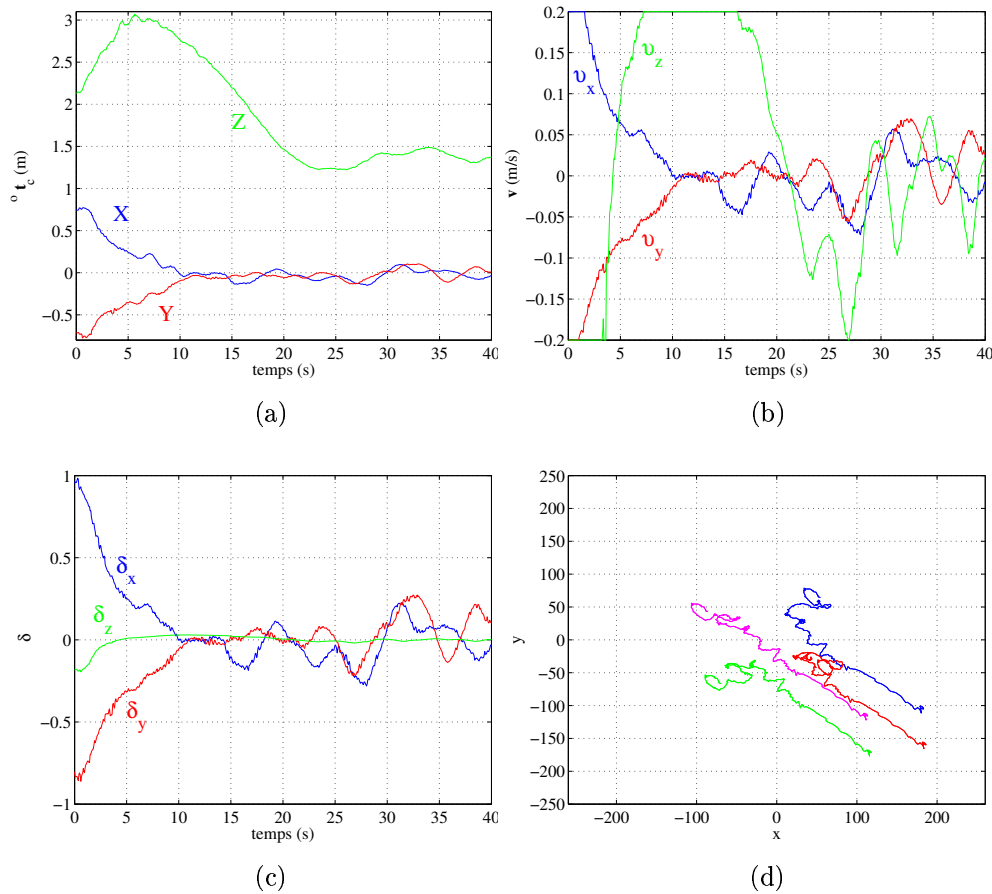


FIG. 2.28 – $\boldsymbol{v} = 0.6\mathbf{Q}^{*-1}\boldsymbol{\delta}$: évolution au cours du temps (en secondes) de la position de la caméra dans le repère fixe (en mètres) (a), et de la consigne en vitesses \boldsymbol{v} issue de la loi de commande d'asservissement visuel (en mètres par secondes) (b). L'évolution de l'erreur sur les informations visuelles est représentée en (c), et les trajectoires des quatre points de la cible dans le plan image sont tracées en (d).

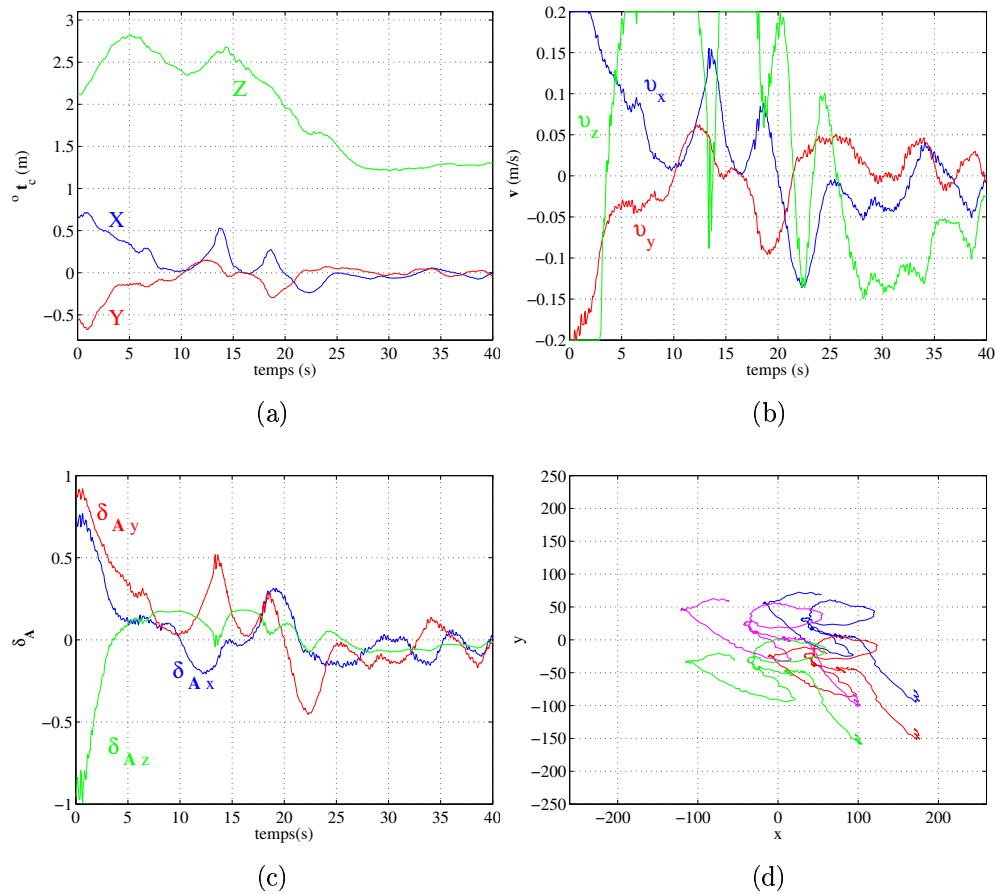


FIG. 2.29 – $\mathbf{v} = 0.3\mathbf{A}(\mathbf{q}_0^*)^\top \delta_{\mathbf{A}}$: évolution au cours du temps (en secondes) de la position de la caméra dans le repère fixe (en mètres) (a), et de la consigne en vitesses \mathbf{v} issue de la loi de commande d'asservissement visuel (en mètres par secondes) (b). L'évolution de l'erreur sur les informations visuelles est représentée en (c), et les trajectoires des quatre points de la cible dans le plan image sont tracées en (d).

pratiques sur le quadrirotor, très similaires à ceux obtenus avec les moments de l'image perspective (voir section 2.6.2).

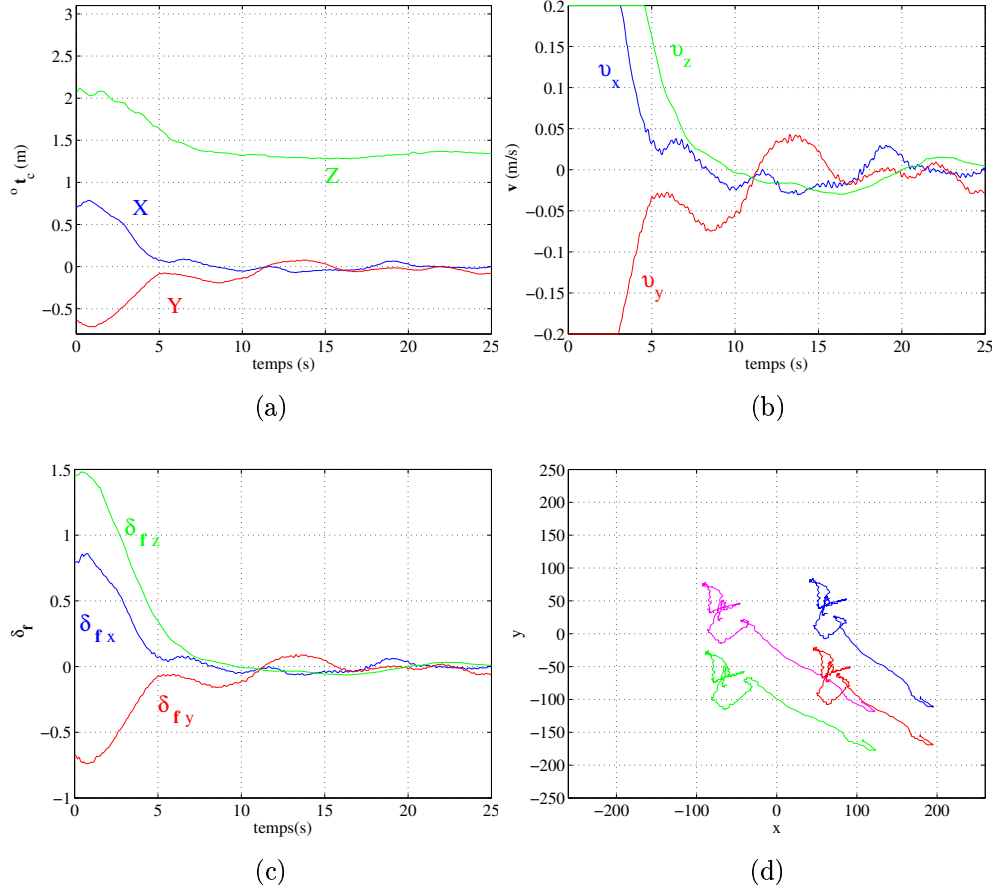


FIG. 2.30 – $v = 0.47\delta_f$: évolution au cours du temps (en secondes) de la position de la caméra dans le repère fixe (en mètres) (a), et de la consigne en vitesses v issue de la loi de commande d'asservissement visuel (en mètres par secondes) (b). L'évolution de l'erreur sur les informations visuelles est représentée en (c), et les trajectoires des quatre points de la cible dans le plan image sont tracées en (d).

2.6.3.4 Commande stable avec nouveau changement d'échelle

La dernière loi de commande proposée est de la forme $v = k_g \frac{\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \delta_g$, $k_g > 0$, avec le nouveau changement d'échelle de l'information visuelle $\mathbf{g} = G(|\mathbf{q}|)\mathbf{q}_n$ (loi de commande δ , voir section 2.4.6). Les résultats obtenus sur le quadrirotor sont très satisfaisants (voir figure 2.31). Comme précédemment, l'évolution transitoire et asymptotique des différentes grandeurs observées montre que le quadrirotor se comporte de

manière tout à fait satisfaisante. Les vitesses de consigne calculées par la loi de commande (voir figure 2.31.b) permettent au drone de se stabiliser à la position désirée (voir figure 2.31.a), et ce sans mouvement transitoire indésirable. Les erreurs sur les informations visuelles convergent correctement vers 0 (voir figure 2.31.c), et les points de la cible dans l'espace image rejoignent leur position désirée (voir figure 2.31.d). Ces résultats sont conformes à nos attentes, et même meilleurs. En effet, les informations visuelles considérées dans cette loi de commande ne sont pas linéairement liées aux positions dans l'espace 3D (voir section 2.4.6, et figure 2.22). Ainsi, bien que le système bouclé soit théoriquement GAS, rien ne garantissait que le comportement transitoire du système serait convenable. En pratique, les expérimentations réalisées avec cette loi de commande n'ont cependant pas mis en évidence un tel cas de mauvais comportement transitoire.

2.6.4 Analyse de la sensibilité au bruit

À première vue, les résultats obtenus avec les lois de commande donnant satisfaction (1, 7, et 8) sont très similaires (voir figures 2.27, 2.30, et 2.31). Cependant, dans les lois de commande utilisant les informations visuelles sphériques (7 et 8), il est nécessaire d'estimer à chaque itération la matrice de rotation ${}^o\mathbf{R}_c$ entre le repère de la caméra et le repère lié à la cible. On pourrait penser que cela introduit du bruit supplémentaire dans la loi de commande. Afin d'évaluer le niveau de bruit présent dans ces trois lois de commande, nous avons calculé l'écart-type des vitesses données par les lois de commande, pendant la période où le drone est stabilisé à la position désirée (entre 10 et 25 secondes).

Les positions de l'engin dans l'espace 3D sont commandées via la commande bas-niveau, à partir des vitesses de translation issues des lois de commande d'asservissement visuel. On peut voir ces vitesses comme étant une estimation de l'erreur de position de l'engin dans l'espace 3D, multipliée par un gain. Notons que les gains λ , $k_{\mathbf{f}}$, et $k_{\mathbf{g}}$, utilisés dans les différentes lois de commande considérées, ne sont pas les mêmes. Pour comparer les lois de commande en terme de bruit sur les positions de l'engin, on a donc considéré \mathbf{v}/k , où $k = \{\lambda, k_{\mathbf{f}}, k_{\mathbf{g}}\}$.

On calcule $\sigma_{\mathbf{v}} = \sqrt{\sigma_{v_x}^2 + \sigma_{v_y}^2 + \sigma_{v_z}^2}$ ainsi que $\sigma_{v_K} = \sqrt{\sum_i (v_{K_i} - \bar{v}_K)^2}$, pour $K \in \{X, Y, Z\}$ et où \bar{v}_K représente la moyenne de v_K entre les instants $t = 10$ et $t = 25$ secondes. Comme le montre le tableau 2.2, le niveau de bruit mesuré à la sortie de toutes les lois de commande est très similaire. Les trois lois de commande 1, 7, et 8 ont donc un comportement très proche par rapport au bruit.

2.7 Conclusion

Dans ce chapitre, nous avons étudié une série de lois de commande cinématiques destinées à un quadrirotor. L'objectif était de stabiliser l'engin au-dessus d'une cible visuelle, en commandant ses mouvements de translation. Pour cela, nous nous sommes intéressés en particulier au choix des informations visuelles et des lois de commande,

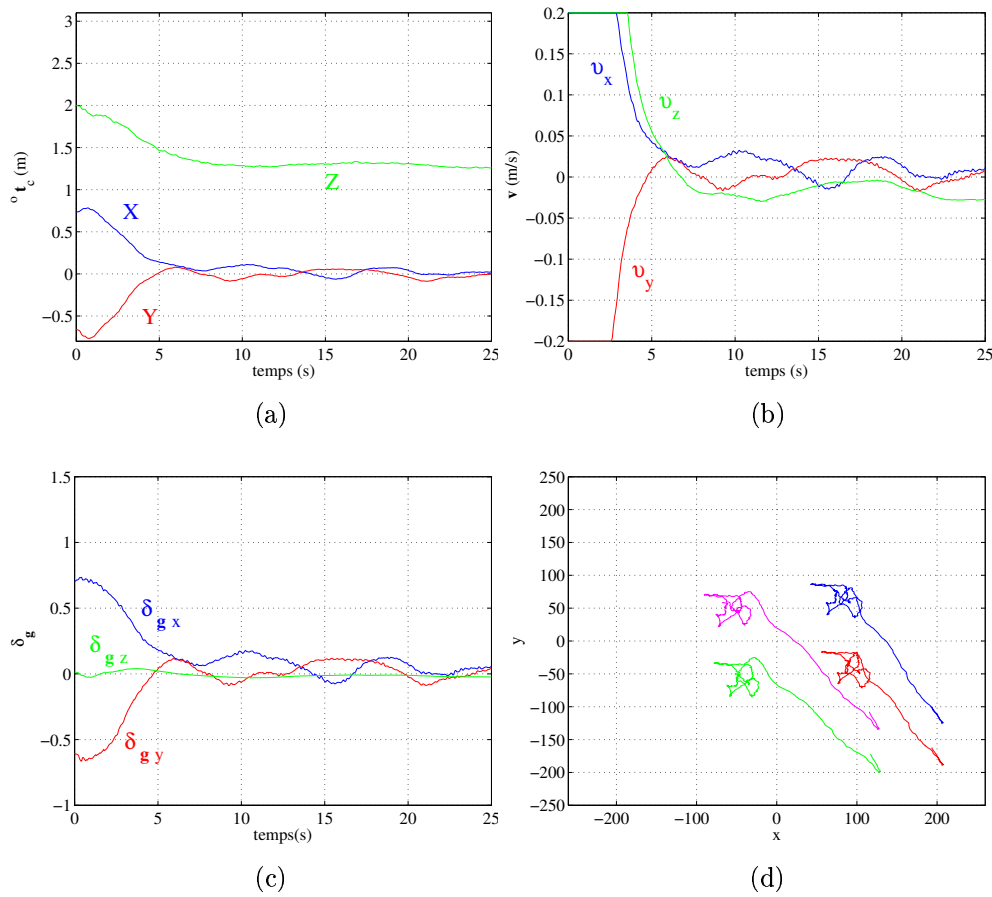


FIG. 2.31 – $\mathbf{v} = 0.3 \frac{\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \boldsymbol{\delta}_g$: évolution au cours du temps (en secondes) de la position de la caméra dans le repère fixe (en mètres) (a), et de la consigne en vitesses \mathbf{v} issue de la loi de commande d'asservissement visuel (en mètres par secondes) (b). L'évolution de l'erreur sur les informations visuelles est représentée en (c), et les trajectoires des quatre points de la cible dans le plan image sont tracées en (d).

	Loi de commande	σ_v	σ_{v_x}	σ_{v_y}	σ_{v_z}
1.	$v = e$	0.95	0.79	0.51	0.15
7.	$v = \delta_f$	1.02	0.51	0.74	0.49
8.	$v = \frac{H}{\alpha^2} \delta_g$	1.01	0.65	0.61	0.47

TAB. 2.2 – Écart-type des vitesses de consigne obtenues avec chacune des trois meilleures lois de commande.

afin que les mouvements de rotation de l'engin ne viennent pas perturber la convergence de la partie translation.

Dans un premier temps, nous avons vu comment l'utilisation des moments d'ordre zéro et un de l'image perspective permet de découpler les translations des rotations, sous l'hypothèse que le plan image de la caméra reste parallèle au plan de la cible. Afin de ne pas se limiter à cette hypothèse et de tenir compte d'éventuelles manœuvres brutales de l'engin, nous nous sommes ensuite intéressés aux informations visuelles sphériques. Nous avons alors proposé une série de lois de commande reposant sur les moments d'ordre un de l'image projetée sur une sphère. En effet, ces informations visuelles permettent de concevoir des schémas de commande tels que la convergence de la partie translation est assurée indépendamment des mouvements de rotation de l'engin. Les propriétés de chacune des lois de commande ont été déterminées et illustrées par des simulations. La présentation de cette série de loi de commandes suit une progression logique, à partir de la loi de commande la plus simple, en montrant à chaque étape les limitations de la loi de commande considérée, motivant ainsi la recherche d'une meilleure solution. Nous nous sommes en particulier intéressés au comportement transitoire de l'engin, pour éviter des déplacements inutiles. La relation entre les informations visuelles et les positions de l'engin dans l'espace 3D est ici essentielle pour qu'un bon comportement dans l'espace image corresponde à un bon comportement dans l'espace 3D. La convergence asymptotique des trois composantes à la même vitesse est également une propriété très utile en pratique : cela évite de devoir "attendre" qu'une composante ne converge alors que les autres ont déjà atteint leur valeurs désirées. Enfin, la propriété de stabilité asymptotique globale constitue une garantie théorique non négligeable pour assurer la convergence du système.

L'étude comparée de toutes les lois de commande présentées a ensuite permis de choisir lesquelles implémenter sur le prototype de quadrirotor. Comme l'analyse théorique et les résultats de simulation nous permettaient de l'espérer, trois lois de commande ont montré de bonnes performances sur le quadrirotor. La première repose sur les moments d'ordre zéro et un de l'image perspective, et fait l'hypothèse que les mouvements de rotation du quadrirotor sont faibles. Son principal atout réside dans sa simplicité. Les deux autres reposent sur les moments du premier ordre de l'image sphérique qui sont utilisés pour concevoir des informations visuelles dotées de la propriété de passivité.

Elles permettent ainsi de s'affranchir de l'hypothèse de faibles mouvements de rotation. Cependant, aucune ne satisfait à toutes les propriétés souhaitées : l'une assure un bon comportement transitoire du fait d'une relation quasi linéaire entre espace image et espace 3D, alors que la propriété de stabilité asymptotique globale est obtenue avec l'autre.

Finalement, reste à mettre au point une loi de commande assurant l'ensemble des propriétés souhaitées (ou à prouver qu'elle n'existe pas). Concernant les trois lois de commande présentées qui ont donné de bons résultats expérimentaux sur un quadrirotor, aucune n'est vraiment meilleure que les autres : selon les propriétés que l'on souhaite assurer, il convient en pratique de choisir l'une ou l'autre.

Chapitre 3

Atterrissage d'un avion par asservissement visuel

Dans cette partie, nous nous intéressons à la commande par asservissement visuel pour l'atterrissage d'avions. Ce travail a été réalisé en collaboration avec Dassault Aviation, en particulier pour les aspects de modélisation et de simulation de l'avion. Après avoir défini l'objectif de notre travail, nous présentons le modèle de l'avion utilisé. Ensuite, nous proposons un choix d'informations visuelles adaptées aux spécificités de l'avion, et nous intégrons ces informations visuelles au sein d'une loi de commande. Les schémas de commande par asservissement visuel mis au point sont validés en simulation.

3.1 Objectif et problématique

Dans le cadre des avions sans pilote, on cherche, autant que possible, à automatiser les manœuvres à réaliser. En particulier, la manœuvre d'atterrissage constitue la tâche la plus difficile pour un opérateur [Pallett 93], et est cause d'une grande part des accidents. Des systèmes d'atterrissage automatiques existent : il s'agit des ILS (*Instrument Landing System*). Un tel système permet à un avion d'atterrir même lorsque les conditions de visibilité sont très mauvaises. Il consiste en un système de radio-navigation, et nécessite que l'aéroport sur lequel on souhaite atterrir soit équipé d'antennes émettrices, permettant à l'avion recevant les informations de se positionner par rapport à la piste. L'utilisation de caméras pourrait permettre d'automatiser la manœuvre d'atterrissage sur tous les aéroports, y compris ceux qui ne sont pas équipés des systèmes d'ILS.

De nombreux ouvrages techniques traitent de la modélisation, la commande, et l'atterrissage des avions [Kermode 01, Bonnet 01, McLean 90, Stevens 92, Etkin 95, Pallett 93, Nelson 98, Tischler 96]. La plupart sont en anglais : pour la traduction des termes techniques anglais-français, on pourra se référer au lexique proposé dans [Kermode 01], ou au lexique en ligne [Lexdas] qui a l'avantage de proposer les termes dans leur contexte. Notons également le site de la NASA [Tutnas] qui propose des tutoriels "interactifs" concernant le vol des avions, et le site de l'université de Carnegie-Mellon, qui traite l'exemple de la commande du tangage d'un avion dans le cadre d'un

tutoriel dédié à la conception de lois de commande avec le logiciel *Matlab* [Tutcar].

Dans cette partie, nous présentons les notations utilisées en aéronautique et dans notre travail, et les principes de fonctionnement et de commande des avions, utiles pour notre étude. Nous décrivons en particulier le comportement de l'avion pendant les différentes phases de l'atterrissage.

3.1.1 Définitions et notations

Position de l'avion. Au cours du vol, la position de l'avion se déplaçant autour du globe terrestre évolue. Cette position, définie par la position du centre de gravité de l'avion, peut être exprimée sous diverses formes, dans différents repères.

La position de l'avion peut être définie sous la forme des coordonnées géographiques : deux angles (longitude θ , latitude ϕ) et une distance (altitude r) (voir figure 3.1). La longitude, notée θ ($\theta \in [-180, 180[$ degrés), représente le positionnement Ouest-Est par rapport au méridien de Greenwich, la latitude, notée ϕ ($\phi \in [-90, 90[$ degrés), représente le positionnement Sud-Nord par rapport à l'équateur, et la distance au centre de la Terre, notée $r = R + H_g + H$, comprend le rayon de la Terre (R , supposé constant), la hauteur du sol (H_g), et l'altitude de l'avion par rapport au sol (H).

On peut également représenter la position de l'avion sous la forme de ses coordonnées (X, Y, Z) dans le repère géocentrique $(0, x_0, y_0, z_0)$, dont l'origine est au centre de la Terre, l'axe z_0 étant orienté vers le Nord, et l'axe x_0 coupant l'équateur et le méridien de Greenwich (voir figure 3.1). On peut obtenir les coordonnées (X, Y, Z) directement à partir des coordonnées de longitude, latitude, altitude, de la manière suivante :

$$X = r \cos \phi \cos \theta \quad (3.1)$$

$$Y = r \cos \phi \sin \theta \quad (3.2)$$

$$Z = r \sin \phi. \quad (3.3)$$

Ces représentations sont bien adaptées lorsque l'on considère des vols de longue distance, où il est nécessaire de tenir compte du fait que la Terre est sphérique. Dans le cas d'une manœuvre de courte durée, sur des distances faibles, on peut considérer que la Terre est localement plate. On utilise alors un repère dont le plan x_t, y_t est horizontal et tangent à la surface de la Terre, tel que l'axe x_t soit orienté vers le Nord, l'axe y_t vers l'Est et l'axe z_t vers le centre de la Terre. L'origine du repère est définie par l'intersection entre l'axe vertical z_t et la surface de la Terre [Stevens 92] (voir figure 3.2). En particulier, dans le cas de la manœuvre d'atterrissage, on pourra orienter le repère selon l'axe de la piste (voir figure 3.3).

On notera ${}^f \mathbf{t}_a$ le vecteur de position de l'avion dans le repère fixe \mathcal{F}_f .

Orientation de l'avion. En ce qui concerne l'orientation de l'avion, elle est définie par la matrice de rotation ${}^f \mathbf{R}_a$, qui contient les coordonnées du repère \mathcal{F}_a lié à l'avion, exprimées dans le repère fixe \mathcal{F}_f . Cette matrice de rotation peut également s'exprimer sous la forme de trois angles. On choisit ici la représentation d'Euler (z, y, x) , telle

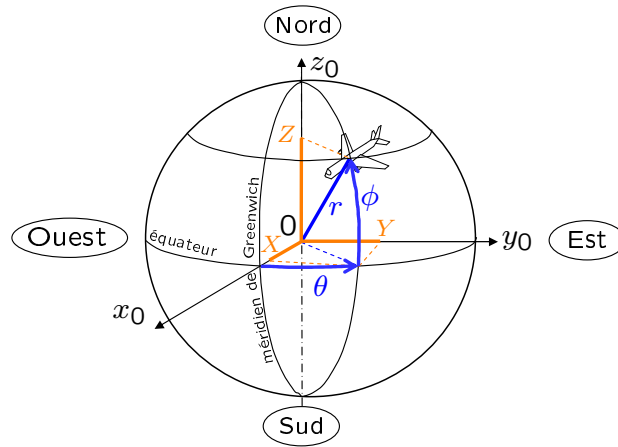


FIG. 3.1 – Coordonnées géographiques θ , ϕ , r et coordonnées (X, Y, Z) dans le repère géocentrique $(0, x_0, y_0, z_0)$.

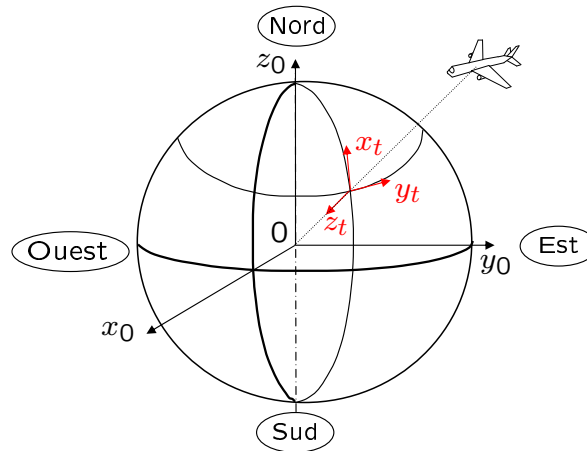


FIG. 3.2 – Repère tangent à la surface de la Terre.

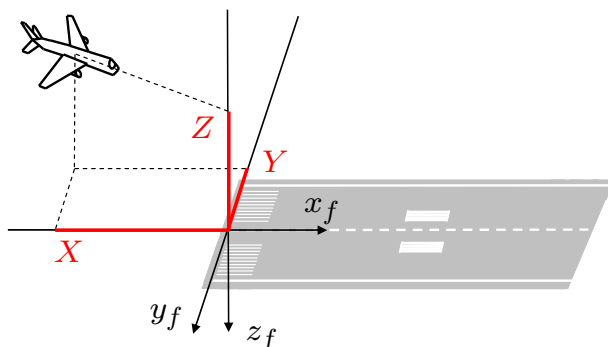


FIG. 3.3 – Coordonnées (X, Y, Z) dans le repère fixe lié à la piste d'atterrissage.

que définie dans l'annexe C. L'intérêt d'un tel choix pour définir l'orientation de l'avion réside dans le fait que les angles de roulis (ϕ), tangage (θ) et lacet (ψ) ainsi obtenus sont ceux que l'on peut mesurer en pratique à l'aide d'un gyroscope embarqué [McLean 90].

Pour représenter l'orientation de l'avion, on introduit souvent les termes suivants [Bonnet 01] :

- l'angle de roulis ϕ est également appelé angle de gîte (ou parfois assiette latérale), il s'agit de l'angle entre plan de symétrie de l'avion et la verticale terrestre (voir figure 3.4.a),
- l'angle de tangage θ est également appelé assiette longitudinale (ou assiette), il s'agit de l'angle entre la référence horizontale du fuselage et l'horizontale terrestre (voir figure 3.4.b),
- l'angle de lacet ψ est également appelé cap (ou azimuth), il s'agit de l'angle entre la projection horizontale de la référence horizontale du fuselage et le Nord (voir figure 3.4.c).

L'interprétation de la représentation d'Euler utilisée est donnée dans l'annexe C. Dans le cas de l'avion, on peut également interpréter ces trois angles en effectuant à partir du repère fixe, les mouvements successifs suivants [Stevens 92] (voir figures 3.4 et 3.5) :

1. rotation de l'avion autour de son axe vertical z_a , de manière à ce que son "nez" dévie vers la droite (angle de lacet ψ positif),
2. rotation de l'avion autour du nouvel axe latéral y_a , de manière à ce que son "nez" monte (angle de tangage θ positif),
3. rotation de l'avion autour du nouvel axe longitudinal x_a , de manière à ce que l'aile droite "descende" (angle de roulis ϕ positif).

Angles aérodynamiques. Lorsque l'avion est en mouvement, sa trajectoire est définie par l'évolution de son vecteur vitesse. Celui-ci n'est pas forcément orienté selon

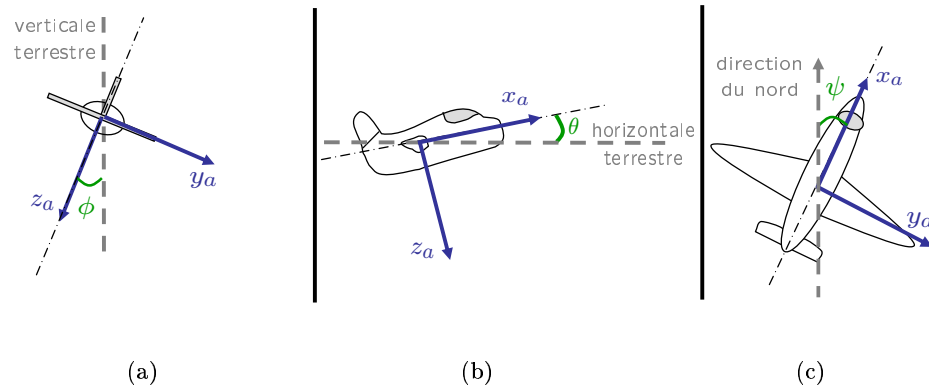


FIG. 3.4 – Angles de roulis (a), tangage (b), lacet (c), tous trois positifs ici.

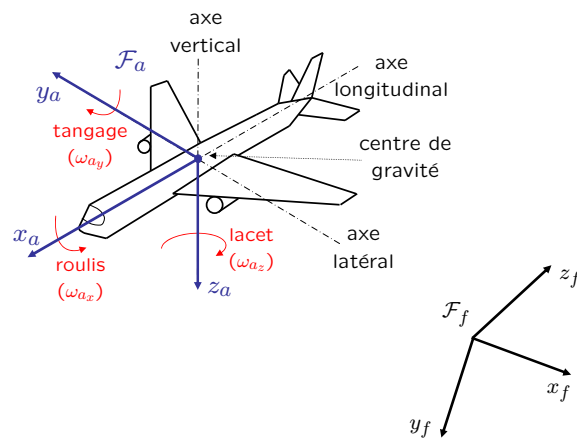


FIG. 3.5 – Repère lié à l'avion \mathcal{F}_a , et repère fixe \mathcal{F}_f .

l'axe longitudinal de l'avion. Son orientation par rapport à l'avion définit les angles aérodynamiques. Dans les ouvrages traitant de la modélisation du mouvement des avions, les vitesses de l'avion sont usuellement notées (u, v, w) pour les vitesses de translation et (p, q, r) pour les vitesses de rotation. Pour utiliser les mêmes notations dans tout le manuscrit, nous les noterons ici $\mathbf{v}_a = (v_{ax}, v_{ay}, v_{az})$ pour les vitesses de translation, et $\boldsymbol{\omega}_a = (\omega_{ax}, \omega_{ay}, \omega_{az})$ pour les vitesses de rotation (voir figure 3.5). Les vitesses de rotation sont liées aux dérivées des angles de roulis-tangage-lacet, comme indiqué dans l'annexe C.

Considérons que l'avion se déplace par rapport à l'air environnant selon le vecteur vitesse \mathbf{v}_a , exprimé dans le repère lié à l'avion \mathcal{F}_a . L'angle entre le vecteur vitesse (qui détermine la trajectoire de l'avion) et l'axe longitudinal de l'avion, noté α , est alors appelé incidence (ou angle d'attaque, voir figure 3.6.a). L'angle entre le vecteur vitesse et le plan de symétrie de l'avion, noté β , est quant à lui appelé dérapage (voir figure 3.6.b). Ces deux angles, incidence et dérapage constituent les angles aérodynamiques de l'avion.

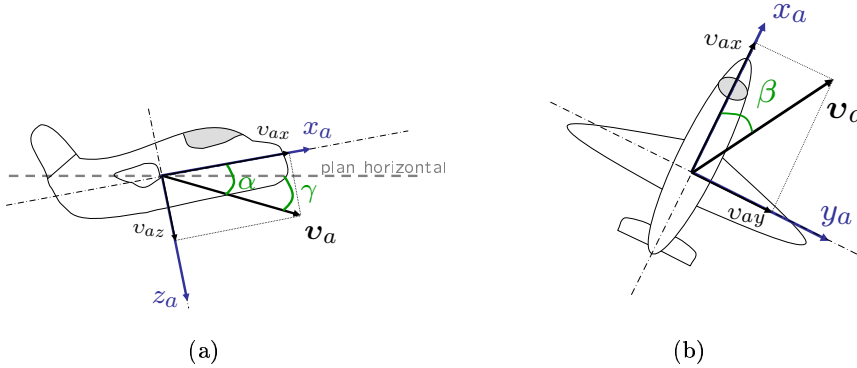


FIG. 3.6 – Incidence α (>0 ici) et pente γ (<0 ici) (a) et dérapage β (>0 ici) (b).

Notons (v_{ax}, v_{ay}, v_{az}) les coordonnées du vecteur vitesse \mathbf{v}_a dans le repère lié à l'avion, et $v_a = \sqrt{v_{ax}^2 + v_{ay}^2 + v_{az}^2}$ son module. Les angles α et β sont définis comme suit :

$$\tan \alpha = \frac{v_{az}}{v_{ax}} \quad (3.4)$$

$$\sin \beta = \frac{v_{ay}}{v_a}. \quad (3.5)$$

Lorsque l'incidence et le dérapage sont faibles (en pratique plus petits que 15 degrés [Nelson 98, McLean 90]), on peut approcher les relations précédentes par :

$$\alpha \simeq \tan \alpha = \frac{v_{az}}{v_a} \quad (3.6)$$

$$\beta \simeq \sin \beta = \frac{v_{ay}}{v_a}. \quad (3.7)$$

On définit également la pente γ : angle entre le plan horizontal et le vecteur vitesse (voir figure 3.6.a). On peut écrire la relation suivante entre tangage θ , incidence α , roulis ϕ et pente γ : $\theta = \alpha \cos \phi + \gamma$, d'où on déduit

$$\theta \simeq \alpha + \gamma \quad \text{lorsque} \quad \phi \simeq 0. \quad (3.8)$$

Repères. Le repère lié à l'avion que nous avons considéré jusque-là est orienté selon les axes du fuselage et le plan de symétrie de l'avion : il est nommé *body axis*, et on l'a noté \mathcal{F}_a . Notons que pour écrire les équations du mouvement de l'avion, d'autres repères peuvent être utilisés.

Par exemple, écrire les équations dans le repère défini par les axes principaux d'inertie de l'avion (repère appelé *principal axis*) peut permettre de simplifier les équations. En effet, un tel repère est défini de manière à ce que les produits d'inertie (termes non diagonaux de la matrice d'inertie) sont nuls dans ce repère [McLean 90, Etkin 95].

Le repère appelé *stability axis* est obtenu à partir du repère \mathcal{F}_a en le faisant tourner de l'angle d'incidence α (voir figure 3.7) [Stevens 92]. Il consiste à orienter le repère selon le vecteur vitesse dans le cas d'un vol symétrique stabilisé (on a alors $\beta = 0$). Dans ce cas, les coordonnées du vecteur vitesse sont alors de la forme $(v_a, 0, 0)$ [McLean 90, Stevens 92, Etkin 95].

Enfin, en faisant à nouveau tourner le repère précédent selon l'angle de dérapage β , on obtient le repère cinématique \mathcal{F}_v , appelé aussi *wind axis* (voir figure 3.7). Ce repère est orienté selon le vecteur vitesse \mathbf{v}_a de l'avion [McLean 90, Stevens 92].

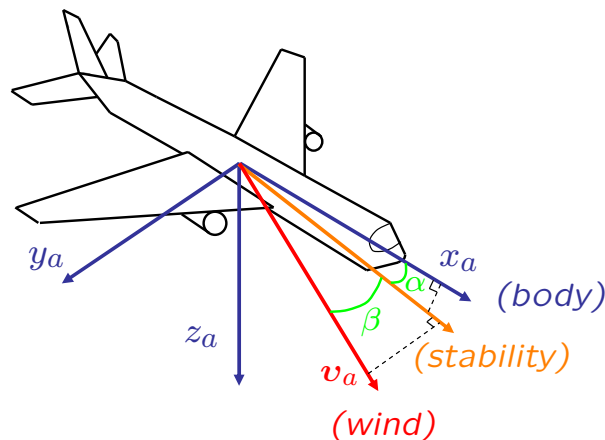


FIG. 3.7 – Différents repères utilisés [Stevens 92].

Notons que l'on définit le gîte cinématique (roulis dans le repère cinématique), noté μ , comme l'angle entre l'axe latéral y_a de l'avion et l'axe y_v du repère cinématique \mathcal{F}_v projeté dans le plan vertical (voir figure 3.8).

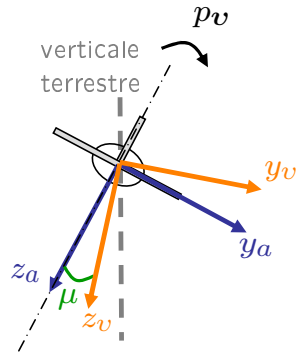


FIG. 3.8 – Repère cinématique \mathcal{F}_v , et gîte (ou roulis) cinématique μ .

Forces et moments entrant en jeu. L'avion subit un ensemble de forces lui permettant de voler [Tutnas, Stevens 92] (voir figure 3.9).

- Le poids (noté F_W , *weight* en anglais), dû à la gravité terrestre, est orienté verticalement vers le centre de la Terre.
- La portance (notée F_L , *lift* en anglais) générée par le déplacement de l'avion dans l'air, et orthogonale à la direction du vol, permet de compenser le poids pour que l'avion ne "tombe" pas.
- La traînée (notée F_D , *drag* en anglais), due à la résistance de l'air, est quant à elle opposée à la direction du vol.
- Enfin, la poussée (notée F_T , *thrust* en anglais) est la force générée par les moteurs.

Notons que les forces de portance et de traînée sont appelées forces aérodynamiques (notées $F_A = F_D + F_L$) : elles sont induites par le fait que l'avion se déplace dans l'air environnant.

Lorsque l'avion se déplace en palier horizontal stabilisé, sa vitesse est constante, et la somme des forces est nulle : si le tangage est nul, la portance compense le poids, et la poussée compense la traînée.

L'avion subit également un ensemble de moments, que l'on peut décomposer sous la forme des moments aérodynamiques, notés M_A , et des moments dûs à la propulsion par les moteurs, notés M_T .

3.1.2 Fonctionnement d'un avion

Actionneurs. En vol, l'orientation de l'avion peut être commandé par l'intermédiaire de surfaces mobiles, les gouvernes. On distingue trois types de gouverne : les ailerons, la gouverne de profondeur, et la gouverne de direction [Pallett 93] (voir figure 3.10).

- Les ailerons sont situés à l'arrière et vers l'extrémité de chaque aile. Leur inclinaison permet de modifier la portance générée par chaque aile, et finalement de

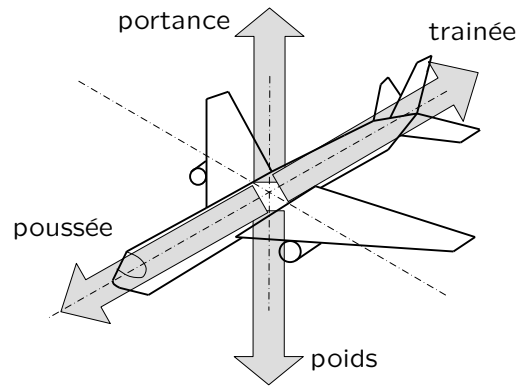


FIG. 3.9 – Forces agissant sur l'avion.

faire pivoter l'avion autour de son axe longitudinal.

- La gouverne de profondeur est située à l'arrière de l'appareil. Elle peut basculer vers le haut ou vers le bas, et faire ainsi pivoter l'avion autour de son axe latéral : le "nez" de l'avion pique ou monte.
- La gouverne de direction est une surface verticale située à l'arrière de l'avion. Elle peut s'orienter vers la droite ou vers la gauche, pour générer un changement de cap : l'avion pivote autour de son axe vertical et son "nez" s'oriente vers la droite ou vers la gauche.
- En plus des gouvernes, les moteurs sont également des actionneurs pouvant être commandés par le pilote, de façon à modifier le comportement de l'appareil.

Commandes pilote. Afin d'actionner ces systèmes, le poste de pilotage est équipé de plusieurs dispositifs de commande.

- Le palonnier se présente sous la forme de deux pédales (voir figure 3.11) : il permet de commander la gouverne de direction. Appuyer sur la pédale droite permet de faire virer l'avion vers la droite, et inversement.
- La commande des ailerons et de la gouverne de profondeur se fait par l'intermédiaire du "manche à balai" (appelé aussi *joystick*), ou du volant (appelé aussi guidonnet, ou *yoke*) — cela dépend des avions (voir figure 3.11). Pour actionner la gouverne de profondeur de façon à cabrer l'avion, le pilote tire sur le manche (ou sur le volant), il pousse pour que l'avion pique. La rotation du volant est équivalente au déplacement latéral du manche : cela permet d'incliner l'avion autour de son axe longitudinal en agissant sur les ailerons.
- Enfin, une manette permet de commander la poussée développée par les moteurs.

Transmission des commandes. La transmission des commandes initiées par le pilote vers les actionneurs peut alors être réalisée de différentes manières. La plus classique

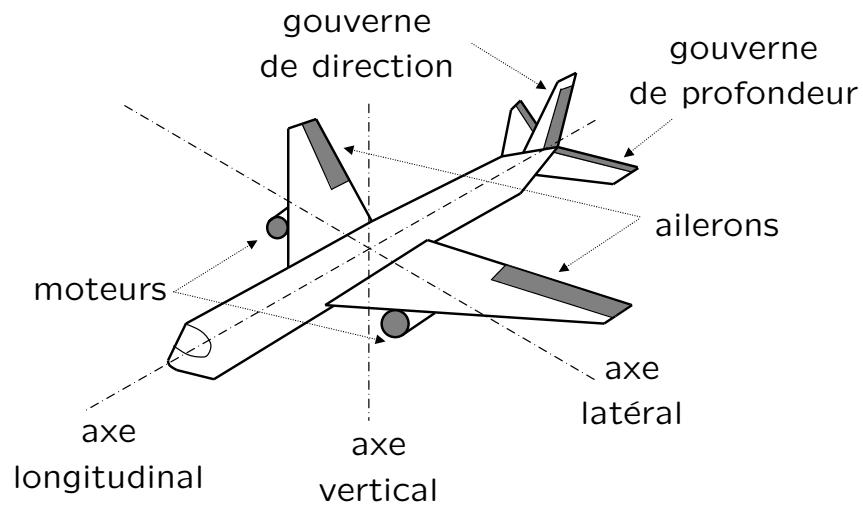


FIG. 3.10 – Dispositifs de commande d'un avion : gouvernes et moteurs.

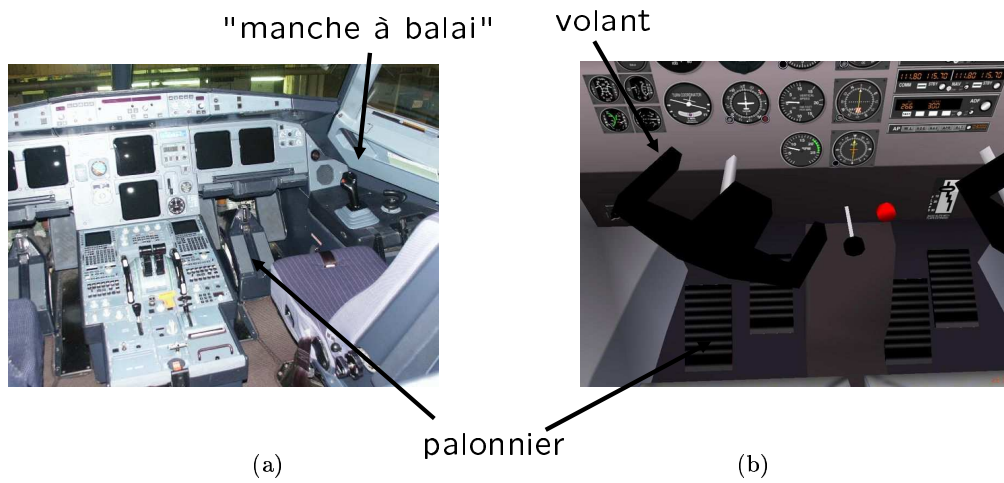


FIG. 3.11 – Différents types de commande pilote : (a) cockpit de l'A321, photo extraite de [Wikafcs]; (b) cockpit d'un C-172, dans le simulateur de vol *FlightGear*, image extraite de [Wiksimu].

consiste en une transmission directe, à l'aide de câbles qui relient mécaniquement les gouvernes au manche et au palonnier [Pallett 93]. Un tel système est cependant inutilisable sur de gros avions, où les forces aérodynamiques agissant sur les gouvernes sont telles que le pilote n'a pas la force de maintenir le manche [McLean 90]. L'action mécanique du pilote peut alors être relayée par des systèmes électrohydrauliques, qui actionnent les gouvernes.

Avec les systèmes de commande de vol électrique (notés FBW, pour *fly-by-wire* en anglais), le pilote n'agit pas directement sur les gouvernes [McLean 90]. Les manœuvres commandées par le pilote sont transmises sous forme de signaux électriques à un système de commande bas niveau (appelé AFCS pour *Automatic Flight Control System*). Ce dernier consiste alors en une boucle de commande bas-niveau qui facilite le pilotage. Les intérêts d'un tel système sont nombreux. Nous en citons ici quelques-uns [Pallett 93].

- Il permet l'utilisation d'un manche de type *joystick*, plus maniable et intuitif.
- Il permet de diminuer la masse de l'avion, en supprimant les câbles, poulies, et systèmes mécaniques, qui sont plus lourds que des systèmes électroniques
- Il permet de diminuer les temps de maintenance de l'avion au sol, puisque les systèmes de commande sont de type numériques.
- Il permet d'améliorer le confort, et de réduire l'impact des perturbations extérieures sur la structure de l'avion, grâce à une réponse plus rapide aux turbulences et aux rafales de vent.
- Il empêche l'avion de dépasser ses limites (en terme par exemple d'angle de roulis, tangage, lacet, vitesse, incidence, accélération, etc.) en limitant les commandes.
- Il permet d'améliorer la manœuvrabilité, en faisant en sorte que l'avion réponde exactement de la façon attendue.

Pour améliorer la fiabilité des systèmes FBW, ceux-ci sont triplés, voire quadruplés. Cela ne pose pas de problème majeur puisqu'il s'agit d'un système léger comparativement à la commande mécanique. De plus, les systèmes ne sont idéalement pas exactement identiques, mais développés séparément, ce qui permet de réduire les risques de défaillance simultanée.

Notons qu'en 1984, l'Airbus A320 était le premier avion de ligne à embarquer un système de commande de vol électrique numérique [Pallett 93, Tischler 96]. Le premier avion d'affaire à intégrer un tel système a vu le jour en 2005 : il s'agit du Falcon 7X de Dassault Aviation [Wikafcs].

Équations du mouvement d'un avion L'avion est considéré comme un corps rigide non déformable, qui peut se déplacer dans l'espace selon six degrés de liberté (trois translations et trois rotations). La Terre est supposée fixe et plate. La masse m de l'avion est supposée constante. Notons \mathbf{I} la matrice d'inertie de l'avion, et $\boldsymbol{\omega}_a$ sa vitesse de rotation dans le repère lié à l'avion \mathcal{F}_a . Les relations classiques permettent de lier les vitesses de l'avion à la dérivée de sa position et de son orientation dans le repère fixe \mathcal{F}_f [Stevens 92, Mahony 05a, Rives 02] :

$${}^f \dot{\mathbf{t}}_a = {}^f \mathbf{R}_a \mathbf{v}_a \quad (3.9)$$

$${}^f \dot{\mathbf{R}}_a = {}^f \mathbf{R}_a [\boldsymbol{\omega}_a]_{\times} \quad (3.10)$$

Nous allons appliquer la seconde loi de Newton à l'avion. Notons que nous nous plaçons dans le repère noté \mathcal{F}_a , appelé précédemment *body axis*, dont l'origine est au centre de gravité de l'avion (voir figure 3.7) [McLean 90]. Les forces et moments agissant sur l'avion ont été décrits précédemment (section 3.1.1). On a :

$$F_A + F_W + F_T = m(\dot{\mathbf{v}}_a + [\boldsymbol{\omega}_a]_{\times} \mathbf{v}_a) \quad (3.11)$$

$$M_a + M_T = \mathbf{I}\dot{\boldsymbol{\omega}}_a + [\boldsymbol{\omega}_a]_{\times} \mathbf{I}\boldsymbol{\omega}_a. \quad (3.12)$$

Notons que l'on peut exprimer la force F_W due à la gravité en fonction du vecteur accélération de la gravité $\mathbf{g} = (0, 0, g)$ dans le repère \mathcal{F}_f . En effet, en projetant \mathbf{g} dans le repère lié à l'avion \mathcal{F}_a , on obtient :

$$F_W = m^a \mathbf{R}_f \mathbf{g} \quad (3.13)$$

À partir de ces relations, on considère un état d'équilibre, et on considère de petites variations des différents paramètres autour de cette position d'équilibre. On linéarise alors les équations obtenues autour de ce point d'équilibre. En considérant que les mouvements latéraux (dans le plan horizontal) et longitudinaux (dans le plan vertical) sont découplés, on peut alors établir les équations de la dynamique de l'avion sous la forme de deux représentations d'états. On a alors des équations du type $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, où \mathbf{x} est le vecteur d'état, \mathbf{u} le vecteur contenant les entrées de commande, et où \mathbf{A} et \mathbf{B} sont appelées matrices d'état du système.

Par exemple, le mouvement longitudinal peut être modélisé par le modèle d'état suivant [Cook 97, Nelson 98] :

$$\begin{bmatrix} \dot{v}_{ax} \\ \dot{v}_{az} \\ \dot{\omega}_{ay} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} X_u & X_w & X_q & X_\theta \\ Z_u & Z_w & Z_q & Z_\theta \\ M_u & M_w & M_q & M_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v_{ax} \\ v_{az} \\ \omega_{ay} \\ \theta \end{bmatrix} + \begin{bmatrix} X_{\delta_m} & X_{\delta_T} \\ Z_{\delta_m} & Z_{\delta_T} \\ M_{\delta_m} & M_{\delta_T} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix} \quad (3.14)$$

où δ_m représente la commande de la variation angulaire de la gouverne de profondeur, et δ_T la commande de la variation de la poussée des moteurs. Les termes des matrices d'évolution sont à identifier selon l'avion que l'on souhaite modéliser, et le point d'équilibre choisi pour la linéarisation. Notons qu'une telle représentation n'est pas unique. Il est par exemple courant de remplacer v_{az} dans le vecteur d'état par l'incidence $\alpha = \frac{v_{az}}{v_a}$ (cf. (3.7)) [McLean 90, Nelson 98]. On peut également ajouter d'autres termes dans l'état, si nécessaire, tels que l'altitude par exemple [McLean 90].

Quant au mouvement latéral, on peut le mettre sous la forme [Cook 97, Nelson 98] :

$$\begin{bmatrix} \dot{v}_{ay} \\ \dot{\omega}_{ax} \\ \dot{\omega}_{az} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} Y_v & Y_p & Y_r & Y_\phi \\ L_v & L_p & L_r & L_\phi \\ N_v & N_p & N_r & N_\phi \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_{ay} \\ \omega_{ax} \\ \omega_{az} \\ \phi \end{bmatrix} + \begin{bmatrix} Y_{\delta_l} & Y_{\delta_r} \\ L_{\delta_l} & L_{\delta_r} \\ N_{\delta_l} & N_{\delta_r} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_l \\ \delta_r \end{bmatrix} \quad (3.15)$$

où δ_l représente la commande de la variation angulaire des ailerons, et δ_r la commande de la variation angulaire de la gouverne de direction. Comme dans le cas longitudinal, on peut modifier le vecteur d'état, en remplaçant \dot{v}_{ay} par $\beta = \frac{v_{ay}}{v_a}$ (cf. (3.7))

[McLean 90, Nelson 98]. On peut également ajouter des états, l'angle de lacet ψ par exemple [Cook 97, McLean 90].

Ces mises en équation étant présentées dans de nombreux ouvrages [McLean 90, Stevens 92, Etkin 95, Cook 97, Nelson 98], nous les détaillons pas plus ici. Nous présenterons dans la section 3.2 le modèle utilisé pour notre étude.

3.1.3 Commande

Comme nous l'avons vu à la section 3.1.2, dans les avions modernes les commandes initiées par le pilote ne sont pas directement transmises aux actionneurs. Des lois de commande "bas-niveau" permettent d'améliorer la stabilité du système et de le rendre plus maniable. De nombreux ouvrages traitent de ces commandes, appelées SAS (*Stability Augmentation System*) et CAS (*Control Augmentation System*) [Stevens 92, McLean 90, Cook 97, Nelson 98]. La conception de telles lois de commande repose le plus souvent sur des modèles LTI (*Linear Time Invariant*) et des techniques de commande classiques, de type placement de pôles, étude du lieu des racines, spécification de la réponse fréquentielle du système bouclé, etc. La commande optimale, appelée aussi commande LQR (*Linear Quadratic Regulator*) est également utilisée, éventuellement couplée avec un observateur de l'état du système pour estimer l'état du système à partir de la sortie [Nelson 98, McLean 90].

Des commandes de plus haut niveau permettant au pilote d'actionner des modes de vol (tels que "conserver une altitude constante", "atterrissage automatique", etc.) constituent également des boucles de commande, appelées "pilote automatique" (souvent abrégé PA) [Nelson 98, Stevens 92].

Notons également les travaux de recherche reposant sur les réseaux de neurones [Jorgensen 90, Iiguni 98, Izadi 03, Li 04], la commande de type LQG / LTR (*Linear Quadratic Gaussian / Loop Transfer Recovery*) [Chatterji 96] ou LQR [Rives 02] pour l'atterrissage automatique. La navigation et le suivi de trajectoire 3D à base de commande floue [Doitsidis 04], et l'utilisation de commande robuste H_∞ dans le cadre de l'atterrissage [Che 01] ou du ravitaillement en vol [Fravolini 03] ont également été proposés.

3.1.4 Comportement d'un avion pendant l'atterrissage

Pendant l'atterrissage, l'avion doit minimiser l'écart latéral avec l'axe de la piste (voir figure 3.12), et suivre une pente d'approche adaptée afin de pouvoir toucher la piste dans de bonnes conditions (voir figure 3.13). Les systèmes d'ILS jouent d'ailleurs sur ces deux aspects : ils fournissent d'une part l'écart de l'avion par rapport à l'axe de la piste (*localizer*), et d'autre part l'écart de l'avion par rapport à la pente nominale d'approche (*glide path*) [McLean 90].

L'atterrissage se décompose en plusieurs phases (voir figure 3.13) [McLean 90, Jorgensen 90, Izadi 03] :

- l'approche, au cours de laquelle l'avion suit à vitesse constante la pente d'approche, inclinée d'un angle γ (de l'ordre de -3 degrés usuellement),

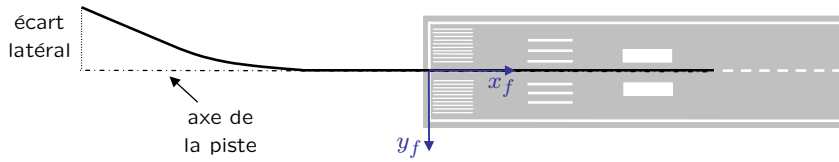
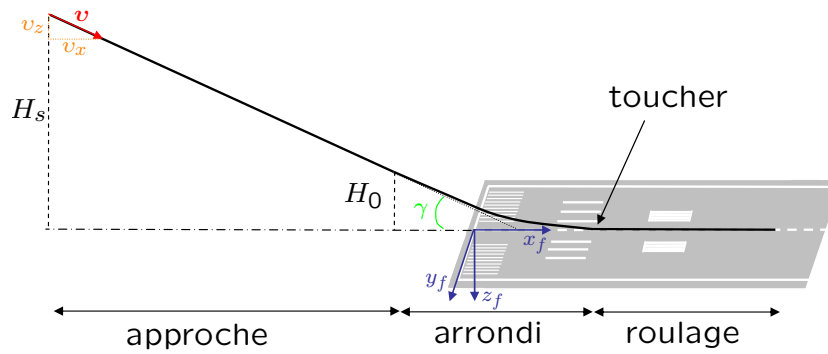


FIG. 3.12 – Atterrissage vu d'en haut.

FIG. 3.13 – Phases d'atterrissage : après l'approche selon une pente γ (< 0 dans le cas de l'atterrissage), l'arrondi débute à l'altitude H_0 .

- l'arrondi, initié à l'altitude H_0 ($\simeq 15$ à 20 mètres usuellement) au cours duquel l'avion réduit sa vitesse verticale, afin de ne pas toucher la piste trop brutalement,
- le toucher de la piste, au moment où le train d'atterrissage touche la piste,
- le roulage, au cours duquel l'avion freine pour atteindre une vitesse nulle avant d'être en bout de piste (ordre de grandeur de la longueur de la piste : 2000 m à 3000 m).

3.1.4.1 Approche

Objectif. Pendant cette phase, l'avion doit rejoindre l'axe de la piste, puis rester aligné avec cet axe. La vitesse v_a et la pente γ sont constantes [McLean 90].

Trajectoire. Après avoir rejoint l'axe de la piste, la trajectoire de l'avion, dans le repère fixe \mathcal{F}_f est telle que

$$\dot{X} = v_a \cos \gamma \quad (3.16)$$

$$\dot{Y} = 0 \quad (3.17)$$

$$\dot{Z} = -v_a \sin \gamma = -\dot{X} \tan \gamma \quad (3.18)$$

et

$$Y(t) = 0 \quad (3.19)$$

$$Z(t) = -X(t) \tan \gamma \quad (3.20)$$

avec $X_s = \frac{H_s}{\tan \gamma}$ la position de l'avion selon l'axe x_f à l'instant $t = t_s$ où son altitude vaut $Z = -H_s$.

Ordres de grandeur numériques. La pente est de l'ordre de $\gamma = -2.5$ à $\gamma = -6$ degrés, et la vitesse d'approche $v_a = 55$ m/s par exemple.

3.1.4.2 Arrondi

Objectif. Pendant cette phase, comme son nom l'indique, la trajectoire s'arrondit. Cela a pour but de diminuer la composante verticale de la vitesse $|\dot{Z}|$ [Izadi 03, Stevens 92, McLean 90].

La phase d'arrondi débute à l'altitude $Z = -H_0$, alors que l'avion suit la pente d'approche définie par l'angle γ . Le prolongement de la pente d'approche coupe la piste en un point appelé point d'aboutissement. Le but est alors de diminuer la pente pour arrondir la trajectoire, et toucher la piste au point d'impact [McLean 90] (voir figure 3.14).

Trajectoire. Supposons que la phase d'arrondi débute à l'instant $t = 0$, lorsque l'altitude de l'avion vaut $Z(0) = -H_0$ et qu'il est à la distance $X(0) = X_0$ du point d'aboutissement (voir figure 3.14). On a avec (3.20) :

$$H(0) = H_0 = X_0 \tan \gamma, \quad (3.21)$$

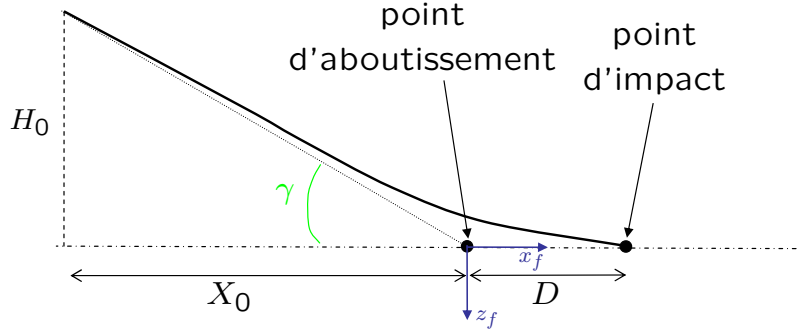


FIG. 3.14 – Phase d'arrondi (X_0 est ici négatif, et D est positif, H_0 est positif).

et avec (3.18), à l'instant $t = 0$:

$$\dot{H}(0) = \dot{H}_0 = \dot{X}_0 \tan \gamma. \quad (3.22)$$

Notons $t = t_T$ l'instant où l'avion va toucher la piste (donc où son altitude est nulle). On a alors à cet instant :

$$H(t_T) = H_T = 0. \quad (3.23)$$

L'objectif est d'arrondir la trajectoire de manière à avoir une vitesse verticale donnée au moment de l'impact. On impose donc que :

$$\dot{H}(t_T) = \dot{H}_T \quad (3.24)$$

où \dot{H}_T est fixée.

À partir des quatre contraintes (3.21), (3.22), (3.23), et (3.24), et en considérant que $\dot{X}_0 = \dot{X}_T$ (vitesse sol constante, ce qui est une approximation raisonnable [McLean 90]), on modélise la trajectoire $H(t)$ sous la forme d'une exponentielle décroissante [Jorgensen 90], [Iiguni 98] :

$$H(t) = \frac{H_0}{\dot{H}_0 - \dot{H}_T} \left(\dot{H}_0 \exp\left(\frac{-X(t) + X_0}{\tau_x}\right) - \dot{H}_T \right) \quad (3.25)$$

avec

$$\tau_x = -\frac{H_0 \dot{X}_0}{\dot{H}_0 - \dot{H}_T}$$

On a alors :

$$\dot{H}(t) = \dot{H}_0 \exp\left(\frac{-X(t) + X_0}{\tau_x}\right). \quad (3.26)$$

On vérifie bien que l'on a au début de l'arrondi $H(0) = H_0$, avec (3.25), et $\dot{H}(0) = \dot{H}_0$ avec (3.26). L'équation (3.25) nous indique aussi que l'on a $H(t_T) = 0$ lorsque

$$\exp\left(\frac{-X(t) + X_0}{\tau_x}\right) = \frac{\dot{H}h_T}{\dot{H}_0}. \quad (3.27)$$

En calculant $\dot{H}(t_T)$ avec (3.26) et en remplaçant $\exp\left(\frac{-X(t)+X_0}{\tau_x}\right)$ par son expression en $t = t_T$ (3.27), on obtient bien $\dot{H}(t_T) = \dot{H}_T$ [Jorgensen 90]. Ainsi, le choix de $H(t)$ sous la forme (3.25) respecte bien les conditions définies initialement aux instants $t = 0$ et $t = t_T$.

Ordres de grandeur numériques. Habituellement, l'arrondi est initié à une altitude de l'ordre de 20 à 15 mètres [Pallett 93, McLean 90]. La pente de la trajectoire est alors la pente d'approche : de l'ordre de $\gamma = -2.5$ à $\gamma = -6$ degrés [Pallett 93, McLean 90]. La vitesse sol est de l'ordre de $\dot{X} = 55$ m/s, et la composante verticale de la vitesse de l'ordre de $\dot{H}_0 = \dot{X} \tan \gamma = -3$ m/s [Stevens 92]. L'objectif est alors de commander l'altitude de l'avion [Stevens 92] de façon à arrondir la trajectoire pour faire en sorte que la composante verticale de la vitesse au moment de l'impact soit de l'ordre de $\dot{H}_T = -0.5$ m/s [Jorgensen 90], ce qui correspond à une pente de l'ordre de $\gamma = -0.5$ degrés.

La distance X_0 découle de la valeur de l'altitude H_0 et de celle de la pente γ . Habituellement, la distance D est de quelques centaines de mètres [McLean 90, Stevens 92] (voir figure 3.14).

3.1.4.3 Toucher de la piste et roulage

Le mode arrondi est en fait inactivé un peu avant le toucher, afin que l'on puisse diminuer l'assiette longitudinale de l'avion (angle de tangage θ , voir figure 3.4). En effet, si l'assiette est trop grande, c'est la "queue" de l'avion qui touche la piste en premier. Mais si l'assiette devient négative, la roue avant impacte la piste en premier, ce qui n'est pas souhaitable non plus ! La gouverne de profondeur est donc utilisée pour régler l'assiette afin que l'impact se fasse dans de bonnes conditions [Bonnet 01].

Après l'impact (roues arrières du train d'atterrissage en premier, puis roulette de "nez"), on active le système de freinage pour stopper l'avion. On peut également utiliser le système d'inversion de la poussée des moteurs, qui génère une force orientée vers l'arrière de l'avion, et l'aide ainsi à s'arrêter [Pallett 93].

3.1.4.4 Rôle des volets

Pendant la procédure d'atterrissage, nous avons vu qu'il est important de réduire la vitesse de l'avion. Mais lorsque la vitesse diminue, la portance diminue (voir figure 3.9). Afin de pouvoir voler à faible vitesse sans risque de décrochage, on utilise des dispositifs hypersustentateurs, appelés volets. Ils permettent d'augmenter la surface des ailes (voir figure 3.15), et donc d'augmenter la portance [Pallett 93, Stevens 92]. Lors

de l'atterrissage, les volets présentent un deuxième avantage : ils augmentent la traînée, ce qui contribue à freiner l'avion sur la piste [Pallett 93].



FIG. 3.15 – Volets sortis pour l'atterrissage : modèle de l'A380 dans le simulateur de vol *Flight Simulator*.

Notons par ailleurs que les volets sont également utilisés lors du décollage, pour les mêmes raisons. Ils sont cependant positionnés différemment, et associés à d'autres dispositifs hypersustentateurs — les becs — de manière à obtenir un bon compromis entre l'augmentation suffisante de la portance, sans augmenter trop la traînée (ce qui serait pénalisant pour le décollage) [Pallett 93].

3.1.5 Atterrissage par asservissement visuel

L'objectif de notre étude est d'automatiser l'atterrissage de l'avion, sur une piste non équipée d'ILS. Pour cela, on considère une caméra embarquée sur l'avion. La phase de traitement d'image, visant à extraire la piste d'atterrissage et les autres informations éventuellement utiles de l'image, est supposée être réalisée. On s'attache donc à concevoir une loi de commande permettant à l'avion d'atterrir. On se place dans le cadre d'un avion équipé d'un système FBW et de boucles de commande bas-niveau. Comme on l'a déjà dit, ces systèmes permettent une meilleure stabilisation de l'avion, et assurent une meilleure sécurité de l'engin en empêchant que d'éventuelles commandes inadaptées ne soient transmises aux actionneurs. On se placera donc "à la place du pilote", pour concevoir un système automatique qui actionnerait les commandes accessibles au pilote, de manière à réaliser un atterrissage à partir des données transmises par la caméra embarquée. Comme dans le cas de l'hélicoptère présenté dans le chapitre précédent, l'avion est un système sous-actionné : il possède plus de degrés de liberté que d'entrées de commande. Il nous faudra donc tenir compte du modèle liant les entrées pilote aux mouvements de l'avion, et des couplages entre les différents degrés de liberté. L'objectif est alors de concevoir des lois de commande reposant sur des informations visuelles pertinentes compte tenu de ces contraintes.

3.2 Modélisation de l'avion

Dassault Aviation nous a fourni un simulateur représentatif du comportement d'un avion, et un modèle non-linéaire approchant le comportement dynamique de l'avion

considéré dans le simulateur. Nous présentons ici le modèle, que nous avons linéarisé par la suite autour d'un point de fonctionnement et séparé en deux parties découplées. Ce modèle nous sert à concevoir des lois de commande adaptées à l'avion considéré.

3.2.1 Modèle non-linéaire

Le modèle fourni se présente sous la forme d'un état, noté \mathbf{x} , et des équations décrivant l'évolution de cet état au cours du temps, selon les entrées de commande, notées \mathbf{u} . Il est de la forme $\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}$. Le dérapage est négligé.

L'état \mathbf{x} contient les paramètres suivants :

- α : incidence (voir figure 3.6.a)
- ω_y : vitesse de tangage (voir figure 3.5)
- γ : pente (voir figure 3.6.a)
- p_v : vitesse de roulis dans le repère cinématique (voir figure 3.8)
- μ : gîte cinématique (voir figure 3.8)
- v_a : module du vecteur vitesse
- n_{x_T} : poussée divisée par mg (où m est la masse de l'avion, et g l'accélération de la gravité, supposées constantes).

Les entrées de commande \mathbf{u} considérées sont :

- δ_m : commande de profondeur
- δ_l : commande de gauchissement
- δ_T : commande de la poussée.

Notons qu'on ne considère pas la gouverne de direction δ_r : le changement de cap est induit par la modification de l'angle de roulis.

Les équations du modèle sont les suivantes, où on a représenté en rouge les termes composant l'état \mathbf{x} , et en vert les entrées de commande du système :

$$\dot{\alpha} = \omega_{a_y} - \frac{g}{v_a} \left[\frac{v_a^2}{v_{a0}^2} (n_{z00} + n_{z\alpha}(\alpha - \alpha_0)) + n_{x_T} \sin \alpha - \cos \mu \cos \gamma \right] \quad (3.28)$$

$$\dot{\omega}_{a_y} = \frac{v_a^2}{v_{a0}^2} M_\alpha (\alpha - \alpha_0) + \frac{v_a}{v_{a0}} M_{q_0} \omega_{a_y} + M_{FCS} \left(1 - \frac{v_a^2}{v_{a0}^2} \right) + \frac{v_a^2}{v_{a0}^2} M_{\delta_{m_0}} \delta_m \quad (3.29)$$

$$\dot{\gamma} = \frac{g}{v_a} \left[\left(\frac{v_a^2}{v_{a0}^2} (n_{z00} + n_{z\alpha}(\alpha - \alpha_0)) + n_{x_T} \sin \alpha \right) \cos \mu - \cos \gamma \right] \quad (3.30)$$

$$\dot{p}_v = \frac{v_a}{v_{a0}} L_{p_0} p_v + \frac{v_a^2}{v_{a0}^2} L_{\delta_{l_0}} \delta_l \quad (3.31)$$

$$\dot{\mu} = p_v \quad (3.32)$$

$$\dot{v}_a = g \left[\frac{v_a^2}{v_{a0}^2} (n_{x00} + n_{x\alpha}(\alpha - \alpha_0)) + n_{x_T} \cos \alpha - \sin \gamma \right] \quad (3.33)$$

$$\dot{n}_{x_T} = \omega_T (-n_{x_T} + X_{\delta_T} \delta_T). \quad (3.34)$$

Les constantes $n_{z\alpha}$, $n_{x\alpha}$, n_{z00} , n_{x00} , M_α , M_{q_0} , $M_{\delta_{m_0}}$, M_{FCS} , L_{p_0} , $L_{\delta_{l_0}}$, ω_T , X_{δ_T} sont fixées. \dot{n}_{x_T} est écriété entre L_v^- et L_v^+ , constantes à fixer également. Ces équations sont

valables autour du point de fonctionnement défini par v_{a_0} et α_0 . Les valeurs numériques sont données dans l'annexe D.1.

On peut également écrire des équations de sortie de la forme $\mathbf{y} = h(\mathbf{x})$, où $\mathbf{y} = (v_{a_x}, v_{a_y}, v_{a_z}, \omega_{a_x}, \omega_{a_y}, \omega_{a_z})$.

$$v_{a_x} = v_a \cos \alpha \quad (3.35)$$

$$v_{a_y} = 0 \quad (3.36)$$

$$v_{a_z} = v_a \sin \alpha \quad (3.37)$$

$$\omega_{a_x} = p_v \cos \alpha - \left(\frac{g \sin \mu}{v_a} \right) \sin \alpha \quad (3.38)$$

$$\omega_{a_y} = q \quad (3.39)$$

$$\omega_{a_z} = \left(\frac{g \sin \mu}{v_a} \right) \cos \alpha + p_v \sin \alpha \quad (3.40)$$

3.2.2 Modèle linéarisé et découplage

Linéarisation. Le modèle précédent est valable en un point de vol particulier, défini par la vitesse nominale v_{a_0} et l'incidence nominale α_0 (pour lesquels on a une pente de γ_0 , une poussée de X_{T_0} , et une commande sur les gaz δ_{T_0}). On sépare chaque paramètre de l'état en deux termes : un terme constant x_0 et une partie contenant les variations autour de ce terme constant dx . On rappelle que certains paramètres ont des valeurs nominales nulles : $\omega_{y_0} = 0$, $p_{v_0} = 0$, $\mu_0 = 0$, $\delta_{l_0} = 0$, $\delta_{m_0} = 0$ (voir annexe D.1).

Sous les hypothèses de variations angulaires de faibles amplitudes autour des valeurs nominales, on peut écrire les approximations classiques suivantes :

$$\cos(\alpha) \simeq \cos(\alpha_0) - \sin(\alpha_0)d\alpha \quad (3.41)$$

$$\cos(\gamma) \simeq \cos(\gamma_0) - \sin(\gamma_0)d\gamma \quad (3.42)$$

$$\cos(\mu) \simeq 1 \quad (3.43)$$

$$\sin(\alpha) \simeq \sin(\alpha_0) + \cos(\alpha_0)d\alpha \quad (3.44)$$

$$\sin(\gamma) \simeq \sin(\gamma_0) + \cos(\gamma_0)d\gamma \quad (3.45)$$

$$\sin(\mu) \simeq d\mu \quad (3.46)$$

En considérant de faibles variations de vitesse autour de v_{a_0} , on écrit :

$$\left(\frac{v_a}{v_{a_0}} \right)^2 \simeq 2 \frac{dv_a}{v_{a_0}} + 1 \quad (3.47)$$

$$\frac{1}{v_a} \simeq \frac{1}{v_{a_0}} - \frac{dv_a}{v_{a_0}^2} \quad (3.48)$$

$$v_a^2 \simeq v_{a_0}^2 + 2v_{a_0}dv_a \quad (3.49)$$

On remplace ensuite les différents termes par leurs expressions (équations (3.41) à 3.49)) dans les équations (3.28) à (3.34) du modèle non linéaire. En écrivant les petites

variations dx sous la forme $dx = x - x_0$, où x reste proche de x_0 , on peut alors écrire le modèle sous la forme matricielle

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A}\tilde{\mathbf{x}} + \mathbf{B}\tilde{\mathbf{u}} \quad (3.50)$$

avec $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_0$, $\dot{\tilde{\mathbf{x}}} = \dot{\mathbf{x}} - \dot{\mathbf{x}}_0$, $\tilde{\mathbf{u}} = \mathbf{u} - \mathbf{u}_0$ et où les expressions des matrices \mathbf{A} et \mathbf{B} et des paramètres de référence \mathbf{x}_0 , $\dot{\mathbf{x}}_0$ et \mathbf{u}_0 sont les suivantes :

$$\mathbf{A} = \begin{bmatrix} -A_1 & 1 & -A_2 & 0 & 0 & -A_3 & -A_4 \\ M_\alpha & M_{q_0} & 0 & 0 & 0 & -\frac{2M_{FCS}}{v_{a_0}} & 0 \\ A_1 & 0 & A_2 & 0 & 0 & A_3 & A_4 \\ 0 & 0 & 0 & L_{p_0} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_5 & 0 & -gc(\gamma_0) & 0 & 0 & \frac{2gn_{x_{00}}}{v_{a_0}} & gc(\alpha_0) \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_T \end{bmatrix}$$

$$\text{avec } \begin{cases} A_1 = \frac{g}{v_{a_0}} [n_{z\alpha} + n_{X_{T_0}} c(\alpha_0)] \\ A_2 = \frac{g}{v_{a_0}} s(\gamma_0) \\ A_3 = \frac{g}{v_{a_0}^2} [n_{z_{00}} - n_{X_{T_0}} s(\alpha_0) + c(\gamma_0)] \\ A_4 = \frac{g}{v_{a_0}} s(\alpha_0) \\ A_5 = g(n_{x_\alpha} - s(\alpha_0)n_{X_{T_0}}) \end{cases}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ M_{\delta_{m_0}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & L_{\delta l_0} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \omega_T X_{T_\delta} \end{bmatrix},$$

$$\mathbf{x}_0 = \begin{bmatrix} \alpha_0 \\ \omega_{a_{y_0}} \\ \gamma_0 \\ p_{v_0} \\ \mu_0 \\ v_{a_0} \\ n_{X_{T_0}} \end{bmatrix}, \quad \dot{\mathbf{x}}_0 = \begin{bmatrix} -\frac{g}{v_{a_0}} [n_{z_{00}} + n_{X_{T_0}} s(\alpha_0) - c(\gamma_0)] \\ 0 \\ \frac{g}{v_{a_0}} [n_{z_{00}} + n_{X_{T_0}} s(\alpha_0) - c(\gamma_0)] \\ 0 \\ 0 \\ g [n_{x_{00}} + n_{X_{T_0}} c(\alpha_0) - s(\gamma_0)] \\ \omega_T (-n_{X_{T_0}} + X_{\delta_T} \delta_{T_0}) \end{bmatrix}, \quad \mathbf{u}_0 = \begin{bmatrix} \delta_{m_0} \\ \delta_{l_0} \\ \delta_{T_0} \end{bmatrix}.$$

De la même façon, à partir des équations (3.35) et (3.40) on écrit le modèle de sortie sous la forme

$$\tilde{\mathbf{y}} = \mathbf{C}\tilde{\mathbf{x}} \quad (3.51)$$

avec $\tilde{\mathbf{y}} = \mathbf{y} - \mathbf{y}_0$ et

$$\mathbf{C} = \begin{bmatrix} -v_{a_0} s(\alpha_0) & 0 & 0 & 0 & c(\alpha_0) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ v_{a_0} c(\alpha_0) & 0 & 0 & 0 & s(\alpha_0) & 0 \\ 0 & 0 & 0 & c(\alpha_0) & \frac{-gs(\alpha_0)}{v_{a_0}} & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & s(\alpha_0) & \frac{gc(\alpha_0)}{v_{a_0}} & 0 \end{bmatrix}, \quad \mathbf{y}_0 = \begin{bmatrix} v_{a_0} c(\alpha_0) \\ 0 \\ v_{a_0} s(\alpha_0) \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (3.52)$$

Pour simplifier les notations, on omettra le $\tilde{\cdot}$ dans la suite, bien que l'on considérera encore les variations des paramètres par rapport à l'état d'équilibre.

Découplage. On remarque que les matrices \mathbf{A} , \mathbf{B} , et \mathbf{C} contiennent des lignes (ou des colonnes) de zéro, et que le modèle obtenu peut donc s'écrire sous la forme de deux parties découplées.

Une partie concerne l'évolution de l'avion dans le plan horizontal (modèle latéral). Cette partie du modèle a la forme suivante :

$$\begin{bmatrix} \dot{p}_v \\ \dot{\mu} \end{bmatrix} = \begin{bmatrix} L_{p0} & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p_v \\ \mu \end{bmatrix} + \begin{bmatrix} L_{\delta_{l_0}} \\ 0 \end{bmatrix} \delta_l \quad (3.53)$$

et

$$\begin{bmatrix} \omega_{a_x} \\ \omega_{a_z} \end{bmatrix} = \begin{bmatrix} c(\alpha_0) & \frac{-gs(\alpha_0)}{v_{a_0}} \\ s(\alpha_0) & \frac{gc(\alpha_0)}{v_{a_0}} \end{bmatrix} \begin{bmatrix} p_v \\ \mu \end{bmatrix}. \quad (3.54)$$

L'autre partie concerne l'évolution de l'avion dans le plan vertical. Il s'agit du modèle longitudinal, qui a la forme suivante :

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\omega}_{a_y} \\ \dot{\gamma} \\ \dot{v}_a \\ \dot{n}_{X_T} \end{bmatrix} = \begin{bmatrix} -A_1 & 1 & -A_2 & -A_3 & -A_4 \\ M_\alpha & M_{q_0} & 0 & -\frac{2M_{FCS}}{V_0} & 0 \\ A_1 & 0 & A_2 & A_3 & A_4 \\ A_5 & 0 & -gc(\gamma_0) & \frac{2gn_{x00}}{V_0} & gc(\alpha_0) \\ 0 & 0 & 0 & 0 & -\omega_T \end{bmatrix} \begin{bmatrix} \alpha \\ \omega_{a_y} \\ \gamma \\ v_a \\ n_{X_T} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ M_{\delta_{m_0}} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \omega_T X_{T_\delta} \end{bmatrix} \begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix} \quad (3.55)$$

et

$$\begin{bmatrix} v_{a_x} \\ v_{a_z} \\ \omega_{a_y} \end{bmatrix} = \begin{bmatrix} -v_{a_0}s(\alpha_0) & 0 & 0 & c(\alpha_0) & 0 \\ v_{a_0}c(\alpha_0) & 0 & 0 & s(\alpha_0) & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \omega_{a_y} \\ \gamma \\ v_a \\ n_{X_T} \end{bmatrix}. \quad (3.56)$$

Simplification. Pour lier l'évolution des paramètres de l'état à l'évolution de la position et de l'orientation de l'avion dans le repère fixe, on a choisi de simplifier le problème en faisant l'hypothèse que l'orientation de l'avion par rapport à la piste reste proche de l'identité :

$${}^f\mathbf{R}_a \simeq \mathbf{I}_3 \quad (3.57)$$

d'où $\phi \simeq 0$, $\theta \simeq 0$ et $\psi \simeq 0$.

Concernant ϕ et ψ , cette hypothèse est équivalente à considérer que l'on reste proche de l'orientation désirée de l'avion (en l'absence de vent de travers), puisque notre objectif est d'aligner l'avion avec la piste. Pour ce qui est de l'angle θ , sa valeur nominale est telle que $\theta_0 = \alpha_0 + \gamma_0 = 2.07$ degrés (voir équation (3.8) et annexe D.1). Considérer l'hypothèse (3.57) est donc réaliste, et simplifie les calculs.

En utilisant la relation (C.3) de l'annexe C, on peut écrire [Gangloff 00] :

$$\dot{\mathbf{p}} \simeq \mathbf{v}_a \quad (3.58)$$

où \mathbf{p} représente la position et l'orientation de l'avion dans le repère fixe :

$$\mathbf{p} = (X, Y, Z, \phi, \theta, \psi),$$

où \mathbf{v}_a est le torseur cinématique de l'avion (vitesses de translation et de rotation de l'avion par rapport au repère fixe, exprimées dans le repère de l'avion) :

$$\mathbf{v}_a = (v_{a_x}, v_{a_y}, v_{a_z}, \omega_{a_x}, \omega_{a_y}, \omega_{a_z}). \quad (3.59)$$

En utilisant la relation (3.58) et les équations initiales du modèle sous la forme de deux parties découplées (voir équations (3.53) à (3.56)), on modifie le vecteur d'état pour obtenir les modèles d'état latéraux et longitudinaux détaillés ci-après.

Modèle latéral. On a ajouté le paramètre Y dans le vecteur d'état, afin de tenir compte de la position latérale de l'avion par rapport à la piste. On a modélisé son évolution par $\dot{Y} = a_5 \psi = v_{a_0} \cos \gamma_0 \psi$ (voir figure 3.16, avec $\dot{Y} \simeq v_{a_y}$ et $\sin \psi \simeq \psi$).

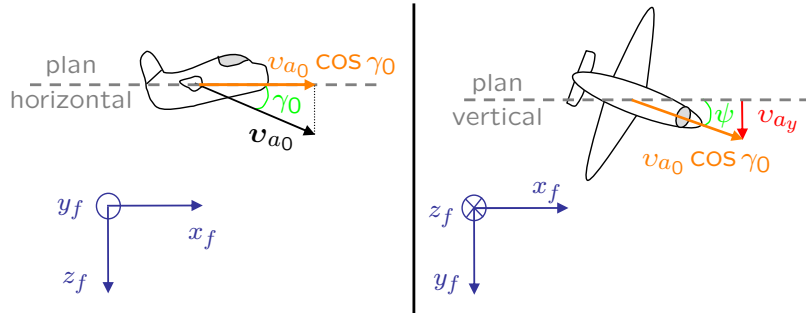


FIG. 3.16 – Décomposition du vecteur vitesse selon l'axe y_f .

Le modèle latéral s'écrit alors sous la forme :

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\psi} \\ \dot{\psi} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & 0 & 0 \\ a_3 & a_4 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a_5 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\psi} \\ \psi \\ Y \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ 0 \\ 0 \end{bmatrix} \delta_l. \quad (3.60)$$

Le modèle latéral fait intervenir la commande de gauchissement δ_l , qui joue sur les ailerons. Il fait également intervenir la position latérale et les rotations de roulis et de lacet, comme dans l'expression générale (3.15). On rappelle qu'on ne considère pas la gouverne de direction δ_r , car dans notre cas, le changement de cap est induit par la modification de l'angle de roulis.

Une version simplifiée du modèle, faisant l'hypothèse que ϕ et $\dot{\psi}$ sont proportionnels, et négligeant les termes a_2 et a_4 dans (3.60) (ce qui est raisonnable, comme on va le voir juste après), donne :

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\psi} \\ \dot{\psi} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} a_1 & 0 & 0 & 0 \\ a_3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a_5 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\psi} \\ \psi \\ Y \end{bmatrix} + \begin{bmatrix} b_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \delta_l. \quad (3.61)$$

Les valeurs numériques des paramètres des matrices sont données dans l'annexe D.2.

Avec (3.61), on voit que la commande de gauchissement agit sur la vitesse de roulis $\dot{\phi}$. L'angle de roulis généré induit alors un changement de cap, et donc modifie $\dot{\psi}$. En l'absence de vent, lorsque le cap n'est pas nul, l'avion ne vole pas dans l'axe de la piste, ce qui induit donc une modification de l'écart latéral Y (voir figures 3.17).

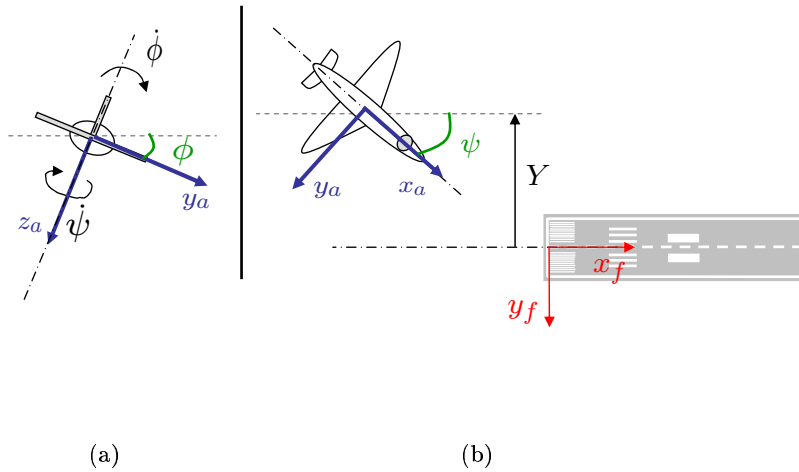


FIG. 3.17 – Paramètres de l'état lors du mouvement latéral de l'avion : (a) vue arrière, (b) vue de dessus.

La figure 3.18 montre le comportement latéral de l'avion, en réponse à deux échelons symétriques successifs de commande. De la mesure de la position latérale Y , on a déduit la dérivée \dot{Y} . On voit sur la figure 3.18 que l'évolution au cours du temps de la dérivée de la position latérale \dot{Y} a la même forme que l'angle de lacet ψ . En dérivant à nouveau, on obtient $\ddot{\psi}$, et on voit que son évolution est similaire à l'évolution de l'angle de roulis ϕ . Enfin, selon le même mécanisme, on constate sur la figure 3.18 que $\dot{\phi}$ évolue comme l'entrée de commande δ_l . Il semble donc raisonnable d'approcher la relation entre la position latérale Y et l'entrée de commande δ_l par une succession de dérivations et de

gains proportionnels (ce qui est équivalent à une succession d'intégrateurs et de gains proportionnels, pour passer de δ_l à Y).

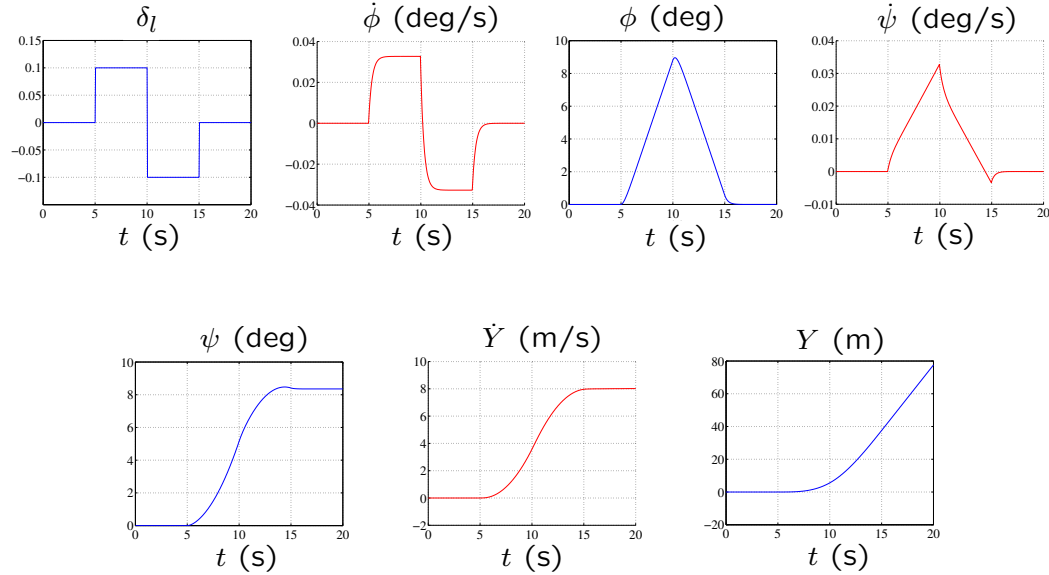


FIG. 3.18 – Réponse du simulateur à la commande de gauchissement.

Après l'avoir illustré avec la réponse du simulateur (figure 3.18), nous allons montrer à présent à partir de l'équation du modèle latéral (3.61) que celle-ci peut être vue comme une cascade d'intégrateurs, et représentée sous forme de schéma bloc (voir figure 3.19).

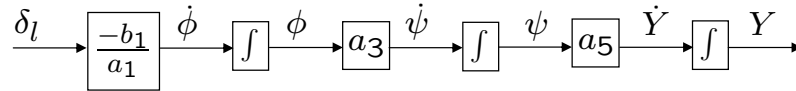


FIG. 3.19 – Schéma bloc du modèle latéral.

- Comme on vient de le voir, la position latérale du manche commande la vitesse de roulis (voir figure 3.18) [Nelson 98]. On peut retrouver cette relation avec la première ligne de (3.61) :

$$\ddot{\phi}(t) = a_1 \dot{\phi}(t) + b_1 \delta_l. \quad (3.62)$$

En supposant que δ_l est constant, et en intégrant cette équation différentielle, on obtient :

$$\dot{\phi} = c_1 e^{a_1 t} - \frac{b_1}{a_1} \delta_l \quad (3.63)$$

avec c_1 constant. Comme $a_1 < 0$ (voir annexe D.2), on peut négliger le terme en exponentielle décroissante [Nelson 98], et on obtient

$$\dot{\phi} = -\frac{b_1}{a_1}\delta_l. \quad (3.64)$$

- Avec la deuxième ligne de (3.61), on a

$$\ddot{\psi} = a_3\dot{\phi}, \quad (3.65)$$

ce qui donne en intégrant cette relation :

$$\dot{\psi} = a_3\phi. \quad (3.66)$$

- Avec la dernière ligne de (3.61), on a directement

$$\dot{Y} = a_5\psi. \quad (3.67)$$

On peut finalement représenter le modèle latéral par le schéma bloc donné sur la figure 3.19.

Modèle longitudinal. On écrit le modèle longitudinal sous la forme suivante :

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\gamma} \\ \dot{v}_a \\ \ddot{v}_a \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ a_6 & a_7 & 0 & a_8 & 0 & 0 \\ a_9 & 0 & a_{10} & a_{11} & a_{12} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ a_{13} & a_{14} & a_{15} & a_{16} & a_{17} & 0 \\ 0 & 0 & a_{18} & a_{19} & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \gamma \\ v_a \\ \dot{v}_a \\ Z \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_3 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & b_4 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix}. \quad (3.68)$$

Les valeurs numériques des paramètres des matrices sont données dans l'annexe D.2.

Comme précédemment avec Y , on a ajouté ici le paramètre Z dans le vecteur d'état, afin de tenir compte de l'altitude de l'avion par rapport au sol. On a modélisé son évolution par $\dot{Z} = a_{19}v_a + a_{18}\gamma = -\sin\gamma_0 v_a - v_{a0} \cos\gamma_0 \gamma$.

On voit alors que la commande de profondeur δ_m agit sur la vitesse de tangage $\dot{\theta}$, et que la commande de la poussée des moteurs δ_T agit sur l'accélération \dot{v}_a . La pente γ et l'altitude Z sont également modifiées (voir figure 3.20). On retrouve également des paramètres du même type dans l'équation générale du modèle longitudinal (3.14) : les composantes de la vitesse dans le plan vertical, et le tangage.

3.2.3 Passage au repère de la caméra

Pour réaliser l'asservissement visuel, on considère une caméra embarquée sur l'avion. Il est alors nécessaire d'établir le lien entre le repère lié à la caméra et celui lié à l'avion. On suppose que la caméra est orientée vers l'avant de l'avion : son axe optique a la même direction que l'axe longitudinal de l'avion. Elle est translatée des distances D_1 et D_2 par rapport au centre de gravité de l'avion (voir figure 3.21).

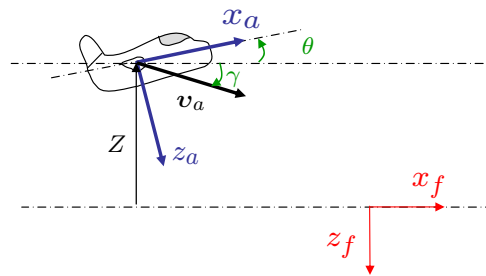


FIG. 3.20 – Paramètres de l'état lors du mouvement longitudinal de l'avion.

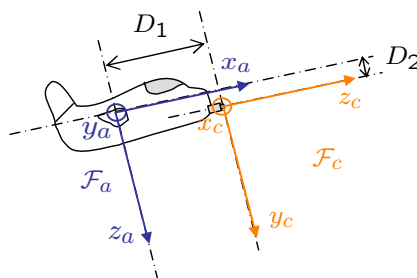


FIG. 3.21 – Positionnement de la caméra sur l'avion.

On passe du repère de la caméra \mathcal{F}_c au repère lié à l'avion \mathcal{F}_a avec la matrice homogène de changement de repère ${}^a\mathbf{M}_c$ telle que :

$${}^a\mathbf{M}_c = \begin{bmatrix} 0 & 0 & 1 & D_1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & D_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.69)$$

En extrayant les parties rotation et translation de cette matrice, on a :

$${}^a\mathbf{R}_c = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{et} \quad {}^a\mathbf{t}_c = \begin{bmatrix} D_1 \\ 0 \\ D_2 \end{bmatrix}. \quad (3.70)$$

On passe alors du torseur cinématique \mathbf{v}_c exprimé dans le repère \mathcal{F}_c au torseur cinématique \mathbf{v}_a exprimé dans le repère \mathcal{F}_a par

$$\mathbf{v}_a = {}^a\mathbf{\Pi}_c \mathbf{v}_c \quad (3.71)$$

avec [Khalil 99] :

$${}^a\mathbf{\Pi}_c = \begin{bmatrix} {}^a\mathbf{R}_c & [{}^a\mathbf{t}_c]_{\times} {}^a\mathbf{R}_c \\ \mathbf{0}_{3 \times 3} & {}^a\mathbf{R}_c \end{bmatrix}. \quad (3.72)$$

On en déduit immédiatement que pour l'opération inverse (passage du torseur cinématique \mathbf{v}_a exprimé dans le repère \mathcal{F}_a au torseur cinématique \mathbf{v}_c exprimé dans le repère \mathcal{F}_c , avec la matrice ${}^c\mathbf{\Pi}_a$), on a

$$\mathbf{v}_c = {}^c\mathbf{\Pi}_a \mathbf{v}_a, \quad (3.73)$$

avec :

$${}^c\mathbf{\Pi}_a = \begin{bmatrix} {}^c\mathbf{R}_a & [{}^c\mathbf{t}_a]_{\times} {}^c\mathbf{R}_a \\ \mathbf{0}_{3 \times 3} & {}^c\mathbf{R}_a \end{bmatrix} \quad (3.74)$$

où

$${}^c\mathbf{R}_a = {}^a\mathbf{R}_c^{\top} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \quad {}^c\mathbf{t}_a = -{}^c\mathbf{R}_a {}^a\mathbf{t}_c = \begin{bmatrix} 0 \\ -D_2 \\ -D_1 \end{bmatrix} \quad (3.75)$$

et

$$[{}^c\mathbf{t}_a]_{\times} {}^c\mathbf{R}_a = \begin{bmatrix} -D_2 & 0 & D_1 \\ 0 & -D_1 & 0 \\ 0 & D_2 & 0 \end{bmatrix}. \quad (3.76)$$

En pratique, on a négligé les termes D_1 et D_2 dans la modélisation. Comme on le verra à la section 3.5.2.3, les résultats obtenus en positionnant la caméra telle que $D_1 =$

$D_2 = 0$ (et en supposant bien entendu que l'on voit la piste) ne sont pas significativement différents de ceux où D_1 et D_2 sont de l'ordre de quelques mètres. Nous avons donc choisi de négliger ces termes, qui introduisent des couplages entre les vitesses de translation et de rotation, afin de simplifier la modélisation.

On a donc considéré :

$${}^c\Pi_a = \begin{bmatrix} {}^c\mathbf{R}_a & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & {}^c\mathbf{R}_a \end{bmatrix} \quad \text{et} \quad {}^a\Pi_c = \begin{bmatrix} {}^c\mathbf{R}_a^\top & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & {}^c\mathbf{R}_a^\top \end{bmatrix} \quad (3.77)$$

3.3 Modélisation d'informations visuelles

3.3.1 Objectif

On cherche à choisir, à partir de l'image de la piste obtenue par la caméra embarquée, des informations visuelles intéressantes compte tenu du modèle de l'avion. Le modèle étant sous la forme de deux parties découplées (modèle latéral (3.61) et modèle longitudinal (3.68)), il est par exemple intéressant de faire en sorte que les informations visuelles liées au mouvement latéral soient découplées des informations visuelles liées au mouvement longitudinal. De plus, il est intéressant de choisir, pour chaque type de mouvement (latéral et longitudinal), des informations visuelles liées à un seul paramètre de l'état, afin de ne pas introduire de nouveaux couplages.

3.3.2 Informations visuelles utilisées

On utilise les lignes délimitant les bords de la piste et marquant son centre. On suppose qu'elles sont espacées de la largeur $L/2$ (voir figure 3.22.a). Elles sont parallèles dans l'espace 3D, et se coupent au point de fuite dans l'image (voir figure 3.22.b). Du point de vue de la modélisation, on ne tient pas compte de leur longueur, et on les considère donc comme trois droites parallèles.

Différentes approches peuvent être envisagées pour réaliser l'asservissement visuel par rapport à des droites. L'utilisation de la représentation de droites sous forme de coordonnées de Plücker binormées [Faugeras 93] est proposée dans [Andreff 02]. Les propriétés de cette représentation sont exploitées pour concevoir un asservissement visuel adapté à la dynamique d'un hélicoptère dans [Mahony 05a].

Plus classiquement, la représentation d'une droite par les paramètres (ρ, θ) , définissant les points (x, y) de la droite considérée par la relation $x \cos \theta + y \sin \theta - \rho = 0$ est aussi utilisée en asservissement visuel [Espiau 92]. Cette représentation est notamment mise en œuvre dans [Rives 02] pour représenter les lignes de la piste d'atterrissage et réaliser l'atterrissage d'un petit avion par asservissement visuel. Les deux paramètres de chacune des trois lignes sont "empilés" pour former le vecteur d'erreur de l'asservissement. Les lignes étant parallèles, seuls cinq ddls peuvent être commandés : il est donc nécessaire de considérer un capteur supplémentaire afin de réguler la vitesse selon l'axe de la piste. Dans [Silveira 03], les paramètres (ρ, θ) de deux lignes de l'image sont utilisés pour permettre à un dirigeable de suivre les lignes correspondantes dans l'espace 3D. Ces paramètres permettent de commander quatre ddls : deux autres capteurs sont

nécessaires pour commander l'ensemble des six ddls de l'engin. Dans [Rives 04], l'objectif est également de suivre des structures linéaires, telles que routes, rivières, etc. avec un dirigeable. Deux lignes sont donc utilisées, et des informations visuelles découplées sont modélisées : les ddls de rotation sont commandés à partir des coordonnées du point de fuite et de l'orientation de la ligne d'horizon dans l'image, et les ddls de translation sont commandés à partir de combinaisons des paramètres θ des lignes de l'image. Là encore, seuls cinq ddls sont commandés par vision : la commande de la vitesse selon l'axe des lignes n'est pas considérée dans le cadre de l'asservissement visuel.

Dans notre travail, on a choisi d'utiliser pour chacune des trois droites du plan image la représentation (ρ, θ) (voir figure 3.22.b). Comme dans [Rives 04], on cherche ensuite à combiner ces informations visuelles "de base" pour modéliser des informations visuelles plus adaptées à la dynamique de l'avion. On a également utilisé les coordonnées (x_f, y_f) du point de fuite (voir figure 3.22.b).

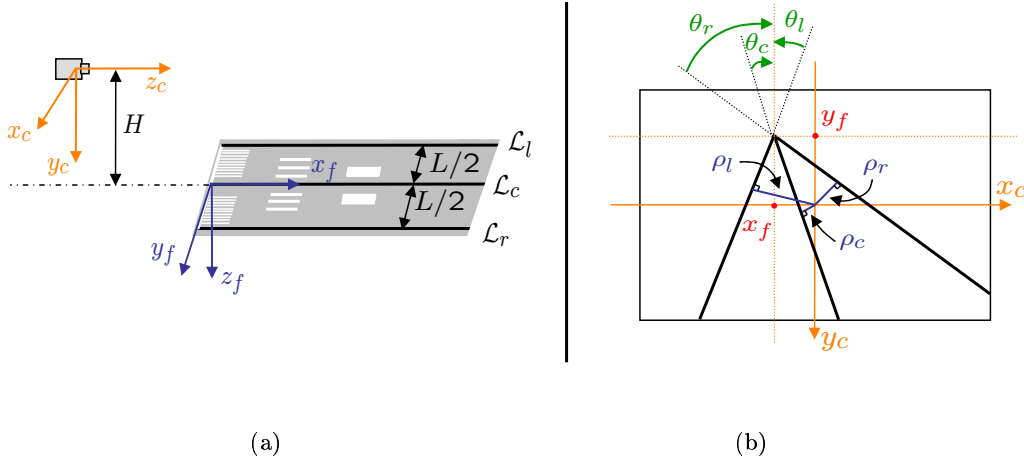


FIG. 3.22 – Positionnement de la caméra par rapport à la piste (a), et projection des lignes de la piste sur le plan image (b).

Droites. Une droite de l'espace 3D peut être définie dans le repère de la caméra \mathcal{F}_c par l'intersection de deux plans :

$$\begin{cases} A_1X + B_1Y + C_1Z + D_1 = 0 & (h_1) \\ A_2X + B_2Y + C_2Z + D_2 = 0 & (h_2). \end{cases} \quad (3.78)$$

La matrice d'interaction associée au paramètre θ de la projection de cette ligne dans le plan image est alors donnée par [Espiau 92] :

$$\mathbf{L}_\theta = [\lambda \cos \theta \quad \lambda \sin \theta \quad -\lambda \rho \quad -\rho \cos \theta \quad -\rho \sin \theta \quad -1], \quad (3.79)$$

où $\lambda = (A_i \sin \theta - B_i \cos \theta) / D_i$,
et $i = 1$ si $D_1 \neq 0$ ou $i = 2$ si $D_2 \neq 0$.

Notons que l'on peut toujours choisir $D_i \neq 0$, sauf dans le cas dégénéré où $D_1 = D_2 = 0$, qui correspond au cas où le centre de projection de la caméra appartient à la droite 3D (ce qui n'est pas le cas en pratique pour notre application).

Par exemple, on peut choisir le plan passant par le centre optique de la caméra et le plan qui lui est orthogonal. Cela permet d'obtenir des équations de plan de la forme

$$\begin{cases} A_1X + B_1Y + C_1Z + D_1 = 0 & (h_1) \\ A_2X + B_2Y + C_2Z = 0 & (h_2) \end{cases} \quad (3.80)$$

L'équation de la droite du plan image est alors donnée par $A_2x + B_2y + C_2 = 0$. La paramétrisation de cette droite avec la représentation (ρ, θ) est alors telle que [Chaumette 90] :

$$\theta = \arctan\left(\frac{B_2}{A_2}\right) \quad (3.81)$$

$$\rho = \frac{C_2}{\sqrt{A_2^2 + B_2^2}}. \quad (3.82)$$

Dans notre cas, on peut également définir les lignes de la piste comme l'intersection du plan horizontal et de plans verticaux, ou — pour la position désirée choisie — comme l'intersection d'un plan passant par l'axe optique et d'un plan qui lui est orthogonal.

On considère la caméra à sa position désirée, c'est-à-dire dans l'axe de la piste, à l'altitude H , le plan image (défini par $(\mathbf{x}_c, \mathbf{y}_c)$) étant vertical, et l'axe des abscisses de l'image \mathbf{x}_c étant horizontal, comme sur la figure 3.22.a.

On note (ρ_l, θ_l) , (ρ_c, θ_c) et (ρ_r, θ_r) les paramètres (ρ, θ) des trois droites de l'image : (ρ_l, θ_l) pour la ligne de gauche, (ρ_c, θ_c) pour celle du milieu, et (ρ_r, θ_r) pour celle de droite (voir figure 3.22.b).

Considérons par exemple les plans passant par l'axe optique de la caméra — située à la position désirée choisie — et dont la droite 3D considérée fait partie. Ces plans ont pour équation dans le repère désiré de la caméra, respectivement $HX + (L/2)Y = 0$ pour la ligne de gauche, $HX - (L/2)Y = 0$ pour la ligne de droite, et $X = 0$ (c'est-à-dire le plan vertical) pour la ligne du milieu. Avec (3.81), on peut alors écrire les relations entre l'altitude H et la largeur de la piste L :

$$\theta_l = \arctan\left(\frac{L}{2H}\right) \quad (3.83)$$

$$\theta_r = -\arctan\left(\frac{L}{2H}\right) \quad (3.84)$$

d'où on déduit :

$$H = \frac{L}{2 \tan\left(\frac{\theta_l - \theta_r}{2}\right)}. \quad (3.85)$$

- Considérons à présent le plan du sol, horizontal. Son équation dans le repère désiré de la caméra est $Y - H = 0$, c'est-à-dire $AX + BY + CZ + D = 0$ avec

$(A, B, C, D) = (0, 1, 0, -H)$. Les termes λ_l , λ_r , et λ_c intervenant respectivement dans les matrices d'interaction associées aux trois droites de l'image gauche, droite, et celle du milieu s'écrivent alors (voir équation (3.79)) :

$$\lambda_l = \frac{1}{H} \cos \theta_l, \quad \lambda_r = \frac{1}{H} \cos \theta_r, \quad \lambda_c = \frac{1}{H} \cos \theta_c. \quad (3.86)$$

- Au lieu du plan horizontal, on peut également choisir d'utiliser les plans verticaux pour calculer λ . Ils ont pour équation $X + L/2 = 0$ (on a donc $(A, B, C, D) = (1, 0, 0, L/2)$) pour \mathcal{L}_l , et $X - L/2 = 0$ (on a donc $(A, B, C, D) = (1, 0, 0, -L/2)$) pour \mathcal{L}_r . En ce qui concerne la ligne du milieu \mathcal{L}_c , le plan vertical correspondant a pour équation $X = 0$ (on a donc $(A, B, C, D) = (1, 0, 0, 0)$). Le paramètre D étant nul dans ce cas, on n'utilisera donc pas le plan vertical pour calculer λ_c . Pour les droites gauche et droite, utiliser les plans verticaux plutôt que le plan horizontal fait intervenir la largeur de la piste L au lieu de l'altitude de la caméra H (qui varie pendant l'atterrissage) dans les expressions de λ :

$$\lambda_l = \frac{2}{L} \sin \theta_l, \quad \lambda_r = -\frac{2}{L} \sin \theta_r. \quad (3.87)$$

Les valeurs désirées des paramètres des droites dans l'image sont (voir figure 3.23) :

$$\rho_c^* = \rho_l^* = \rho_r^* = 0 \quad \text{et} \quad \theta_c^* = 0. \quad (3.88)$$

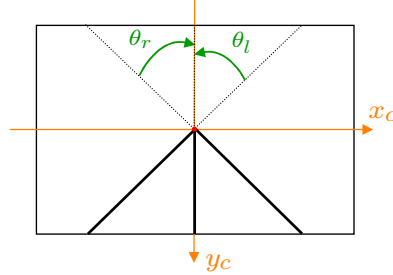


FIG. 3.23 – Projection des lignes de la piste sur le plan image, à la position désirée.

On obtient alors, en remplaçant les paramètres (ρ, θ) par leurs valeurs désirées dans (3.79) :

$$\mathbf{L}_{\theta_l^*} = [\lambda_l^* \cos \theta_l \quad \lambda_l^* \sin \theta_l \quad 0 \quad 0 \quad 0 \quad -1] \quad (3.89)$$

$$\mathbf{L}_{\theta_r^*} = [\lambda_r^* \cos \theta_r \quad \lambda_r^* \sin \theta_r \quad 0 \quad 0 \quad 0 \quad -1] \quad (3.90)$$

$$\mathbf{L}_{\theta_c^*} = [\lambda_c^* \cos \theta_c \quad \lambda_c^* \sin \theta_c \quad 0 \quad 0 \quad 0 \quad -1]. \quad (3.91)$$

où, avec (3.86),

$$\lambda_l^* = \frac{1}{H} \cos \theta_l, \quad \lambda_r^* = \frac{1}{H} \cos \theta_r, \quad \lambda_c^* = \frac{1}{H}. \quad (3.92)$$

De même, (3.87) donne directement les valeurs de λ_l^* et λ_r^* utilisant L :

$$\lambda_l^* = \frac{2}{L} \sin \theta_l, \quad \lambda_r^* = -\frac{2}{L} \sin \theta_r. \quad (3.93)$$

Afin d'obtenir des matrices d'interaction creuses (avec de nombreux termes nuls) et donc, autant que possible d'associer une information visuelle à un ddl, on choisit les informations visuelles δ et σ telles que :

$$\delta = \theta_l + \theta_r - 2\theta_c, \quad (3.94)$$

$$\sigma = \theta_l - \theta_r. \quad (3.95)$$

Avec (3.79) et (3.86) on obtient alors la matrice d'interaction associée à δ :

$$\mathbf{L}_\delta = \begin{bmatrix} \frac{1}{H}(c^2\theta_l + c^2\theta_r - 2c^2\theta_c) & \frac{1}{H}(c\theta_l s\theta_l + c\theta_r s\theta_r - 2c\theta_c s\theta_c) & -\frac{1}{H}(\rho_l c\theta_l + \rho_r c\theta_r - 2\rho_c c\theta_c) \\ -\rho_l c\theta_l - \rho_r c\theta_r + 2\rho_c c\theta_c & -\rho_l s\theta_l - \rho_r s\theta_r + 2\rho_c s\theta_c & 0 \end{bmatrix}.$$

En considérant les matrices d'interaction désirées (3.89), (3.90) (3.91) et avec (3.92), on obtient la matrice d'interaction associée à δ^* :

$$\mathbf{L}_{\delta^*} = \begin{bmatrix} \frac{1}{H}(\cos^2\theta_l + \cos^2\theta_r - 2) & \frac{1}{H}(\cos\theta_l \sin\theta_l + \cos\theta_r \sin\theta_r) & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Comme $\theta_l = -\theta_r$ à la position désirée (voir figure 3.23), on a $c\theta_l s\theta_l + c\theta_r s\theta_r = -c\theta_r s\theta_r + c\theta_r s\theta_r = 0$. Finalement, on obtient pour la matrice d'interaction désirée associée à δ :

$$\mathbf{L}_{\delta^*} = \begin{bmatrix} \frac{1}{H}(\cos^2\theta_l + \cos^2\theta_r - 2) & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.96)$$

Avec (3.79) et (3.87) on obtient de même la matrice d'interaction associée à σ :

$$\mathbf{L}_\sigma = \begin{bmatrix} \frac{2}{L}(s\theta_l c\theta_l + s\theta_r c\theta_r) & \frac{2}{L}(s^2\theta_l + s^2\theta_r) & -\frac{2}{L}(\rho_l s\theta_l - \rho_r s\theta_r) \\ -\rho_l c\theta_l + \rho_r c\theta_r & -\rho_l s\theta_l + \rho_r s\theta_r & 0 \end{bmatrix}.$$

En considérant les matrices d'interaction désirées (3.89), (3.90) (3.91) et avec (3.93), on obtient la matrice d'interaction associée à σ^* :

$$\mathbf{L}_{\sigma^*} = \begin{bmatrix} \frac{2}{L}(\sin\theta_l \cos\theta_l + \sin\theta_r \cos\theta_r) & \frac{2}{L}(\sin^2\theta_l + \sin^2\theta_r) & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Et comme précédemment, avec $\theta_l = -\theta_r$, on en déduit la matrice d'interaction désirée associée à σ :

$$\mathbf{L}_{\sigma^*} = \begin{bmatrix} 0 & \frac{2}{L}(\sin^2\theta_l + \sin^2\theta_r) & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.97)$$

L'intérêt de ce choix d'informations visuelles réside dans l'obtention des matrices d'interaction creuses (3.96) et (3.97), qui vont permettre, comme on va le voir par la suite, d'obtenir par la vision des informations sur la position latérale de l'avion par rapport à la piste (à partir de l'information visuelle δ), et sur son altitude (à partir de l'information visuelle σ). Dans [Rives 02], les paramètres (ρ, θ) de trois droites sont utilisés directement dans le vecteur d'informations visuelles considéré. Nos choix d'informations visuelles découplées sont proches de ceux proposés dans [Rives 04]. Cependant, au lieu de l'information visuelle $\frac{\theta_r + \theta_l}{2}$, nous proposons $\delta = \theta_r + \theta_l - 2\theta_c$. Cela nous permet d'annuler le dernier terme de la matrice d'interaction, et ainsi d'obtenir une information visuelle découplée (à la position désirée) de la rotation autour de l'axe optique. Concernant $\sigma = \theta_l - \theta_r$, elle est identique à celle proposée dans [Rives 04]. Cependant, nous avons choisi d'écrire sa matrice d'interaction en fonction de la largeur de la piste, pour ne pas y faire intervenir l'altitude de la caméra. En effet, σ étant destinée à commander l'altitude, on a préféré ne pas faire intervenir l'altitude dans sa matrice d'interaction associée.

Point de fuite. L'utilisation des coordonnées du point de fuite en asservissement visuel a été proposée dans [Rives 04]. Il s'agit de la projection sur le plan image d'un point situé à l'infini dans l'espace 3D. Son mouvement dans l'image est indépendant des translations de la caméra, il n'est fonction que des mouvements de rotation [Rives 04]. Cette propriété est particulièrement intéressante lorsque l'on souhaite utiliser des informations visuelles découplées.

Les matrices d'interaction associées aux coordonnées du point de fuite peuvent être obtenues à partir de la matrice d'interaction pour un point (1.30), en faisant tendre la distance Z vers l'infini. On obtient :

$$\mathbf{L}_{x_f} = \begin{bmatrix} 0 & 0 & 0 & x_f y_f & -(1 + x_f^2) & y_f \end{bmatrix} \quad (3.98)$$

$$\mathbf{L}_{y_f} = \begin{bmatrix} 0 & 0 & 0 & (1 + y_f^2) & -x_f y_f & -x_f \end{bmatrix}. \quad (3.99)$$

À la position désirée décrite précédemment (voir figure 3.23), le point de fuite se situe au centre de l'image. Ses coordonnées sont $(x_f, y_f) = (0, 0)$. On en déduit l'expression des matrices d'interaction calculées à la position désirée :

$$\mathbf{L}_{x_f^*} = \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix} \quad (3.100)$$

$$\mathbf{L}_{y_f^*} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \quad (3.101)$$

Relations avec le mouvement de l'avion. Les matrices d'interaction \mathbf{L}_{δ^*} (3.96), \mathbf{L}_{σ^*} (3.97), $\mathbf{L}_{x_f^*}$ (3.100) et $\mathbf{L}_{y_f^*}$ (3.101) lient le torseur cinématique de la caméra \mathbf{v}_c aux variations des informations visuelles δ , σ , x_f et y_f (voir (1.12), section 1.1.1.2). En utilisant (3.73) pour remplacer \mathbf{v}_c en fonction du torseur cinématique de l'avion \mathbf{v}_a , on obtient :

$$\dot{s} \simeq \mathbf{L}_{s^*}{}^c \mathbf{\Pi}_a \mathbf{v}_a \quad (3.102)$$

où s est une information visuelle. Puisque \mathbf{L}_{s^*} et ${}^c\Pi_a$ sont constants, on peut intégrer cette expression, pour de petites variations autour de la position nominale considérée [Azinheira 02, Gangloff 00]. On obtient avec (3.58) :

$$\tilde{s} \simeq \mathbf{L}_{s^*} {}^c\Pi_a \tilde{\mathbf{P}}. \quad (3.103)$$

où la notation $\tilde{}$ représente les variations entre les valeurs courantes et les valeurs nominales, comme à la section 3.2.2. De la même manière, pour simplifier les notations, on omettra le $\tilde{}$ dans la suite, bien que l'on considérera encore les variations des paramètres par rapport à la position nominale.

En remplaçant \mathbf{L}_{s^*} par les matrices d'interaction associées aux informations visuelles δ^* (3.96), σ^* (3.97), x_f^* (3.100) et y_f^* (3.101), et en utilisant l'expression de ${}^c\Pi_a$ (3.77), on obtient :

$$\delta \simeq \frac{1}{H}(\cos^2\theta_l + \cos^2\theta_r - 2)Y \quad (3.104)$$

$$\sigma \simeq \frac{2}{L}(\sin^2\theta_l + \sin^2\theta_r)Z \quad (3.105)$$

$$x_f \simeq -\psi \quad (3.106)$$

$$y_f \simeq \theta. \quad (3.107)$$

De plus, en dérivant (3.106) et (3.107), on obtient :

$$\dot{x}_f \simeq -\dot{\psi} \quad (3.108)$$

$$\dot{y}_f \simeq \dot{\theta}. \quad (3.109)$$

Notons que l'on peut a priori rencontrer deux cas de singularité, avec (3.104) et (3.105) :

- $H \rightarrow 0$: en pratique, cela n'arrive pas puisque l'altitude de la caméra ne sera pas nulle (à l'atterrissage, les roues de l'avion touchent la piste, mais pas la caméra).
- $(\theta_l, \theta_r) \rightarrow (k_1\pi, k_2\pi)$: les droites sont parallèles dans le plan image, voir figure 3.24). Notons que le point de fuite est alors rejeté à l'infini, ce qui pose également un problème dans (3.106), (3.107) et (3.108), (3.109) : cela pourrait se produire dans une situation où l'axe optique de la caméra est vertical, ou si la caméra était rejetée à une altitude infinie. Dans le cadre de l'application considérée, on ne risque bien évidemment pas de rencontrer ces situations.

L'inversion des équations (3.104) à (3.109) donne une relation directe entre les paramètres de l'état de l'avion et les informations visuelles :

$$Y \simeq \frac{H}{\cos^2\theta_l + \cos^2\theta_r - 2}\delta, \quad (3.110)$$

$$Z \simeq \frac{L}{2(\sin^2\theta_l + \sin^2\theta_r)}\sigma \quad (3.111)$$

$$\psi \simeq -x_f \quad (3.112)$$

$$\theta \simeq y_f \quad (3.113)$$

$$\dot{\psi} \simeq -\dot{x}_f \quad (3.114)$$

$$\dot{\theta} \simeq \dot{y}_f. \quad (3.115)$$

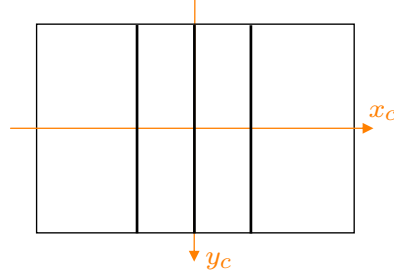


FIG. 3.24 – Projection des lignes de la piste sur le plan image : cas particulier où $(\theta_l, \theta_r) \rightarrow (k_1\pi, k_2\pi)$.

Notons que l'on a par ailleurs, une autre expression de Z , avec (3.85) :

$$Z \simeq -H = -\frac{L}{2 \tan\left(\frac{\sigma}{2}\right)}. \quad (3.116)$$

On retrouve bien, comme dans (3.111) que l'information visuelle σ est liée essentiellement à l'altitude Z . Avec (3.116), on a une singularité lorsque $\tan\left(\frac{\sigma}{2}\right) = \tan\left(\frac{\theta_l - \theta_r}{2}\right) \rightarrow 0$, ce qui correspond au cas où $\theta_r = \theta_l$. Les droites sont alors parallèles dans l'image. Comme déjà dit, cela correspond à une configuration où la caméra est à une altitude infinie, ou alors à une configuration où l'axe optique est vertical, ce qui n'arrive pas en pratique dans le cadre de l'application considérée. Un autre cas de singularité apparaît dans (3.116) lorsque $\frac{\sigma}{2} \rightarrow \frac{\pi}{2}$, c'est-à-dire $\theta_l \rightarrow \theta_r + \pi$. Cela correspond à une configuration où les droites sont quasi superposées sur le plan image, lorsque l'altitude de la caméra tend vers zéro. Cela ne nous pose pas problème, car comme déjà dit précédemment, la caméra reste à une altitude suffisante dans notre application.

Du modèle latéral, on a extrait la relation suivante (3.66) : $\dot{\psi} = a_3\phi$, d'où on déduit, avec (3.114) :

$$\phi = -\frac{1}{a_3}\dot{x}_f. \quad (3.117)$$

De même, du modèle longitudinal (3.68), on extrait la relation suivante :

$$\dot{Z} = a_{18}\gamma + a_{19}v_a,$$

d'où on déduit une expression de γ :

$$\gamma = \frac{1}{a_{18}}\dot{Z} - \frac{a_{19}}{a_{18}}v_a. \quad (3.118)$$

Pour exprimer \dot{Z} , on a envisagé deux possibilités :

- soit repartir de la matrice d'interaction de σ^* (3.97), ce qui donne avec (3.102) et (3.77) :

$$\dot{Z} \simeq \frac{L}{2} \frac{1}{(\sin^2\theta_l + \sin^2\theta_r)} \dot{\sigma}. \quad (3.119)$$

- soit dériver (3.116) par rapport au temps, et obtenir ainsi :

$$\dot{Z} \simeq \frac{L}{2} \frac{1}{2 \sin^2(\frac{\sigma}{2})} \dot{\sigma}. \quad (3.120)$$

On remarque que les deux expressions sont très proches : elles lient \dot{Z} à $\dot{\sigma}$, avec un terme de proportionnalité en $L/2$, et l'inverse de :

- $\sin^2\theta_l + \sin^2\theta_r$, la somme des carrés des sinus des angles (3.119),
- $\sin^2(\frac{\sigma}{2}) = \sin^2(\frac{\theta_l - \theta_r}{2})$, le sinus au carré de la demi différence des angles dans (3.120).

En pratique, on a utilisé dans la suite (3.116) et (3.120), qui ont donné de meilleurs résultats que les expressions (3.111) et (3.119). Cependant, notons que l'estimation de γ obtenue en combinant (3.118) et (3.120) dépend de la dérivée de l'information visuelle σ , ce qui, comme nous le verrons par la suite, peut produire en pratique un signal très perturbé.

Ces relations entre paramètres de l'état de l'avion et informations visuelles seront exploitées dans la section suivante pour mettre au point une loi de commande par asservissement visuel.

3.4 Commande

3.4.1 Loïs de commande

Ayant considéré un modèle de l'avion sous forme découplée (modèle latéral et modèle longitudinal, voir section 3.2.2), des lois de commande séparées sont également proposées. L'une est conçue à partir du modèle latéral pour commander les mouvements de l'avion dans le plan horizontal (commande latérale), et l'autre, conçue à partir du modèle longitudinal, permet de commander les mouvements de l'avion dans le plan vertical (modèle longitudinal).

3.4.1.1 Commande latérale

Première loi de commande. Au vu du modèle latéral (figures 3.18 et 3.19) on constate qu'en l'absence de vent, lorsque l'erreur latérale Y est stabilisée à 0, l'angle de lacet ψ est également stabilisé à 0, et cela implique que l'angle de roulis ϕ est aussi nécessairement nul (voir figures 3.18 et 3.25). Notre idée consiste alors à concevoir une loi de commande en liant l'entrée de commande δ_l à la position latérale Y , afin de stabiliser la position Y à 0. Avec (3.64) et (3.65), on a :

$$\ddot{\psi} = -\frac{a_3 b_1}{a_1} \delta_l. \quad (3.121)$$

Et en dérivant une fois (3.67), on obtient

$$\dot{Y} = a_5 \dot{\psi}, \quad (3.122)$$

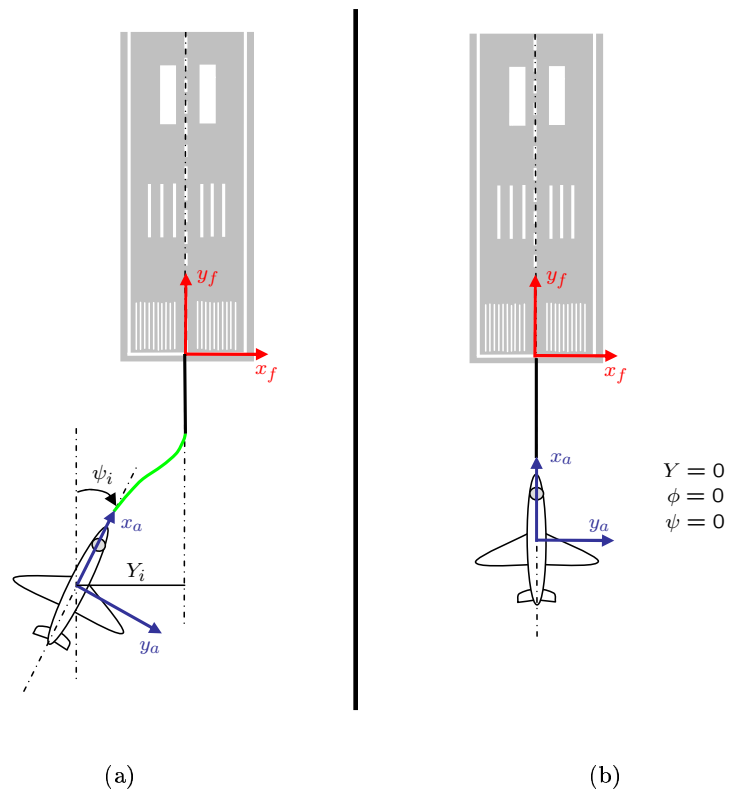


FIG. 3.25 – Alignement de l'avion dans l'axe de la piste au cours de l'atterrissage : position initiale (a) et désirée (b).

et en dérivant à nouveau, on a :

$$\ddot{Y} = a_5 \ddot{\psi}. \quad (3.123)$$

En combinant (3.121) et (3.123), on a finalement :

$$\ddot{Y} = -\frac{a_5 a_3 b_1}{a_1} \delta_l, \quad (3.124)$$

d'où on déduit

$$\delta_l = -\frac{a_1}{a_5 a_3 b_1} \ddot{Y}. \quad (3.125)$$

En choisissant \ddot{Y} de la forme $\ddot{Y} = -k_1 \ddot{Y} - k_2 \dot{Y} - k_3 Y$, on obtient la loi de commande suivante :

$$\delta_l = -n_1 \ddot{Y} - n_2 \dot{Y} - n_3 Y \quad (3.126)$$

avec $n_i = (-k_i a_1)/(a_5 a_3 b_1)$, $i \in \{1, 2, 3\}$, et où k_1 , k_2 , et k_3 sont choisis de manière à ce que Y soit stable (on fait en sorte que les racines de l'équation $s^3 + k_1 s^2 + k_2 s + k_3 = 0$ soient négatives).

En injectant dans (3.126) les informations visuelles proposées à la section précédente (3.110), (3.112), (3.114), et en utilisant les relations (3.67) et (3.122), on obtient la loi de commande par asservissement visuel :

$$\delta_l = n_1 a_5 \dot{x}_f + n_2 a_5 x_f - n_3 \frac{H}{c^2 \theta_l + c^2 \theta_r - 2} \delta. \quad (3.127)$$

Outre les informations visuelles x_f , \dot{x}_f , θ_l , θ_r , $\delta = \theta_l + \theta_r - 2\theta_c$ (3.94), l'altitude H intervient ici. On pourra considérer qu'un altimètre est présent dans l'avion et fournit cette mesure. Une autre manière pourra consister estimer H à partir de la connaissance a priori de la largeur L de la piste.

Deuxième loi de commande. Dassault Aviation nous a par ailleurs proposé le schéma de commande suivant :

$$\delta_l = k p_c \quad (3.128)$$

tel que

$$p_c = \omega_3 (\phi_c - \phi) \quad \text{et} \quad p_c \in [-p_m, p_m], \quad (3.129)$$

tel que

$$\phi_c = \omega_2 \frac{v_a}{g} (\psi_c - \psi) \quad \text{et} \quad \phi_c \in [-\phi_m, \phi_m], \quad (3.130)$$

tel que

$$\psi_c = \omega_1 \frac{1}{v_a} (Y_c - Y) \quad \text{et} \quad \psi_c \in [-\psi_m, \psi_m], \quad (3.131)$$

tel que

$$Y_c = 0, \quad (3.132)$$

et avec

$k = 2.9$, $\omega_1 = 0.08$ rad/s, $\omega_2 = 0.2$ rad/s, $\omega_3 = 0.5$ rad/s, $p_m = \pi/9$ rad/s, $\phi_m = \pi/3$ rad, $\psi_m = \pi/2$ rad.

Par ailleurs, on remarque que dans le modèle latéral, $\frac{-b_1}{a_1} \simeq \frac{1}{k}$, $a_3 \simeq \frac{g}{v_a}$, $a_5 \simeq v_a$. Si on utilise ce modèle simple, le schéma de commande proposé devient celui de la figure 3.26.

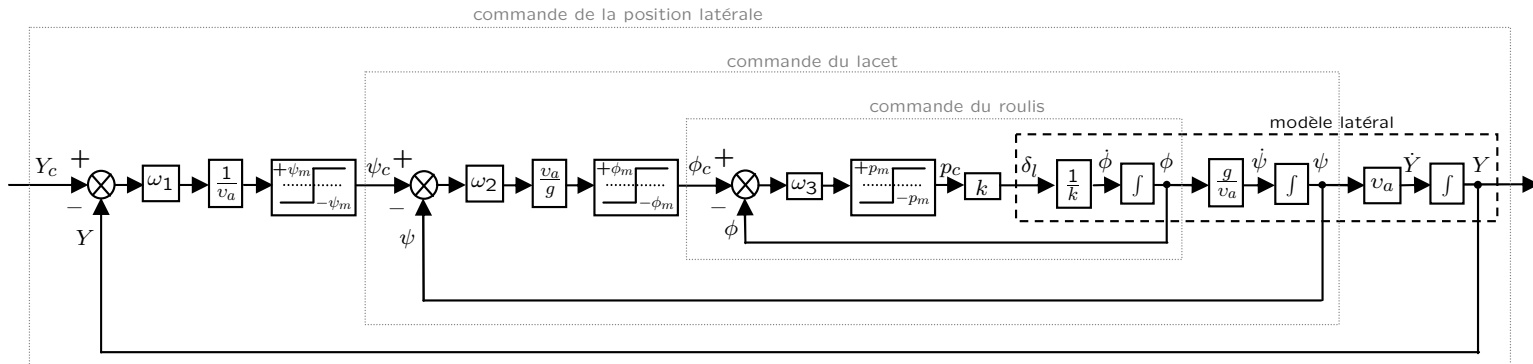


FIG. 3.26 – Schéma de commande latéral.

On propose dans l'annexe E une analyse du système bouclé ainsi obtenu, en termes de fonction de transfert et de réponse du système, montrant l'intérêt du choix proposé ici pour les gains ω_i , $i \in 1, 2, 3$.

Comme précédemment, on peut injecter des informations visuelles dans ce schéma de commande, en utilisant les relations (3.110), (3.112), et (3.117). Au lieu d'utiliser des estimations de Y , ψ et ϕ , la loi de commande repose alors sur des mesures issues de l'image (voir figure 3.27).

$$\delta_l = kp_c$$

tel que

$$p_c = -\omega_3 \frac{v_a}{g} (\dot{x}_{f_c} - \dot{x}_f) \quad \text{et} \quad p_c \in [-p_m, p_m],$$

tel que

$$\dot{x}_{f_c} = \omega_2 (x_{f_c} - x_f) \quad \text{et} \quad \dot{x}_{f_c} \in [-\dot{x}_{f_{cm}}, \dot{x}_{f_{cm}}],$$

tel que

$$x_{f_c} = -\omega_1 \frac{1}{v_a} \frac{H}{c^2 \theta_l + c^2 \theta_r - 2} (\delta_c - \delta) \quad \text{et} \quad x_f \in [-x_{f_{cm}}, x_{f_{cm}}],$$

tel que

$$\delta_c = 0,$$

et avec

$$k = 2.9, \omega_1 = 0.08 \text{ rad/s}, \omega_2 = 0.2 \text{ rad/s}, \omega_3 = 0.5 \text{ rad/s}, p_m = \pi/9 \text{ rad/s}, \dot{x}_{f_{cm}} = \frac{g}{v_a} \phi_m = \frac{g}{v_a} \frac{\pi}{3} \text{ rad}, x_{f_{cm}} = \psi_m = \frac{\pi}{2} \text{ rad}.$$

FIG. 3.27 – Schéma de commande latérale par asservissement visuel.

Remarque :

Supposons que l'on reste dans le domaine linéaire (on ne tient donc pas compte des saturations). En rassemblant les équations (3.128) à (3.132), on peut écrire la loi de

commande sous la forme :

$$\delta_l = k\omega_3 \left(\omega_2 \frac{v_a}{g} \left(\omega_1 \frac{-1}{v_a} Y - \psi \right) - \phi \right) \quad (3.133)$$

$$= -\frac{k\omega_3\omega_2\omega_1}{g} Y - \frac{k\omega_3\omega_2 v_a}{g} \psi - k\omega_3 \phi. \quad (3.134)$$

Et si on utilise alors, comme précédemment, les équations du modèle, en remplaçant ϕ par $v_a/g\dot{\psi}$ ((3.66), avec $a_3 \simeq \frac{g}{v_a}$) puis ψ par \dot{Y}/v_a ((3.67), avec $a_5 \simeq v_a$), et $\dot{\psi}$ par \ddot{Y}/v_a (en dérivant la relation précédente), on a finalement :

$$\delta_l = -\frac{k\omega_3\omega_2\omega_1}{g} Y - \frac{k\omega_3\omega_2}{g} \dot{Y} - \frac{k\omega_3}{g} \ddot{Y} \quad (3.135)$$

qui est de la même forme que la loi de commande proposée initialement (3.126). L'avantage de la formulation avec saturations à chaque étape (3.128) à (3.132) permet explicitement d'empêcher que les consignes en roulis et lacet ne soient trop importantes, évitant ainsi de déstabiliser l'avion.

3.4.1.2 Commande longitudinale.

Pour la partie longitudinale, on n'a pas comme précédemment mis en évidence de modèle simple, avec un lien intuitif entre les différents paramètres de l'état. À partir de l'équation d'évolution de l'état (3.68), on propose donc une loi de commande classique de type LQR (voir section 1.2.3.2 et figure 1.21).

Comme avec la commande latérale, on peut décliner la loi de commande obtenue sous deux formes : une première en supposant que l'on sait mesurer directement les paramètres de l'état de l'avion, puis une autre en substituant des informations visuelles. Les mesures nécessaires sont alors essentiellement issues de l'image.

Notons l'équation d'évolution de l'état (3.68) sous la forme

$$\dot{\mathbf{x}}_{lo} = \mathbf{A}_{lo}\mathbf{x}_{lo} + \mathbf{B}_{lo}\mathbf{u}_{lo} \quad (3.136)$$

où $\mathbf{x}_{lo} = (\theta, \dot{\theta}, \gamma, v_a, \dot{v}_a, Z)$ est le vecteur d'état longitudinal, \mathbf{A}_{lo} la matrice de système, \mathbf{B}_{lo} la matrice d'entrée et $\mathbf{u}_{lo} = (\delta_m, \delta_T)$ le vecteur d'entrées.

La loi de commande optimale par retour d'état est alors de la forme

$$\mathbf{u}_{lo} = \mathbf{K}\mathbf{x}_{lo}, \quad (3.137)$$

Considérons l'équation de sortie suivante :

$$\mathbf{y}_{lo} = \mathbf{I}_6\mathbf{x}_{lo} \quad (3.138)$$

où $\mathbf{y}_{lo} = (\theta, \dot{\theta}, \gamma, v_a, \dot{v}_a, Z)$ représente le vecteur de sortie. La loi de commande (3.137) devient alors :

$$\mathbf{u}_{lo} = \mathbf{K}\mathbf{y}_{lo}. \quad (3.139)$$

En utilisant les relations (3.113), (3.115), (3.111), (3.118) et (3.120) on peut écrire les relations suivantes entre les composantes de \mathbf{y}_{lo} et les informations visuelles y_f , \dot{y}_f et σ :

$$\theta = y_f \quad (3.140)$$

$$\dot{\theta} = \dot{y}_f \quad (3.141)$$

$$\gamma = \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2\left(\frac{\sigma}{2}\right)} - \frac{a_{19}}{a_{18}} v_a \quad (3.142)$$

$$Z = \frac{-L}{2 \tan\left(\frac{\sigma}{2}\right)} \quad (3.143)$$

et la loi de commande (3.139) devient :

$$\begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix} = \mathbf{K} \begin{bmatrix} y_f \\ \dot{y}_f \\ \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2\left(\frac{\sigma}{2}\right)} - \frac{a_{19}}{a_{18}} v_a \\ v_a \\ \dot{v}_a \\ \frac{-L}{2 \tan\left(\frac{\sigma}{2}\right)} \end{bmatrix}. \quad (3.144)$$

Outre les informations visuelles y_f , \dot{y}_f , $\sigma = \theta_l - \theta_r$ (3.95), la vitesse v_a de l'avion et son accélération \dot{v}_a interviennent dans la loi de commande. Les informations visuelles de base que nous avons choisies étant trois droites parallèles, il est clair que l'on ne peut pas commander le déplacement de l'avion selon l'axe de ces lignes. On pourrait pour cela considérer le seuil de la piste, et d'autres caractéristiques de la piste, telles que les marquages blancs (comme ceux représentés sur le schéma de piste de la figure 3.25 par exemple). Dans notre cas, on a supposé que l'on connaît la distance longitudinale initiale X_i de l'avion par rapport à la piste, et la vitesse de l'avion est supposée être obtenue par un autre capteur que la caméra. Notons également que la largeur de la piste L est ici supposée connue.

3.4.2 Génération de trajectoires

Les lois de commande par asservissement visuel proposées reposent sur des modèles valables lorsque la consigne est relativement proche de la position initiale. Afin d'améliorer leur comportement, les valeurs désirées sont spécifiées sous forme de trajectoire. Néanmoins, il est nécessaire de spécifier des trajectoires réalisables par l'avion, et donc de tenir compte des couplages dus au modèle de l'avion.

3.4.2.1 Trajectoire latérale

On considère que initialement, l'avion n'est pas forcément aligné avec l'axe de la piste. Pendant la phase d'approche, il doit donc rejoindre cet axe, puis rester aligné pendant toute la suite de la procédure d'atterrissage (voir figure 3.25). Notons t_0 l'instant initial considéré, et t_f l'instant final où l'avion s'est aligné dans l'axe de la piste. On

cherche à concevoir une trajectoire $Y^*(t)$ que l'avion devra suivre, et qui lui permettra de rejoindre la position désirée $Y_f = 0$ à partir de la position initiale Y_i . Les contraintes sur $Y^*(t)$ sont donc

$$Y^*(t_0) = Y_i \quad \text{et} \quad Y^*(t_f) = Y_f = 0. \quad (3.145)$$

Ces contraintes font en sorte de "lisser" la position latérale, au début et à la fin de la trajectoire, en prenant en compte les conditions aux limites.

Afin d'améliorer la trajectoire, considérons à présent les dérivées successives de Y .

- Avec (3.67), on voit que contraindre \dot{Y} est équivalent à contraindre l'angle de lacet ψ . En écrivant (3.67) aux instants initial et final, on a :

$$\dot{Y}^*(t_0) = a_5\psi_i \quad \text{et} \quad \dot{Y}^*(t_f) = a_5\psi_f = 0, \quad (3.146)$$

où ψ_i et $\psi_f = 0$ sont respectivement les valeurs initiales et finales de l'angle de lacet. Ces contraintes font en sorte que la trajectoire de l'angle de lacet soit "lisse".

- En remplaçant $\dot{\psi}$ dans (3.122) avec (3.65), on obtient

$$\ddot{Y} = a_5a_3\phi \quad (3.147)$$

et donc :

$$\ddot{Y}^*(t_0) = a_5a_3\phi_i \quad \text{et} \quad \ddot{Y}^*(t_f) = a_5a_3\phi_f = 0, \quad (3.148)$$

où ϕ_i et $\phi_f = 0$ sont respectivement les valeurs initiales et finales de l'angle de roulis. Comme précédemment, ces contraintes font en sorte que la trajectoire de l'angle de roulis soit "lisse".

- En dérivant la relation précédente et en remplaçant $\dot{\phi}$ par son expression (3.64), on a $\ddot{Y} = -\frac{b_1}{a_1}a_5a_3\delta_l$ et donc :

$$\ddot{Y}^*(t_0) = -\frac{b_1}{a_1}a_5a_3\delta_{l_i}(t_0) = 0 \quad \text{et} \quad \ddot{Y}^*(t_f) = -\frac{b_1}{a_1}a_5a_3\delta_{l_f}(t_f) = 0 \quad (3.149)$$

où $\delta_{l_i} = 0$ et $\delta_{l_f} = 0$ sont respectivement les valeurs initiales et finales de la commande de gauchissement. Ces contraintes font en sorte que l'entrée de commande (et donc la vitesse de roulis (3.64)) soit "lisse".

Les relations (3.145) à (3.149) tiennent compte du modèle de l'avion, en fixant huit contraintes à respecter pour $Y^*(t)$. Une manière simple de concevoir $Y^*(t)$ consiste alors à l'exprimer sous la forme d'un polynôme de degré sept :

$$Y^*(t) = at^7 + bt^6 + ct^5 + dt^4 + et^3 + ft^2 + gt + h \quad (3.150)$$

où les coefficients a, b, c, d, e, f, g, h sont déterminés de manière à satisfaire les huit conditions (voir annexe F.1). Ils sont donc fonctions de la position initiale Y_i , de l'orientation ϕ_i, ψ_i de l'avion, et des instants initiaux t_0 et finaux t_f .

On peut alors également déterminer une trajectoire pour les angles de roulis et de lacet, avec (3.67), (3.147) et les dérivées de $Y^*(t)$:

$$\psi^*(t) = \frac{1}{a_5}(7at^6 + 6bt^5 + 5ct^4 + 4dt^3 + 3et^2 + 2ft + g) \quad (3.151)$$

$$\phi^*(t) = \frac{1}{a_3a_5}(42at^5 + 30bt^4 + 20ct^3 + 12dt^2 + 6et + 2f) \quad (3.152)$$

Les paramètres initiaux sont supposés connus. Par contre, on a besoin de fixer au départ l'instant final t_f , c'est-à-dire de décider en combien de temps l'avion va rejoindre l'axe de la piste. Pour ce faire, on propose une relation simple prenant en compte l'erreur latérale initiale et l'angle de lacet initial (voir figure 3.28) :

$$t_f = t_0 + k|Y_i + D \tan \psi_i| \quad (3.153)$$

où D et k sont des constantes, fixées de manière à ce que la trajectoire soit réalisable.

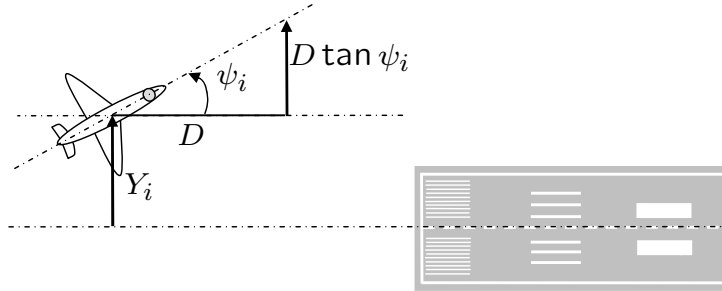


FIG. 3.28 – Position initiale de l'avion par rapport à l'axe de la piste.

3.4.2.2 Trajectoire longitudinale

Concernant les mouvements longitudinaux de l'avion, il s'agit tout d'abord de rejoindre la pente d'approche, de la suivre à vitesse constante, puis d'effectuer la manœuvre d'arrondi avant de toucher la piste, rouler puis s'arrêter (voir figure 3.29).

Pour ce qui est de la trajectoire à suivre pendant les phases d'approche et d'arrondi, nous l'avons présentée à la section 3.1.4. Mais, comme dans le cas de la trajectoire latérale, on suppose que l'avion ne suit pas initialement la pente d'approche désirée et n'a pas la vitesse, la pente, l'angle de tangage, etc. désirés. L'avion étant initialement à l'altitude Z_i , avec une pente γ_i , nous proposons une trajectoire $Z^*(X)$ dans le plan vertical lui permettant de rejoindre la pente d'approche définie par (3.20) :

$$Z(X) = X \tan(-\gamma_0) \quad (3.154)$$

où γ_0 est l'angle de pente souhaité. Comme précédemment pour la partie latérale, on s'intéresse aux conditions aux limites.

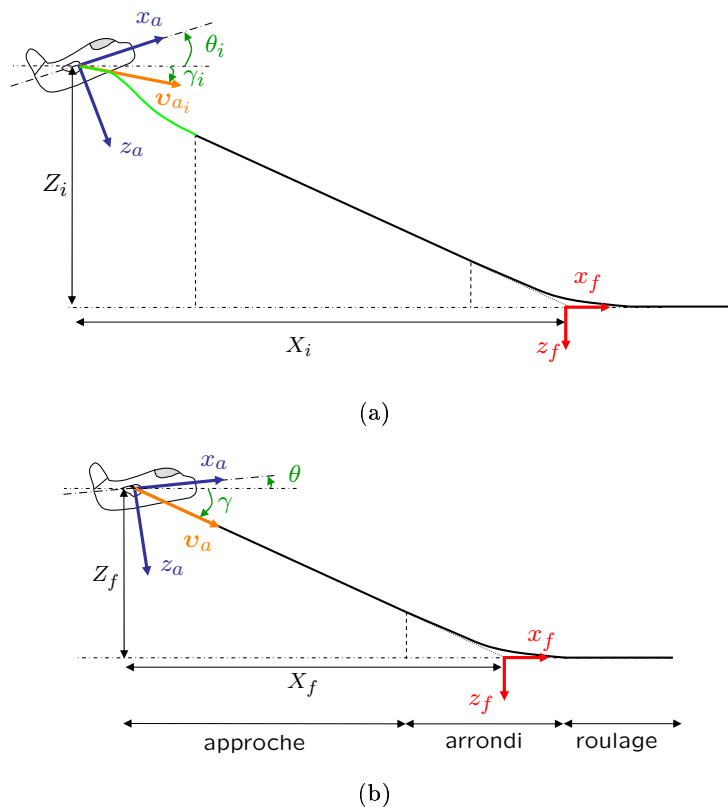


FIG. 3.29 – Mouvements longitudinaux de l'avion pendant l'atterrissage : (a) l'avion n'est pas initialement sur la trajectoire d'approche, (b) l'avion a rejoint la trajectoire d'approche.

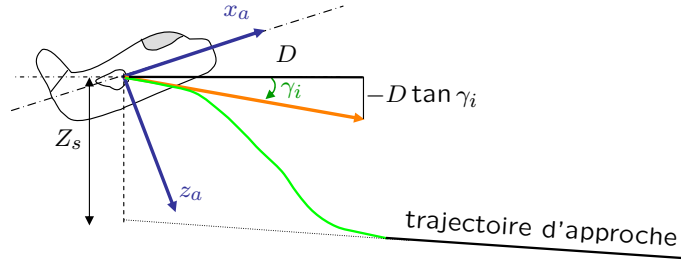


FIG. 3.30 – Position initiale de l'avion par rapport à la trajectoire d'approche.

- La trajectoire $Z^*(X)$ doit passer par la position initiale (X_i, Z_i) et la position finale (X_f, Z_f) de l'avion :

$$Z^*(X_i) = Z_i \quad (3.155)$$

$$Z^*(X_f) = Z_f = X_f \tan(-\gamma_0) \quad (3.156)$$

- La pente définie par la trajectoire $Z^*(X)$ doit correspondre à la pente initiale γ_i de l'avion au départ et à la pente de la trajectoire d'approche $\gamma_f = \gamma_0$ à la fin. On a donc, en dérivant $Z(X)$ par rapport à X (voir (3.154)) :

$$\frac{\partial Z^*}{\partial X}(X_i) = \tan(-\gamma_i) \quad (3.157)$$

$$\frac{\partial Z^*}{\partial X}(X_f) = \tan(-\gamma_0) \quad (3.158)$$

Avec les quatre conditions (3.155) à (3.157), on a choisi comme précédemment de déterminer $Z^*(X)$ sous la forme d'un polynôme, de degré trois cette fois-ci :

$$Z^*(X) = aX^3 + bX^2 + cX + d, \quad (3.159)$$

où les paramètres a, b, c, d sont déterminés de la même manière que précédemment (voir annexes F.1 et F.2). Ils dépendent des conditions initiales et de la distance $X_f - X_i$ parcourue pour rejoindre la trajectoire d'approche.

Le choix de $X_f - X_i$ est du même type que précédemment (voir figure 3.30) :

$$X_f - X_i = k|Z_s + D \tan \gamma_i| \quad (3.160)$$

où k et D sont des constantes.

Notons également que nous avons choisi pour l'angle de tangage désiré une expression de la forme

$$\theta^* = k(Z - Z^*) + \theta_p, \quad (3.161)$$

où k et θ_p sont fixés [Jorgensen 90], la vitesse v_a^* désirée est quant à elle conservée constante, tout comme la pente γ^* .

3.4.3 Lois de commande pour le suivi de trajectoires

À partir des trajectoires définies précédemment à la section (3.4.2) pour la position et l'orientation de l'avion dans l'espace 3D, on peut d'une part injecter directement ces valeurs désirées dans les lois de commande initiales (3.139) et (3.126) pour obtenir :

$$\mathbf{u}_{l_0} - \mathbf{u}_{l_{00}} = \mathbf{K}(\mathbf{y}_{l_0} - \mathbf{y}_{l_0}^*), \quad (3.162)$$

et

$$\delta_l = -n_1(\ddot{Y} - \ddot{Y}^*) - n_2(\dot{Y} - \dot{Y}^*) - n_3(Y - Y^*) \quad (3.163)$$

où $\mathbf{y}_{l_0}^*$, Y^* , \dot{Y}^* et \ddot{Y}^* ne sont plus fixes, mais contiennent les expressions des trajectoires latérales et longitudinales (3.150), (3.151), (3.152), (3.159), (3.161), et (3.25) pour l'altitude pendant la phase d'arrondi. On rappelle que l'on avait précédemment — dans le modèle de l'avion puis dans les lois de commande — allégé les notations en omettant les termes de référence, et en notant x les termes \tilde{x} , où $\tilde{x} = x - x_0$. Nous avons ici explicité à nouveau entièrement les expressions des paramètres sous la forme $x - x_0$ et $x - x^*$.

La largeur de la piste étant supposée connue, ainsi que la position de la caméra sur l'avion, on peut également, à chaque itération, connaître la position et l'orientation désirées de la caméra sous forme de trajectoire. On peut ensuite projeter le modèle de la piste dans le plan image de la caméra placée à la position et avec l'orientation désirées, et en déduire les informations visuelles correspondantes. Les trajectoires désirées de l'avion dans l'espace 3D vont ainsi correspondre à des trajectoires pour les informations visuelles dans le plan image. Les lois de commande (3.144) et (3.127) deviennent :

$$\begin{bmatrix} \delta_m \\ \delta_T - \delta_{T0} \end{bmatrix} = \mathbf{K} \begin{bmatrix} y_f - y_f^* \\ \dot{y}_f - \dot{y}_f^* \\ \frac{L}{4a_{18}} \left(\frac{1}{\sin^2(\frac{\sigma}{2})} \dot{\sigma} - \frac{1}{\sin^2(\frac{\sigma^*}{2})} \dot{\sigma}^* \right) - \frac{a_{19}}{a_{18}} (v_a - v_a^*) \\ v_a - v_a^* \\ \dot{v}_a - \dot{v}_a^* \\ \frac{-L}{2 \tan(\frac{\sigma}{2})} - \frac{-L}{2 \tan(\frac{\sigma^*}{2})} \end{bmatrix}, \quad (3.164)$$

et

$$\delta_l = n_1 a_5 (\dot{x}_f - \dot{x}_f^*) + n_2 a_5 (x_f - x_f^*) - n_3 \left(\frac{H}{c^2 \theta_l + c^2 \theta_r - 2} \delta - \frac{H^*}{c^2 \theta_l^* + c^2 \theta_r^* - 2} \delta^* \right) \quad (3.165)$$

où les informations visuelles désirées varient en fonction de la trajectoire désirée. Notons que les trajectoires étant définies dans l'espace 3D, il n'est pas garanti que pour les informations visuelles désirées ainsi obtenues, les lignes de la piste restent dans l'image. En pratique, ce problème ne s'est cependant pas posé dans les simulations réalisées.

3.5 Résultats de simulation

Les schémas de commande proposés ont été testés en simulation. Nous présentons ici le simulateur utilisé, puis les résultats obtenus.

3.5.1 Simulateur utilisé

Simulation de l'avion. Le cœur du simulateur utilisé est constitué d'une bibliothèque de fonctions comportant un modèle de vol et permettant de simuler le comportement d'un avion (voir figure 3.31). Cette bibliothèque, écrite en langage C++, nous a été fournie par Dassault Aviation. Parmi le panel de fonctions permettant de paramétrer et de simuler le vol d'un avion, nous utilisons les fonctions suivantes :

- À l'initialisation du simulateur :
 - activation des dispositifs hypersustentateurs (utiles pendant la phase d'atterrissage, voir section 3.1.4.4),
 - spécification de la position initiale de l'avion, du cap initial et de la vitesse initiale.
- À chaque itération (on a choisi la cadence vidéo d'une caméra "classique", soit 40 ms) :
 - spécification des entrées de commande (positions du manche et manette des gaz),
 - mesure de la position de l'avion (longitude, latitude, altitude, voir section 3.1.1, figure 3.1)
 - mesure de l'orientation de l'avion (voir section 3.1.1, figure 3.1)
 - mesures de diverses autres grandeurs, telles que la vitesse, la pente, l'incidence (voir section 3.1.1, figure 3.6).

À partir des mesures de la position (sous forme de longitude, latitude, altitude) et de l'orientation (sous forme d'une matrice de rotation) de l'avion fournies par le simulateur, on calcule sa position et son orientation par rapport à une piste d'atterrissage (positions dans le repère lié à la piste d'atterrissage, et angles de roulis-tangage-lacet), comme indiqué à la section 3.1.1 et à l'annexe B. On peut alors exprimer la pose de l'avion sous la forme d'une matrice homogène :

$${}^f\mathbf{M}_a = \begin{bmatrix} {}^f\mathbf{R}_a & {}^f\mathbf{t}_a \\ \mathbf{O}_{1 \times 3} & 1 \end{bmatrix} \quad (3.166)$$

où ${}^f\mathbf{R}_a$ représente l'orientation de l'avion dans le repère fixe lié à la piste d'atterrissage, et ${}^f\mathbf{t}_a$ la position de l'avion dans ce repère.

On peut alors spécifier la pose de la caméra par rapport à l'avion avec la matrice homogène ${}^a\mathbf{M}_c$ (3.69), puis déterminer la pose de la caméra dans le repère fixe ${}^f\mathbf{M}_c$, avec la relation classique :

$${}^f\mathbf{M}_c = {}^f\mathbf{M}_a {}^a\mathbf{M}_c. \quad (3.167)$$

Visualisation. Un visualisateur 3D a été couplé avec le modèle de vol : il s'agit de la partie visualisation du simulateur de vol *Flight Gear* (voir figure 3.31). Nous utilisons des fonctions permettant de positionner une caméra (longitude, latitude et altitude, et orientation), et le visualisateur fournit l'image de la scène ainsi observée. Dans notre cas, on se place à proximité d'une piste d'atterrissage.

Notons que l'on peut, dans la simulation, positionner la caméra exactement là où on le souhaite dans le repère fixe. Il est bien sûr possible de considérer une caméra

embarquée sur l'avion (c'est ce qu'on utilise par la suite pour l'asservissement visuel), mais on peut également considérer une caméra déportée, qui observe l'avion, ce qui permet de visualiser le comportement de l'avion avec un point de vue externe (voir figure 3.31). Dans ce cas, on ajoute au "monde 3D" du visualisateur un modèle 3D d'avion (décrit par un fichier *XML* chargé dans *Flight Gear*).

Dans le cadre de nos simulations, on n'a pas considéré de module de traitement d'image. Le visualisateur n'est donc utilisé que pour visualiser la scène du point de vue de la caméra (embarquée ou externe). Pour construire les lignes de la piste dans le plan image, on a projeté un modèle de la piste d'atterrissage dans le plan image de la caméra (voir figure 3.31). On obtient ainsi directement les paramètres des lignes de la piste dans la plan image.

Commande. On peut ensuite déterminer les informations visuelles courantes, obtenir du modèle de vol les autres mesures utilisées dans nos lois de commande, et boucler le système en injectant les valeurs calculées pour les entrées de commande (position du manche et de la manette des gaz) dans le simulateur de vol (voir figure 3.31).

Notons qu'il est également possible de choisir un mode "manuel", où l'on spécifie les valeurs des entrées de commande avec le clavier et la souris (voir figure 3.31).

Autres outils. Nous avons implémenté nos schémas de commande en C/C++, et utilisé également la bibliothèque d'asservissement visuel ViSP-2 [Marchand 05, Visp], qui propose de nombreux outils très utiles pour la mise au point de telles lois de commande. En particulier, nous avons utilisé les fonctionnalités de :

- calculs avec des matrices de rotation, des matrices homogènes, des angles d'Euler,
- projection des lignes de la piste sur le plan image
- affichage des lignes dans l'image.

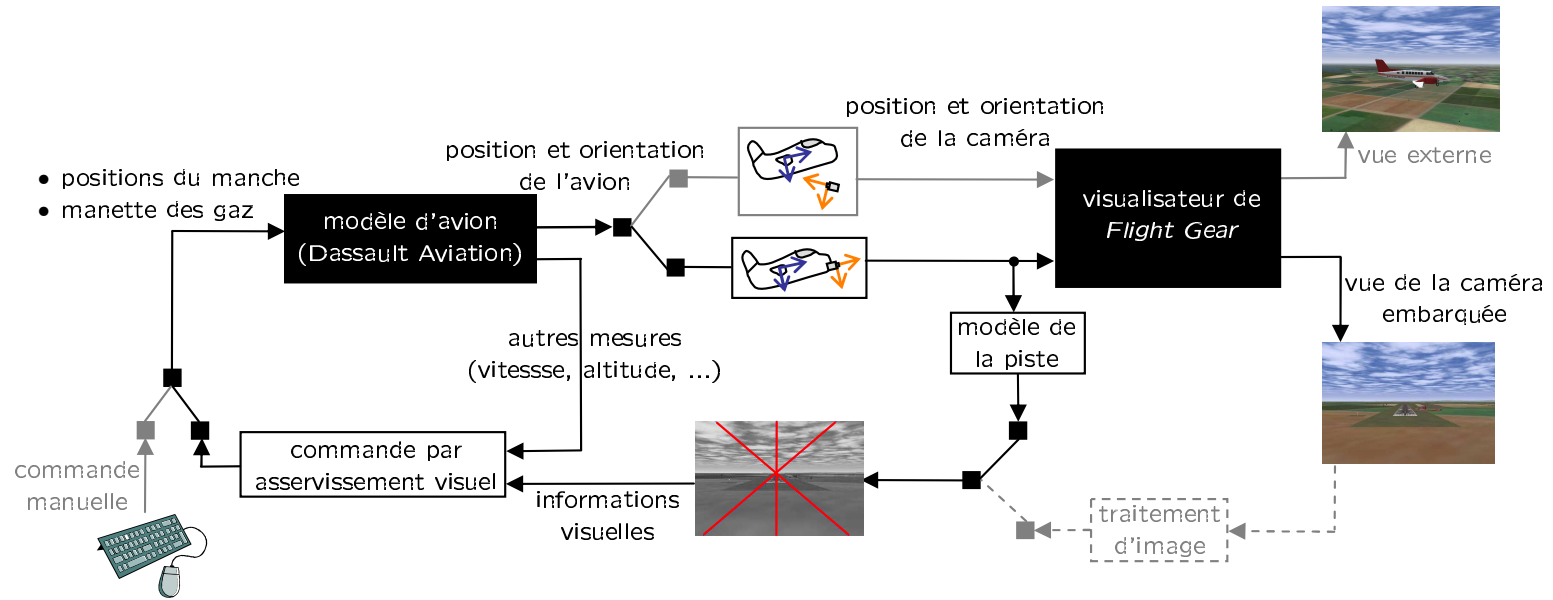


FIG. 3.31 – Schéma du simulateur utilisé : en traits noirs, les éléments utilisés pour tester nos schémas de commande, et en gris les fonctionnalités disponibles, au choix de l'utilisateur, et en tirets gris les fonctionnalités qui pourraient être facilement intégrées.

3.5.2 Résultats

Dans le simulateur, la piste considérée a une largeur $L = 60$ mètres.

Pendant la phase d'approche considérée, les valeurs nominales des paramètres du modèle de l'avion sont les suivantes :

$$\begin{aligned}
 \theta_0 &= -3 \text{ deg} \\
 \gamma_0 &= -5.5 \text{ deg} \\
 v_{a_0} &= 55.28 \text{ m/s} \\
 Y_0 &= 0 \text{ m} \\
 \phi_0 &= 0 \text{ deg} \\
 \psi_0 &= 0 \text{ deg} \\
 \delta_{l_0} &= 0 \\
 \delta_{m_0} &= 0 \\
 \delta_{T_0} &= 0.1.
 \end{aligned}$$

Pour la phase d'arrondi, on a utilisé la trajectoire verticale (3.25) présentée à la section 3.1.4.2. On lance la manœuvre d'arrondi à l'altitude $H_0 = 20$ m, pour atteindre la vitesse verticale $\dot{H}(t_T) = -0.5$ m/s au moment de toucher la piste. La vitesse désirée est gardée constante à v_{a_0} pendant cette phase. La pente désirée est calculée à partir de la vitesse verticale désirée $\dot{H}^*(t)$ par : $\gamma^*(t) = \text{atan} \frac{\dot{H}^*(t)}{\dot{X}_0}$ où $\dot{X}_0 = \frac{-\dot{H}_0}{\tan \gamma_0}$ est la composante de la vitesse v_{a_0} selon l'axe longitudinal de la piste. Comme dans [Jorgensen 90], on a considéré un angle de tangage désiré de la forme $\theta^* = -k(H - H^*) + \theta_p$. On a pris $k = 0.15$ et $\theta_p = 3.517$ deg.

3.5.2.1 Commande longitudinale

On ne considère ici que les mouvements dans le plan vertical : l'avion est initialement placé dans l'axe de la piste ($Y = 0$), et ses angles de roulis et de lacet sont nuls ($\phi = \psi = 0$). On conserve également une entrée de commande de gauchissement nulle ($\delta_l = 0$), ce qui — en l'absence de perturbations extérieures — permet à l'avion de rester aligné avec la piste. L'avion est placé initialement trop haut par rapport à la pente d'approche nominale. L'objectif est alors de rejoindre la pente d'approche nominale, et de réguler les différents paramètres (pente, vitesse, tangage) aux valeurs nominales considérées (voir figures 3.29 et 3.30).

Les entrées commandant les mouvements longitudinaux sont la gouverne de profondeur et la poussée des moteurs : δ_m et δ_T , notées sous la forme du vecteur

$$\mathbf{u}_{lo} = (\delta_m, \delta_T). \quad (3.168)$$

Pour chaque simulation, on représentera les paramètres du modèle longitudinal de l'avion : les entrées de commande δ_m et δ_T , l'angle de tangage θ , la pente γ , la vitesse de l'avion v_a , et sa position $Z = f(X)$ dans le plan vertical. On a également représenté \dot{Z} ainsi que l'évolution des informations visuelles.

On considère une commande de type LQR, comme présentée aux sections 1.2.3.2 et 3.4.1.2. Pour calculer la matrice optimale \mathbf{K} , on a pris $\mathbf{Q} = \mathbf{I}_6$ et $\mathbf{R} = \mathbf{I}_6$. En pratique, on a utilisé les fonctionnalités du logiciel *Matlab* et de la *Control System Toolbox*, dédiée à la commande, pour calculer \mathbf{K} . On obtient :

$$\mathbf{K} = \begin{bmatrix} 13.6 & 3.5 & 50.8 & -0.178 & -0.084 & -0.991 \\ 2.66 & -0.282 & 9.496 & 0.978 & 1.420 & -0.131 \end{bmatrix}$$

Commande avec mesures directes. L'avion se trouve initialement à la distance $X_i = -4100$ m de la piste, et à l'altitude $Z_i = -491$ m. À cette distance, pour être sur la trajectoire d'approche, l'avion devrait se trouver à l'altitude $Z_{i_0} = -X_i * \tan(\gamma_0) = -395$ m. L'erreur initiale en altitude est donc de $Z_s = Z_{i_0} - Z_i = 96$ m.

On utilise ici la loi de commande (3.139) :

$$\mathbf{u}_{l_0} = \mathbf{K}(\mathbf{y}_{l_0} - \mathbf{y}_{l_0}^*),$$

et on considère qu'une mesure directe des paramètres $\mathbf{y}_{l_0} = (\theta, \dot{\theta}, \gamma, v_a, \dot{v}_a, Z)$ est disponible. En pratique la bibliothèque permettant de simuler le comportement de l'avion fournit θ, γ, v_a et Z . Les dérivées par rapport au temps sont estimées à partir du taux d'accroissement entre deux mesures successives, aux instants t et $t + dt$: $\dot{x}(t + dt) = \frac{x(t+dt) - x(t)}{dt}$, où $x(t)$ est la mesure à l'instant t , et dt la période séparant deux mesures. Notons que la planification de trajectoire n'est pas utilisée pour cette première simulation.

La figure 3.32 présente les résultats obtenus. On voit que l'avion rejoint la pente d'approche après environ 40 secondes et 1500 mètres. Puis il reste stabilisé sur la pente d'approche, jusqu'à ce que la manœuvre d'arrondi soit initiée, à l'instant $t = 67$ ($X = -210$ m). On peut voir sur la figure 3.32 que des oscillations d'amplitude importante sont générées au début, le temps de rejoindre la trajectoire nominale, notamment sur la pente (figure 3.32.d) et l'angle de tangage (figure 3.32.c). La commande de profondeur δ_m oscille également (figure 3.32.a). Néanmoins, l'avion rejoint la pente d'approche (figure 3.32.b) et sa vitesse verticale se stabilise (figure 3.32.f). Concernant la vitesse, son évolution est plus douce (figure 3.32.e), tout comme celle de la commande de poussée δ_T (figure 3.32.a). Au départ, l'avion "pique" et sa vitesse augmente, puis il rejoint la pente d'approche et conserve une vitesse constante (figure 3.32.e).

Concernant la manœuvre d'arrondi, on peut voir que la trajectoire "s'arrondit" effectivement sur la figure 3.32.b. Ce mouvement est essentiellement réalisé en levant le "nez" de l'avion, avec la commande de profondeur δ_m (figure 3.32.a). L'objectif est réalisé : de la vitesse verticale $\dot{Z} = 5.3$ m/s, on passe à $\dot{Z} = 0.7$ m/s (figure 3.32.f), pour toucher la piste avec une pente $\gamma = -0.7$ deg. L'angle de tangage augmente pendant cette phase (figure 3.32.c) : il doit permettre à l'avion d'être légèrement cabré pour toucher la piste dans de bonnes conditions (voir section 3.1.4.3). Pour que l'angle de tangage n'augmente pas trop, on a conservé une consigne de vitesse constante (figure 3.32.e), ce qui conduit à une commande non nulle sur la poussée des moteurs (figure 3.32.a). Dans un deuxième temps (après 6 s), on force la commande des moteurs à zéros,

afin de réduire la poussée des moteurs et pouvoir toucher le sol puis s'arrêter dans de bonnes conditions (figure 3.32.a).

Commande visuelle. Au lieu de considérer comme précédemment que toutes les mesures sont directement disponibles, on considère ici la caméra comme capteur principal, fournissant les informations visuelles nécessaires à la loi de commande. Dans la loi de commande précédente, on remplace alors \mathbf{y}_{lo} par $(y_f, \dot{y}_f, \gamma, v_a, \dot{v}_a, \frac{-L}{2 \tan(\frac{\sigma}{2})})$. Cela donne la loi de commande suivante :

$$\begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix} = \mathbf{K} \begin{bmatrix} y_f - y_f^* \\ \dot{y}_f - \dot{y}_f^* \\ \gamma - \gamma^* \\ v_a - v_a^* \\ \dot{v}_a - \dot{v}_a^* \\ \frac{-L}{2 \tan(\frac{\sigma}{2})} - \frac{-L}{2 \tan(\frac{\sigma^*}{2})} \end{bmatrix}. \quad (3.169)$$

On remarque que l'on n'a pas utilisé exactement la loi de commande (3.144). En effet, en pratique et comme nous allons le voir, remplacer $\gamma - \gamma^*$ par $\xi - \xi^*$ avec $\xi = \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2(\frac{\sigma}{2})} - \frac{a_{19}}{a_{18}} v_a$ ne permet pas d'obtenir de bons résultats.

Les conditions initiales et les consignes sont les mêmes que précédemment. Le comportement des entrées de commande, la trajectoire, l'angle de tangage, la pente, la vitesse et la vitesse verticale ont un comportement très similaire à celui obtenu avec les mesures directes : à la vue des courbes de résultats, on remarque seulement une légère différence au niveau du comportement transitoire de l'altitude Z et de la vitesse v_a (comparer les figures 3.32 et 3.33).

Sur la figure 3.34, on a représenté les informations visuelles considérées : l'ordonnée du point de fuite y_f (figure 3.34.a), les paramètres $\theta_{\{l,r\}}$ des lignes gauche et droite (figure 3.34.b), et $\sigma = \theta_l - \theta_r$ (figure 3.34.c).

La figure 3.35 montre des images de la caméra embarquée pendant la simulation. Les angles des droites sont faibles au départ, et tendent à s'approcher de 90 degrés au fur et à mesure que l'avion descend. En pratique, la caméra reste toujours au dessus de la piste, et les valeurs de $\theta_{\{l,r\}}$ tendent vers 83 deg (figure 3.34.b). On peut voir que l'ordonnée du point de fuite, y_f , évolue exactement de la même façon que l'angle de tangage θ (comparer les figures 3.34.a et 3.33.c). Sur la figure 3.34.e, on a représenté l'évolution de l'erreur sur θ , et l'évolution de l'erreur sur y_f : on voit que les deux erreurs sont identiques. On a également représenté sur la figure 3.34.d l'évolution de l'erreur sur l'altitude mesurée Z , avec l'évolution de l'erreur sur l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$. On voit que l'erreur sur W ne correspond pas exactement à l'erreur sur Z , surtout au départ. Néanmoins leur comportement est similaire : cela explique les petites différences observées sur le comportement transitoire de l'altitude lorsque l'on compare les résultats précédents utilisant la mesure directe de Z (figures 3.32.b) et les résultats avec les informations visuelles (figures 3.33.b). Sur la figure 3.34.f, on a représenté l'évolution de l'erreur sur la pente mesurée γ et l'erreur sur ξ . On remarque

que si l'erreur sur ξ correspond correctement à l'erreur sur γ lorsque la commande a convergé, elle est très perturbée (et présente notamment des "pics") pendant la phase transitoire. C'est la raison pour laquelle on n'a en pratique pas utilisé ξ dans la loi de commande, afin de ne pas déstabiliser l'avion.

Suivi de trajectoire. Afin d'améliorer le comportement pendant la phase d'approche, on considère à présent la génération de trajectoires présentée à la section 3.4.2.2. Les paramètres D et k de (3.160) sont choisis empiriquement : on a pris $k = 2$ et $D = 10000$, et saturé $X_f - X_i$ entre 1000 et 2000 m pour que la trajectoire ne soit jamais trop courte (ce qui pourrait conduire à des manœuvres brusques) ni trop longue (ce qui pourrait ne pas laisser le temps à l'avion de rejoindre la pente d'approche correcte suffisamment tôt avant de toucher la piste). Pour l'angle de tangage, on a pris une consigne évolutive de la même forme que pendant l'arrondi : $\theta^* = -k(H - H^*) + \theta_p$, avec cette fois $k = 0.02$ et $\theta_p = 3.517$ deg. Pour éviter les consignes trop importantes, on a également saturé θ^* entre -20 et 20 degrés. Comme nous l'avons présenté à la section 3.4.3, les consignes sous forme d'informations visuelles sont obtenues en projetant simplement les lignes de la piste dans le plan image de la caméra, placée à la position et avec l'orientation désirée (celles-ci étant données par les trajectoires présentées à la section 3.4.2.2). Finalement, la loi de commande utilisée repose sur (3.164), où on a comme précédemment remplacé $\xi - \xi^*$ par $\gamma - \gamma^*$. Il s'agit donc de (3.169), avec des consignes évolutives.

Les résultats sont globalement meilleurs que précédemment, au sens où les paramètres évoluent de manière beaucoup plus "lisse". Les dépassements et oscillations observés précédemment pendant la phase transitoire ont maintenant quasiment disparu (comparer les figures 3.33 et 3.36). Les entrées de commande sont moins perturbées que précédemment, et n'atteignent pas les seuils de saturation (comparer les figures 3.33.a et 3.36.a). L'avion rejoint la pente d'approche aux environs de l'instant $t = 40$ s, comme précédemment. Sur la figure 3.37, on a représenté les informations visuelles. Les mêmes remarques que précédemment peuvent être faites concernant l'ordonnée du point de fuite, dont l'évolution de l'erreur correspond à l'évolution de l'erreur sur le tangage θ (figure 3.37.a et 3.37.e). Concernant les paramètres des lignes de la piste, on peut observer que les petits écarts initiaux visibles entre les valeurs courantes et les valeurs désirées ont disparu (figure 3.37). Cela est dû à la trajectoire, qui "lisse" le comportement des informations visuelles. Comme précédemment, l'erreur sur l'information visuelle $W - W^*$ correspond bien à l'erreur sur l'altitude mesurée $Z - Z^*$ (figure 3.37.d). On peut remarquer que ces erreurs sont beaucoup plus faibles qu'avant pendant la phase transitoire (de l'ordre de quelques mètres au lieu de 100 mètres, voir figures 3.34.d et 3.37.d) : du fait de la trajectoire, on n'est jamais "très loin" de la consigne. On peut remarquer sur la figure 3.37.e, le "pic" d'erreur sur l'angle de tangage à $t \simeq 76$ s : en fait il s'agit du moment où l'avion touche la piste : la consigne est donc très faible, mais il faut quelques secondes pour que le "nez" de l'avion se pose à son tour. En fait, la commande est à ce moment désactivée (les entrées δ_m et δ_T sont nulles), puisque l'avion s'est posé : on se contente de freiner pour que l'avion s'arrête. On observe également un "pic" au même instant au niveau de l'erreur ξ et sur γ (figure 3.37.f). L'erreur sur ξ n'est

cependant pas bien adaptée pour commander la pente, notamment pendant la phase transitoire : on peut voir sur la figure 3.37.f qu'elle représente mal le comportement de l'erreur sur γ . Comme précédemment, cette information visuelle n'est donc pas intégrée dans la loi de commande.

De bons résultats sont finalement obtenus avec la loi de commande (3.169), intégrant les informations visuelles y_f , \dot{y}_f , et $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$ et avec des consignes suivant les trajectoires présentées à la section 3.4.2.2 : l'avion rejoint doucement la pente d'approche nominale, puis arrondit sa trajectoire et se pose sur la piste.

Limites de la commande visuelle. On considère à présent une autre position initiale, plus éloignée de la pente d'approche nominale. L'avion se trouve initialement à la distance $X_i = -4100$ m de la piste, et à l'altitude $Z_i = -571$ m. À cette distance, pour être sur la trajectoire d'approche, il devrait se trouver à l'altitude $Z_{i_0} = -X_i \tan(\gamma_0) = -395$ m. L'erreur initiale en altitude est donc de $Z_s = Z_{i_0} - Z_i = 176$ m.

On utilise dans un premier temps la loi de commande (3.169), et les trajectoires présentées à la section 3.4.2.2, comme précédemment. Les résultats sont présentés sur les figures 3.38 et 3.39. Les consignes sont très perturbées (figure 3.38.a) tout comme l'angle de tangage (figure 3.38.c), la pente (figure 3.38.d) et la vitesse verticale (figure 3.38.f). La trajectoire de l'avion dans le plan vertical a tendance à osciller autour de la consigne (figure 3.38.b), et la vitesse ne rejoint la vitesse nominale que très tard (figure 3.38.e). À la vue du comportement des informations visuelles (figure 3.39), on remarque que pendant la phase transitoire, l'erreur sur l'information visuelle W s'éloigne de l'erreur sur l'altitude Z (figure 3.39.d).

Considérons la même expérience, en remplaçant l'erreur visuelle $W - W^*$ par $Z - Z^*$ (obtenue par mesure directe de l'altitude) dans la loi de commande (3.169) :

$$\begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix} = \mathbf{K} \begin{bmatrix} y_f - y_f^* \\ \dot{y}_f - \dot{y}_f^* \\ \gamma - \gamma^* \\ v_a - v_a^* \\ \dot{v}_a - \dot{v}_a^* \\ Z - Z^* \end{bmatrix}. \quad (3.170)$$

On obtient les résultats présentés sur les figures 3.40 et 3.41. Cette fois, le comportement est satisfaisant : les paramètres convergent vers leurs valeurs désirées et évoluent de manière "lisse" (figure 3.40). Les entrées de commande ne sont pas saturées (figure 3.40.a), et la trajectoire de l'avion dans le plan vertical suit la trajectoire désirée sans oscillations (figure 3.40.b). La figure 3.41 présente l'évolution des informations visuelles.

Il s'avère que l'utilisation de l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$ pour commander l'altitude n'est pas adaptée ici, le comportement de l'avion n'étant pas satisfaisant (figure 3.38). En fait, W représente bien l'altitude de la caméra (et donc de l'avion) pour une position et une orientation données (voir section 3.3.2 et figure 3.22.a). Lorsque l'erreur sur Z augmente, la consigne en tangage θ^* augmente. Lors de l'étape de projection des lignes de la piste dans le plan image de la caméra (orientée avec cet angle de tangage

désiré), l'information visuelle désirée W^* n'est plus représentative de l'altitude désirée Z^* si l'on est trop loin de l'orientation de référence utilisée dans la modélisation (section 3.3.2 et figure 3.22.a). Il s'en suit que l'erreur $W - W^*$ augmente fortement, ce qui fait augmenter θ^* , et ainsi de suite. Notons que cela se produit au départ pendant la phase transitoire. L'avion parvient tout de même à rejoindre la pente d'approche (voir figure 3.38).

On a mis en évidence ici une limite de l'utilisation de l'information visuelle W : si l'on s'éloigne trop de la position de référence utilisée dans la modélisation, de fortes oscillations apparaissent.

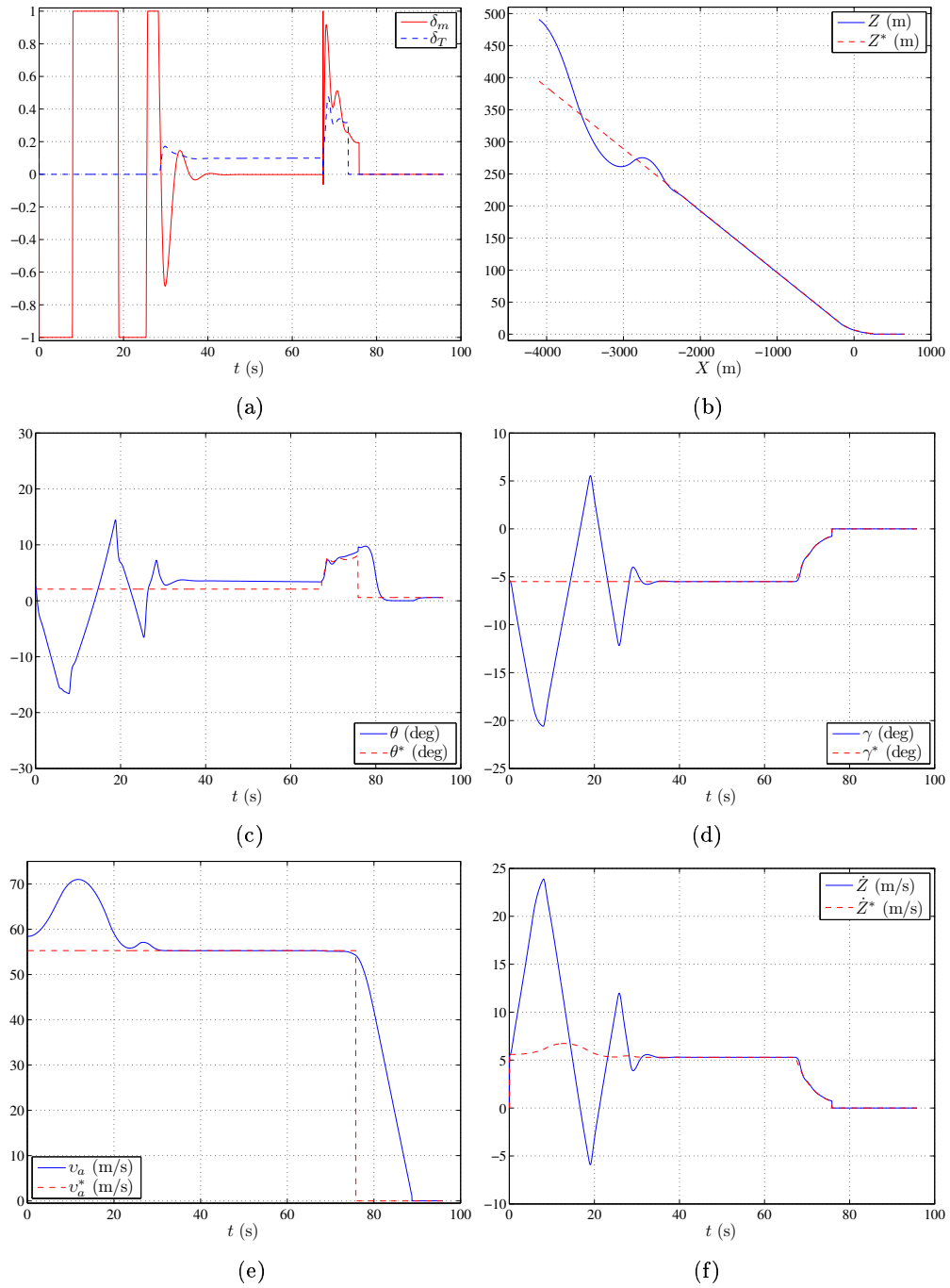


FIG. 3.32 – Résultats de la commande longitudinale avec mesures directes. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et vitesse verticale de l'avion (f).

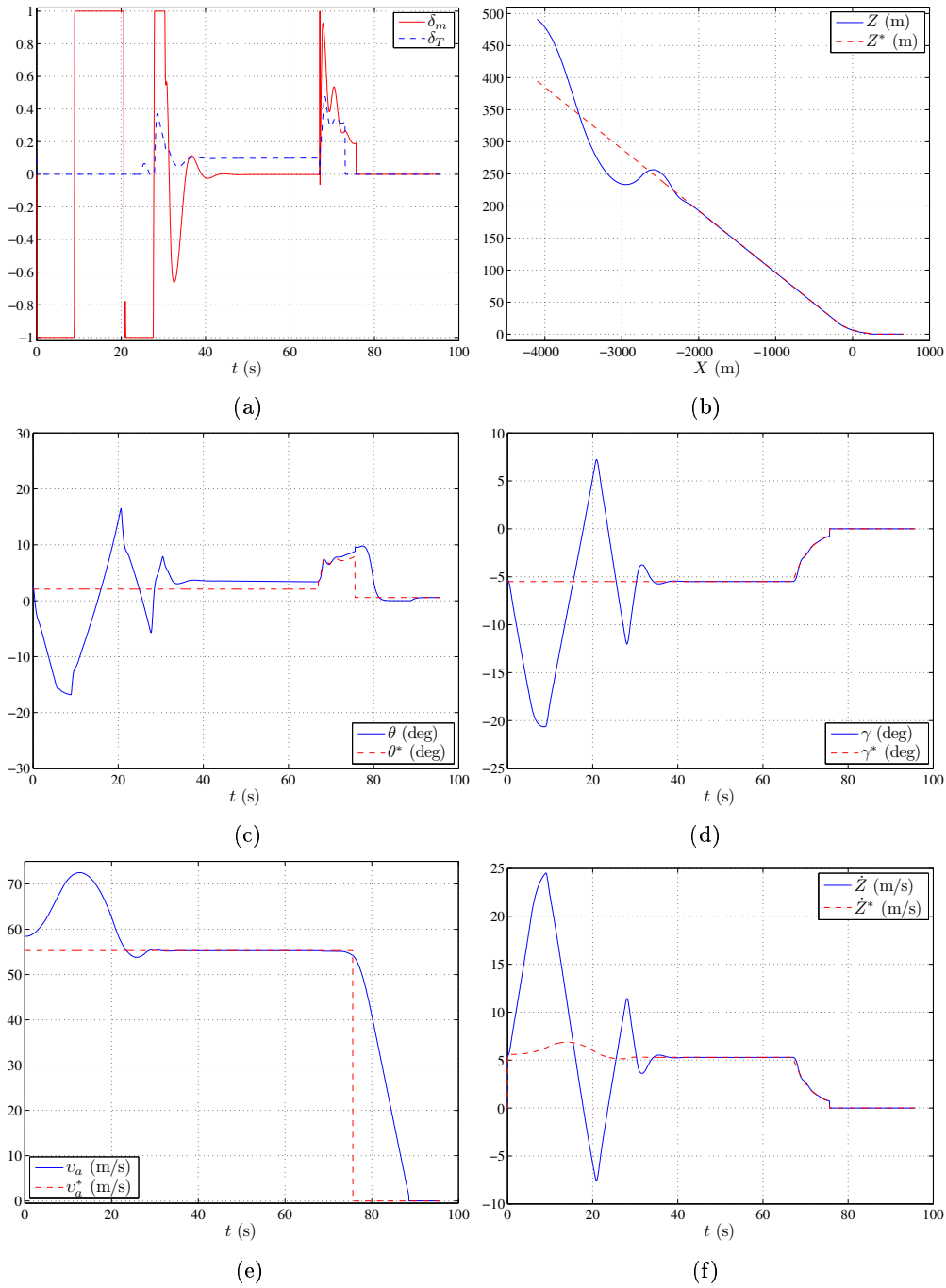


FIG. 3.33 – Résultats de la commande longitudinale avec mesures visuelles. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et vitesse verticale de l'avion (f).

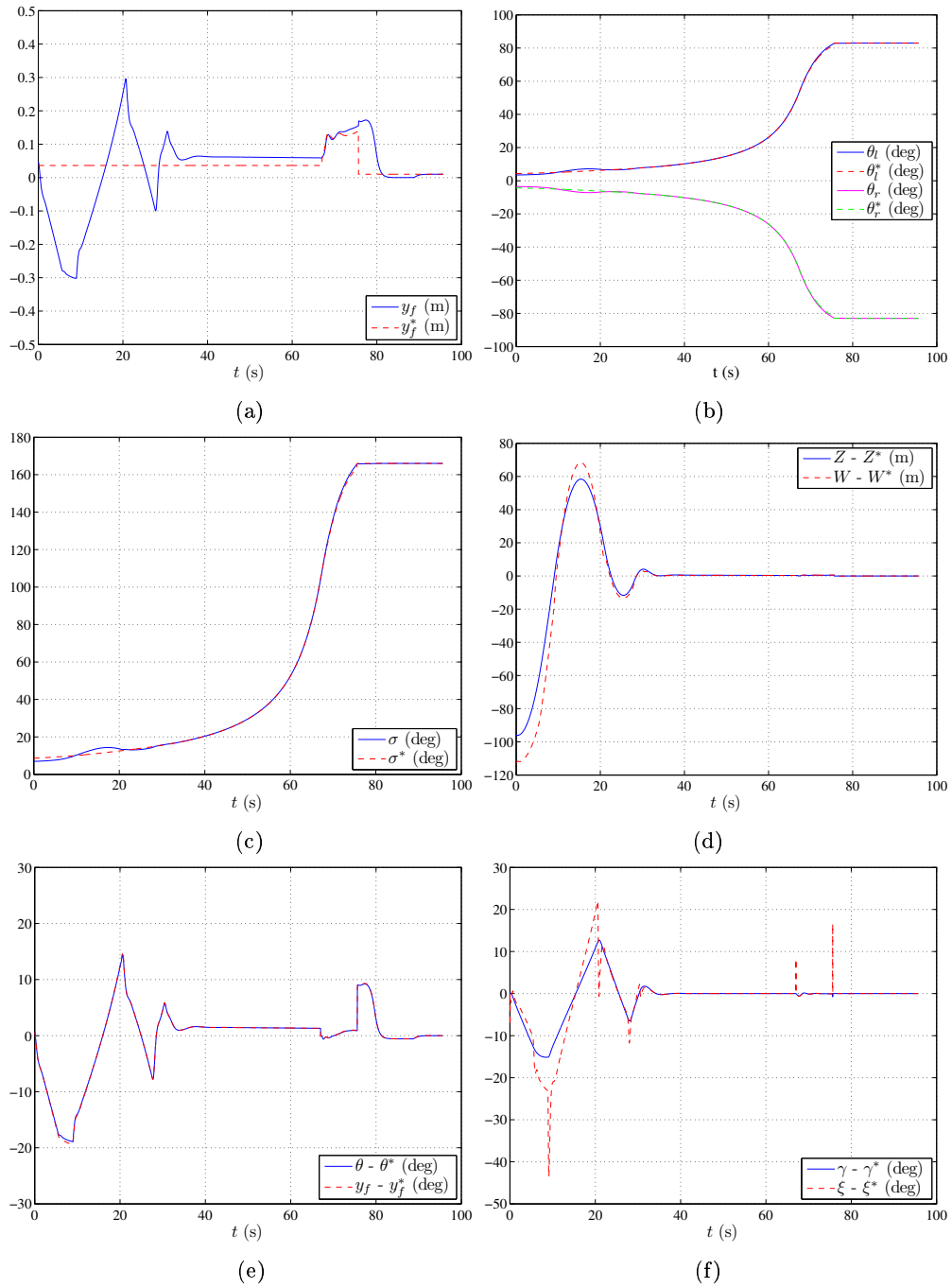


FIG. 3.34 – Résultats de la commande longitudinale avec mesures visuelles. Évolution au cours du temps de la position verticale du point de fuite dans l'image (a), des paramètres θ de la projection des lignes gauche et droite de la piste dans l'image (b), de l'information visuelle σ (c). Évolution de l'erreur sur l'altitude Z et sur l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$, utilisée dans la loi de commande (d), évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (e), évolution de l'erreur sur la pente γ et sur l'information visuelle $\xi = \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2(\frac{\sigma}{2})} - \frac{a_{19}}{a_{18}} v_a$ (f).

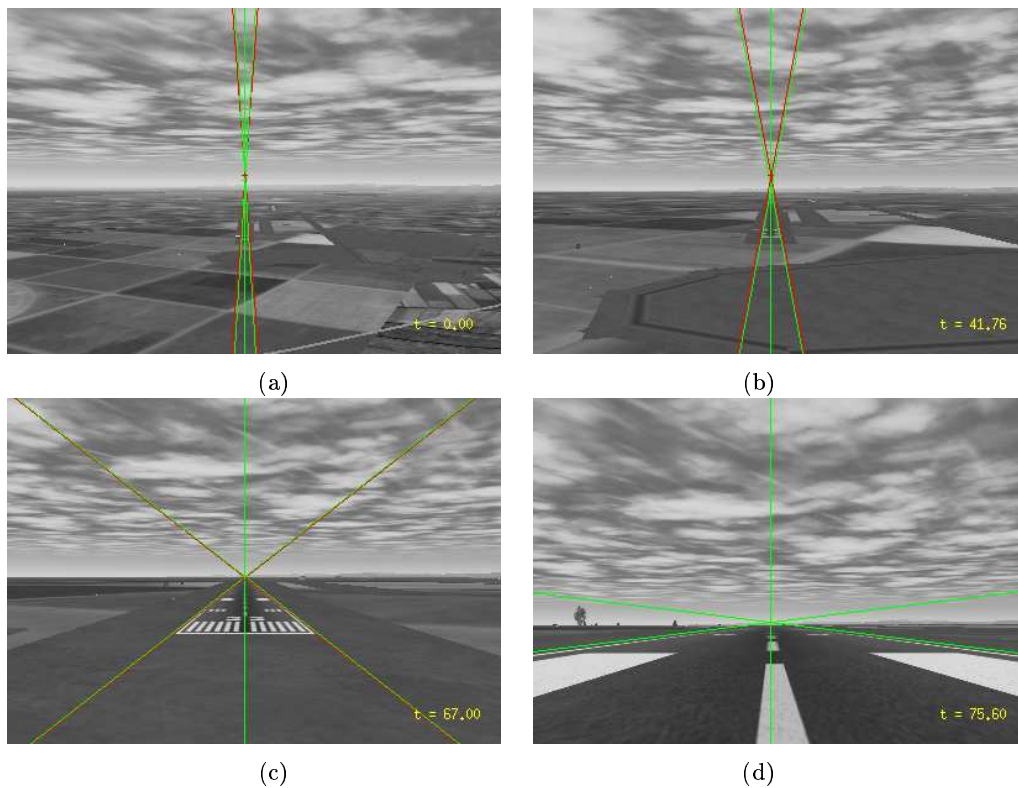


FIG. 3.35 – Vues de la caméra embarquée, aux étapes-clé de l'atterrissage. (a) : position initiale ($t = 0$ s); (b) l'avion a rejoint la pente d'approche ($t = 41.76$ s); (c) : début de la manœuvre d'arrondi ($t = 67$ s); (d) : toucher de la piste ($t = 75.6$ s). Les lignes de la piste sont ici assimilées à des droites, c'est pourquoi elles se poursuivent au-delà du point de fuite. Le point de fuite est représenté par une croix. En vert les informations visuelles courantes sont représentées, et leurs positions désirées dans l'image sont tracées en rouge.

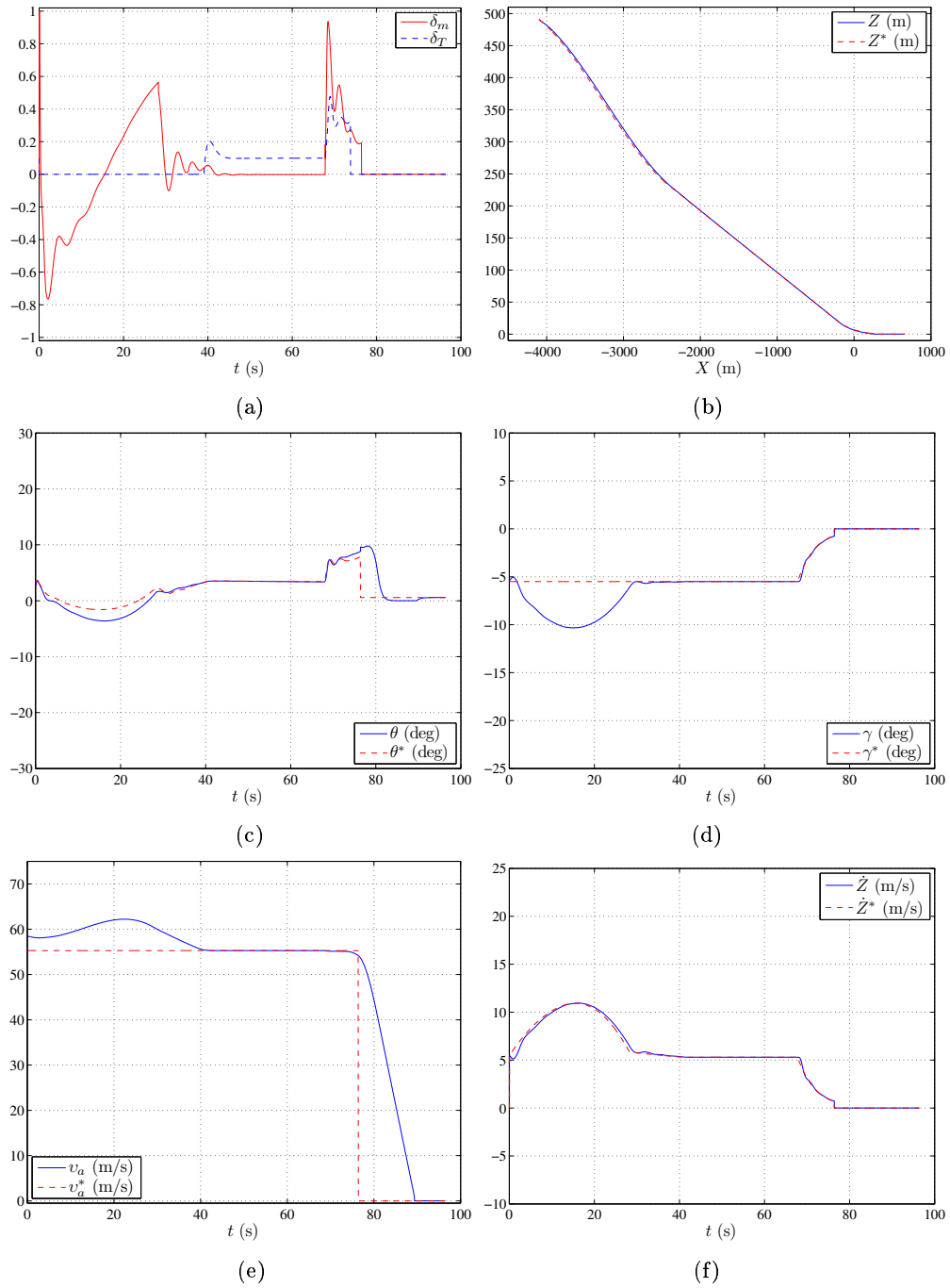


FIG. 3.36 – Résultats de la commande longitudinale avec mesures visuelles et trajectoire. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et vitesse verticale de l'avion (f).

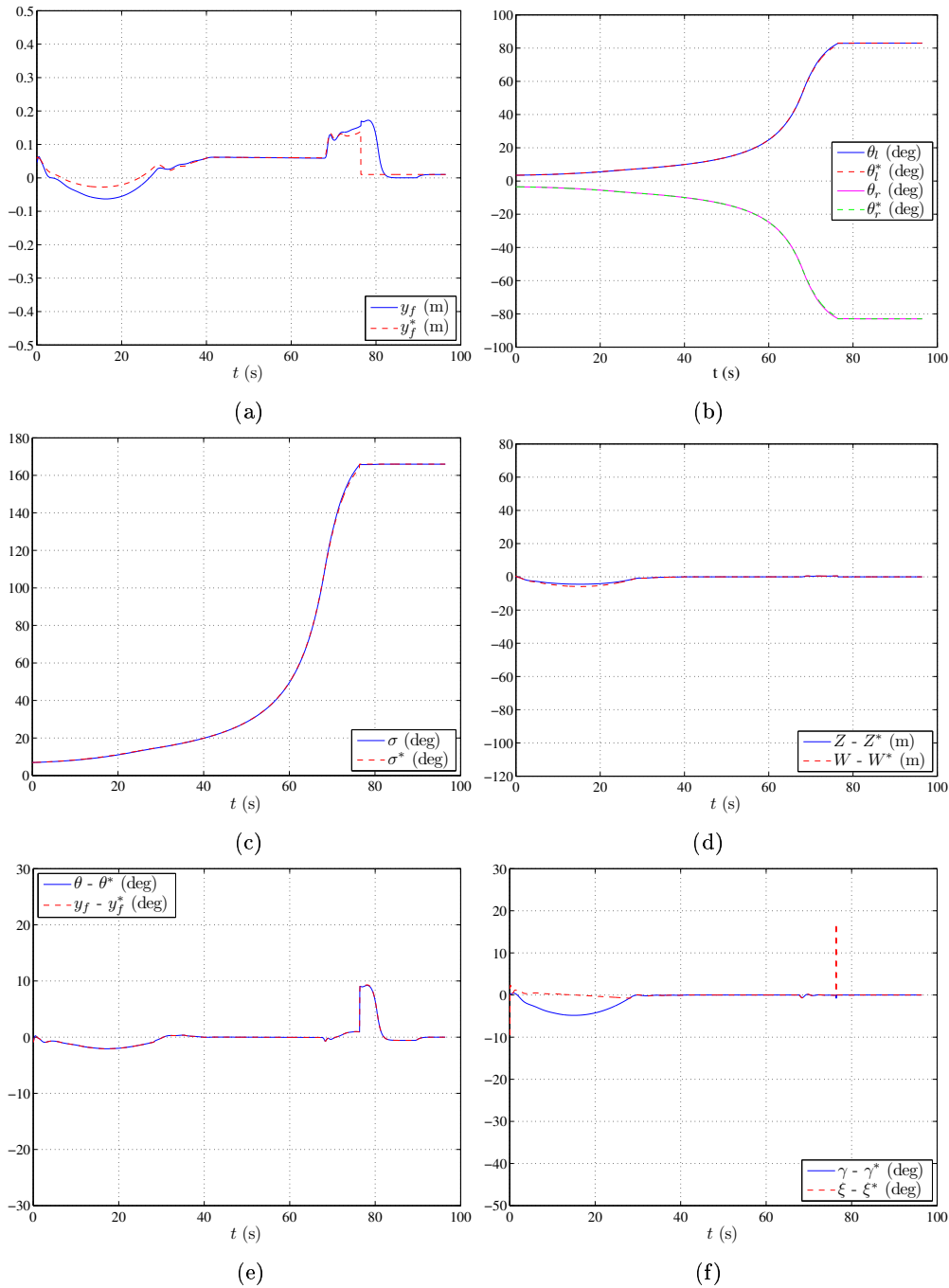


FIG. 3.37 – Résultats de la commande longitudinale avec mesures visuelles et trajectoire. Résultats de la commande longitudinale avec mesures visuelles partielles et trajectoire. Évolution au cours du temps de la position verticale du point de fuite dans l'image (a), des paramètres θ de la projection des lignes gauche et droite de la piste dans l'image (b), de l'information visuelle σ (c). Évolution de l'erreur sur l'altitude Z et sur l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$, utilisée dans la loi de commande (d), évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (e), évolution de l'erreur sur la pente γ et sur l'information visuelle $\xi = \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2(\frac{\sigma}{2})} - \frac{a_{19}}{a_{18}} v_a$ (f).

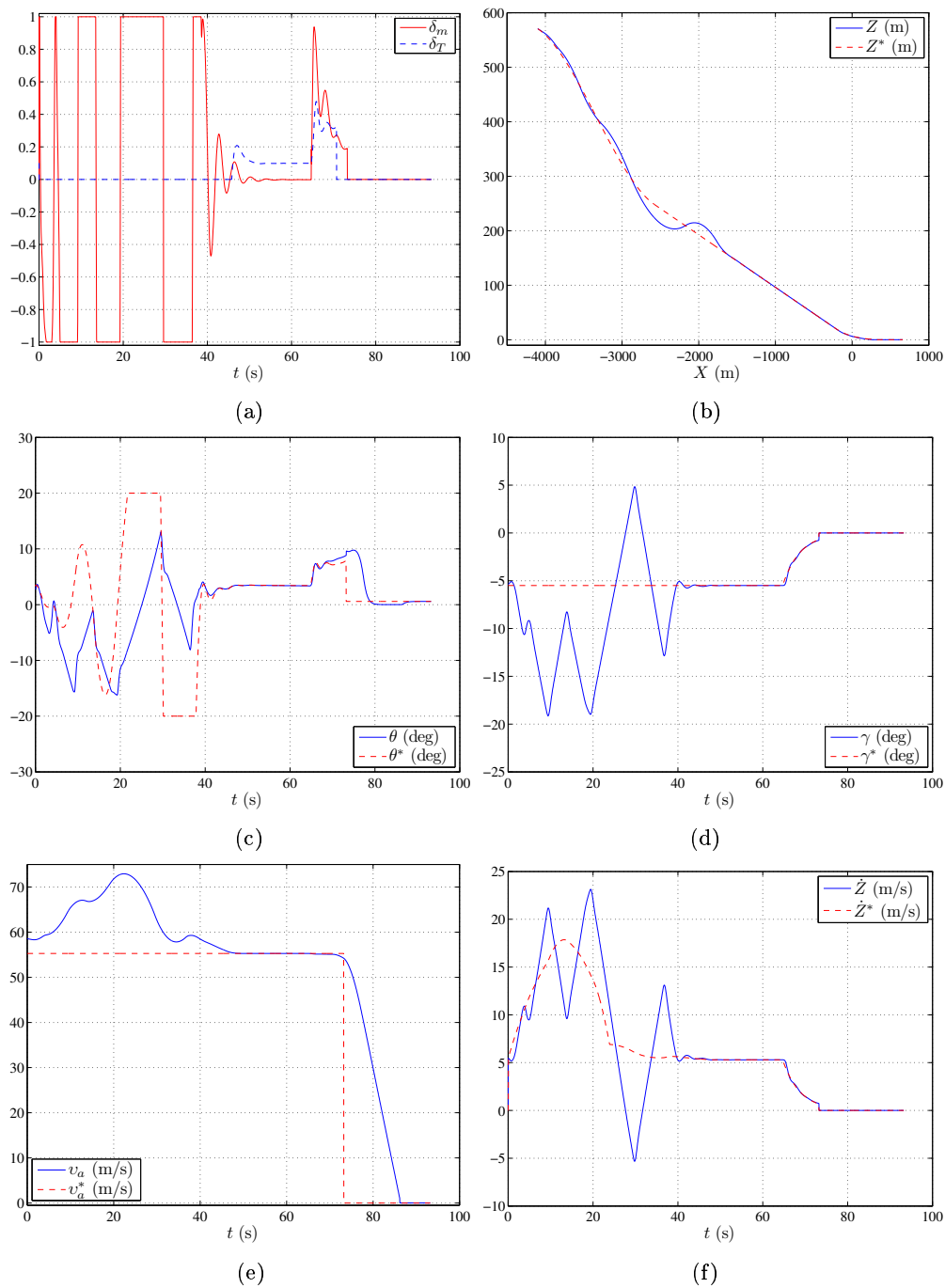


FIG. 3.38 – Résultats de commande longitudinale avec mesures visuelles et trajectoire. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et vitesse verticale de l'avion (f).

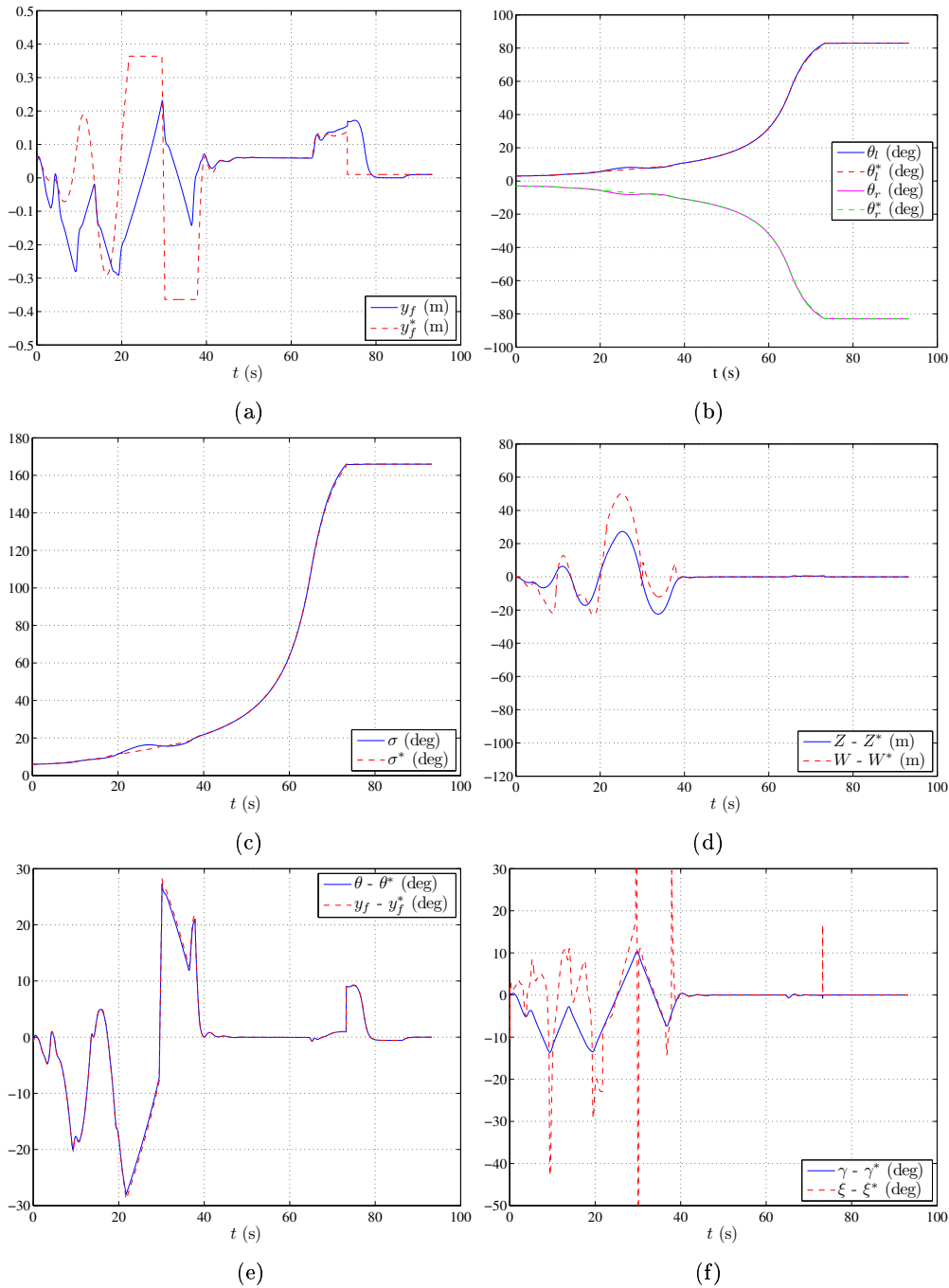


FIG. 3.39 – Résultats de commande longitudinale avec mesures visuelles et trajectoire. Évolution au cours du temps de la position verticale du point de fuite dans l'image (a), des paramètres θ de la projection des lignes gauche et droite de la piste dans l'image (b), de l'information visuelle σ (c). Évolution de l'erreur sur l'altitude Z et sur l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$, utilisée dans la loi de commande (d), évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (e), évolution de l'erreur sur la pente γ et sur l'information visuelle $\xi = \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2(\frac{\sigma}{2})} - \frac{a_{19}}{a_{18}} v_a$ (f).

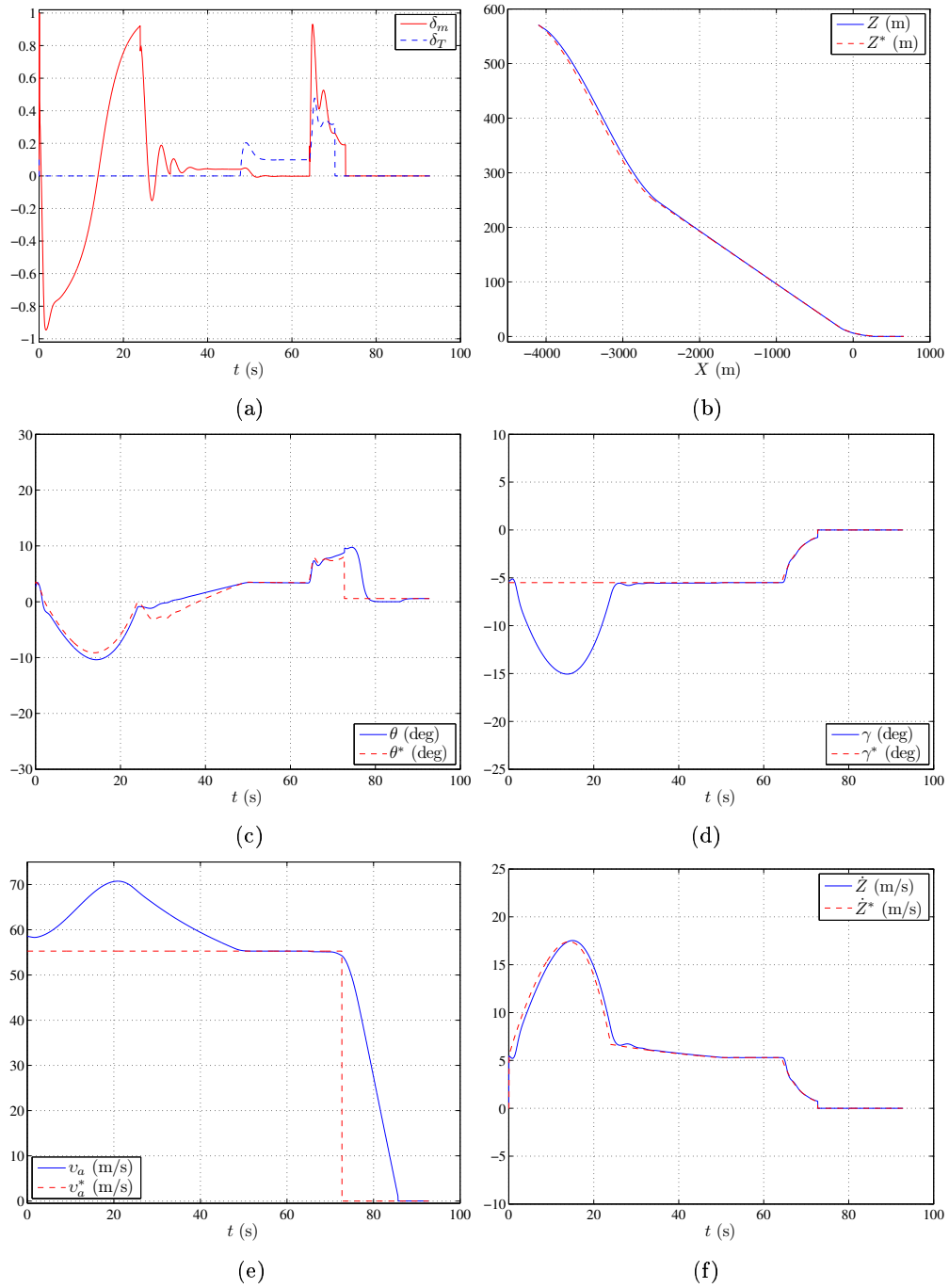


FIG. 3.40 – Résultats de commande longitudinale avec mesures visuelles partielles et trajectoire. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et vitesse verticale de l'avion (f).

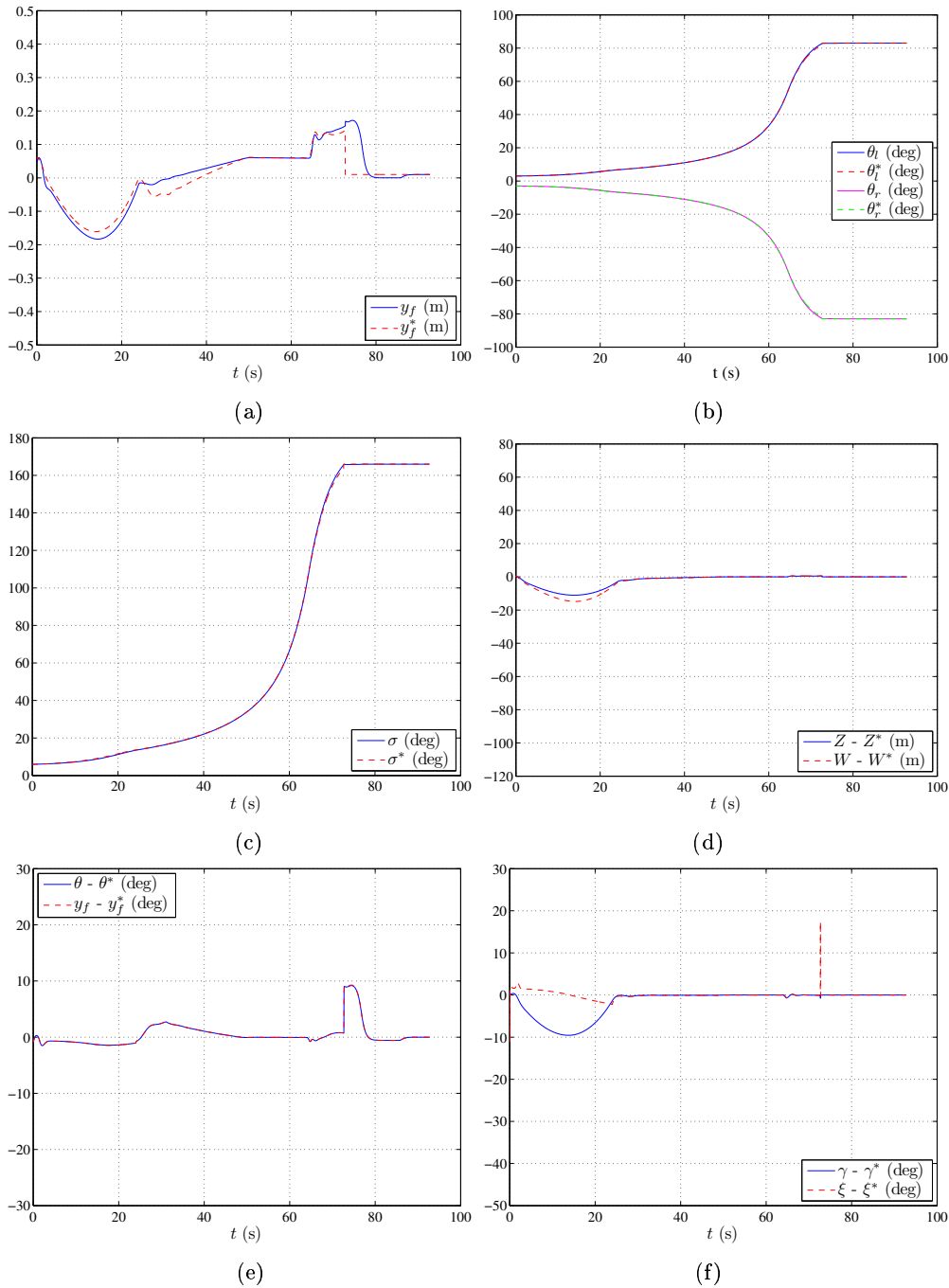


FIG. 3.41 – Résultats de la commande longitudinale avec mesures visuelles partielles et trajectoire. Évolution au cours du temps de la position verticale du point de fuite dans l'image (a), des paramètres θ de la projection des lignes gauche et droite de la piste dans l'image (b), de l'information visuelle σ (c). Évolution de l'erreur sur l'altitude Z et sur l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$ (d), évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (e), évolution de l'erreur sur la pente γ et sur l'information visuelle $\xi = \frac{L}{4a_{18} \sin^2(\frac{\sigma}{2})} - \frac{a_{19}}{a_{18}} v_a$ (f).

3.5.2.2 Commande latérale

On s'intéresse à présent aux mouvements de l'avion dans le plan horizontal : l'avion n'est plus initialement aligné avec l'axe de la piste ($Y \neq 0$). L'objectif consiste alors à aligner l'avion avec l'axe de la piste (voir figure 3.25). Pour cela, on considère la commande de gauchissement, δ_l . Pour chaque simulation, on représentera les paramètres du modèle latéral : l'entrée de commande δ_l , la position latérale Y , et les angles de roulis ϕ et de lacet ψ de l'avion. On représentera également les informations visuelles, pour les simulations qui les utilisent.

On considère les commandes présentées à la section 3.4.1.1.

- D'une part on a considéré la loi de commande (3.126) :

$$\delta_l = -n_1(\ddot{Y} - \ddot{Y}^*) - n_2(\dot{Y} - \dot{Y}^*) - n_3(Y - Y^*) \quad (3.171)$$

avec $n_1 = 0.48$, $n_2 = 0.24$ et $n_3 = 0.04$. Cela permet d'avoir $k_1 = 1.5$, $k_2 = 0.75$, $k_3 = 0.125$, ce qui correspond à un polynôme de la forme $s^3 + k_1s^2 + k_2s + k_3$ ayant -0.5 pour racine triple (on rappelle que $n_i = (-k_ia_1)/(a_5a_3b_1)$, $i \in \{1, 2, 3\}$: voir section 3.4.1.1).

- D'autre part, on a utilisé la loi de commande (3.128) :

$$\delta_l = kp_c \quad (3.172)$$

$$\begin{aligned} \text{tel que } p_c &= \omega_3(\phi_c - \phi) & \text{et } p_c &\in [-p_m, p_m], \\ \text{tel que } \phi_c &= \omega_2 \frac{v_a}{g}(\psi_c - \psi) & \text{et } \phi_c &\in [-\phi_m, \phi_m], \\ \text{tel que } \psi_c &= \omega_1 \frac{1}{v_a}(Y_c - Y) & \text{et } \psi_c &\in [-\psi_m, \psi_m], \\ \text{tel que } Y_c &= 0, \end{aligned}$$

et avec $k = 2.9$, $\omega_1 = 0.08$ rad/s, $\omega_2 = 0.2$ rad/s, $\omega_3 = 0.5$ rad/s, $p_m = \pi/9$ rad/s, $\phi_m = \pi/3$ rad, $\psi_m = \pi/2$ rad.

On a ensuite utilisé les commandes avec informations visuelles ((3.127) et figure 3.27), en conservant les mêmes valeurs de gains n_i et ω_i ($i \in \{1, 2, 3\}$).

Dans toute cette partie, on s'intéresse essentiellement aux mouvements latéraux. Concernant les mouvements longitudinaux, la loi de commande (3.170) est appliquée afin de maintenir les conditions nominales et de réaliser la manœuvre d'arrondi. Les conditions initiales relatives aux mouvements longitudinaux sont très proches des conditions nominales, ce qui limite fortement les mouvements longitudinaux. On peut donc négliger les mouvements longitudinaux et se focaliser sur les mouvements latéraux.

Commande avec mesures directes. L'avion se trouve initialement à $Y_i = -150$ m à gauche de la piste, et est orienté de $\psi_i = -3$ deg vers la gauche (comme sur la figure 3.28).

Dans un premier temps, on applique la loi de commande (3.171), en supposant que la mesure de Y est disponible directement. On estime alors \dot{Y} et \ddot{Y} en dérivant successivement Y et \dot{Y} , comme on l'a fait précédemment pour obtenir des termes dérivés

dans la commande longitudinale. Les résultats obtenus sont représentés sur la figure 3.42.

On voit que tous les paramètres convergent bien vers zéro, assez rapidement (après environ 25 secondes). Cependant, les angles de roulis et de lacet deviennent très importants pendant la phase transitoire (figure 3.42.c et 3.42.d). De plus, la commande est également très importante et est donc saturée pendant la phase transitoire (figure 3.42.a). À la vue de ces résultats, il n'est donc pas souhaitable d'utiliser cette loi de commande en pratique.

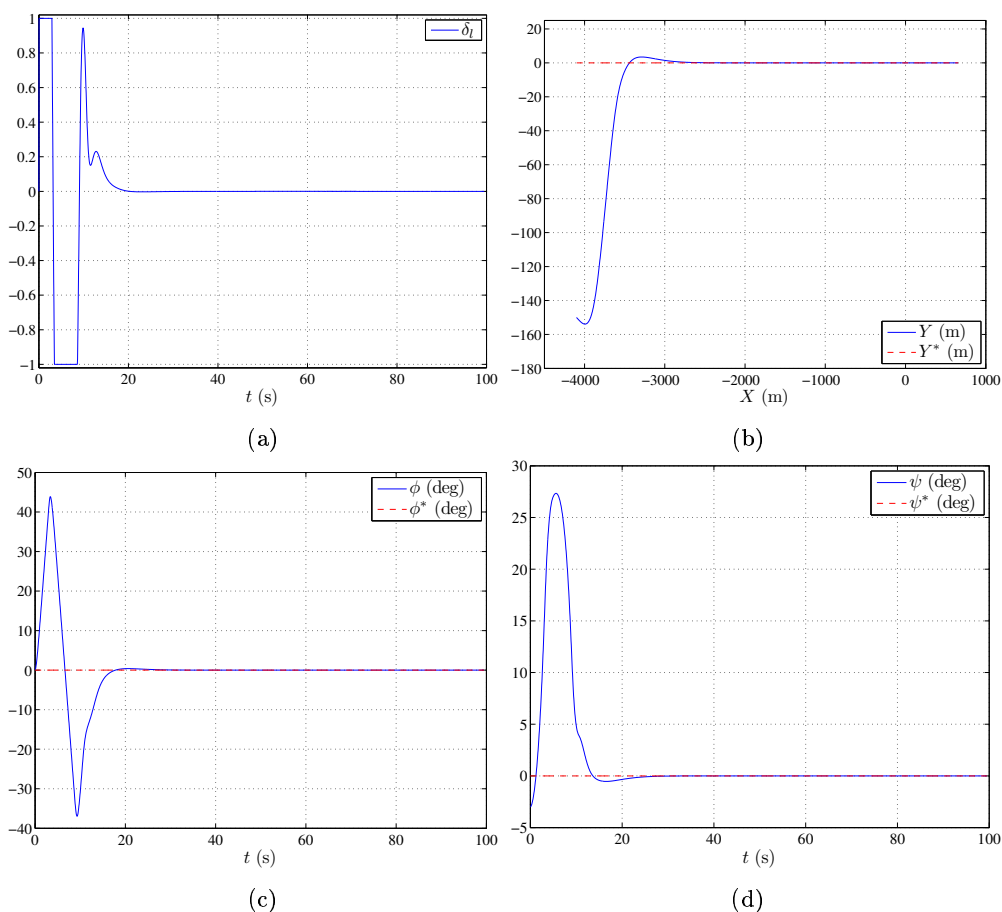


FIG. 3.42 – Résultats de la commande latérale (3.171) avec mesures directes. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et angle de lacet (d).

On a ensuite appliqué la commande (3.172), avec les mêmes conditions initiales : $Y_i = -150$ m et $\psi_i = -3$ deg. Les angles ϕ et ψ , ainsi que la mesure de Y sont supposés être mesurés directement : en pratique, le simulateur utilisé donne accès à ces paramètres. On obtient les résultats de la figure 3.43.

Les paramètres latéraux convergent à nouveau tous vers zéro, en un temps de l'ordre

de 60 secondes. L'entrée de commande n'est pas saturée, et présente une évolution douce (figure 3.43.a). Les angles de roulis et de lacet atteignent des valeurs de l'ordre de 10 degrés, ce qui semble acceptable en comparaison des 25 à 40 degrés obtenus précédemment (comparer les figures 3.42.c, 3.42.d et figures 3.43.c, 3.43. d).

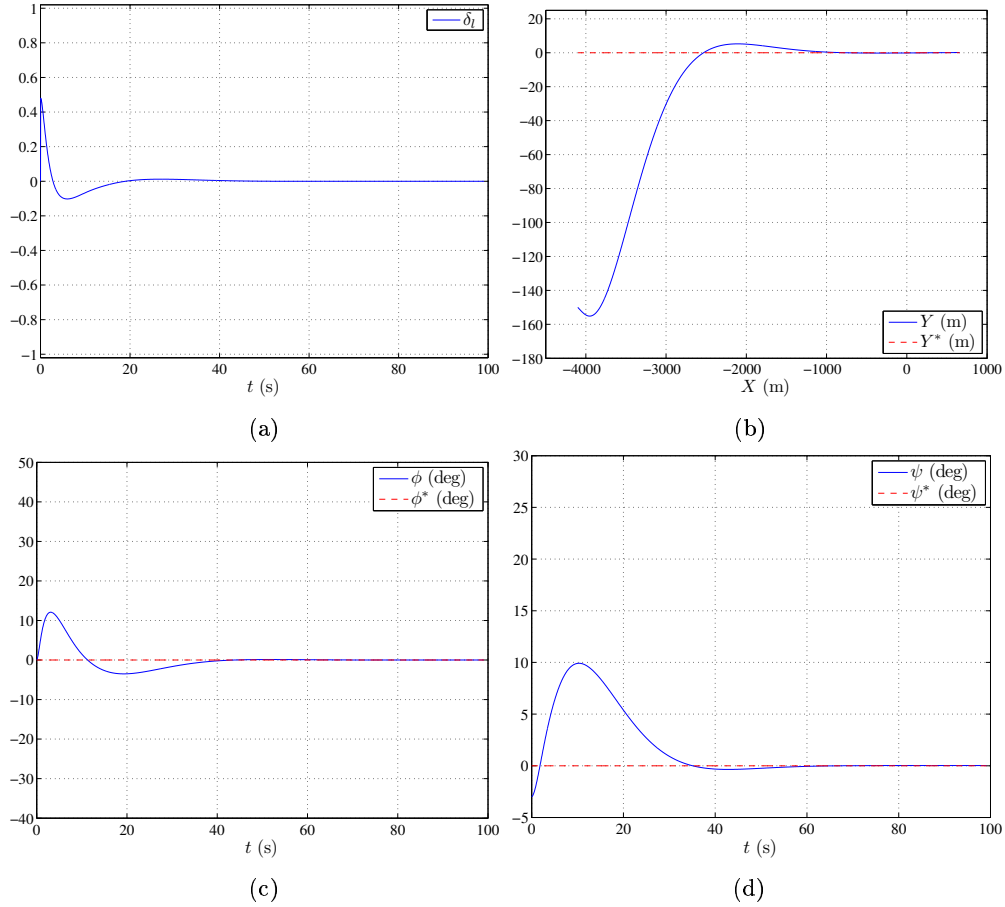


FIG. 3.43 – Résultats de la commande latérale (3.172) avec mesures directes. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

Commande visuelle. Rappelons que l'objectif de notre travail est de mettre au point des lois de commande visuelles. On s'intéresse donc à présent aux lois de commande latérales intégrant des mesures issues de la caméra. Les conditions initiales utilisées sont toujours les mêmes que précédemment : $Y_i = -150$ m et $\psi_i = -3$ deg.

Considérons la loi de commande (3.127) :

$$\delta_l = n_1 a_5 (\dot{x}_f - \dot{x}_f^*) + n_2 a_5 (x_f - x_f^*) - n_3 \left(\frac{H}{c^2 \theta_l + c^2 \theta_r - 2} \delta - \frac{H^*}{c^2 \theta_l^* + c^2 \theta_r^* - 2} \delta^* \right). \quad (3.173)$$

L'abscisse x_f du point de fuite, les valeurs des paramètres $\theta_{\{l,r,c\}}$, ainsi que l'information visuelle $\delta = \theta_l + \theta_r - 2\theta_c$ sont déterminées à chaque itération en projetant les lignes de la piste sur le plan image de la caméra. La dérivée \dot{x}_f est estimée à partir de deux mesures successives de x_f , de la même manière que \dot{y}_f dans le cas longitudinal. On n'a pas utilisé la génération de trajectoire : les informations visuelles désirées \dot{x}_f^* , x_f^* , θ_l^* , θ_r^* , δ^* sont fixes. L'altitude courante H est supposée être mesurée par un altimètre, H^* est obtenue à partir de la pente nominale d'approche.

Les résultats sont présentés sur les figures 3.44 et 3.45. La figure 3.46 montre la vue de la caméra embarquée à différents instants de la manœuvre.

On voit sur la figure 3.44.a que l'entrée de commande converge vers zéro, assez rapidement (après environ 10 secondes). Cependant, la position latérale, et les angles de roulis et de lacet ne se stabilisent pas à zéro : ils continuent à évoluer (voir figures 3.44.b, 3.44.c, 3.44.d). Notons que l'avion touche le sol à l'instant $t = 78.92$ s : c'est pour cette raison que le roulis est soudainement ramené à 0, et que le lacet reste constant à partir de cet instant (voir figure 3.44.c, 3.44.d). Pour comprendre ce qui se passe, il faut regarder l'évolution des informations visuelles utilisées (figure 3.45). L'abscisse du point de fuite, les paramètres $\theta_{\{l,r,c\}}$ des droites et l'information visuelle δ ne convergent pas vers leurs valeurs désirées (voir figures 3.45.a, 3.45.b, 3.45.c). La figure 3.46 illustre bien ce phénomène : les lignes courantes ne se superposent pas aux lignes désirées. Les erreurs sur les informations visuelles $\eta = \frac{1}{a_3}\dot{x}_f$ et x_f coïncident bien avec les erreurs sur les angles de roulis et de lacet (voir figures 3.45.e, 3.45.f). Cependant, elles ne convergent pas vers zéro pendant la phase de descente (avant $t = 78$ s). Quant à l'erreur sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2} \delta$, elle ne correspond pas à l'erreur sur la position latérale Y (voir figure 3.45.d). Cela est dû au fait que la position initiale est éloignée de la position nominale utilisée lors de l'étape de modélisation présentée à la section 3.3.2. De plus, comme la position désirée est fixe, l'avion se trouve initialement "loin" de la position désirée (dans l'axe de la piste), et il ne parvient pas à s'aligner avec l'axe de la piste. L'entrée de commande converge vers 0, mais les informations visuelles ne rejoignent pas leurs positions désirées : les différents termes de la loi de commande se compensent. En particulier, les valeurs désirées des paramètres $\theta_{\{l,r,c\}}$ sont telles que $\delta = \theta_l + \theta_r - 2\theta_c = 0$, avec $\theta_l = -\theta_r$ et $\theta_c = 0$ (voir figures 3.45.b, 3.45.c : les lignes désirées sont centrées et placées symétriquement dans l'image, comme on peut le voir sur la figure 3.46). Or, on voit sur la figure 3.45.b (et sur la figure 3.46) que les lignes courantes tendent à converger vers une position telle que $\theta_l = \theta_r = \theta_c$. On a alors également $\delta = \theta_l + \theta_r - 2\theta_c \simeq 0$ dans cette configuration, mais cela ne correspond plus à une position latérale Y nulle.

On a représenté sur les figures 3.47 et 3.48 les résultats obtenus en appliquant la loi de commande (3.172), avec des mesures visuelles :

$$\delta_l = kp_c \quad (3.174)$$

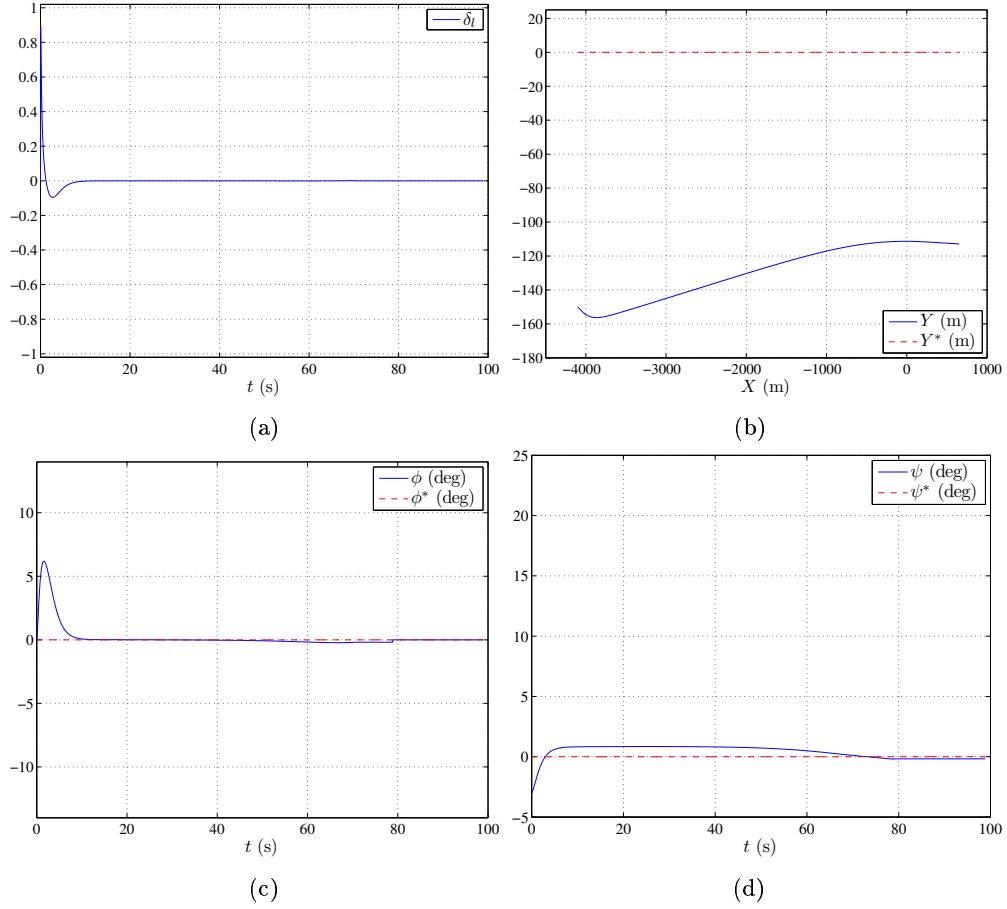


FIG. 3.44 – Résultats de la commande latérale (3.173) avec mesures visuelles. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

$$\begin{aligned}
 \text{tel que } p_c &= -\omega_3 \frac{v_a}{g} (\dot{x}_{f_c} - \dot{x}_f) & \text{et } p_c &\in [-p_m, p_m], \\
 \text{tel que } \dot{x}_{f_c} &= \omega_2 (x_{f_c} - x_f) & \text{et } \dot{x}_{f_c} &\in [-\dot{x}_{f_{cm}}, \dot{x}_{f_{cm}}], \\
 \text{tel que } x_{f_c} &= -\omega_1 \frac{1}{v_a} \frac{H}{c^2 \theta_l + c^2 \theta_r - 2} (\delta_c - \delta) & \text{et } \psi_c &\in [-x_{f_{cm}}, x_{f_{cm}}], \\
 \text{tel que } \delta_c &= 0,
 \end{aligned}$$

et avec $k = 2.9$, $\omega_1 = 0.08$ rad/s, $\omega_2 = 0.2$ rad/s, $\omega_3 = 0.5$ rad/s, $p_m = \pi/9$ rad/s, $\dot{x}_{f_{cm}} = \frac{g}{v_a} \phi_m = \frac{g}{v_a} \frac{\pi}{3}$ rad, $x_{f_{cm}} = \psi_m = \frac{\pi}{2}$ rad.

Les résultats sont très similaires à ceux obtenus avec la loi de commande (3.173). Pour conclure, on voit que l'utilisation d'informations visuelles dans la loi de commande pose problème lorsque la consigne est trop éloignée de l'état initial, ce qui est somme toute logique, au vu des simplifications effectuées durant la phase de modélisation.

Suivi de trajectoire. Pour que la consigne reste proche de l'état courant, on utilise les trajectoires déterminées à la section 3.4.2.1 pour les valeurs désirées de Y , ϕ , et ψ . Les paramètres D et k sont choisis empiriquement : on a pris $k = 0.26$ et $D = 40$, et saturé t_f entre 20 et 40 secondes. Comme dans le cas longitudinal, les consignes pour les informations visuelles sont obtenues en projetant les lignes de la piste dans le plan image de la caméra. Comme décrit à la section 3.4.3, la caméra est pour cela placée à chaque instant à la position et avec l'orientation désirée, celles-ci étant issues de la génération de trajectoire présentée à la section 3.4.2.1.

On a utilisé la loi de commande (3.165), où les consignes évoluent au cours du temps. Les résultats obtenus sont présentés sur les figures 3.49, 3.50, et 3.51. On voit immédiatement que l'utilisation de la trajectoire permet à l'avion de se placer dans l'alignement de l'axe de la piste (voir figure 3.49). La consigne converge vers zéro (voir figure 3.49.a), et permet d'annuler la position latérale Y , l'angle de roulis ϕ et l'angle de lacet ψ en un peu plus de 40 secondes (voir figure 3.49.b, 3.49.c, 3.49.d). Les valeurs courantes suivent globalement les valeurs désirées. On peut voir sur les figures 3.50.a, 3.50.b, 3.50.c que les informations visuelles convergent vers leurs positions désirées. On voit également sur la figure 3.51 que les lignes courantes rejoignent les lignes désirées, et que l'avion se trouve aligné avec la piste à $t = 50$ s. Notons que δ semble diverger à partir de $t \simeq 78$ s. Mais on rappelle que l'avion se pose à l'instant $t \simeq 78$ s : la commande latérale est alors désactivée, et l'évolution de δ n'est donc plus prise en compte dans la commande. Sur les figures 3.50.e et 3.50.f, on voit comme précédemment que les erreurs sur les informations visuelles coïncident assez bien avec les erreurs sur le roulis et le lacet. En ce qui concerne l'erreur latérale $Y - Y^*$, (figure 3.50.d), l'erreur sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$ n'en donne une bonne représentation qu'à partir de $t \simeq 30$ s. On voit que cet instant correspond au moment où les lignes de la piste se répartissent symétriquement dans l'image ($\theta_l > 0$, $\theta_r \simeq -\theta_l$, et $\theta_c \simeq 0$, voir figures 3.51.c et 3.51.d). Cela est dû au fait que l'on a considéré, pour la modélisation des informations visuelles, une position nominale particulière, dont l'avion se trouve très éloigné au départ. On voit cependant que cela entraîne uniquement un léger "soubresaut" dans la commande et sur l'angle de roulis (voir figures 3.49.a et 3.49.c), mais n'empêche pas l'avion de s'aligner avec l'axe de la piste.

Limites de la commande visuelle. On considère à présent une nouvelle position initiale : $Y_i = -300$ m, $\phi = 0$ deg, et $\psi_i = 20$ deg.

Dans un premier temps, la même loi de commande que précédemment est utilisée (3.165), avec les informations visuelles et les valeurs désirées définies par les trajectoires de la section 3.4.2.1. Les résultats sont représentés sur les figures 3.52, 3.53, et 3.54. Les résultats sont très similaires à ceux obtenus précédemment (comparer les figures 3.49, 3.50, et figures 3.52, 3.53). On remarque cependant que le pic lié à l'information visuelle V est ici beaucoup plus marqué que précédemment (voir figures 3.50.d et 3.53.d). Il conduit à un pic sur la loi de commande et donc sur l'angle de roulis (voir figures 3.52.a, 3.52.c). On voit sur la figure 3.54 que les lignes de la piste sont initialement très fortement inclinées (paramètres $\theta_{\{l,r,c\}}$ de l'ordre de 40 degrés, voir figure 3.53.b). Notons que malgré le phénomène transitoire mentionné, le système converge à la position

désirée.

Nous allons voir qu'en remplaçant l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$ par la mesure directe de Y dans la loi de commande (3.165), le phénomène de pic pendant la phase transitoire disparaît.

On utilise donc à présent la loi de commande suivante :

$$\delta_l = n_1 a_5 (\dot{x}_f - \dot{x}_f^*) + n_2 a_5 (x_f - x_f^*) - n_3 (Y - Y^*). \quad (3.175)$$

Les résultats sont présentés sur les figures 3.55 et 3.56. L'avion converge cette fois "en douceur" vers sa position désirée (voir figure 3.55). Bien que seules les informations visuelles x_f et $\eta = \frac{1}{a_3} \dot{x}_f$ soient utilisées dans cette expérience, on a quand même représenté l'évolution de toutes les informations visuelles. On constate le même phénomène que précédemment : les erreurs sur le roulis et le lacet sont correctement représentées par les erreurs sur les informations visuelles η et x_f . Par contre, lorsque les droites de l'image se trouvent trop loin de leur position nominale, l'erreur sur V représente mal l'évolution de l'erreur latérale $Y - Y^*$.

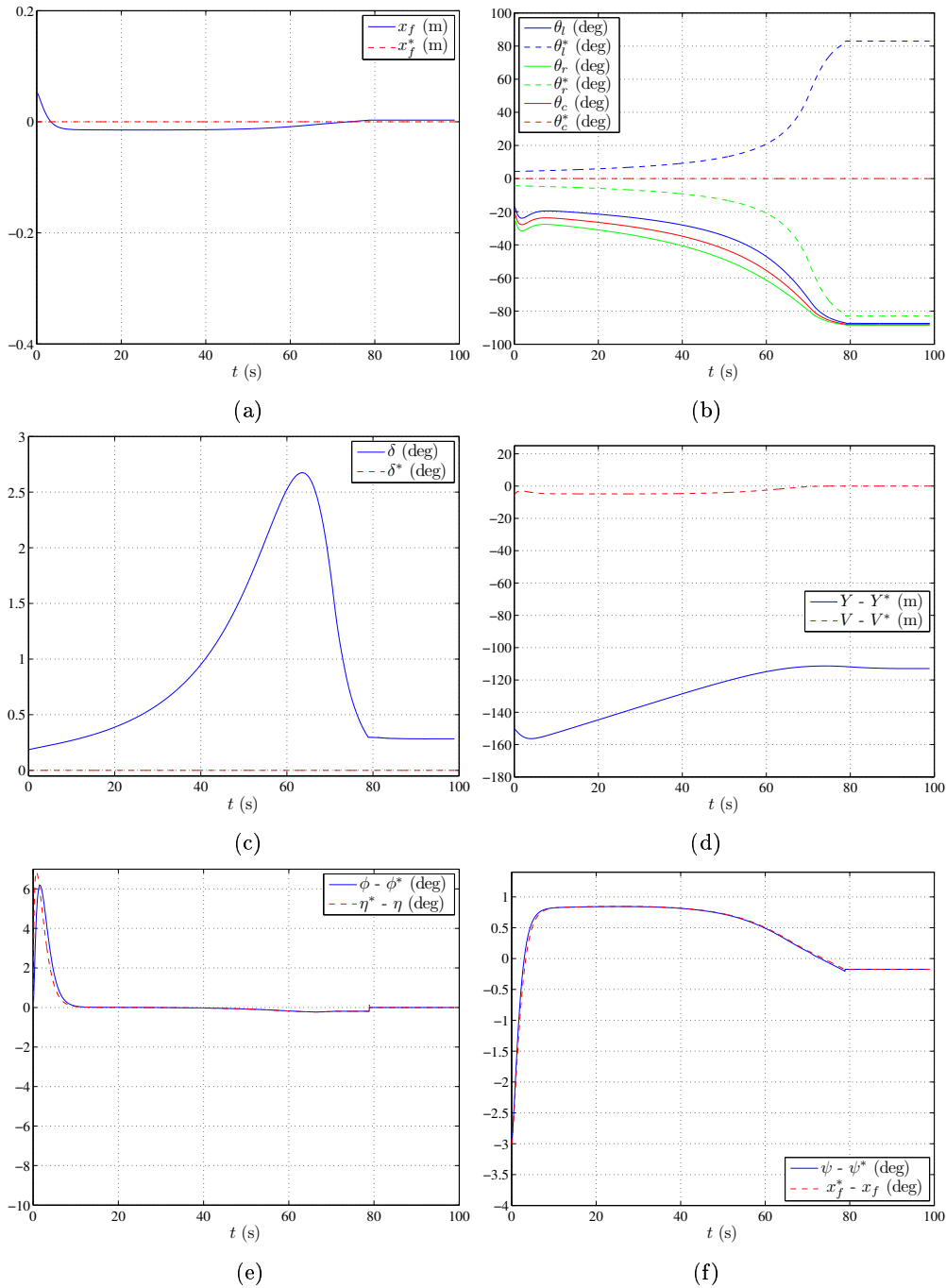


FIG. 3.45 – Résultats de la commande latérale (3.173) avec mesures visuelles. Évolution au cours du temps de la position horizontale du point de fuite dans l'image (a), des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (b), de l'information visuelle δ (c). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$, utilisée dans la loi de commande (d), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3}\dot{x}_f$ (e), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (f).

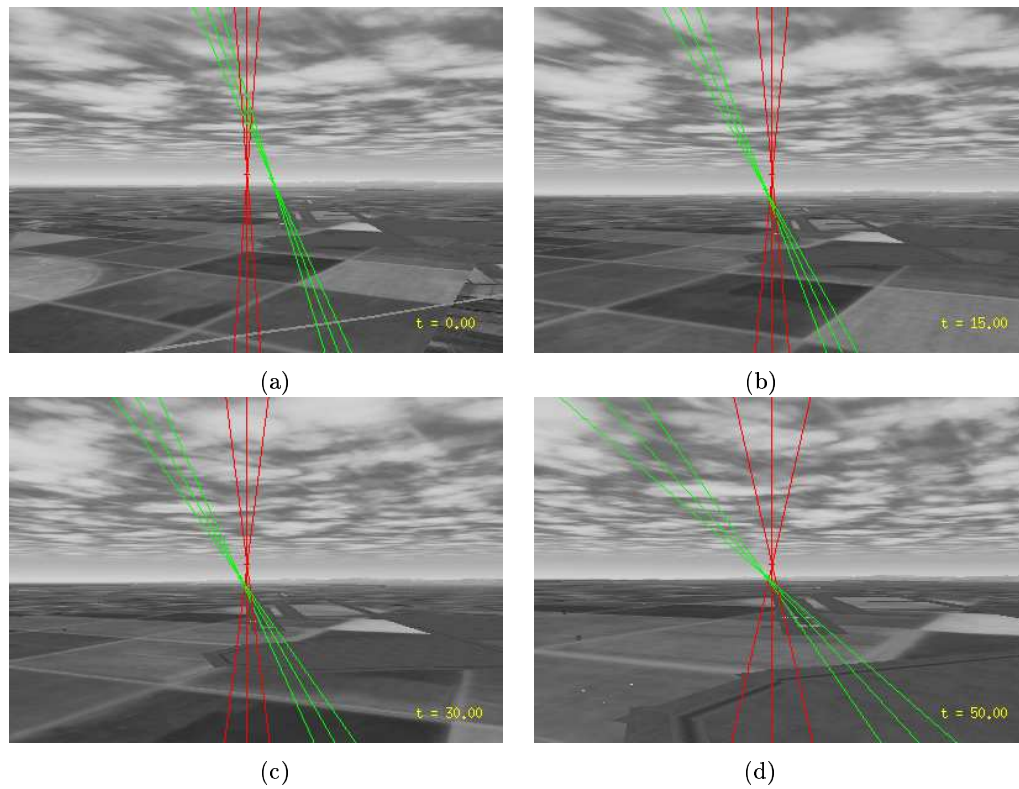


FIG. 3.46 – Vues de la caméra embarquée, aux étapes-clé de l'atterrissage, lorsque la consigne est fixe. (a) : position initiale ($t = 0$ s); (b) ($t = 15$ s); (c) : ($t = 30$ s); (d) : toucher de la piste ($t = 50$ s). Les lignes de la piste sont ici assimilées à des droites, c'est pourquoi elles se poursuivent au-delà du point de fuite. Le point de fuite est représenté par une croix. En vert les informations visuelles courantes sont représentées, et leurs positions désirées dans l'image sont tracées en rouge.

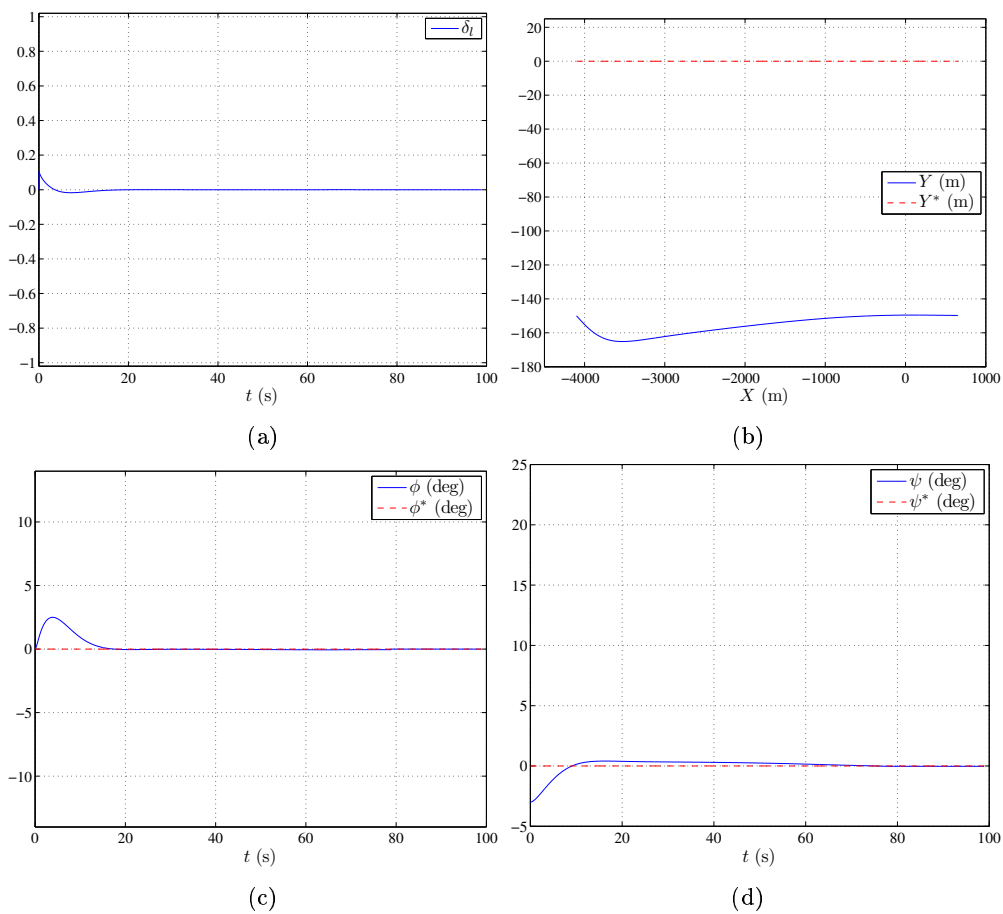


FIG. 3.47 – Résultats de la commande latérale (3.174) avec mesures visuelles. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

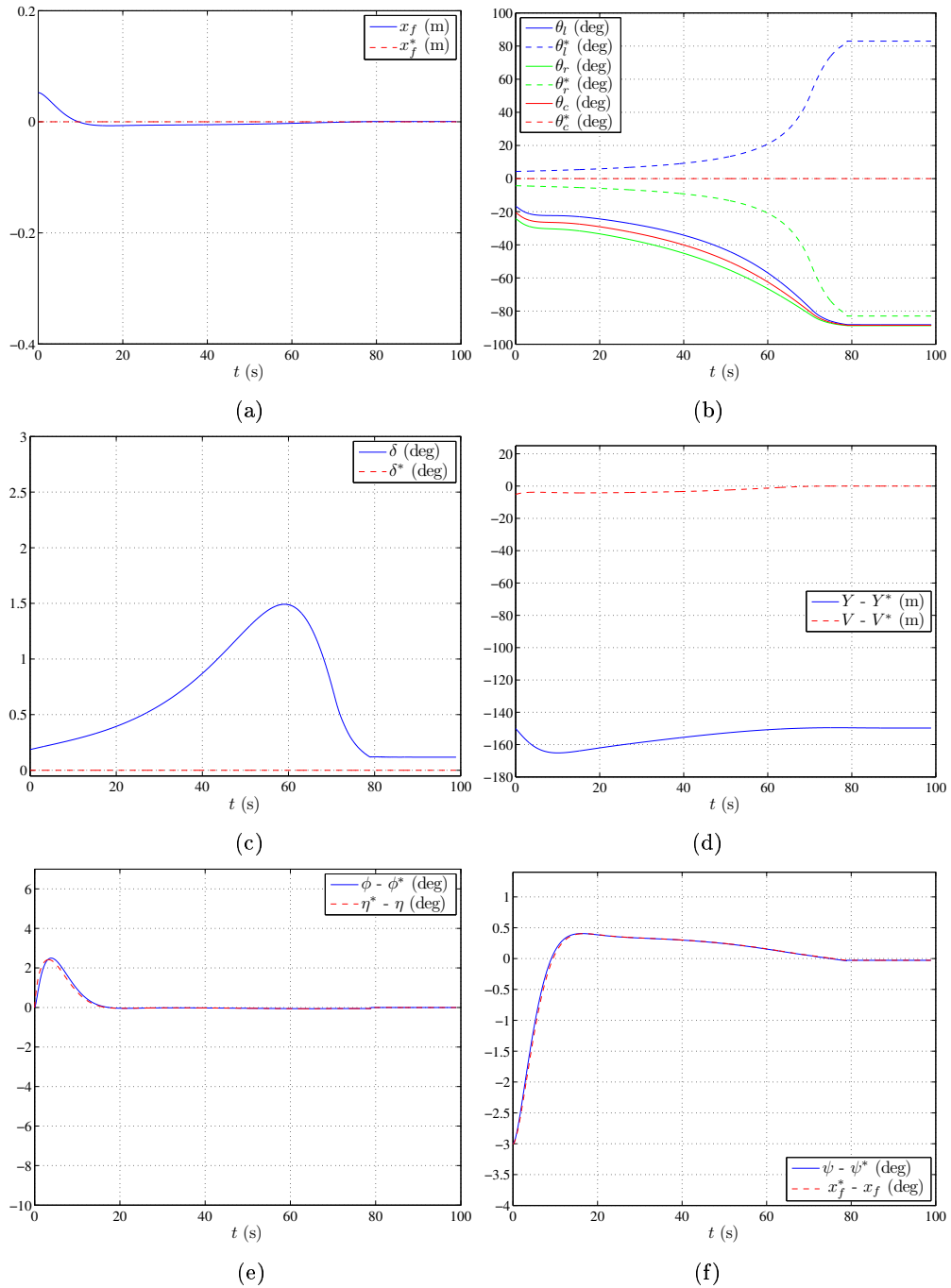


FIG. 3.48 – Résultats de la commande latérale (3.174) avec mesures visuelles. Évolution au cours du temps de la position horizontale du point de fuite dans l'image (a), des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (b), de l'information visuelle δ (c). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$, utilisée dans la loi de commande (d), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3}\dot{x}_f$ (e), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (f).

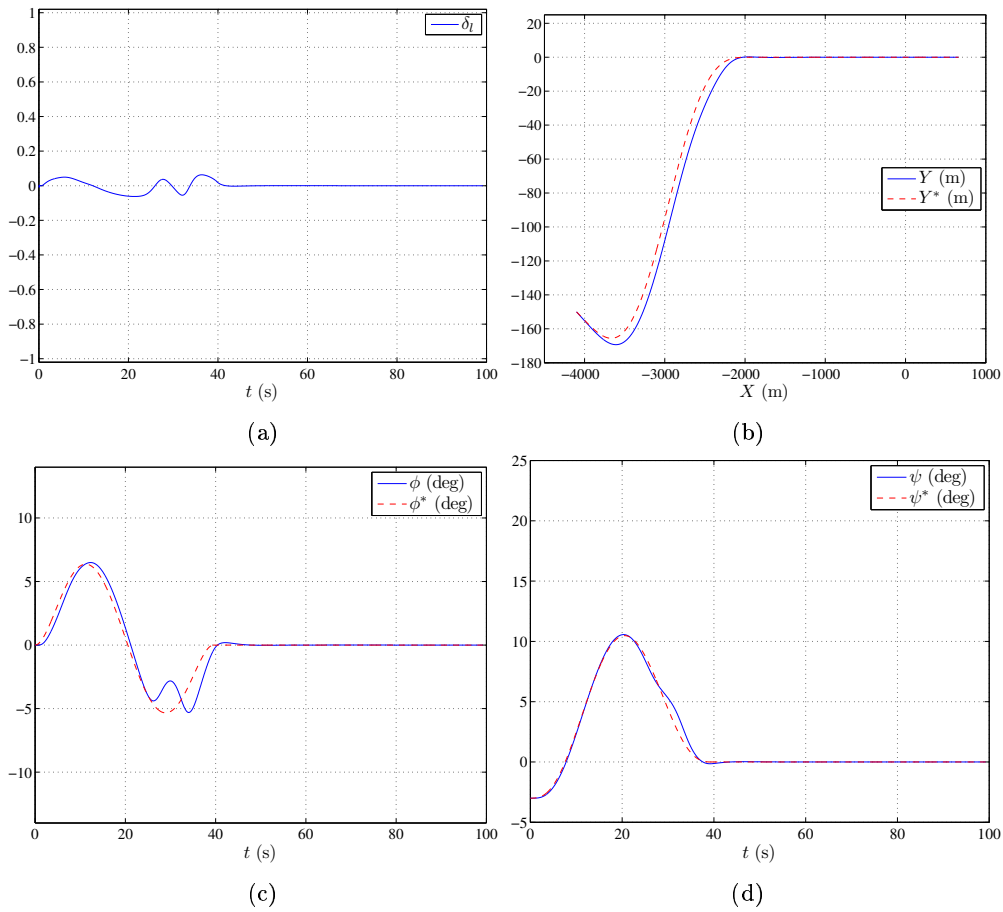


FIG. 3.49 – Résultats de la commande latérale (3.165) avec mesures visuelles et trajectoire. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

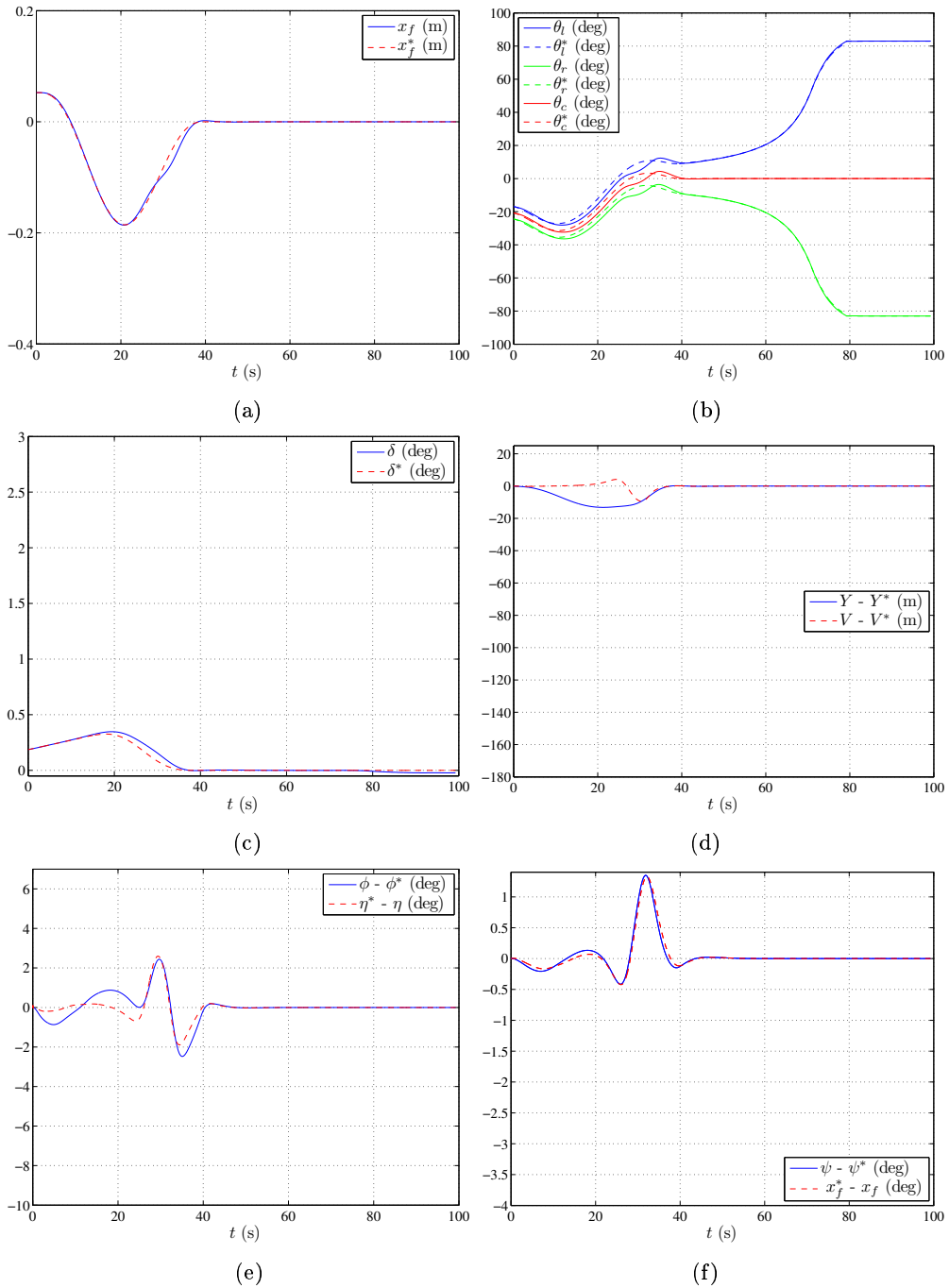


FIG. 3.50 – Résultats de la commande latérale (3.165) avec mesures visuelles et trajectoire. Évolution au cours du temps de la position horizontale du point de fuite dans l'image (a), des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (b), de l'information visuelle δ (c). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$, utilisée dans la loi de commande (d), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3}\dot{x}_f$ (e), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (f).

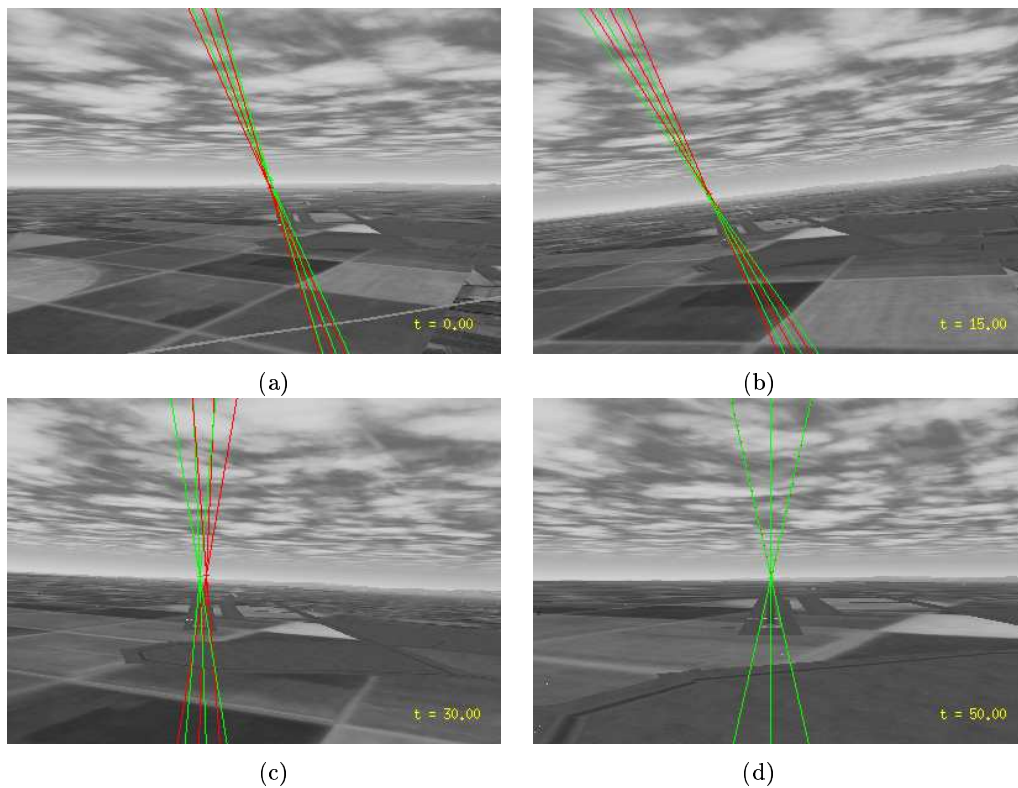


FIG. 3.51 – Vues de la caméra embarquée, aux étapes-clef de l'atterrissage, lorsque la consigne est spécifiée sous forme de trajectoire. (a) : position initiale $t = 0$ s ; (b) $t = 15$ s ; (c) : $t = 30$ s ; (d) : toucher de la piste $t = 50$ s. Les lignes de la piste sont ici assimilées à des droites, c'est pourquoi elles se poursuivent au-delà du point de fuite. Le point de fuite est représenté par une croix. En vert les informations visuelles courantes sont représentées, et leurs positions désirées dans l'image sont tracées en rouge.

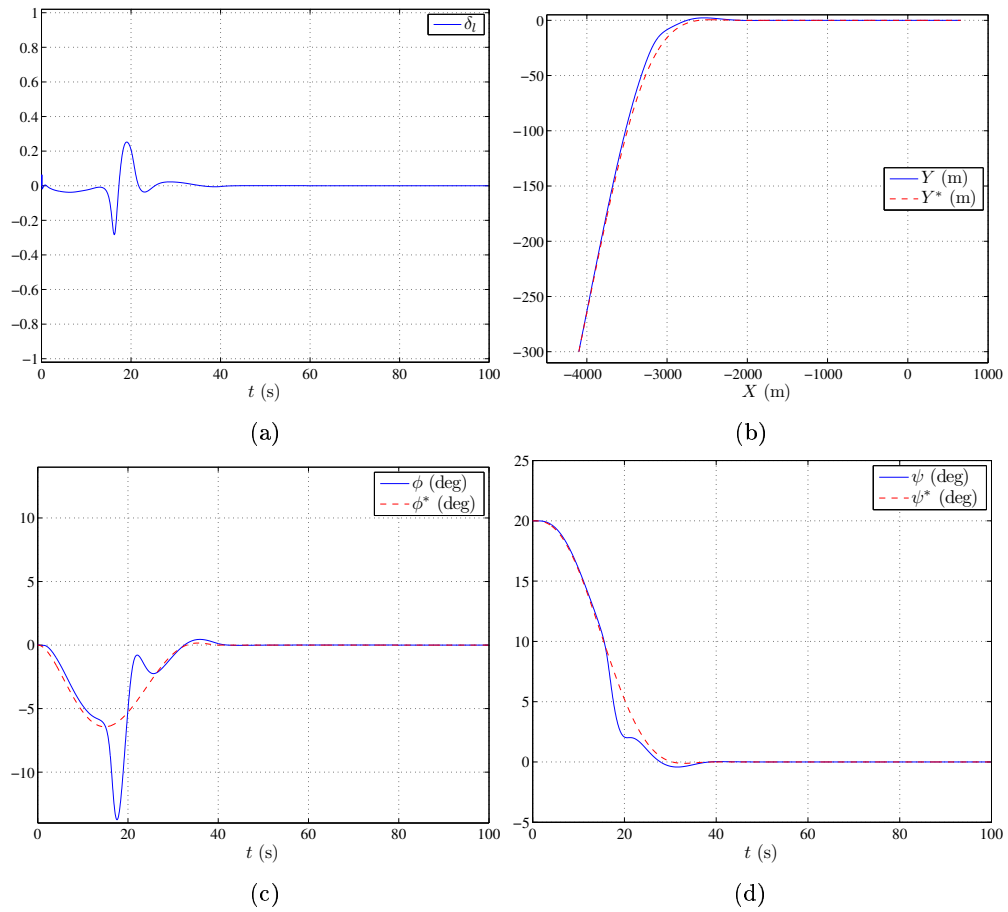


FIG. 3.52 – Résultats de la commande latérale (3.165) avec mesures visuelles et trajectoire. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

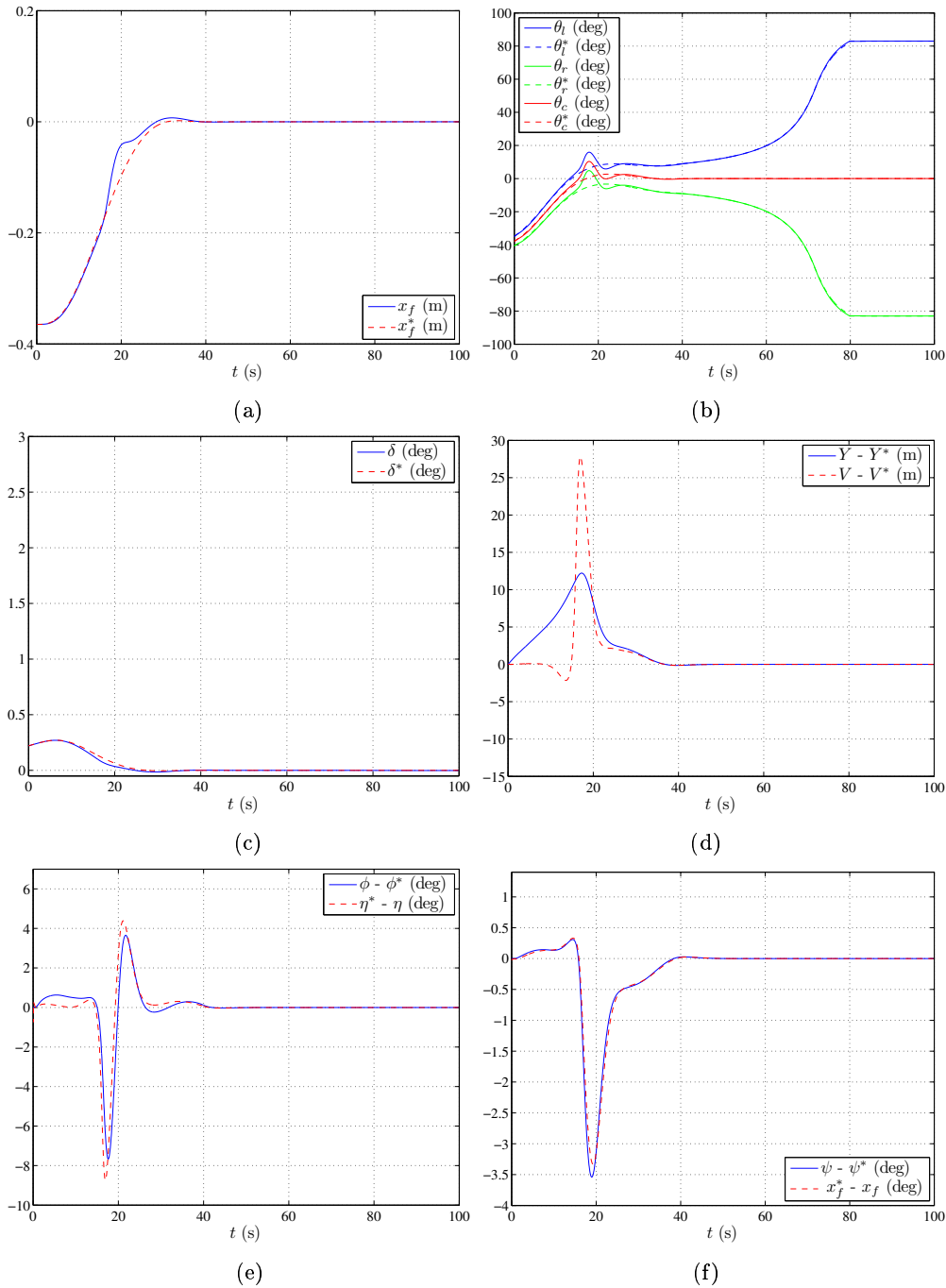


FIG. 3.53 – Résultats de la commande latérale (3.165) avec mesures visuelles et trajectoire. Évolution au cours du temps de la position horizontale du point de fuite dans l'image (a), des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (b), de l'information visuelle δ (c). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$, utilisée dans la loi de commande (d), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3}\dot{x}_f$ (e), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (f).

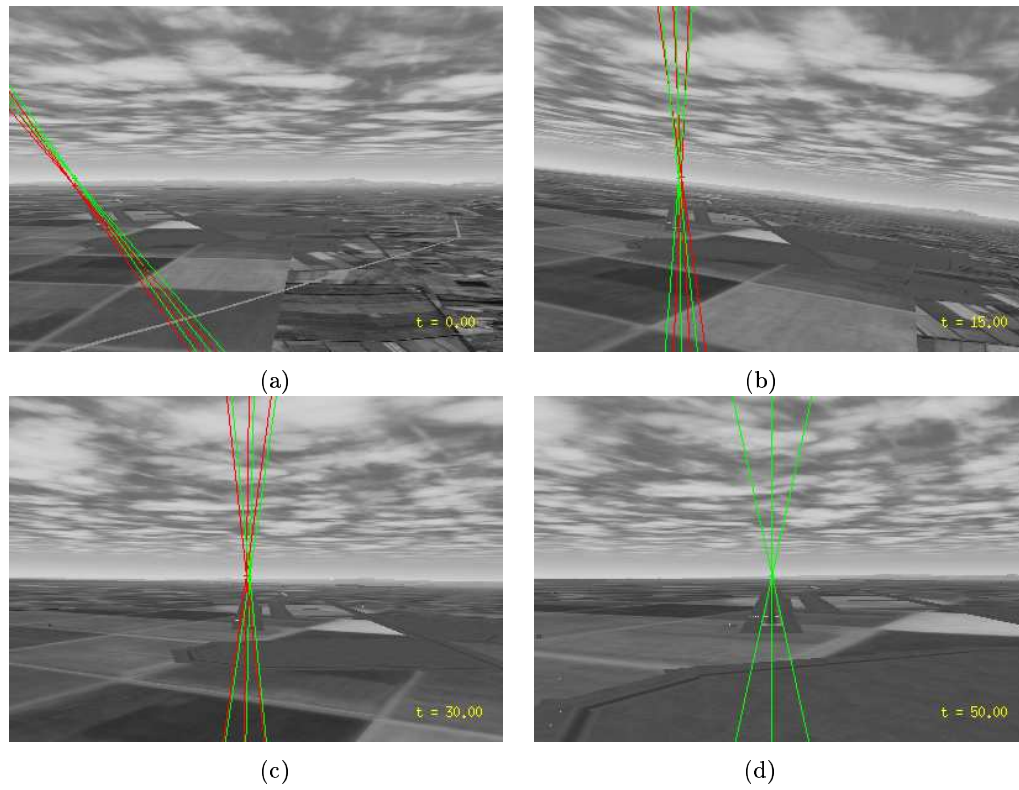


FIG. 3.54 – Vues de la caméra embarquée, aux étapes-clé de l’atterrissage, lorsque la consigne est spécifiée sous forme de trajectoire. (a) : position initiale $t = 0$ s ; (b) $t = 15$ s ; (c) : $t = 30$ s ; (d) : toucher de la piste $t = 50$ s. Les lignes de la piste sont ici assimilées à des droites, c’est pourquoi elles se poursuivent au-delà du point de fuite. Le point de fuite est représenté par une croix. En vert les informations visuelles courantes sont représentées, et leurs positions désirées dans l’image sont tracées en rouge.

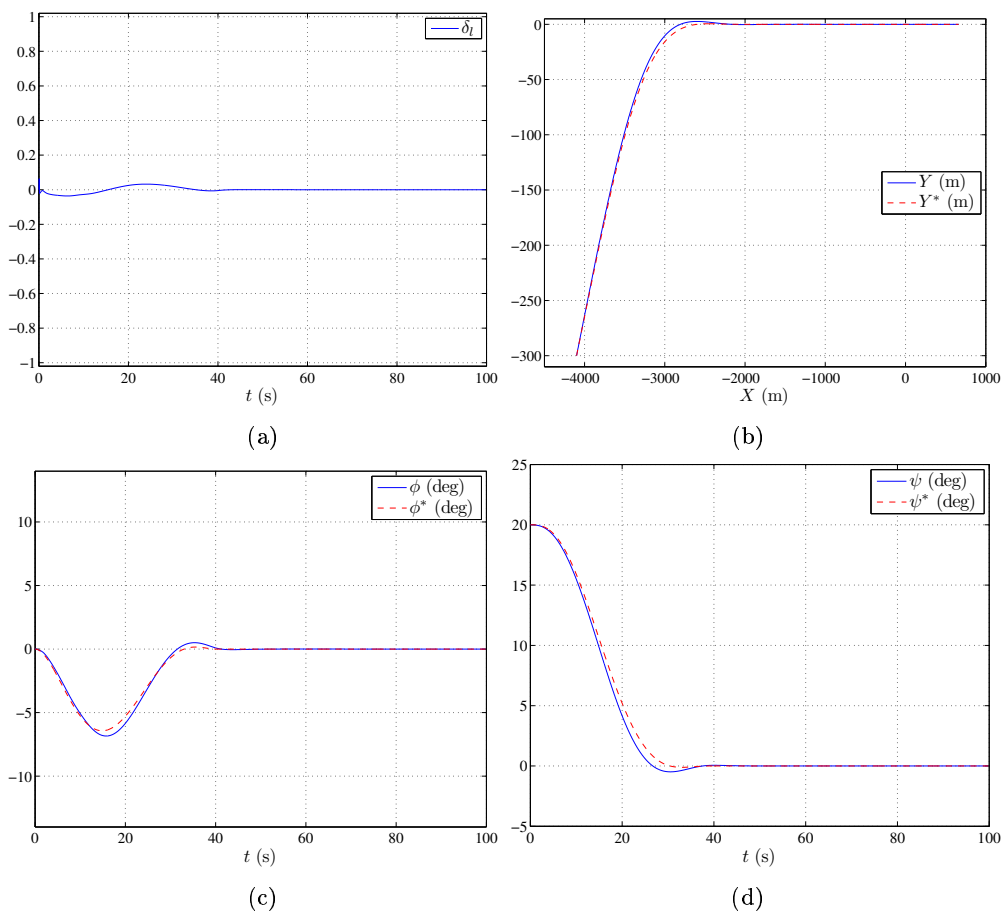


FIG. 3.55 – Résultats de la commande latérale (3.175) avec mesures visuelles partielles et trajectoire. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

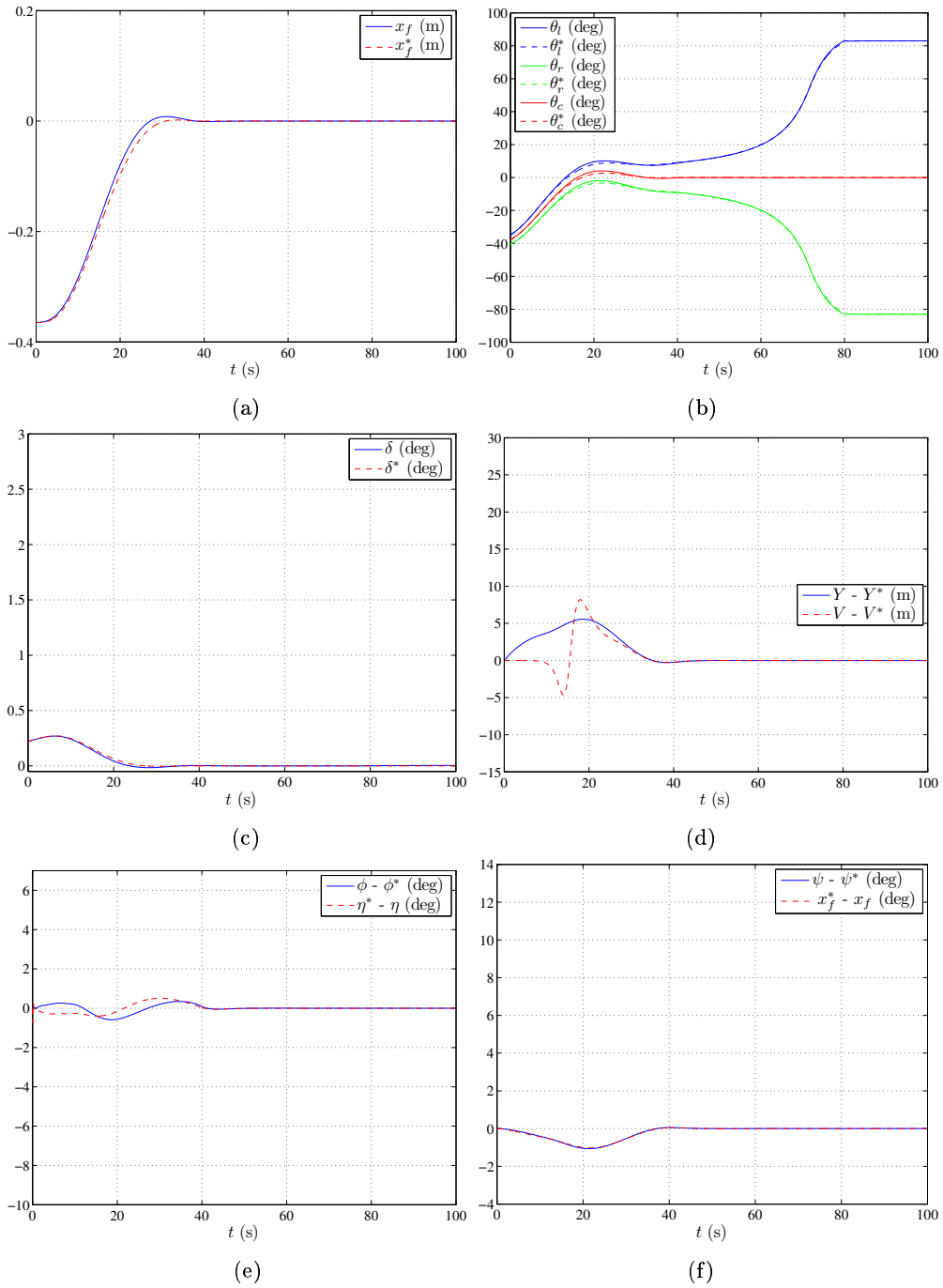


FIG. 3.56 – Résultats de la commande latérale (3.175) avec mesures visuelles partielles et trajectoire. Évolution au cours du temps de la position horizontale du point de fuite dans l'image (a), des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (b), de l'information visuelle δ (c). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$ (d), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3}\dot{x}_f$ (e), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (f).

3.5.2.3 Commande complète avec suivi de trajectoire

Dans cette partie, on considère simultanément les lois de commande longitudinale et latérale.

Les conditions initiales sont les suivantes : $X_i = -4100$ m, $Y_i = -150$ m, $Z_i = -491$ m, $\phi_i = 0$ m, $\psi_i = -3$ deg. On utilise la loi de commande longitudinale (3.170), et la loi de commande latérale (3.165). On va comparer les résultats obtenus avec ceux des figures 3.36, 3.37, et 3.49, 3.50. Cela permettra d'avoir une idée de l'effet des mouvements longitudinaux sur le comportement latéral de l'avion, et vice versa. En effet, on a négligé les couplages entre les deux mouvements dans toute la modélisation.

Pour les mouvements latéraux, on compare les figures 3.57 et 3.58 avec les figures 3.36 et 3.37. Les résultats sont globalement très similaires concernant les entrées de commande et le comportement de l'avion dans l'espace (figures 3.57 et 3.36). On peut noter que l'erreur sur l'information visuelle W est significativement perturbée pendant la phase transitoire (voir figures 3.58.d et 3.37.d). C'est la raison pour laquelle on a utilisé ici la loi de commande (3.170), qui n'utilise pas W . On peut également noter que, bien qu'elle corresponde encore très bien à l'erreur sur l'angle de tangage θ , l'erreur sur l'ordonnée du point de fuite y_f ne lui est plus égale (léger écart aux alentours de l'instant $t = 20$ s, voir figures 3.58.e et 3.37.e). Cela est dû au fait que les angles de roulis et de lacet ne sont plus nuls, à cause des mouvements latéraux.

Pour les mouvements latéraux, on compare les figures 3.59 et 3.60 avec les figures 3.49 et 3.50. Là encore, les résultats sont globalement très proches. On remarque cependant de légères différences : les couplages perturbent un peu l'entrée de commande δ_l , qui est moins "lisse" (voir figures 3.59.a et 3.49.a). On voit également que les comportements transitoires de la position latérale Y , et des angles de roulis et de lacet sont légèrement différents, avec notamment l'apparition d'un léger dépassement (voir figures 3.59.b, 3.59.c, 3.59.d et 3.49.b, 3.49.c, 3.49.d). En comparant les figures 3.60 et 3.50, on observe également de légères différences transitoires, notamment en ce qui concerne l'erreur latérale $Y - Y^*$ et l'information visuelle $V - V^*$. Dans les deux cas, l'erreur sur l'information visuelle V correspond bien à l'erreur latérale à partir de l'instant $t \simeq 30$ s. On remarque cependant que les couplages avec les mouvements longitudinaux font que cette "correspondance" est moins bonne (voir figures 3.60.d et 3.50.d). Il en est de même pour les erreurs sur les angles de roulis et lacet, et sur les informations visuelles η et x_f (voir figures 3.60.e, 3.60.f, et 3.50.e, 3.50.f).

Pour conclure, avec les conditions initiales considérées ici, les couplages entre mouvements longitudinaux et latéraux ne sont pas très significatifs. Notons que lorsque l'on s'éloigne trop de la trajectoire nominale, les problèmes mis en évidence précédemment pour chaque commande séparément sont cependant accentués par les effets de couplage.

Autres conditions initiales. On considère à présent différentes conditions initiales, plus ou moins difficiles.

Les figures 3.61 et 3.62 montrent des résultats obtenus avec $Y_i = 350$ m, $\psi_i = 5$ deg, et $Z_i \simeq -500$ m. Les figures 3.63 et 3.64 montrent des résultats obtenus avec $Y_i = 50$ m, $\psi_i = 5$ deg, et $Z_i \simeq -570$ m : les contraintes latérales sont donc moins importantes,

mais l'avion se trouve plus haut que dans le cas précédent. Dans les deux cas, la tâche est bien réalisée : l'avion s'aligne avec la piste, et rejoint la pente d'approche. Notons que les conditions initiales doivent être choisies de manière à ce que la piste soit "visible", c'est-à-dire, puisqu'on est en simulation, au moins que les lignes de la piste se projettent sur le plan image.

Avec D_1 et D_2 non nuls. Jusqu'à présent, on a considéré que la caméra est placée au centre de gravité de l'avion (les paramètres D_1 et D_2 — voir figure 3.21 — étaient donc nuls). On place maintenant la caméra en avant et en haut de l'avion, de manière à ce que $D_1 = 8$ m, et $D_2 = -2$ m, mais on garde les mêmes commandes et génération de trajectoire que précédemment, considérant $D_1 = D_2 = 0$.

On voit, en comparant les figures 3.65, 3.66 et 3.57 3.58, 3.59, 3.60, que les résultats sont très similaires.

Avoir négligé les paramètres D_1 et D_2 durant la modélisation ne change donc presque rien au comportement du système bouclé.

Avec une mauvaise estimation de la largeur de la piste. La piste considérée a une largeur de 60 mètres. Comme on l'a vu, pour calculer les informations visuelles désirées à partir des trajectoires sur les paramètres de la pose de l'avion, on projette le modèle de la piste sur le plan image de la caméra. Jusqu'à présent, on a utilisé pour cette projection un modèle fidèle de la piste réellement observée par la caméra. On suppose ici que la largeur de la piste est mal connue : on utilise comme modèle pour le calcul des informations visuelles désirées une piste large de 90 mètres.

Les résultats sont donnés sur les figures 3.68, 3.69 et 3.70. On peut les comparer à ceux représentés sur les figures 3.57 3.58, et 3.59 : un modèle parfait de la piste était alors utilisé. Pour la partie longitudinale, y_f et sa dérivée \dot{y}_f sont les seules informations visuelles prises en compte dans la loi de commande. On peut voir sur les figures 3.68.f et 3.58.e que, comme on s'y attend, son comportement est quasi identique dans les deux cas.

En ce qui concerne les mouvements latéraux, on voit que les paramètres des droites $\theta_{\{l,r\}}$ ne convergent pas sur les valeurs désirées (figure 3.70.a). Cela perturbe l'évolution de l'erreur visuelle $V - V^*$ (figure 3.70.b), et entraîne un retard sur l'évolution de la position latérale de l'engin (figure 3.69.b). Finalement, l'avion rejoint l'axe de la piste : la principale conséquence de la mauvaise estimation de la largeur de la piste résulte en un temps de convergence légèrement plus important.

Avec une mauvaise calibration de la caméra. Jusqu'à présent, on a supposé que la caméra est correctement calibrée (on connaît précisément les paramètres f/l_x , f/l_y , u_0 , et v_0 de la caméra — voir section 1.1.1.1). En pratique, on a considéré des erreurs de 10% sur les différents paramètres. Comme on réalise des simulations, et qu'on ne considère pas de traitement d'image, on a en pratique considéré une caméra C_1 pour "construire" la projection des lignes de la piste sur le plan image en pixels, puis une autre caméra C_2 (dont les paramètres diffèrent de 10% de ceux de la première) pour

déterminer les paramètres des lignes de la piste et les coordonnées du point de fuite. La caméra C_1 est donc utilisée pour créer les images ; dans les calculs liés à la génération de trajectoire et à l'asservissement visuel, c'est la caméra C_2 qui est utilisée.

Les résultats obtenus sont présentés sur les figures 3.71, 3.72, 3.73, et 3.74. On peut les comparer aux résultats considérant une caméra correctement calibrée, présentés sur les figures 3.57, 3.58, 3.59, 3.60. On constate que les entrées de commande et le comportement de l'avion sont identiques, que la caméra soit correctement calibrée ou non (comparer les figures 3.71 avec 3.57, et 3.72 avec 3.59). En fait, on constate sur les figures 3.71.f, 3.73.b, 3.73.c, et 3.73.d que les erreurs sur les informations visuelles tendent vers zéros, et sont identiques à celles obtenues avec une bonne calibration (figures 3.58.e, 3.60.d, 3.60.e, 3.60.f). En fait, l'observation de la figure 3.74 nous montre que du fait de la mauvaise calibration, les informations visuelles courantes sont erronées, mais les informations visuelles désirées également (les coordonnées du point de fuite ne tendent pas vers zéro comme précédemment, voir figures 3.58.a et 3.60.a). On voit également que les paramètres $\theta_{\{l,r\}}$ des droites ne tendent pas exactement vers les mêmes valeurs que précédemment (comparer les figures 3.58.b et 3.73.a). En conclusion, les informations visuelles sont modifiées du fait de la mauvaise calibration, mais cela n'empêche pas de ramener les informations visuelles courantes sur les informations visuelles désirées, ce qui correspond à amener l'avion à la position désirée.

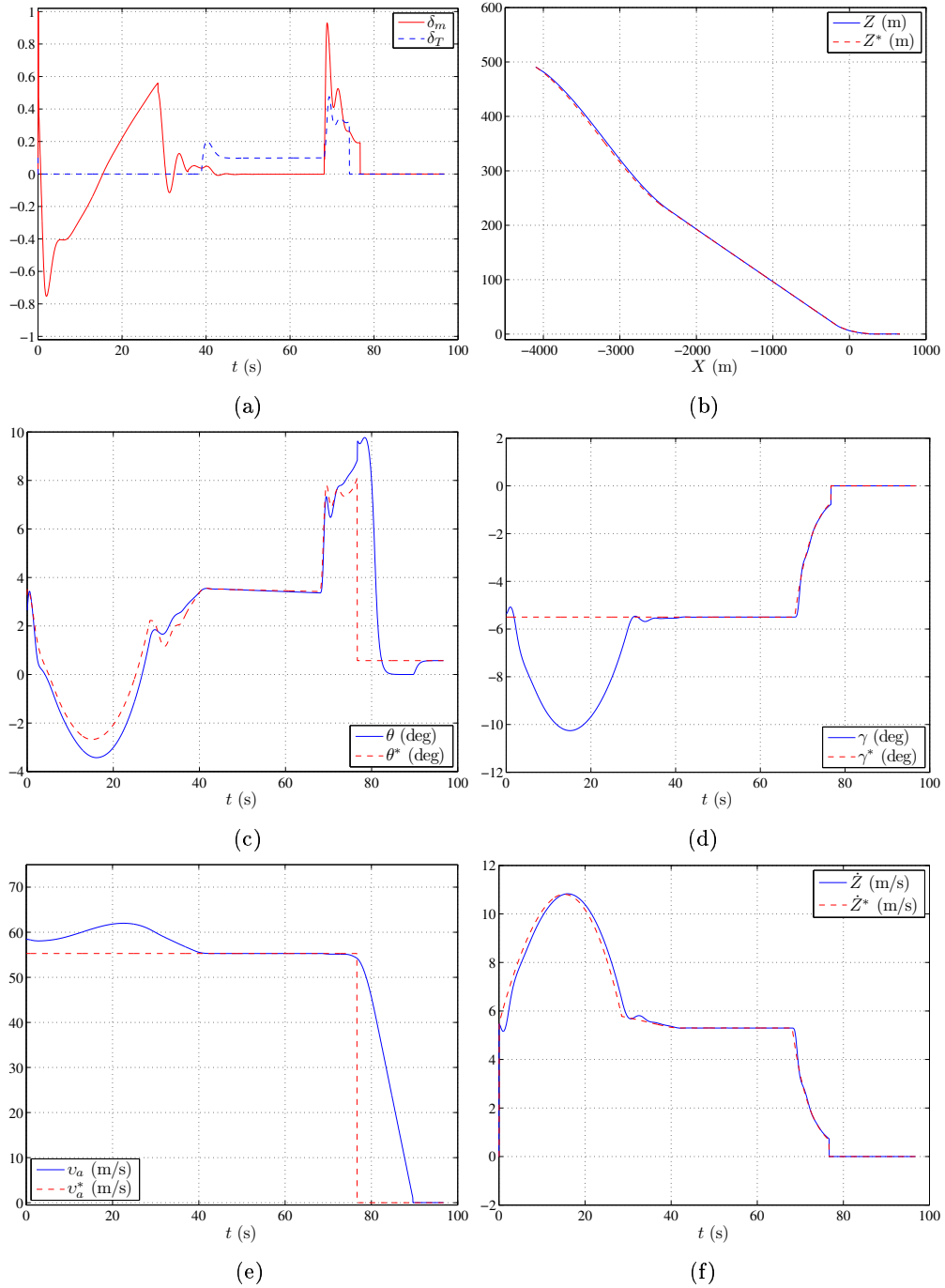


FIG. 3.57 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et vitesse verticale de l'avion (f).

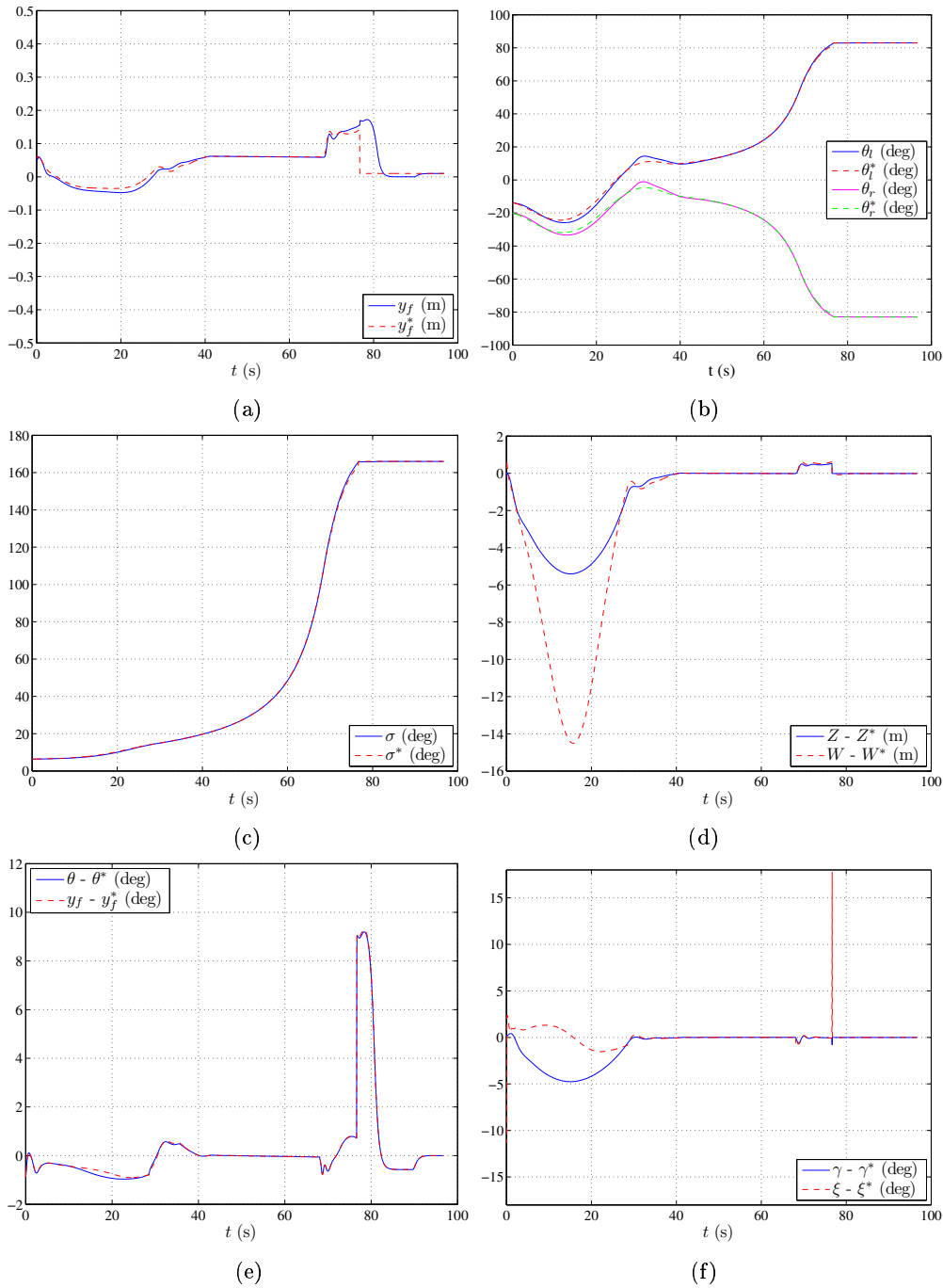


FIG. 3.58 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps de la position verticale du point de fuite dans l'image (a), des paramètres θ de la projection des lignes gauche et droite de la piste dans l'image (b), de l'information visuelle σ (c). Évolution de l'erreur sur l'altitude Z et sur l'information visuelle $W = \frac{-L}{2 \tan \frac{\sigma}{2}}$ (d), évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (e), évolution de l'erreur sur la pente γ et sur l'information visuelle $\xi = \frac{L}{4a_{18}} \frac{\dot{\sigma}}{\sin^2(\frac{\sigma}{2})} - \frac{a_{19}}{a_{18}} \nu_a$ (f).

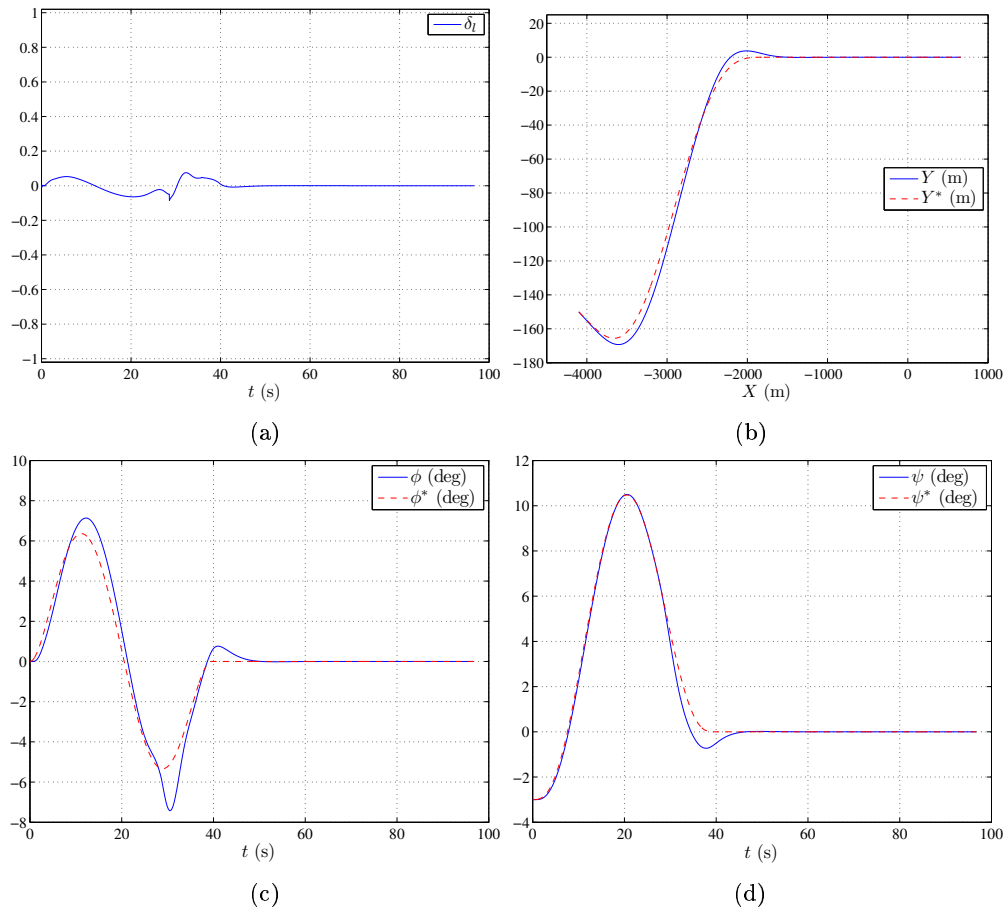


FIG. 3.59 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

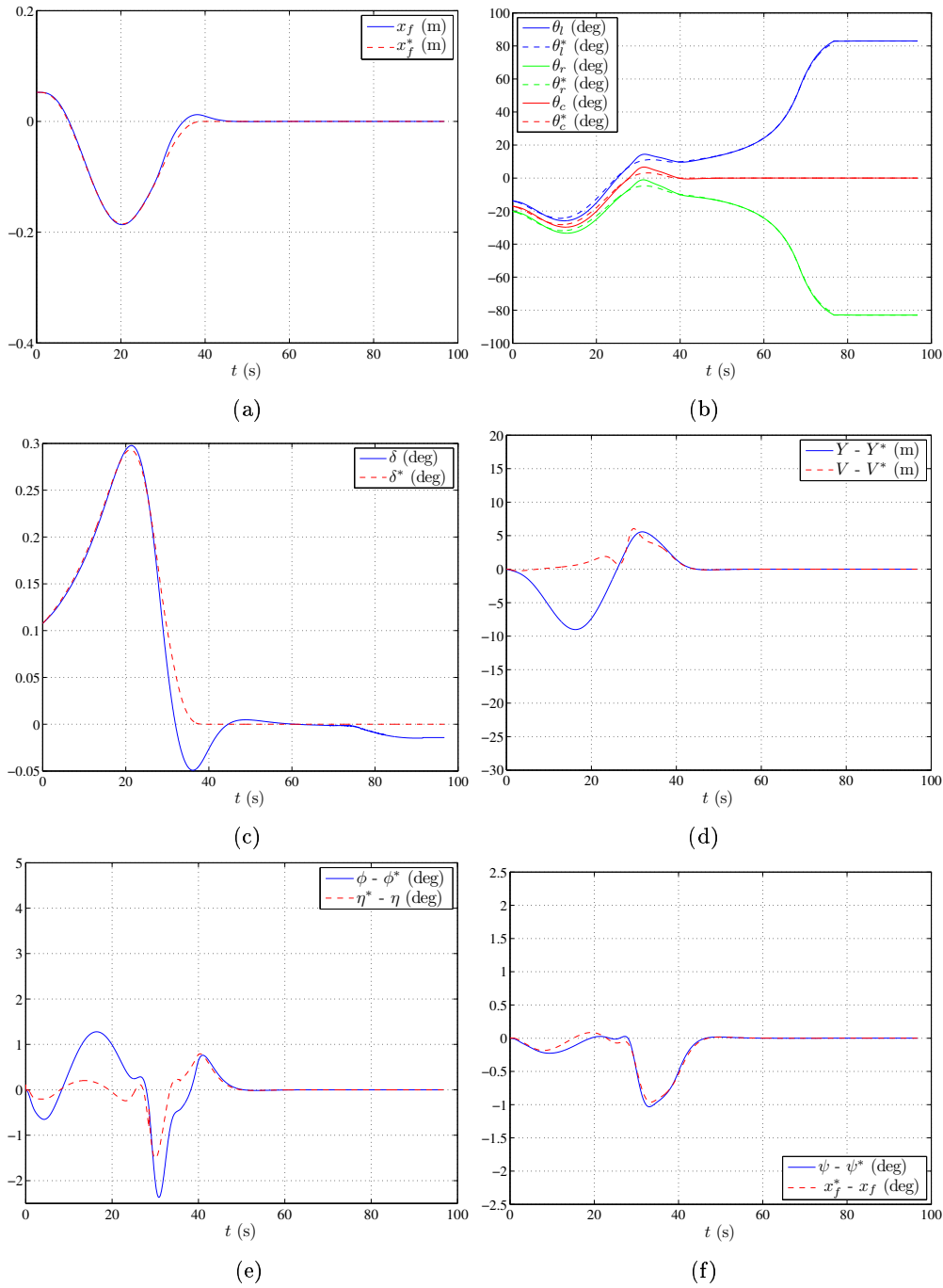


FIG. 3.60 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps de la position horizontale du point de fuite dans l'image (a), des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (b), de l'information visuelle δ (c). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2}\delta$, utilisée dans la loi de commande (d), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3}\dot{x}_f$ (e), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (f).

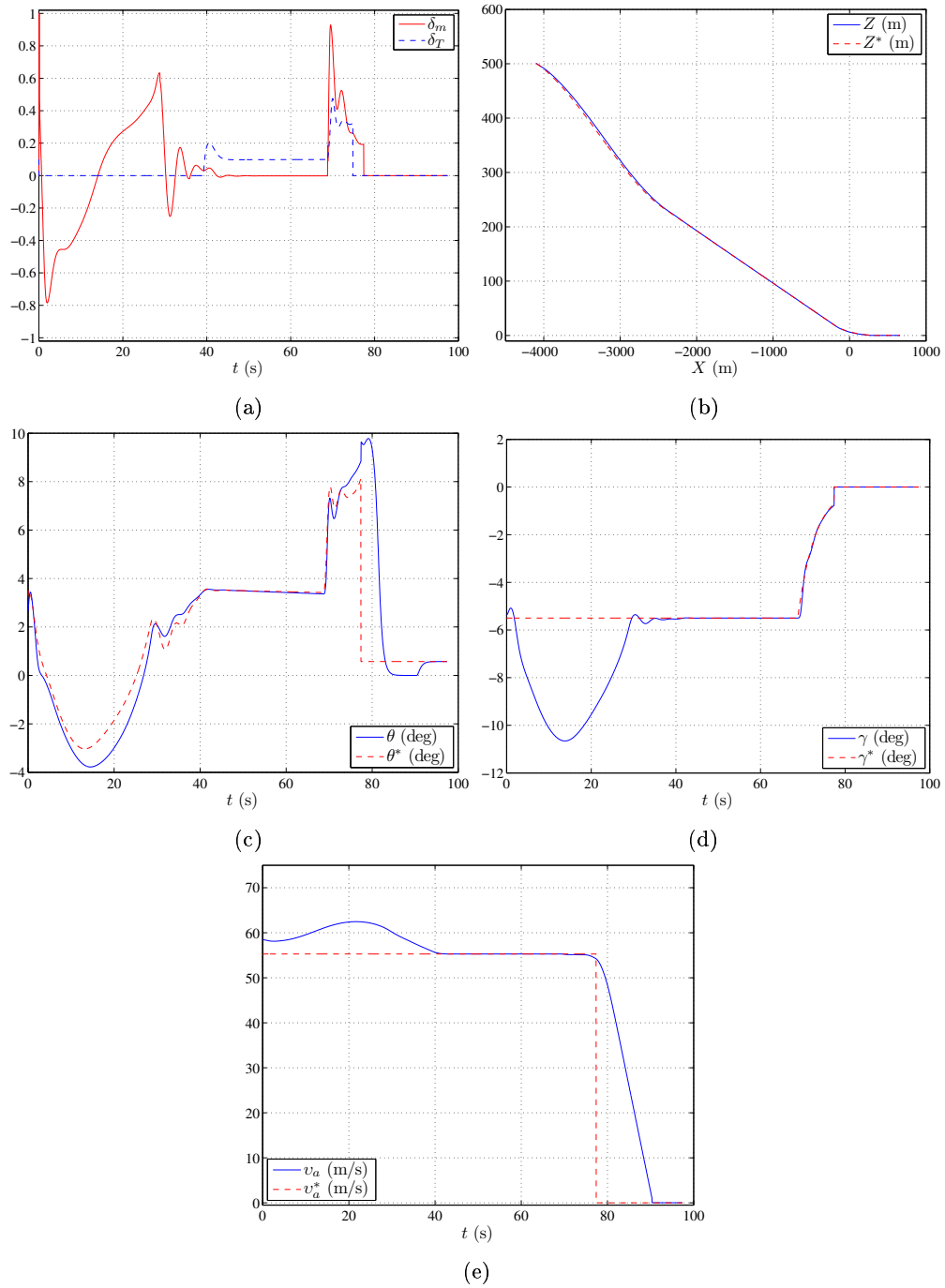


FIG. 3.61 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e).

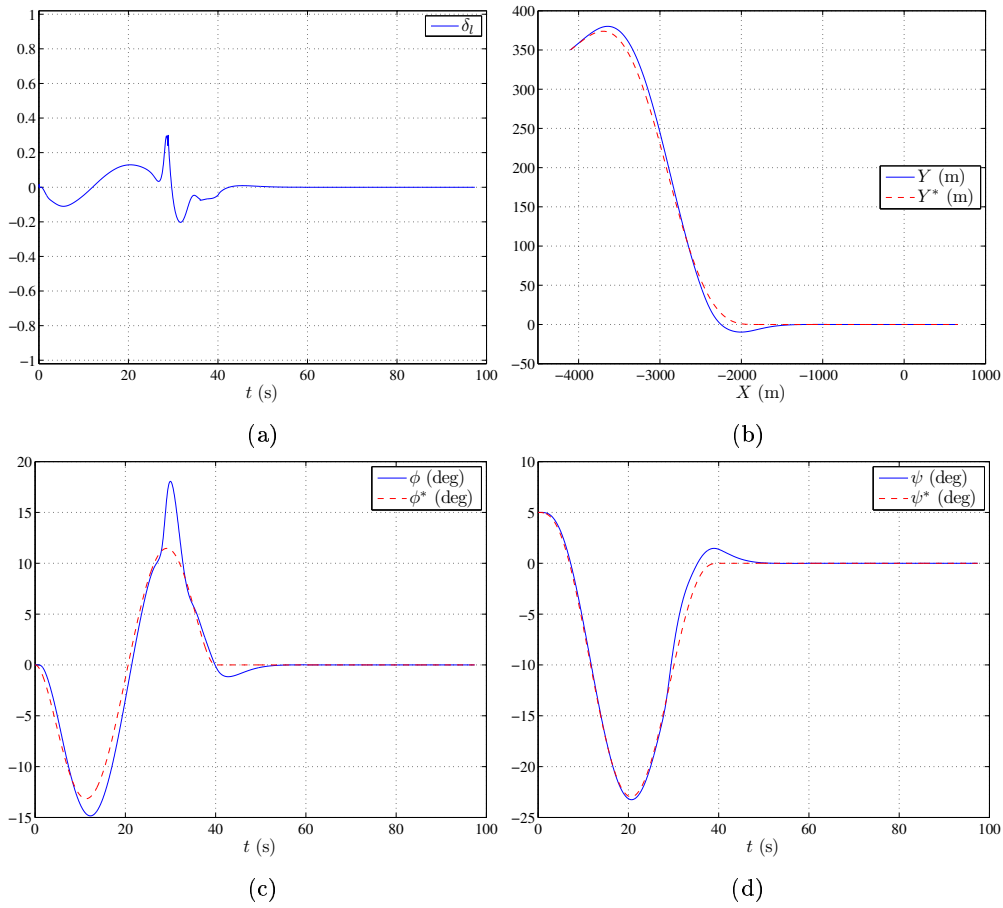


FIG. 3.62 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

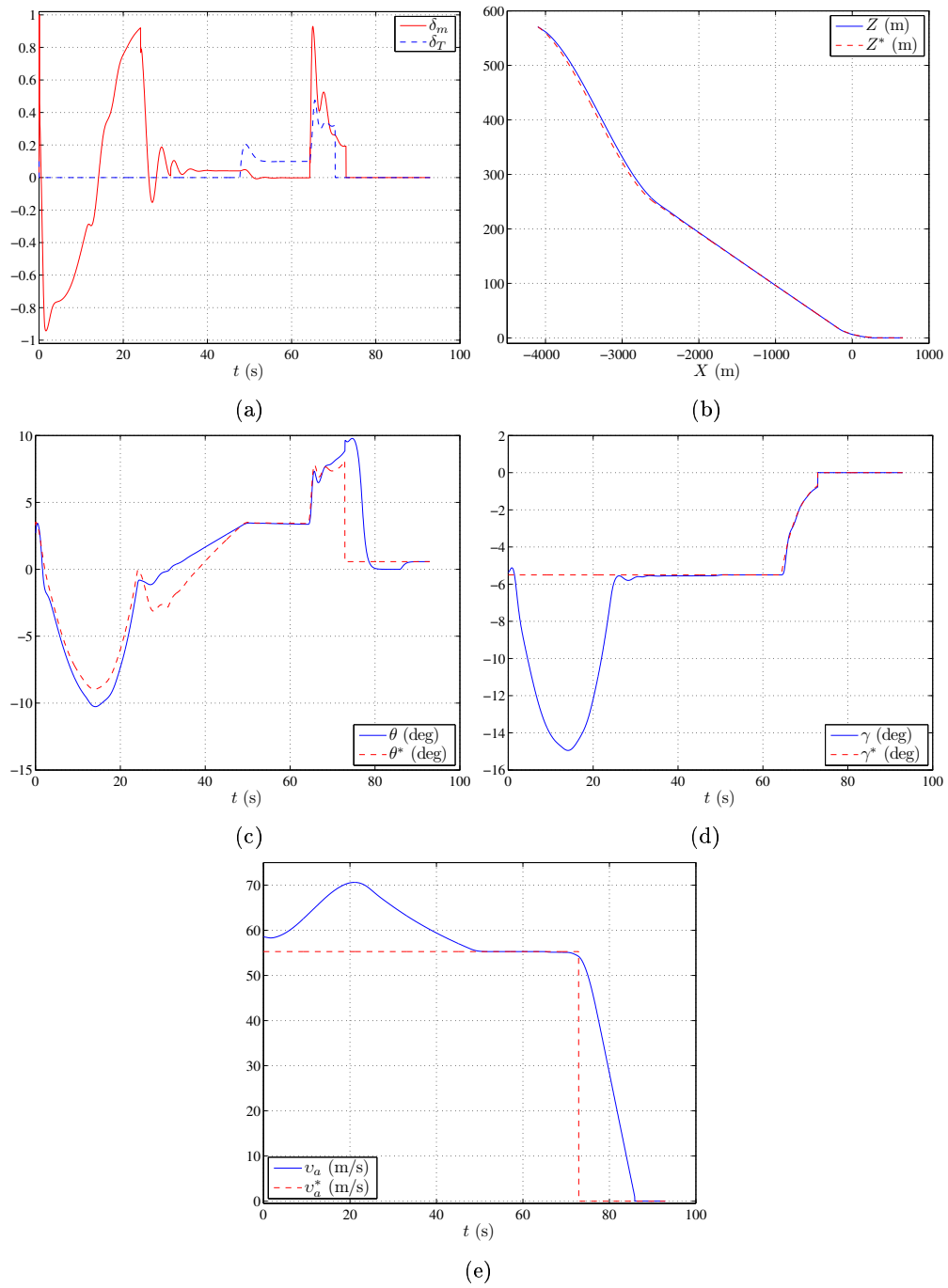


FIG. 3.63 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e).

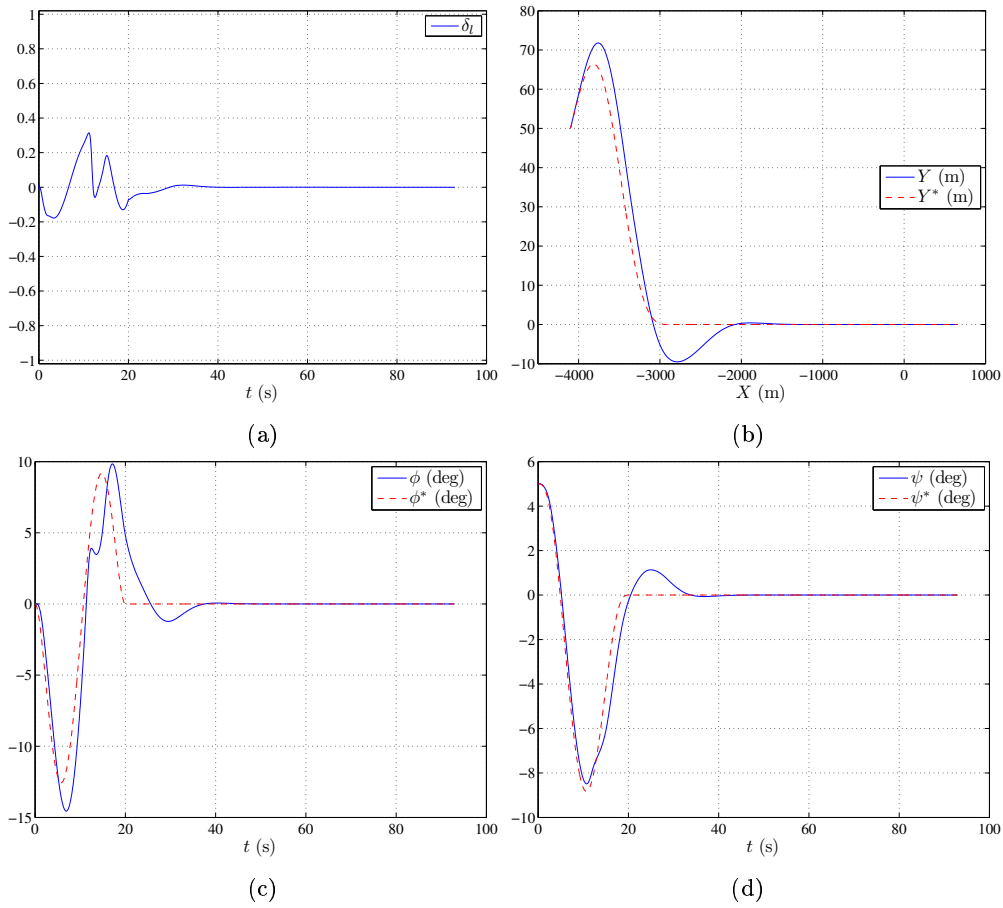


FIG. 3.64 – Résultats avec les commandes (3.165) et (3.170). Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

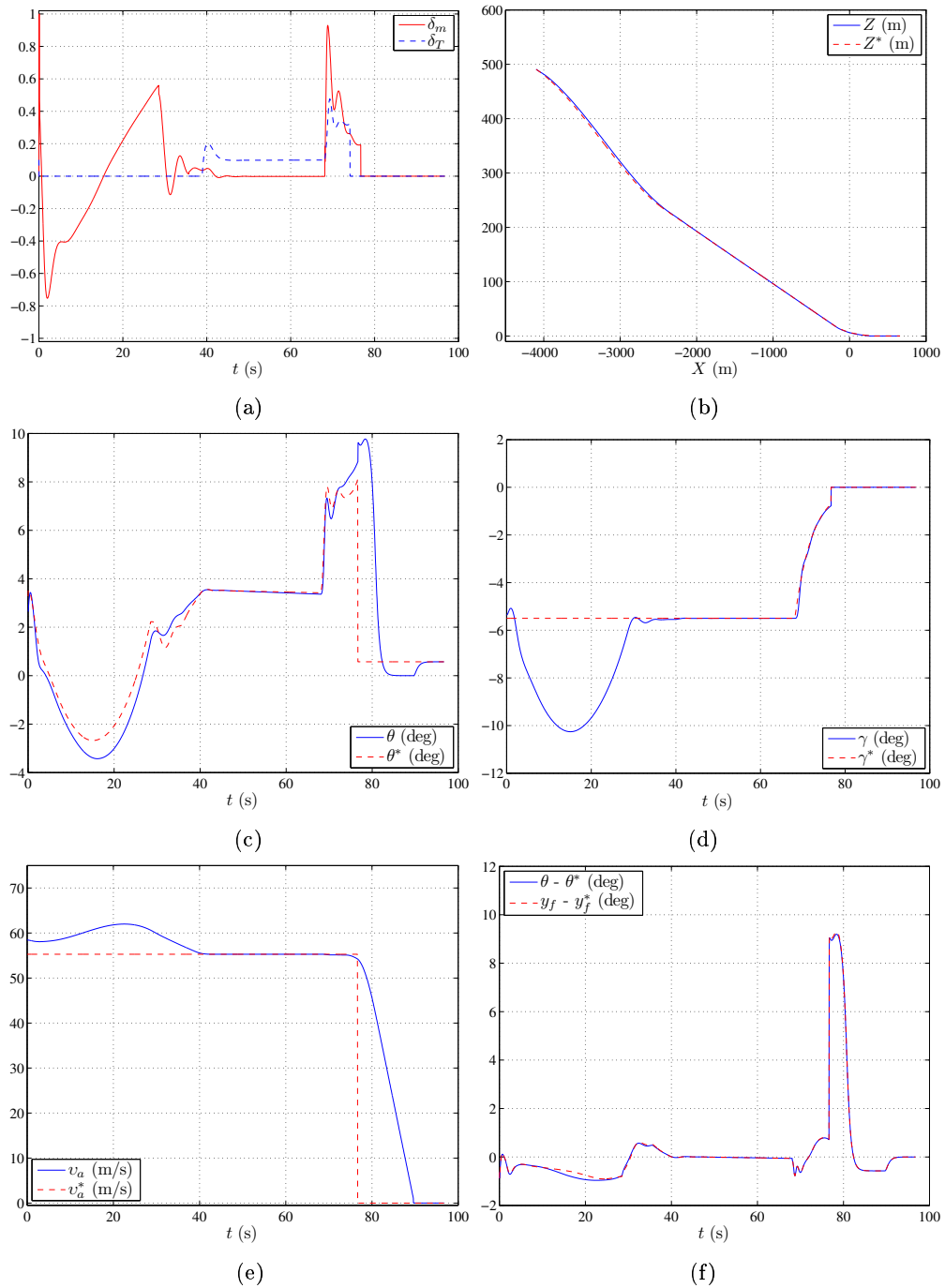


FIG. 3.65 – Résultats avec les commandes (3.165) et (3.170), avec la caméra déplacée par rapport au centre de gravité de l'avion. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (f).

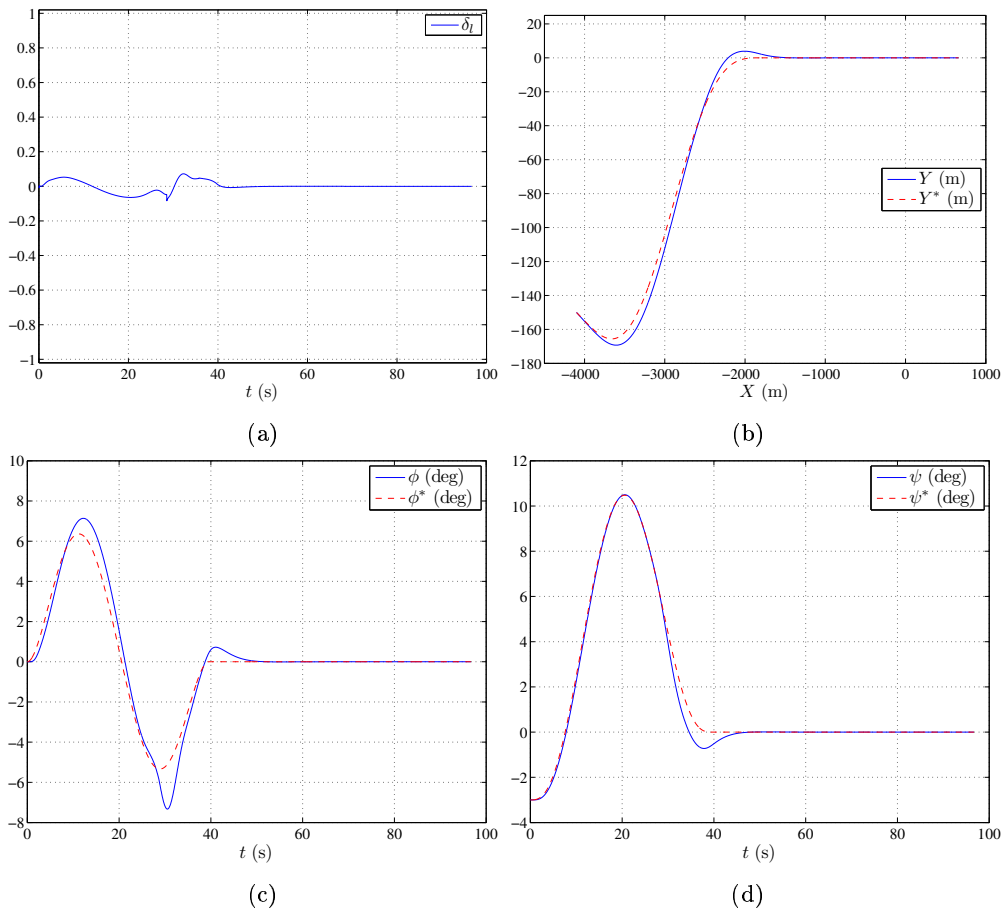


FIG. 3.66 – Résultats avec les commandes (3.165) et (3.170), avec la caméra déplacée par rapport au centre de gravité de l'avion. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

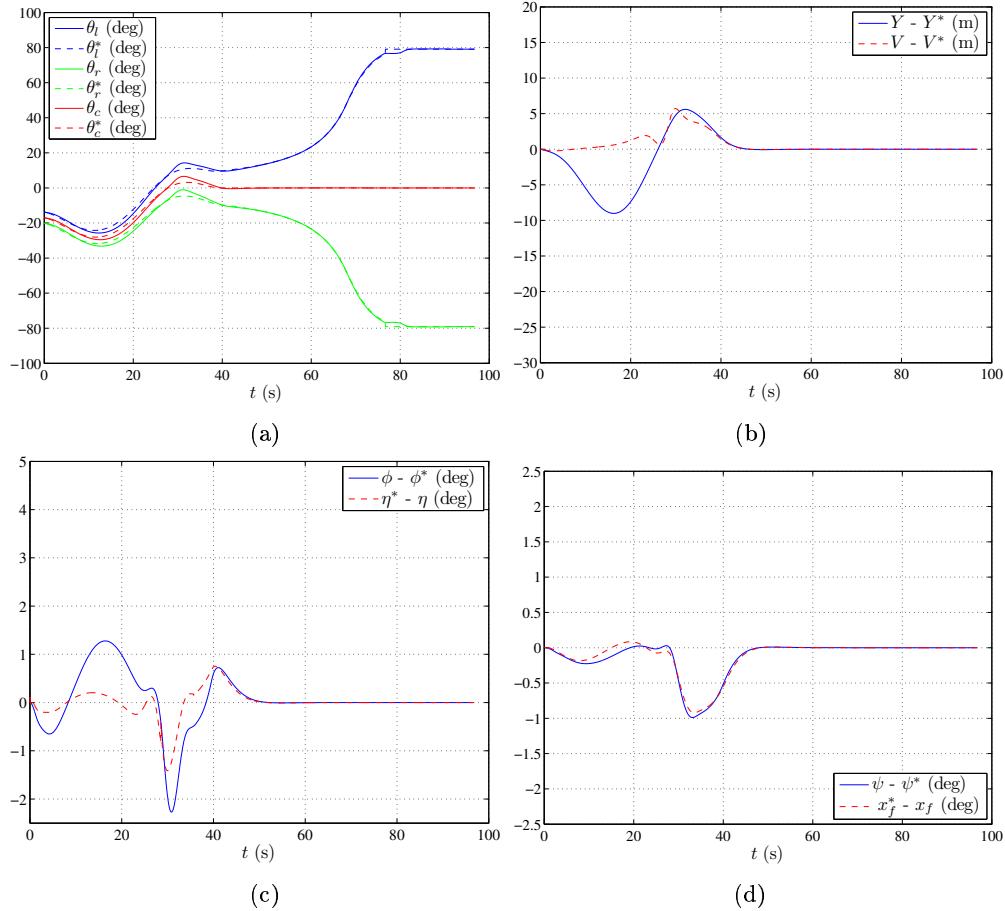


FIG. 3.67 – Résultats avec les commandes (3.165) et (3.170), avec la caméra déplacée par rapport au centre de gravité de l'avion. Évolution au cours du temps des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (a). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2} \delta$, utilisée dans la loi de commande (b), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3} \dot{x}_f$ (c), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (d).

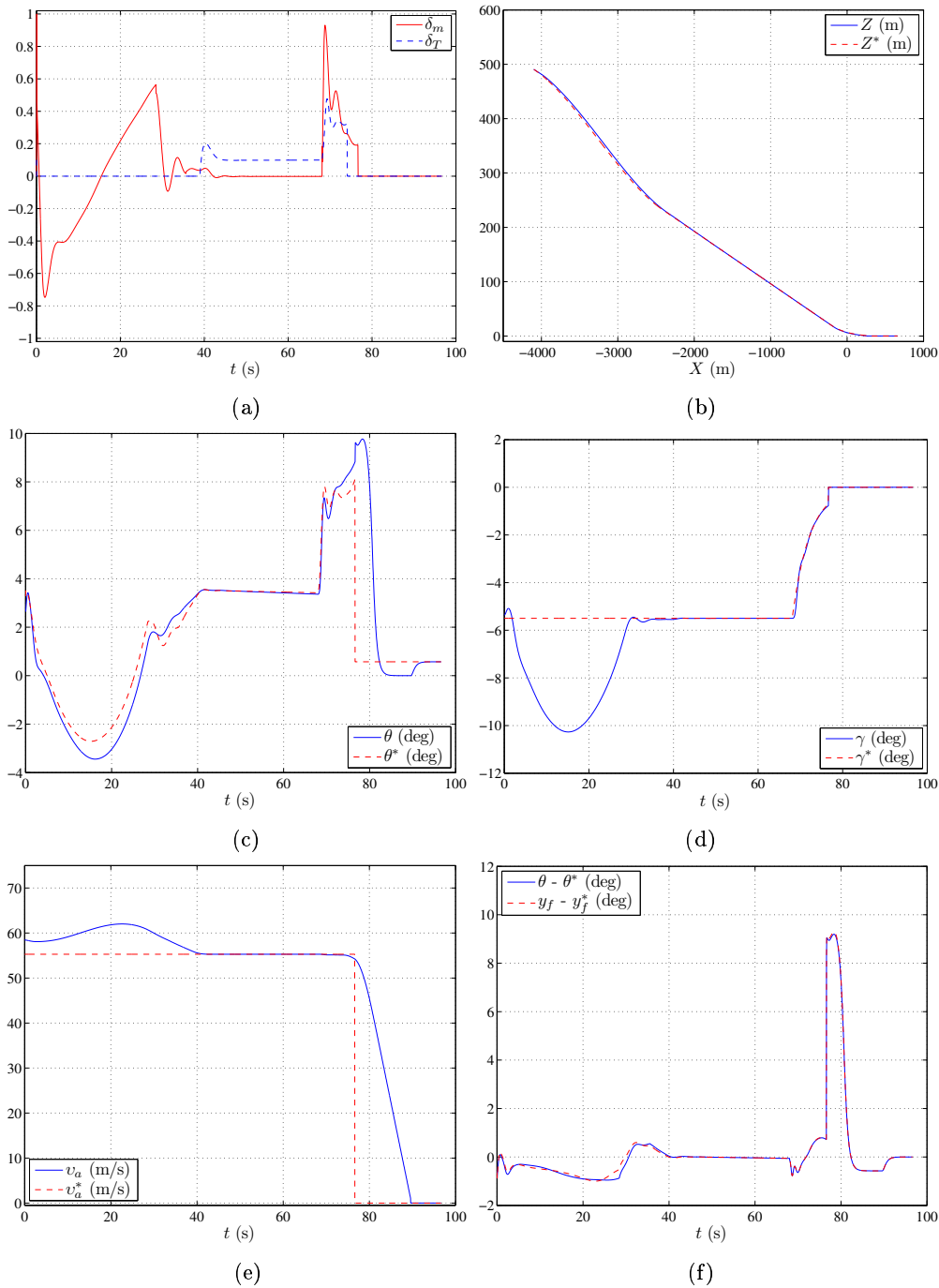


FIG. 3.68 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise connaissance du modèle de la piste. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (f).

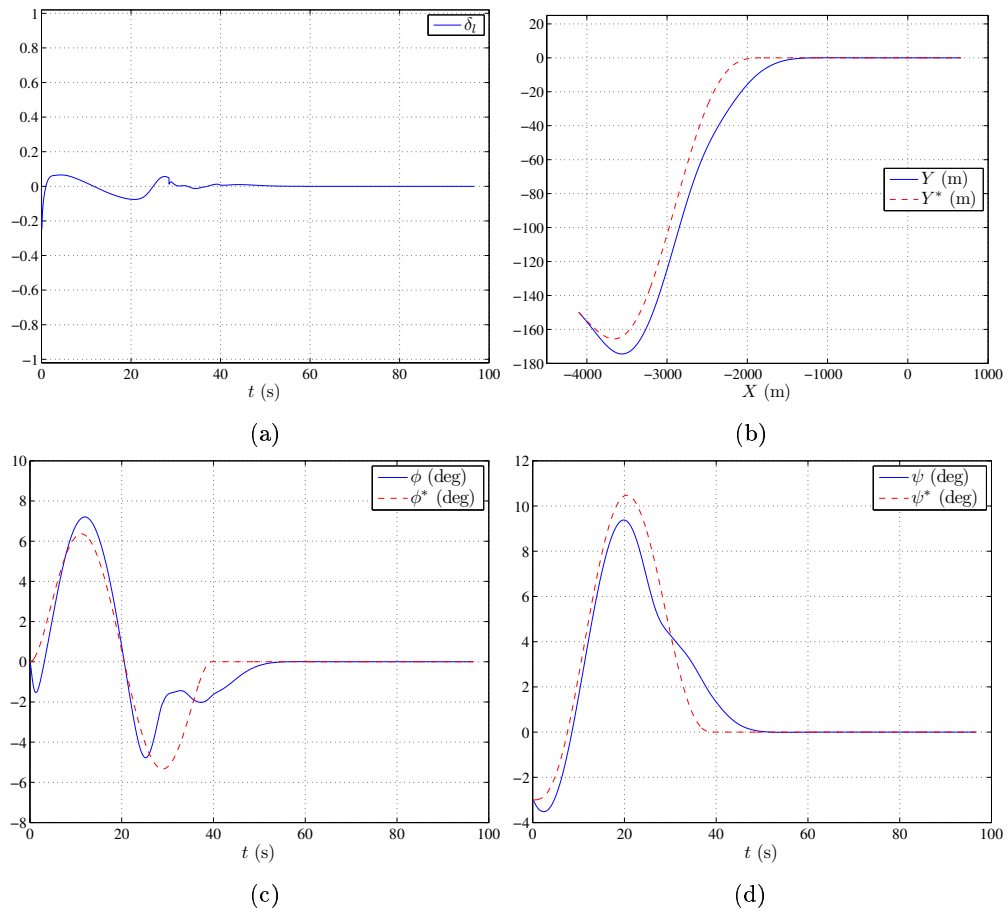


FIG. 3.69 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise connaissance du modèle de la piste. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

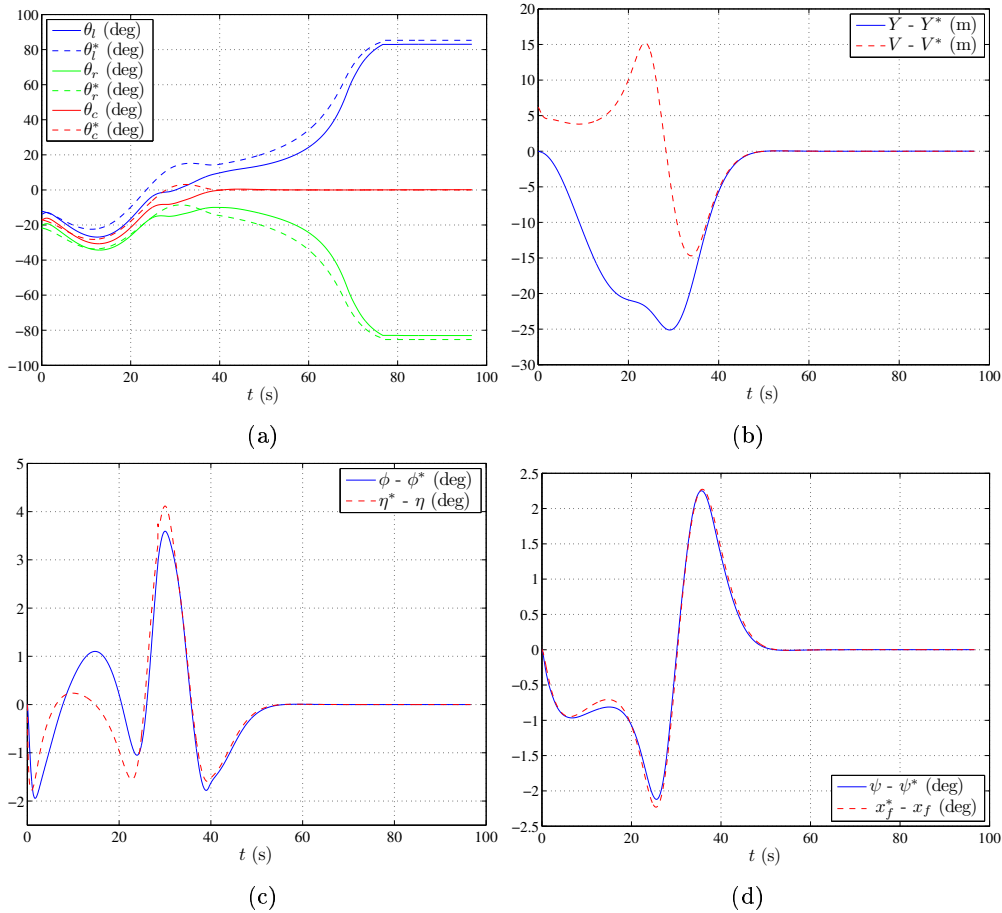


FIG. 3.70 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise connaissance du modèle de la piste. Évolution au cours du temps des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (a). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2} \delta$, utilisée dans la loi de commande (b), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3} \dot{x}_f$ (c), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (d).

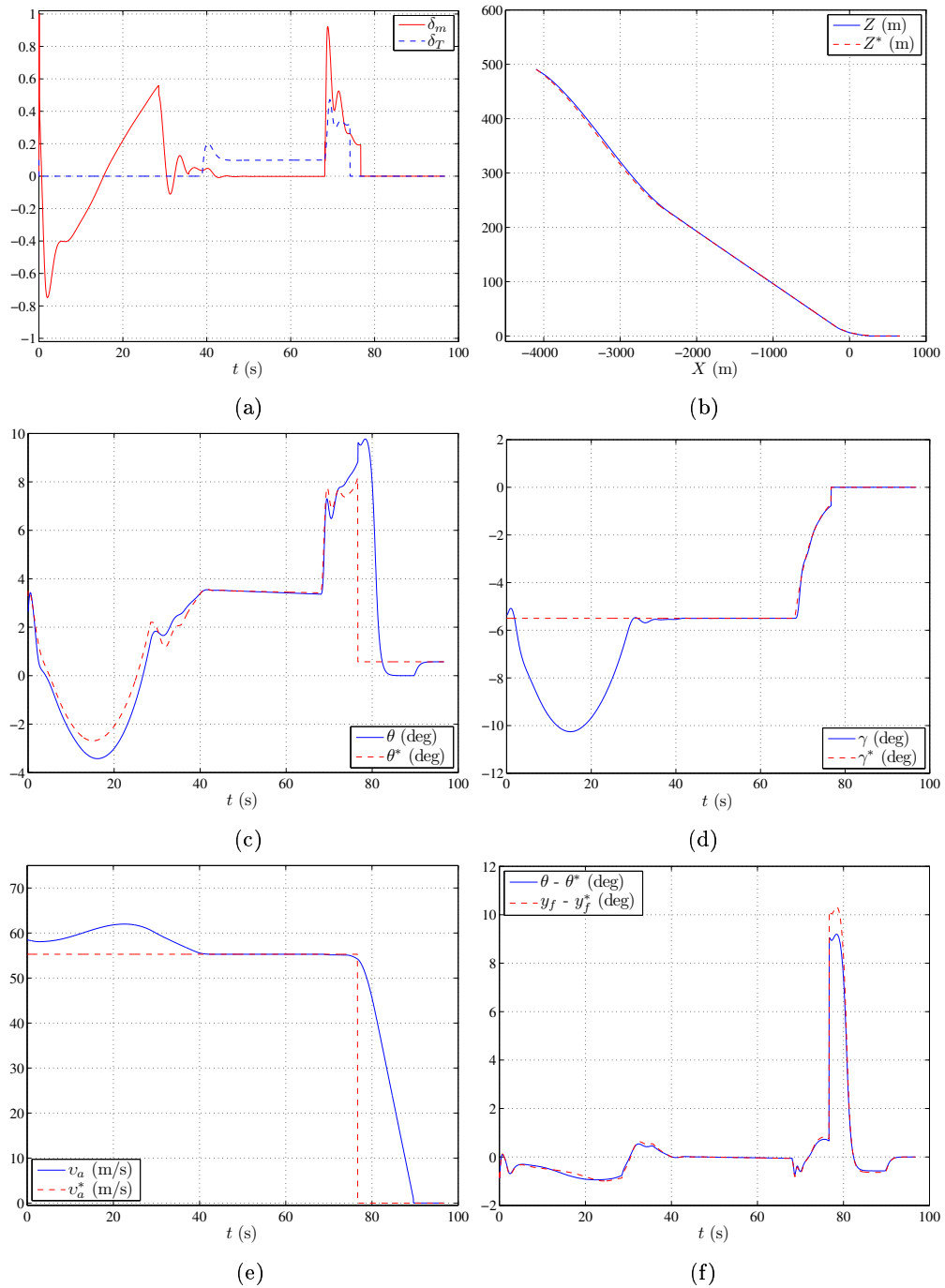


FIG. 3.71 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra. Évolution au cours du temps des entrées de commande (a), trajectoire de l'avion dans le plan vertical (b), angle de tangage (c), pente (d), vitesse de l'avion (e), et évolution de l'erreur sur le tangage θ et sur l'information visuelle y_f , utilisée dans la loi de commande (f).

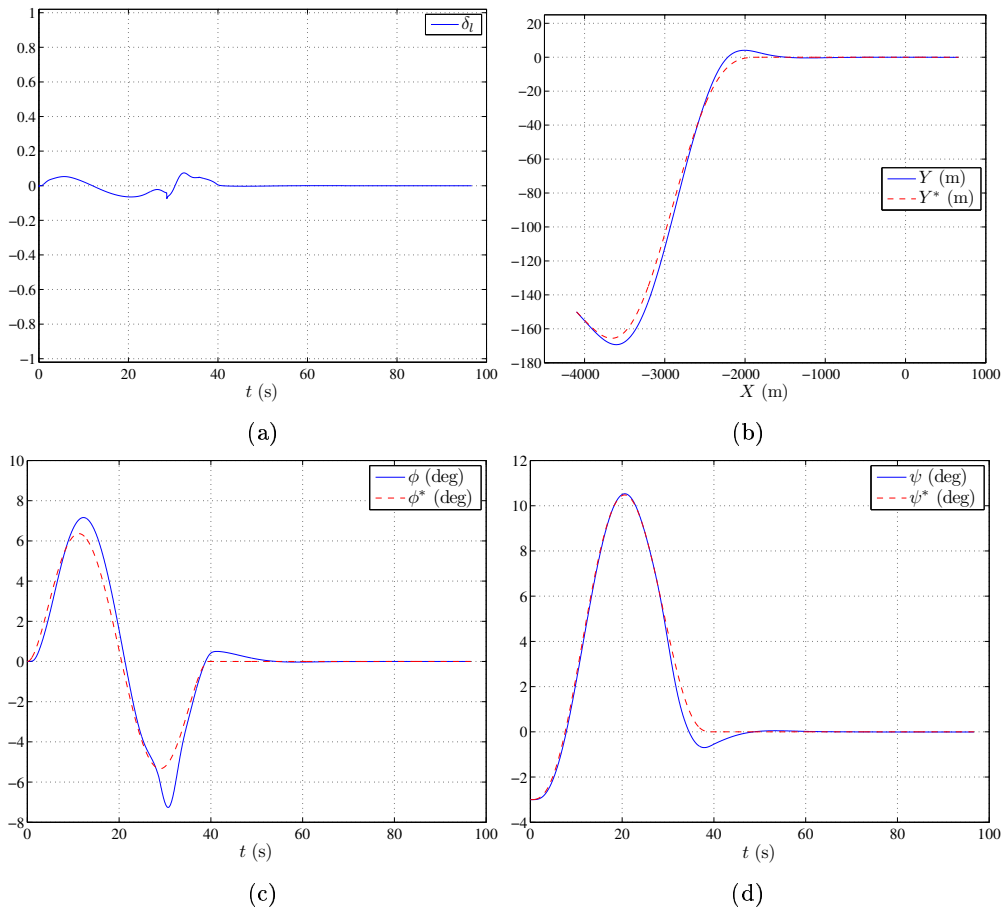


FIG. 3.72 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

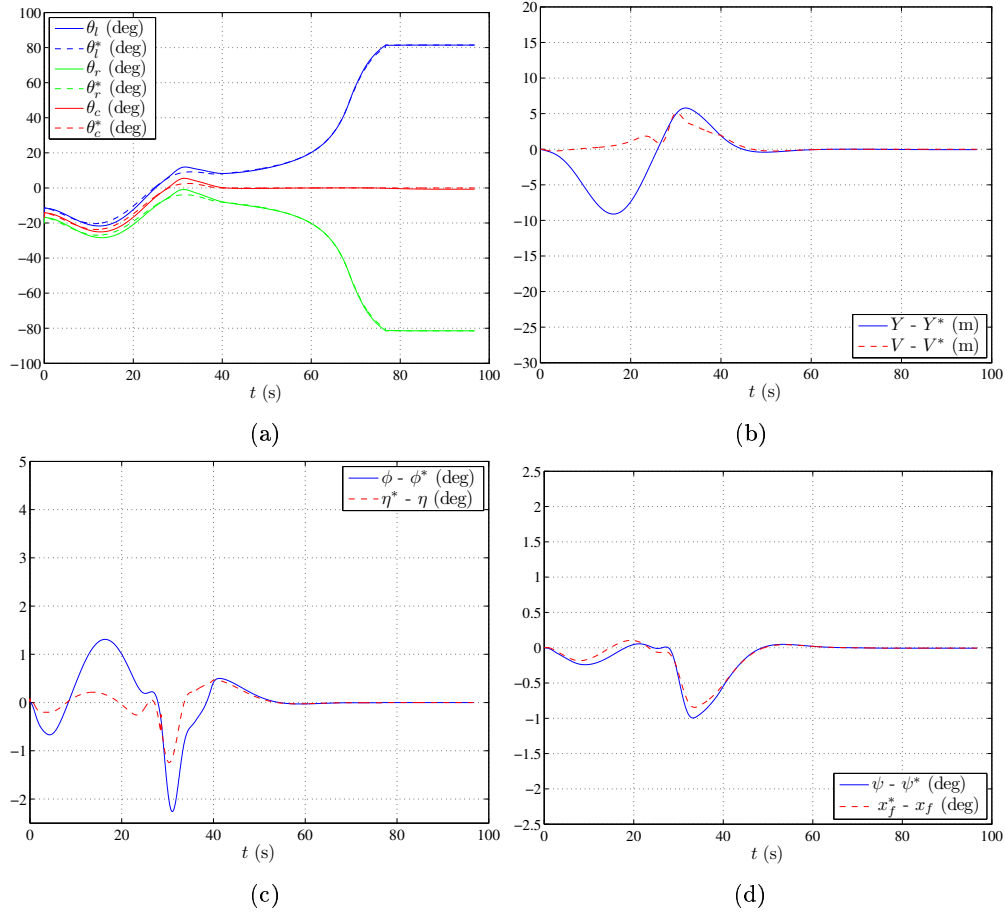


FIG. 3.73 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra. Évolution au cours du temps des paramètres $\theta_{\{l,r,c\}}$ de la projection des lignes gauche, droite et centrale de la piste dans l'image (a). Évolution au cours du temps de l'erreur sur la position latérale Y et sur l'information visuelle $V = \frac{H}{c^2\theta_l + c^2\theta_r - 2} \delta$, utilisée dans la loi de commande (b), évolution de l'erreur sur le roulis ϕ et sur l'information visuelle $\eta = \frac{1}{a_3} \dot{x}_f$ (c), évolution de l'erreur sur le lacet ψ et sur l'information visuelle x_f (d).

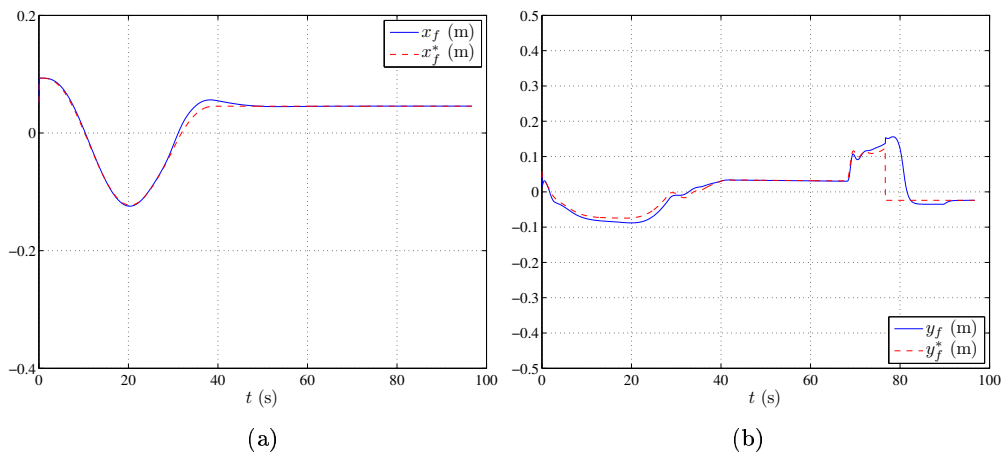


FIG. 3.74 – Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra. Évolution au cours du temps des informations visuelles x_f et y_f .

Avec du vent de travers. Jusqu'à présent, on a supposé qu'il n'y a pas de perturbations extérieures, en particulier pas de vent. On considère ici qu'il y a du vent de travers : vent latéral de l'ordre de 5 m/s (qui souffle dans le sens des Y positifs, et tend à pousser l'avion à droite de la piste).

On a représenté les résultats concernant les mouvements latéraux sur la figure 3.75. On remarque que, bien que l'entrée de commande converge vers zéro (figure 3.75.a), l'avion ne parvient pas à s'aligner avec l'axe de la piste (figures 3.75.b, 3.75.c, 3.75.d), le vent de travers l'en empêche.

En fait, dans la loi de commande par vision (3.165), les termes s'annulent (l'entrée de commande δ_l converge bien vers 0, voir figure 3.75.a), mais cela ne permet pas d'annuler l'erreur latérale Y dans l'espace 3D (figure 3.75.b).

Si on utilise la loi de commande (3.171) avec des mesures directes et pas de données de vision, on obtient les résultats de la figure 3.77. On voit que cette fois la position latérale converge vers 0 (figure 3.77.b), et que l'avion garde un angle de cap non nul pour "lutter" contre le vent (figure 3.77.d). Le problème rencontré précédemment vient des estimations à partir des informations visuelles, qui ne tiennent pas compte des perturbations extérieures. On a fait l'hypothèse que : "lorsque l'erreur latérale Y est stabilisée à 0, c'est que l'angle de lacet ψ et aussi stabilisé à 0, et cela implique que l'angle de roulis ϕ est aussi nécessairement nul (voir figures 3.18 et 3.25)". Cependant, cette hypothèse n'est valable qu'en l'absence de vent. En effet, lorsqu'il y a du vent de travers, pour pouvoir rester aligné avec la piste (garder $Y = 0$), l'avion doit se déplacer avec un angle de lacet non nul ($\psi \neq 0$). Ainsi, la relation (3.67) : $\dot{Y} = a_5\psi$ n'est plus valable telle quelle dans ce cas. On devrait considérer $\dot{Y} = a_5(\psi - \psi^*)$ où ψ^* est la valeur de l'angle de lacet que l'avion doit prendre pour "contrer" l'effet du vent de travers (voir figure 3.76).

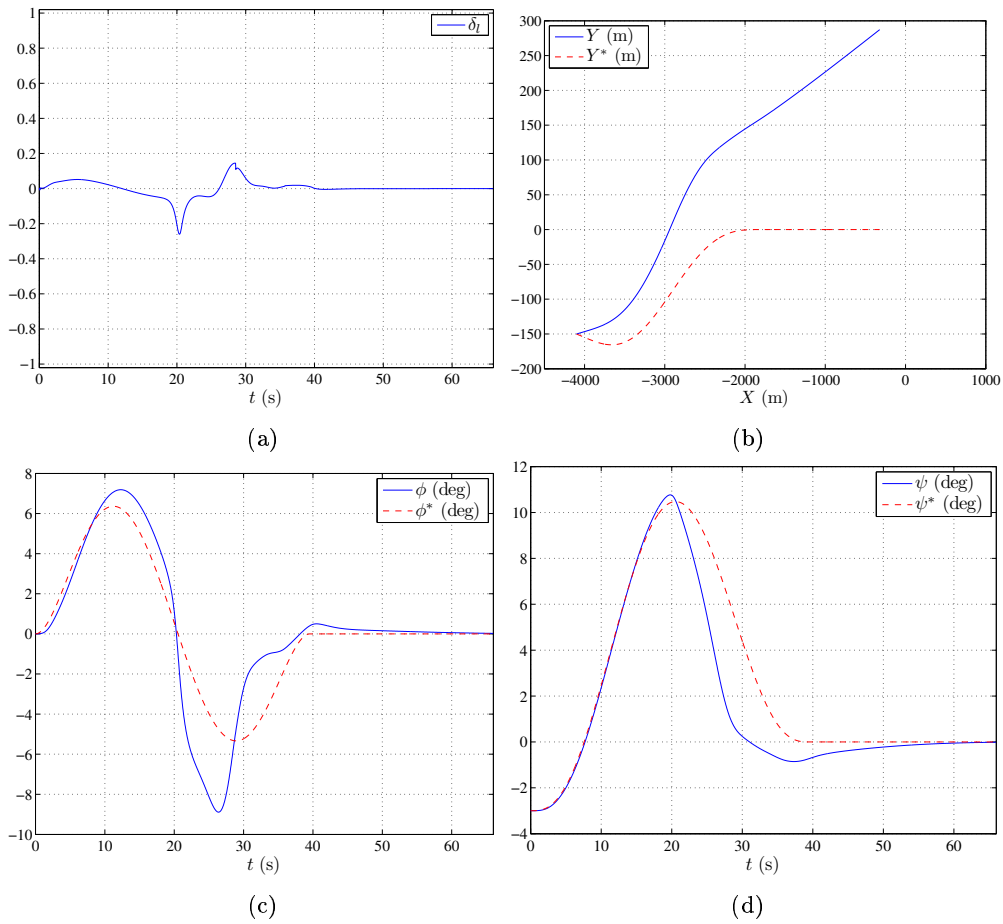


FIG. 3.75 – Résultats avec les commandes (3.165) et (3.170), et avec du vent de travers. Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

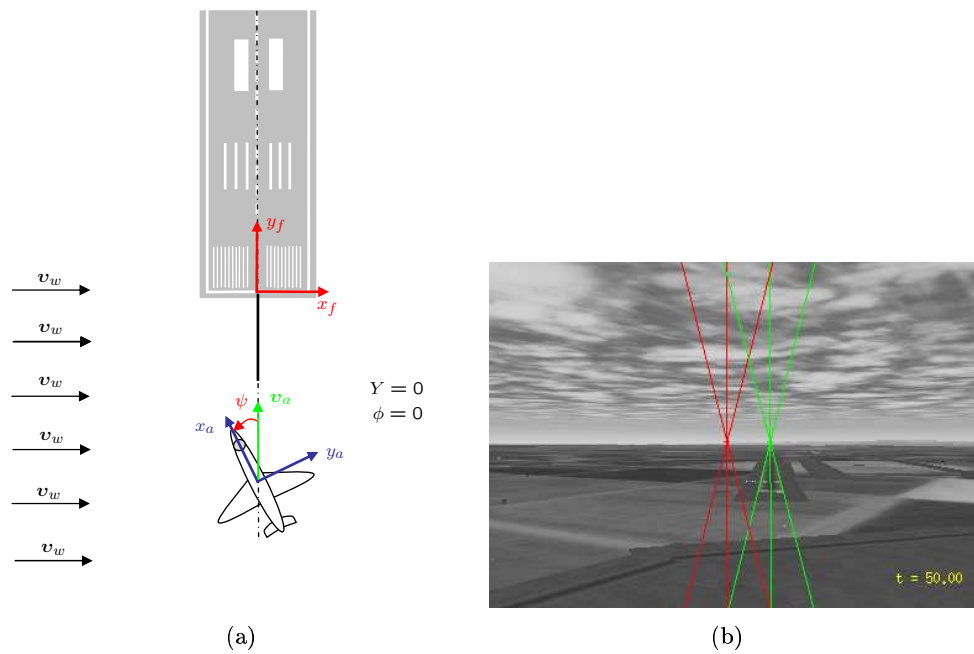


FIG. 3.76 – Alignement de l'avion dans l'axe de la piste au cours de l'atterrissage, en présence de vent de travers v_w (a), et vue de la caméra embarquée après alignement : en vert, la position courante des lignes de la piste, et en rouge la position "centrée" des lignes de la piste (b).

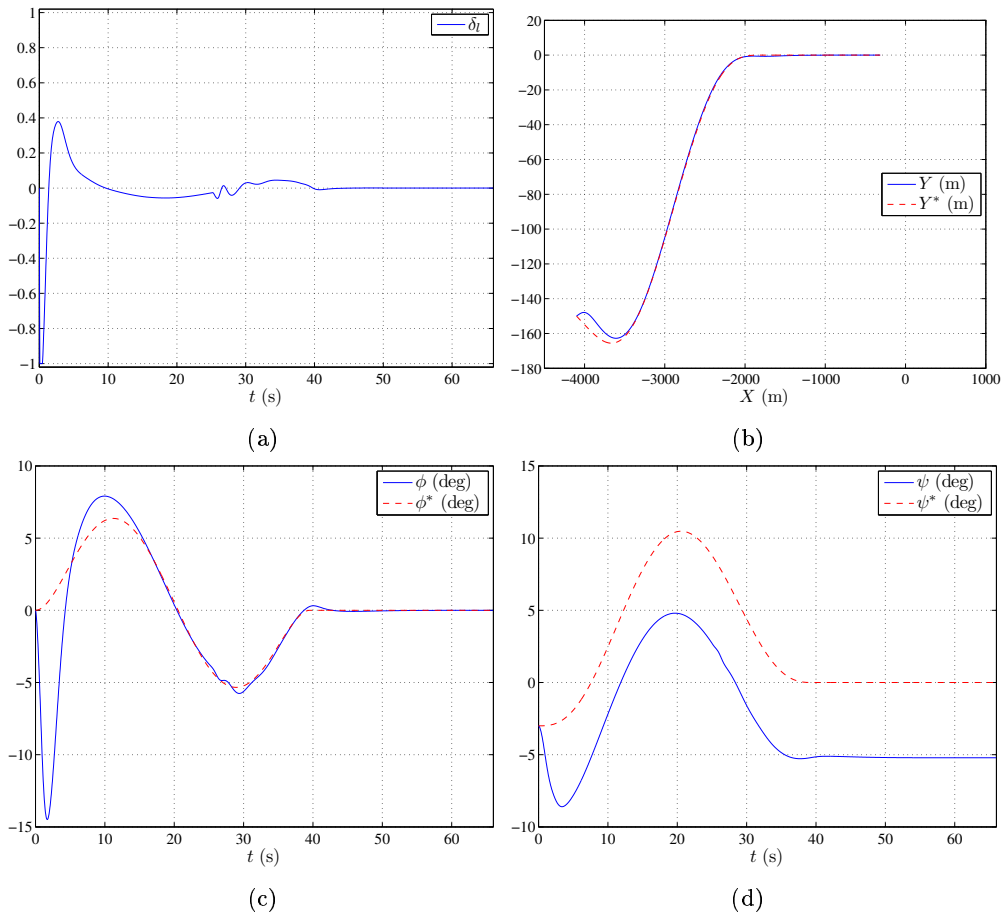


FIG. 3.77 – Alignement de l'avion avec la piste, avec du vent de travers et la loi de commande (3.171). Évolution au cours du temps de l'entrée de commande (a), trajectoire de l'avion dans le plan horizontal (b), évolution au cours du temps de l'angle de roulis (c) et de l'angle de lacet (d).

3.6 Conclusion

Dans ce chapitre, nous nous sommes intéressés à l'asservissement visuel d'un avion pendant sa phase d'atterrissage. L'objectif était de permettre à l'avion de s'aligner avec l'axe de la piste et de rejoindre une bonne trajectoire d'approche, pour ensuite arrondir sa trajectoire avant de toucher la piste. Pour cela, nous avons considéré une caméra embarquée sur l'avion, et nous nous sommes intéressés au choix d'informations visuelles tenant compte du modèle de l'avion en vol. Nous avons ensuite mis au point des lois de commande d'asservissement visuel, et défini des trajectoires à suivre pour les consignes visuelles. Enfin, nous avons réalisé une série de simulations utilisant un modèle réaliste d'avion, afin de mettre en évidence les performances et les limitations des méthodes proposées.

Dans un premier temps, nous avons linéarisé un modèle non-linéaire de l'avion, autour d'une position nominale correspondant à la trajectoire d'approche. Nous avons ainsi obtenu un modèle simplifié sous forme de représentation d'état linéaire, et découplant les mouvements latéraux et longitudinaux. Nous avons ensuite cherché à modéliser des informations visuelles présentant les mêmes propriétés de découplage, afin de commander séparément les deux types de mouvements. Pour cela, nous avons considéré les deux lignes des bords de la piste et la ligne centrale. On a alors modélisé la relation entre les mouvements de l'avion et l'évolution des paramètres des lignes et les coordonnées du point de fuite dans le plan image. On a mis au point des lois de commande, utilisant les informations visuelles et avec le modèle linéarisé de l'avion. On a également proposé une génération de trajectoire pour éviter que les informations visuelles courantes ne soient trop éloignées des informations visuelles désirées à chaque instant, et ainsi obtenir un meilleur comportement de l'avion.

Finalement, les stratégies de commande proposées ont été expérimentées à l'aide d'un simulateur réaliste. On a d'abord expérimenté séparément les parties latérales et longitudinales, afin de mettre en évidence l'intérêt des trajectoires utilisées, les performances et les limitations de l'utilisation des informations visuelles : lorsque l'on s'éloigne trop de la position nominale utilisée dans la modélisation, les erreurs visuelles ne sont plus toutes représentatives des erreurs réelles de positionnement, et on peut obtenir des comportements transitoires très perturbés, ou rejoindre un minimum local au lieu de converger vers la position souhaitée. On a ensuite testé simultanément les schémas de commande latéraux et longitudinaux. On s'est en particulier intéressés aux effets d'un mauvais modèle de la piste, aux effets d'une mauvaise calibration de la caméra, et du vent de travers. Si les algorithmes semblent robustes aux deux premiers problèmes de calibration et de modélisation, le vent de travers n'est pour l'instant pas pris en compte dans la loi de commande. Plusieurs pistes restent donc à exploiter pour améliorer les performances des lois de commande proposées, notamment en prenant en compte les perturbations extérieures.

Conclusions et perspectives

L'objectif de notre travail a consisté à mettre en œuvre des techniques de commande par asservissement visuel dans le cadre particulier des engins volants. Nous nous sommes intéressés à deux types d'engins volants : un quadrirotor (petit hélicoptère à quatre rotors contrarotatifs) et un avion. Nous avons montré pourquoi dans les deux cas, les schémas d'asservissement visuel classiques existant pour des robots de type bras manipulateur ne sont pas bien adaptés, du fait des effets de couplage entre les degrés de liberté. Nous avons alors conçu des lois de commande reposant sur des informations visuelles bien choisies, et nous avons montré leurs performances dans le cadre de simulations et d'expérimentations.

Asservissement visuel d'un quadrirotor

Contributions

Concernant le positionnement par asservissement visuel d'un quadrirotor, nous nous sommes attachés à la conception d'une commande haut-niveau fournissant des consignes sous forme de vitesses de translation à une boucle de commande bas-niveau. Des mouvements de rotation sont générés par la commande bas-niveau afin de permettre à l'engin d'évoluer en translation. Il apparaît alors intéressant de choisir, pour commander les ddl de translation, des informations visuelles découplées des mouvements de rotation de l'engin.

Nous avons dans un premier temps considéré les moments de l'image perspective, et les informations visuelles proposées dans [Tahri 05]. À partir de deux hypothèses sur le système — plan image de la caméra parallèle au plan de la cible observée et vitesses de rotation négligeables — nous avons proposé une première loi de commande pour le quadrirotor. Si ces approximations sont assez réalistes pour l'application que l'on a considérée, elles risquent de ne plus être valables pour d'autres types de quadrirotors.

Dans un second temps, nous avons considéré les moments du premier ordre de l'image sphérique, qui permettent de commander par asservissement visuel les mouvements de translation d'un quadrirotor, sans que les mouvements de rotation induits ne viennent perturber la convergence exponentielle de l'erreur considérée (propriété de passivité) [Hamel 02]. Si l'on se contente d'utiliser une loi de commande proportionnelle assurant la propriété de stabilité, on rencontre un problème de sensibilité dans la direction de l'axe optique : la matrice d'interaction est mal conditionnée, ce qui conduit à voir les trois

composantes de l'erreur visuelle — et celles de l'erreur de positionnement — converger à des vitesses différentes. Cela peut être très gênant en pratique. Une compensation asymptotique n'a pas non plus donné entière satisfaction, puisqu'elle a conduit à un comportement transitoire inadapté. Nous avons alors envisagé de résoudre le problème avec une loi de commande partitionnée, en séparant l'information visuelle en deux parties, pour appliquer un gain spécifique à la composante qui pose problème. Cela n'a pas été satisfaisant non plus du point de vue du comportement transitoire dans l'espace 3D. Ensuite, nous avons proposé de modifier l'information visuelle utilisée (tout en conservant la propriété de passivité), et nous avons envisagé plusieurs lois de commande. L'objectif était alors d'assurer la stabilité globale asymptotique du système, une décroissance de l'erreur visuelle identique pour les trois composantes, et un bon comportement de l'engin dans l'espace 3D tout au long de l'asservissement. Pour cela, on a cherché également à faire en sorte que les informations visuelles 2D soient "aussi proches que possible des positions 3D", afin que la loi de commande assurant la décroissance correcte des informations visuelles corresponde également à un comportement convenable dans l'espace 3D.

Nous avons présenté la série de lois de commande envisagées, et commenté leurs atouts et défauts respectifs, en illustrant leurs performances sur des simulations. Nous avons également montré et analysé les résultats expérimentaux obtenus lors du positionnement d'un quadrirotor au-dessus d'une cible visuelle. Parmi les lois de commande étudiées, trois d'entre elles ont donné de bons résultats en pratique : la loi de commande proportionnelle utilisant les moments de la projection perspective, un correcteur proportionnel, avec information visuelle sphérique et changement d'échelle, et une loi de commande stable reposant sur une information visuelle sphérique modifiée. Nous avons également pu écarter l'hypothèse selon laquelle un bruit important serait introduit dans le système lors du calcul de l'information visuelle de référence. Par ailleurs, bien qu'elle n'ait pas la propriété de passivité, la loi de commande en translation basée sur les moments de l'image perspective n'est pas perturbée de manière très significative par les mouvements de rotations générés par le quadrirotor utilisé. Nous avons pu montrer que les différents changements d'échelle envisagés pour les informations visuelles sphériques ont des comportements très similaires en ce qui concerne le bruit issu de l'image. Celui-ci se retrouve en effet au niveau de la loi de commande et finalement sur les positions 3D de la caméra. Avec un bruit image identique au départ, le bruit sur les positions de la caméra est identique quelle que soit la loi de commande choisie parmi les trois meilleures. Finalement, les trois lois de commande donnant des résultats satisfaisants en pratique ont également un comportement similaire vis-à-vis du bruit. L'avantage des informations visuelles sphériques réside plutôt dans des considérations théoriques, puisqu'elles permettent d'obtenir la propriété de passivité et même la stabilité asymptotique globale du système. Le choix de l'une ou de l'autre loi de commande en pratique dépendra des propriétés que l'on souhaite assurer pour le système.

L'objectif pratique consistant à mettre en œuvre un schéma d'asservissement visuel 2D permettant au quadrirotor de se positionner correctement par rapport à une cible connue est réalisé. Nous avons même proposé trois solutions différentes : trois lois de commande, aux propriétés différentes, donnent de bons résultats.

Perspectives

Concernant les expérimentations réalisées, il serait bien sûr intéressant de multiplier les expérimentations, avec différentes conditions expérimentales, différents types de quadrirotors, etc.

Le but de notre étude était aussi de concevoir une loi de commande qui satisfasse à la fois aux contraintes de bon comportement transitoire et asymptotique du système, dans l'image et aussi en 3D, avec une relation approximativement linéaire entre l'espace image et l'espace 3D, tout en conservant la propriété de passivité, et en assurant la stabilité asymptotique globale du système. Or, il est difficile de conjuguer toutes ces propriétés au sein d'une même loi de commande. Parmi les trois lois de commande donnant de bons résultats, pour l'une on n'a pas la propriété de passivité, pour une autre on n'a pas prouvé que le système est GAS, et dans la troisième, la relation linéaire entre espace image et espace 3D est brisée. Proposer une loi de commande respectant l'ensemble des propriétés souhaitées reste donc un problème que nous n'avons pas résolu. La question de savoir s'il est possible de la résoudre se pose également.

Nous nous sommes intéressés à une tâche de positionnement et stabilisation au-dessus d'une cible visuelle connue. Cette tâche peut être considérée comme la première phase d'un processus d'atterrissage, pour lequel notre approche pourrait être exploitée et généralisée.

Notre travail a porté sur l'utilisation d'une cible visuelle connue, constituée de quatre points. L'étude théorique pourrait être reprise et adaptée dans le cas d'une cible différente, constituée d'autres formes géométriques, d'un motif continu, de scènes plus réalistes, etc.

Atterrissage d'un avion par asservissement visuel

Contributions

En ce qui concerne l'atterrissage d'un avion par asservissement visuel, nous avons considéré un modèle d'avion équipé d'un système de commande de vol électrique, et nous avons proposé une loi de commande par asservissement visuel agissant au niveau des entrées disponibles pour un pilote.

À partir d'un modèle non linéaire du comportement de l'avion — sous forme de représentation d'état — nous avons obtenu un modèle linéarisé découplant les mouvements latéraux (dans le plan horizontal) des mouvements longitudinaux (dans le plan vertical). Compte tenu de la scène observée — une piste d'atterrissage — nous avons considéré une caméra embarquée, orientée vers l'avant de l'avion. Nous avons utilisé des informations visuelles, issues de la projection des lignes du bord et du milieu de la piste d'atterrissage sur le plan image de la caméra. Pour cela nous avons utilisé les paramètres θ des droites projetées dans le plan image et les coordonnées du point de fuite [Rives 04]. L'intérêt de ces informations visuelles est qu'elles sont découplées lorsque l'avion est aligné avec l'axe de la piste et que les rotations sont nulles.

Nous avons alors proposé des lois de commande pour la commande latérale et pour

la commande longitudinale, tenant compte du modèle d'état de l'avion, et intégrant les informations visuelles choisies. Pour la commande latérale, nous avons utilisé un modèle simplifié, reposant sur une cascade d'intégrateurs, et des considérations de stabilité par placement de pôle. Pour la commande longitudinale, une approche de type LQR a été choisie. Afin d'améliorer le comportement de l'asservissement visuel, et pour que les consignes ne soient pas trop éloignées des valeurs courantes des informations visuelles, nous avons également proposé des consignes sous forme de trajectoires à suivre. Ces trajectoires tiennent compte du modèle de l'avion, et sont exprimées dans l'espace 3D sous la forme de polynômes. La projection du modèle de la piste dans le plan image lorsque l'avion est à la position désirée permet alors d'obtenir les trajectoires désirées pour les informations visuelles.

Les schémas de commande proposés ont été implémentés sur un simulateur réaliste d'avion. Des simulations utilisant des mesures 3D exclusivement sont comparées aux résultats obtenus avec les informations visuelles proposées. Nous montrons également l'intérêt des trajectoires mises au point. Des simulations intégrant des erreurs sur le modèle (calibration de la caméra, connaissance du modèle de la piste, positionnement de la caméra sur l'avion, vent de travers, etc.) permettent également de discuter les performances et les limites des schémas de commande proposés.

Finalement, nous avons mis au point un schéma de commande permettant à un avion d'atterrir automatiquement. Ce schéma de commande repose essentiellement sur des données de vision, intègre une planification de trajectoire utilisant le modèle de l'avion, et se place au même niveau qu'un pilote en agissant sur les mêmes entrées de commande.

Perspectives

Le schéma de commande que nous avons proposé a donné des résultats assez satisfaisants compte tenu des hypothèses réalisées. On pourrait envisager de relâcher ces hypothèses, afin d'obtenir de meilleurs résultats et d'élargir le champ d'application de la loi de commande. En particulier, les hypothèses considérées pour modéliser les informations visuelles sont assez restrictives. Avec les informations visuelles que l'on a choisies ici, relâcher ces hypothèses conduit à rompre la propriété de découplage. D'autres informations visuelles pourraient alors être proposées. Découpler complètement les informations visuelles n'est sans doute pas nécessaire. Il nous semble que se contenter dans un premier temps d'un découplage entre les informations visuelles dédiées à la commande latérale et celles dédiées à la commande longitudinale serait un bon compromis. Par ailleurs, l'approche proposée dans [Mahony 05a], utilisant la représentation de Plücker des droites pourrait peut-être être mise à profit dans le cadre de notre application pour concevoir des informations visuelles différentes.

La modélisation des informations visuelles a été réalisée pour une position désirée particulière de la caméra, où l'axe optique est parallèle à la piste. Dans le cas où la position désirée de la caméra ne respecte pas cette hypothèse, on pourrait envisager de se ramener à cette configuration, en calculant une image virtuelle.

Le modèle que l'on a utilisé ne tient pas compte de l'influence du vent de travers, et

les lois de commande ne permettent pas de compenser son effet. Revoir la modélisation et les lois de commande en tenant compte du vent permettrait d'améliorer les performances de la commande.

Il pourrait également être intéressant de réduire les mesures externes nécessaires, et de concevoir une loi de commande purement visuelle. On pourrait pour cela utiliser non pas seulement les lignes de la piste, mais d'autres éléments visuels disponibles sur l'image d'une piste (seuil de la piste, autres marquages, etc.).

Nous avons proposé des consignes sous forme de trajectoire permettant à l'avion de rejoindre la position et l'orientation nominales pour l'atterrissage. Pour cela nous avons pris en compte le modèle de l'avion afin que la trajectoire soit "réalisable" dans l'espace en 3D. Cependant, en l'état, rien ne garantit que les trajectoires correspondantes pour les informations visuelles sont correctes, et notamment que la piste ne sorte pas de l'image. Il serait alors intéressant d'envisager de mettre au point des trajectoires directement dans l'espace image — tout en conservant les contraintes liées au modèle de l'avion. Des techniques de planification dans l'image [Mezouar 02] pourraient être exploitées à cette fin.

Dans notre étude, nous avons considéré que l'étape de traitement d'image était préalablement réalisée, et que les lignes de la piste sont donc déjà localisées dans l'image. Cependant, on voit bien que lorsque l'avion se trouve loin de la piste, celle-ci est très petite dans le plan image, et les méthodes classiques de traitement d'image seraient inadéquates. On pourrait alors concevoir une caméra équipée d'un zoom et montée sur une tourelle pan-tilt. Ainsi, un asservissement bas-niveau permettrait à la caméra de suivre la piste, et il serait alors nécessaire de tenir compte de ces nouveaux degrés de liberté entre le repère lié à la caméra et le repère de l'avion dans la loi de commande.

Enfin, nous nous sommes intéressés à l'atterrissage de l'avion. Concevoir des schémas d'asservissement visuel pourrait être intéressant dans d'autres cadres, comme le ravitaillement en vol.

Conclusion

L'un des objectifs du projet de recherche Lagadic de l'IRISA à Rennes — où se sont déroulés nos travaux — est "d'élaborer des stratégies de perception et d'action à partir d'images pour des applications en robotique" [Laga]. Cette thèse s'est déroulée dans le cadre de différents projets et collaborations (Robea-Robvolint, financement de thèse DGA et collaboration avec Dassault Aviation, PICS), qui ont maintenant pris fin. Nos travaux ont constitué pour l'équipe Lagadic une première approche de l'application de son expertise en asservissement visuel dans le cadre des engins volants.

L'équipe Lagadic est à présent impliquée dans d'autres projets d'envergure liés à la commande d'engins volants par vision.

- Le projet SCUAV (*Sensory Control of Unmanned Aerial Vehicles*) regroupe cinq laboratoires de recherche français (l'I3S et l'INRIA de Sophia Antipolis, l'IRISA à Rennes, le CEA/LIST de Fontenay-aux-Roses, Heudiasyc à Compiègne) et un industriel (Bertin Technologies à Montigny le Bretonneux). L'objectif est de pro-

poser de nouvelles techniques de commande, de stabilisation, de téléopération et de fusion de capteurs pour des drones de type VTOL. L'équipe Lagadic travaille en particulier sur des aspects de fusion de capteurs et d'asservissement visuel.

- Le projet européen PEGASE (*helicoPter and aEronef naviGation Airborne System Experimentations*) [Pega] regroupe pour sa part une quinzaine de partenaires industriels et académiques. L'objectif est de fournir de nouvelles solutions pour l'asservissement des avions et des hélicoptères dans le cadre de l'atterrissage et du décollage automatiques. L'équipe Lagadic travaille sur les aspects de suivi et localisation 3D à partir des images, et sur des lois de commande d'asservissement visuel.

Nous espérons que les travaux réalisés durant cette thèse pourront être exploités dans le cadre de ces projets.

Annexe A

Calculs relatifs à la modélisation des informations visuelles utilisées pour la commande de l'hélicoptère

A.1 Calcul de la dérivée de la fonction $\mathcal{L} = \frac{1}{2}(|\boldsymbol{\delta}_{11}|^2 + \lambda^2 \delta_{12}^2)$

On cherche à montrer que (cf. (2.48)) :

$$\dot{\mathcal{L}} = -\boldsymbol{\delta}_{\mathbf{A}}^\top \mathbf{A}(\mathbf{q}_n^*) \mathbf{Q} \mathbf{v}, \quad (\text{A.1})$$

avec

$$\mathbf{A}(\mathbf{q}_n^*) = [\mathbf{q}_n^*]_\times + \lambda \mathbf{q}_n^* \mathbf{q}_n^{*\top}. \quad (\text{A.2})$$

et où l'erreur visuelle $\boldsymbol{\delta}_{\mathbf{A}}$ est définie par (cf. (2.44)) :

$$\boldsymbol{\delta}_{\mathbf{A}} = \boldsymbol{\delta}_{11} + \lambda \mathbf{q}_n^* \delta_{12}. \quad (\text{A.3})$$

Rappelons les expressions de $\dot{\boldsymbol{\delta}}_{11}$ et $\dot{\delta}_{12}$ (cf. (2.46) et (2.47)) :

$$\dot{\boldsymbol{\delta}}_{11} = -[\boldsymbol{\omega}]_\times \boldsymbol{\delta}_{11} - [\mathbf{q}_n^*]_\times \mathbf{Q} \mathbf{v}, \quad (\text{A.4})$$

$$\dot{\delta}_{12} = -\mathbf{q}_n^{*\top} \mathbf{Q} \mathbf{v}. \quad (\text{A.5})$$

Avec l'expression de départ de \mathcal{L} (donnée par (2.45) : $\mathcal{L} = \frac{1}{2}(|\boldsymbol{\delta}_{11}|^2 + \lambda^2 \delta_{12}^2)$), on déduit par dérivation :

$$\dot{\mathcal{L}} = \boldsymbol{\delta}_{11}^\top \dot{\boldsymbol{\delta}}_{11} + \lambda^2 \delta_{12} \dot{\delta}_{12}. \quad (\text{A.6})$$

Calculons alors chacun des termes de $\dot{\mathcal{L}}$.

D'une part, on a :

$$\boldsymbol{\delta}_{11}^\top \dot{\boldsymbol{\delta}}_{11} = \boldsymbol{\delta}_{11}^\top (-[\boldsymbol{\omega}]_\times \boldsymbol{\delta}_{11} - [\mathbf{q}_n^*]_\times \mathbf{Q} \mathbf{v}).$$

Comme par définition δ_{11} est orthogonal à \mathbf{q}_n^* (cf. 2.43), le terme $-\delta_{11}^\top[\boldsymbol{\omega}]_\times \delta_{11}$ est nul. On en déduit :

$$\delta_{11}^\top \dot{\delta}_{11} = -\delta_{11}^\top[\mathbf{q}_n^*]_\times \mathbf{Q}\mathbf{v}. \quad (\text{A.7})$$

Et d'autre part,

$$\begin{aligned} \lambda^2 \delta_{12} \dot{\delta}_{12} &= \lambda^2 \delta_{12} (-\mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v}) \\ &= -\lambda^2 \delta_{12} \mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v}. \end{aligned} \quad (\text{A.8})$$

En substituant (A.7) et (A.8) dans (A.6), on obtient :

$$\dot{\mathcal{L}} = -\delta_{11}^\top[\mathbf{q}_n^*]_\times \mathbf{Q}\mathbf{v} - \lambda^2 \delta_{12} \mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v}. \quad (\text{A.9})$$

On va alors montrer que l'expression (A.1) est vraie, en la développant de manière à obtenir l'expression (A.9). Dans (A.1), on remplace \mathbf{A} et $\delta_{\mathbf{A}}$ par leurs expressions ((A.2) et (A.3)). On obtient :

$$\dot{\mathcal{L}} = -(\delta_{11} + \lambda \mathbf{q}_n^* \delta_{12})^\top ([\mathbf{q}_n^*]_\times + \lambda \mathbf{q}_n^* \mathbf{q}_n^{*\top}) \mathbf{Q}\mathbf{v}$$

et en développant, on a alors :

$$\dot{\mathcal{L}} = -\delta_{11}^\top[\mathbf{q}_n^*]_\times \mathbf{Q}\mathbf{v} - \lambda \delta_{12} \mathbf{q}_n^{*\top} [\mathbf{q}_n^*]_\times \mathbf{Q}\mathbf{v} - \lambda (\delta_{11}^\top \mathbf{q}_n^*) \mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v} - \lambda^2 \delta_{12} (\mathbf{q}_n^{*\top} \mathbf{q}_n^*) \mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v}$$

On remarque alors que les termes du milieu sont nuls :

- $-\lambda \delta_{12} \mathbf{q}_n^{*\top} [\mathbf{q}_n^*]_\times \mathbf{Q}\mathbf{v} = 0$ (propriété du produit mixte), et
- $-\lambda (\delta_{11}^\top \mathbf{q}_n^*) \mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v} = 0$ (car δ_{11} est orthogonal à \mathbf{q}_n^* (cf. 2.43)).

De plus, on remarque que $(\mathbf{q}_n^{*\top} \mathbf{q}_n^*) = 1$ (module du vecteur normalisé \mathbf{q}_n^*).

On retrouve donc, à partir de (A.1), l'expression (A.9) :

$$\dot{\mathcal{L}} = -\delta_{11}^\top[\mathbf{q}_n^*]_\times \mathbf{Q}\mathbf{v} - \lambda^2 \delta_{12} \mathbf{q}_n^{*\top} \mathbf{Q}\mathbf{v}.$$

A.2 Calcul de la dérivée de l'information visuelle $\mathbf{f} = F(|\mathbf{q}|) \mathbf{q}_n$

Dans cette annexe, nous présentons les calculs menant à l'expression de $\dot{\mathbf{f}}$, la dérivée par rapport au temps de l'information visuelle sphérique \mathbf{f} introduite à la section 2.4.5.

On rappelle que \mathbf{f} a été définie à l'équation (2.54) par : $\mathbf{f} = F(|\mathbf{q}|) \mathbf{q}_n$, avec $\mathbf{q}_n = \frac{\mathbf{q}}{|\mathbf{q}|}$ et avec $F(|\mathbf{q}|) = \frac{R|\mathbf{q}|}{\sqrt{n^2 - |\mathbf{q}|^2}}$ (cf. (2.53)). Sa dérivée s'écrit donc sous la forme :

$$\dot{\mathbf{f}} = \frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|} |\dot{\mathbf{q}}| \mathbf{q}_n + F(|\mathbf{q}|) \dot{\mathbf{q}}_n. \quad (\text{A.10})$$

- Calculons $|\dot{\mathbf{q}}| = \frac{d}{dt}(|\mathbf{q}|) = \frac{1}{|\mathbf{q}|} \mathbf{q}^\top \dot{\mathbf{q}}$.

En remplaçant $\dot{\mathbf{q}}$ par son expression (2.21), et en utilisant la propriété du produit vectoriel $\mathbf{q}^\top [\boldsymbol{\omega}]_\times \mathbf{q} = 0$, on obtient :

$$\begin{aligned} |\dot{\mathbf{q}}| &= -\frac{\mathbf{q}^\top}{|\mathbf{q}|} \mathbf{Q}\mathbf{v} \\ &= -\mathbf{q}_n^\top \mathbf{Q}\mathbf{v}. \end{aligned} \quad (\text{A.11})$$

- Calculons $\dot{\mathbf{q}}_n = \frac{d}{dt} \left(\frac{\mathbf{q}}{|\mathbf{q}|} \right)$.

En utilisant le résultat précédent (A.11) et en remplaçant à nouveau $\dot{\mathbf{q}}$ par son expression (2.21), on obtient :

$$\begin{aligned} \dot{\mathbf{q}}_n &= \frac{\dot{\mathbf{q}}}{|\mathbf{q}|} - \frac{\mathbf{q}|\dot{\mathbf{q}}|}{|\mathbf{q}|^2} \\ &= \frac{-[\boldsymbol{\omega}]_{\times}\mathbf{q} - \mathbf{Q}\mathbf{v}}{|\mathbf{q}|} - \frac{-\mathbf{q}\mathbf{q}_n^{\top}\mathbf{Q}\mathbf{v}}{|\mathbf{q}|^2}. \end{aligned}$$

En regroupant les termes en \mathbf{v} , on a finalement :

$$\dot{\mathbf{q}}_n = -[\boldsymbol{\omega}]_{\times}\mathbf{q}_n - \frac{1}{|\mathbf{q}|}(\mathbf{I}_3 - \mathbf{q}_n\mathbf{q}_n^{\top})\mathbf{Q}\mathbf{v}. \quad (\text{A.12})$$

On n'a plus qu'à injecter les expressions (A.11) et (A.12) dans (A.10), remplacer le terme $F(|\mathbf{q}|)\mathbf{q}_n$ par \mathbf{f} (c'est la définition de \mathbf{f}), et à factoriser les termes en $\mathbf{Q}\mathbf{v}$ pour obtenir

$$\begin{aligned} \dot{\mathbf{f}} &= \frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}(-\mathbf{q}_n^{\top}\mathbf{Q}\mathbf{v})\mathbf{q}_n + F(|\mathbf{q}|) \left(-[\boldsymbol{\omega}]_{\times}\mathbf{q}_n - \frac{1}{|\mathbf{q}|}(\mathbf{I}_3 - \mathbf{q}_n\mathbf{q}_n^{\top})\mathbf{Q}\mathbf{v} \right) \\ &= -\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}\mathbf{q}_n\mathbf{q}_n^{\top}\mathbf{Q}\mathbf{v} - [\boldsymbol{\omega}]_{\times}\mathbf{f} - \frac{F(|\mathbf{q}|)}{|\mathbf{q}|}(\mathbf{I}_3 - \mathbf{q}_n\mathbf{q}_n^{\top})\mathbf{Q}\mathbf{v} \\ &= -[\boldsymbol{\omega}]_{\times}\mathbf{f} - \left(\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}\mathbf{q}_n\mathbf{q}_n^{\top} + \frac{F(|\mathbf{q}|)}{|\mathbf{q}|}(\mathbf{I}_3 - \mathbf{q}_n\mathbf{q}_n^{\top}) \right) \mathbf{Q}\mathbf{v} \end{aligned} \quad (\text{A.13})$$

ce qu'on note sous la forme donnée par les équations (2.57) et (2.58) :

$$\dot{\mathbf{f}} = -[\boldsymbol{\omega}]_{\times}\mathbf{f} - \mathbf{M}\mathbf{Q}\mathbf{v}, \quad (\text{A.14})$$

avec

$$\mathbf{M}(\mathbf{q}) = \frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}\mathbf{q}_n\mathbf{q}_n^{\top} + \frac{F(|\mathbf{q}|)}{|\mathbf{q}|}(\mathbf{I}_3 - \mathbf{q}_n\mathbf{q}_n^{\top}). \quad (\text{A.15})$$

On peut remarquer que \mathbf{M} est une matrice symétrique. Le terme $\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}$ affecte la projection selon la direction de \mathbf{q}_n , et $\frac{F(|\mathbf{q}|)}{|\mathbf{q}|}$ la projection sur le plan orthogonal à \mathbf{q}_n .

Notons que le calcul est le même pour l'information visuelle $\mathbf{g} = G(|\mathbf{q}|)\mathbf{q}_n$ de la section 2.4.6. On obtient dans ce cas (équations (2.70) et (2.71)) :

$$\dot{\mathbf{g}} = -[\boldsymbol{\omega}]_{\times}\mathbf{g} - \mathbf{H}\mathbf{Q}\mathbf{v} \quad (\text{A.16})$$

avec

$$\mathbf{H}(\mathbf{q}) = \frac{\partial G(|\mathbf{q}|)}{\partial |\mathbf{q}|}\mathbf{q}_n\mathbf{q}_n^{\top} + \frac{G(|\mathbf{q}|)}{|\mathbf{q}|}(\mathbf{I}_3 - \mathbf{q}_n\mathbf{q}_n^{\top}). \quad (\text{A.17})$$

A.3 Calcul de $G(|\mathbf{q}|)$

Dans cette annexe, nous présentons les calculs permettant d'obtenir l'expression du paramètre $G(|\mathbf{q}|)$, définissant l'information visuelle $\mathbf{g} = G(|\mathbf{q}|)\mathbf{q}_n$ proposée dans la section 2.4.6. On souhaite que les deux propriétés suivantes soient satisfaites :

1. passivité : on a cette propriété si $\dot{\mathbf{g}} = -\boldsymbol{\omega} \times \mathbf{g} - \mathbf{H}\mathbf{Q}\mathbf{v}$ où \mathbf{H} est la matrice issue du facteur d'échelle G (cf. (A.17))

$$\mathbf{H}(\mathbf{q}) = \frac{\partial G(|\mathbf{q}|)}{\partial |\mathbf{q}|} \mathbf{q}_n \mathbf{q}_n^\top + \frac{G(|\mathbf{q}|)}{|\mathbf{q}|} (\mathbf{I}_3 - \mathbf{q}_n \mathbf{q}_n^\top). \quad (\text{A.18})$$

2. GAS avec même temps de convergence sur les trois composantes de l'erreur $\boldsymbol{\delta}_g$, où $\boldsymbol{\delta}_g = G(|\mathbf{q}|)\mathbf{q}_n - G(|\mathbf{q}^*|)\mathbf{q}_n^*$ (cf. (2.72)) : on a cette propriété si \mathbf{H} est de la forme

$$\mathbf{H}(|\mathbf{q}|) = \alpha(|\mathbf{q}|) \sqrt{\mathbf{M}}. \quad (\text{A.19})$$

En effet, en choisissant \mathbf{v} de la forme $\mathbf{v} = k_g \frac{\mathbf{H}(\mathbf{q})}{\alpha(|\mathbf{q}|)^2} \boldsymbol{\delta}_g$, avec $k_g > 0$, on a alors pour la dérivée de la fonction de Lyapunov $\mathcal{L} = \frac{1}{2} |\boldsymbol{\delta}_g|^2$: $\dot{\mathcal{L}} = -k_g \boldsymbol{\delta}_g^\top \frac{\mathbf{H}\mathbf{Q}\mathbf{H}}{\alpha^2} \boldsymbol{\delta}_g$. Comme $\mathbf{Q} > 0$, $\alpha^2 > 0$ et \mathbf{H} est une matrice symétrique non singulière (voir (A.18)), la théorie de Lyapunov nous permet de conclure à la stabilité asymptotique globale du système. De plus, comme $\mathbf{M} \simeq \mathbf{Q}^{-1}$ (cf. (2.65)), la contrainte (A.19) donne $\frac{\mathbf{H}\mathbf{Q}\mathbf{H}}{\alpha^2} \simeq \mathbf{I}_3$, ce qui conduit à avoir des temps de convergence sur du même ordre sur les trois composantes de l'erreur $\boldsymbol{\delta}_g$.

La forme imposée par la contrainte 2) donne, en remplaçant \mathbf{M} par son expression (2.58) :

$$\begin{aligned} \mathbf{H}(\mathbf{q}) &= \alpha(|\mathbf{q}|) \sqrt{\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}} \mathbf{q}_n \mathbf{q}_n^\top \\ &\quad + \alpha(|\mathbf{q}|) \sqrt{\frac{F(|\mathbf{q}|)}{|\mathbf{q}|}} (\mathbf{I}_3 - \mathbf{q}_n \mathbf{q}_n^\top). \end{aligned} \quad (\text{A.20})$$

En identifiant les termes de (A.18) et de (A.20) on obtient alors :

$$G(|\mathbf{q}|) = \alpha(|\mathbf{q}|) \sqrt{|\mathbf{q}| F(|\mathbf{q}|)}, \quad (\text{A.21})$$

$$\frac{\partial G(|\mathbf{q}|)}{\partial |\mathbf{q}|} = \alpha(|\mathbf{q}|) \sqrt{\frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}}. \quad (\text{A.22})$$

Puis en dérivant (A.21) et en injectant l'expression obtenue dans (A.22), on obtient :

$$\alpha' = \frac{\alpha(|\mathbf{q}|)}{\sqrt{|\mathbf{q}| F(|\mathbf{q}|)}} \left(\sqrt{F'} - \frac{F(|\mathbf{q}|) + |\mathbf{q}| F'}{2\sqrt{|\mathbf{q}| F(|\mathbf{q}|)}} \right) \quad (\text{A.23})$$

où $\alpha' = \frac{\partial \alpha(|\mathbf{q}|)}{\partial |\mathbf{q}|}$ et $F' = \frac{\partial F(|\mathbf{q}|)}{\partial |\mathbf{q}|}$.

Nous dérivons ensuite $F(|\mathbf{q}|)$ (défini par (2.53)), et nous remplaçons F' dans l'équation précédente. On obtient l'équation différentielle ordinaire suivante :

$$\alpha' = \alpha(|\mathbf{q}|) \left(\frac{2n\sqrt{D} - 2n^2 + |\mathbf{q}|^2}{2|\mathbf{q}|D} \right), \quad (\text{A.24})$$

avec $D = n^2 - |\mathbf{q}|^2$.

La solution de cette équation est alors donnée par

$$\alpha(|\mathbf{q}|) = C(n, |\mathbf{q}^*|) \frac{D^{1/4}}{|\mathbf{q}|} \sqrt{\frac{n - \sqrt{D}}{n + \sqrt{D}}} \quad (\text{A.25})$$

où $C(n, |\mathbf{q}^*|)$ est tel que $\alpha(|\mathbf{q}^*|) = 1$.

Finalement, on obtient l'expression du facteur d'échelle $G(|\mathbf{q}|)$ en remplaçant α et F par leurs expressions respectives (équations (2.53) et (A.25) respectivement) dans (A.21). On obtient :

$$G(|\mathbf{q}|) = C(n, |\mathbf{q}^*|) \sqrt{R} \sqrt{\frac{n - \sqrt{D}}{n + \sqrt{D}}}. \quad (\text{A.26})$$

Annexe B

Passage du repère géocentrique au repère lié à la piste d'atterrissage

Les matrices de changement de repère permettant de passer du repère géocentrique \mathcal{F}_0 au repère tangent à la piste \mathcal{F}_f sont décrites ici (voir figure B.1). On suppose que la piste d'atterrissage est située à la latitude ϕ , longitude θ , altitude r , et orientée d'un angle ψ par rapport à la direction du Nord. Ses coordonnées (X, Y, Z) dans le repère géocentrique \mathcal{F}_0 sont données par (voir figure B.1) :

$$X = r \cos \phi \cos \theta \quad (\text{B.1})$$

$$Y = r \cos \phi \sin \theta \quad (\text{B.2})$$

$$Z = r \sin \phi. \quad (\text{B.3})$$

Pour passer du repère \mathcal{F}_0 au repère \mathcal{F}_f , on va alors effectuer une série de transformations intermédiaires simples, et considérer successivement les repères $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$, et \mathcal{F}_4 (voir figure B.1). Pour chaque transformation du repère \mathcal{F}_i vers le repère \mathcal{F}_j , on écrit la matrice homogène de changement de repère correspondante ${}^j\mathbf{M}_i$.

- On passe de \mathcal{F}_0 à \mathcal{F}_1 par une translation de $r \sin \phi$ selon l'axe z_0 , ce qui s'écrit sous la forme de la matrice homogène de changement de repère :

$${}^1\mathbf{M}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -r \sin \phi \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{B.4})$$

- On passe de \mathcal{F}_1 à \mathcal{F}_2 par une rotation d'angle θ autour de l'axe z_1 , ce qui s'écrit sous la forme de la matrice homogène de changement de repère :

$${}^2\mathbf{M}_1 = \begin{bmatrix} \cos \theta & \sin \theta & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{B.5})$$

- On passe de \mathcal{F}_2 à \mathcal{F}_3 par une translation de $r \cos \phi$ le long de l'axe x_2 , ce qui s'écrit sous la forme de la matrice homogène de changement de repère :

$${}^3\mathbf{M}_2 = \begin{bmatrix} 1 & 0 & 0 & -r \cos \phi \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{B.6})$$

- On passe de \mathcal{F}_3 à \mathcal{F}_4 par une rotation d'angle $-(\pi/2 + \phi)$ autour de l'axe y_3 , ce qui s'écrit sous la forme de la matrice homogène de changement de repère :

$${}^4\mathbf{M}_3 = \begin{bmatrix} -\sin \phi & 0 & \cos \phi & 0 \\ 0 & 1 & 0 & 0 \\ -\cos \phi & 0 & -\sin \phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{B.7})$$

- On passe de \mathcal{F}_4 à \mathcal{F}_f par une rotation d'angle ψ autour de l'axe z_4 (on suppose que la piste est orientée selon la direction ψ par rapport au Nord), ce qui s'écrit sous la forme de la matrice homogène de changement de repère :

$${}^f\mathbf{M}_4 = \begin{bmatrix} \cos \psi & \sin \psi & 0 & 0 \\ -\sin \psi & \cos \psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{B.8})$$

Finalement, la matrice homogène de changement de repère (du repère \mathcal{F}_0 au repère \mathcal{F}_f) est la suivante :

$${}^f\mathbf{M}_0 = {}^f\mathbf{M}_4 {}^4\mathbf{M}_3 {}^3\mathbf{M}_2 {}^2\mathbf{M}_1 {}^1\mathbf{M}_0 \quad (\text{B.9})$$

c'est-à-dire :

$${}^f\mathbf{M}_0 = \begin{bmatrix} -\cos \psi \sin \phi \cos \theta - \sin \psi \sin \theta & -\cos \psi \sin \phi \sin \theta + \sin \psi \cos \theta & \cos \psi \cos \phi & 0 \\ \sin \psi \sin \phi \cos \theta - \cos \psi \sin \theta & \sin \psi \sin \phi \sin \theta + \cos \psi \cos \theta & -\sin \psi \cos \phi & 0 \\ -\cos \phi \cos \theta & -\cos \phi \sin \theta & -\sin \phi & r \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Lorsque la position de l'avion est décrite sous la forme de sa latitude ϕ_a , longitude θ_a , et altitude r_a , on peut alors facilement exprimer ses coordonnées homogènes $(X_a, Y_a, Z_a, 1)$ dans le repère \mathcal{F}_f lié à la piste :

$$\begin{bmatrix} X_a \\ Y_a \\ Z_a \\ 1 \end{bmatrix} = {}^f\mathbf{M}_0 \begin{bmatrix} r_a \cos \phi_a \cos \theta_a \\ r_a \cos \phi_a \sin \theta_a \\ r_a \sin \phi_a \\ 1 \end{bmatrix} \quad (\text{B.10})$$

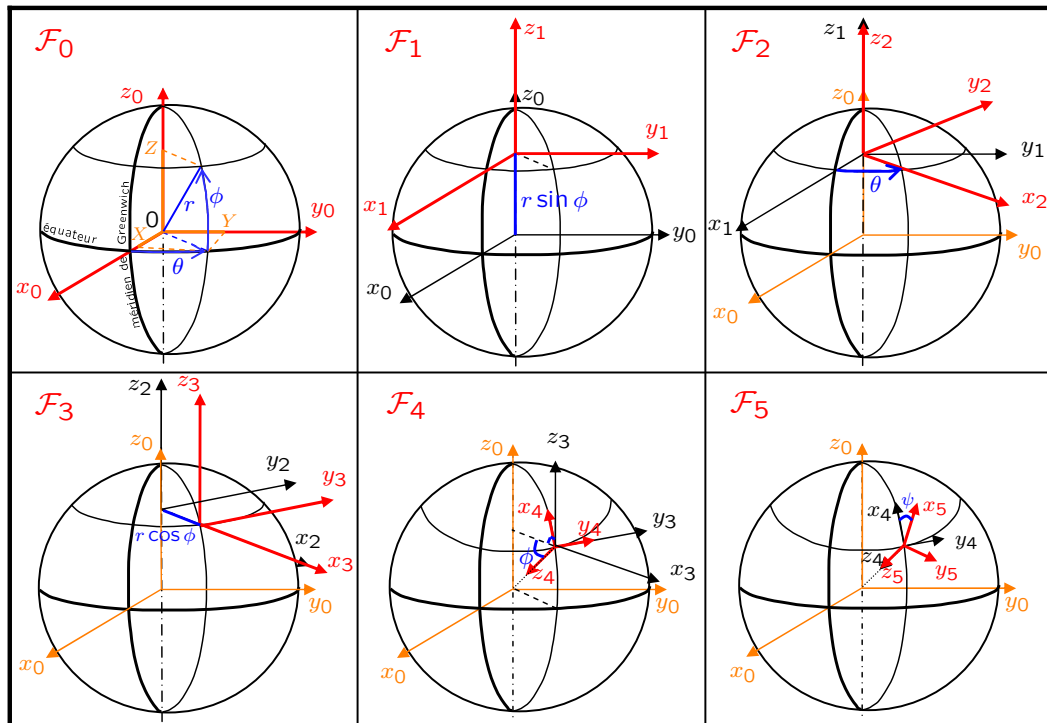


FIG. B.1 – Repères successifs considérés pour passer du repère géocentrique \mathcal{F}_0 au repère \mathcal{F}_f , tangent à la surface de la Terre, orienté selon l'axe de la piste.

Annexe C

Représentation des rotations

La matrice de rotation ${}^f\mathbf{R}_a$ (orientation du repère \mathcal{F}_a par rapport à un repère \mathcal{F}_f , figure C.1) peut être représentée sous la forme d'angles d'Euler [Dombre 88]. Cependant, il existe plusieurs conventions d'angles d'Euler, selon les axes autour desquels on tourne, et dans quel ordre. Par exemple, les angles d'Euler (z, x, z) correspondent à une première rotation autour de z , suivie d'une rotation autour du nouvel axe x , et enfin d'une rotation autour du nouvel axe z [Dombre 88]. Nous présentons ici les angles d'Euler avec la convention (z, y, x) .

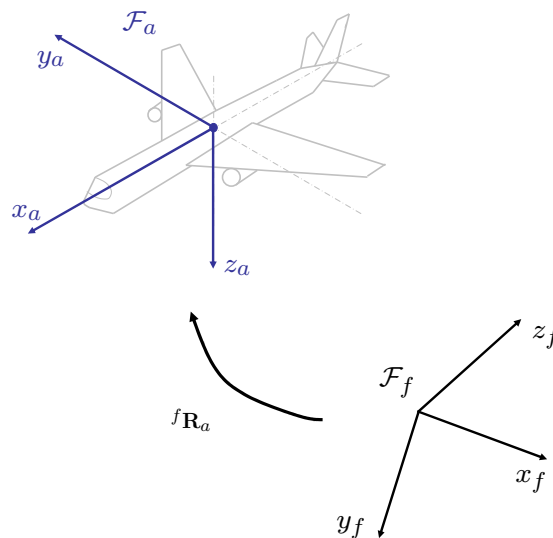


FIG. C.1 – Orientation du repère \mathcal{F}_a par rapport au repère \mathcal{F}_f

On peut interpréter cette représentation de deux façons équivalentes.

Première interprétation : On passe du repère \mathcal{F}_f au repère \mathcal{F}_a par des rotations successives autour des axes des repères successifs \mathcal{F}_f , \mathcal{F}_0 et \mathcal{F}_1 (figure C.2), comme suit :

1. rotation d'un angle ψ autour de l'axe $z_f (= z_0)$: passage du repère \mathcal{F}_f au repère \mathcal{F}_0
2. rotation d'un angle θ autour de l'axe $y_0 (= y_1)$: passage du repère \mathcal{F}_0 au repère \mathcal{F}_1
3. rotation d'un angle ϕ autour de l'axe $x_1 (= x_a)$: passage du repère \mathcal{F}_1 au repère \mathcal{F}_a

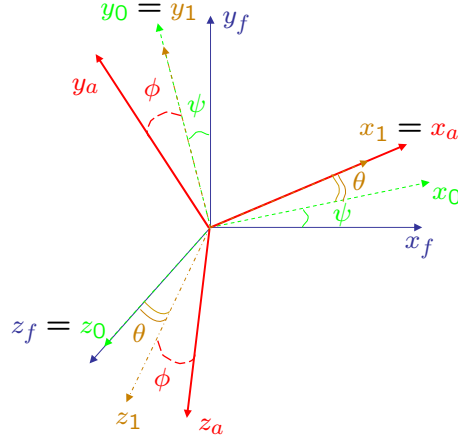


FIG. C.2 – Angles d'Euler (z, y, x) : roulis (ϕ), tangage (θ), lacet (ψ).

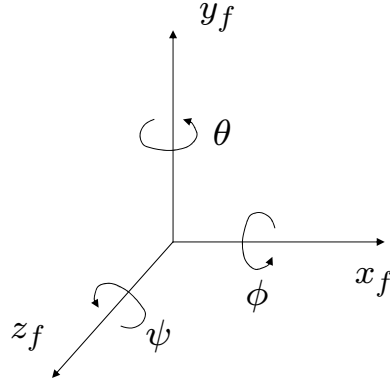
On peut alors décomposer ${}^f\mathbf{R}_a$ sous la forme ${}^f\mathbf{R}_a = {}^f\mathbf{R}_0{}^0\mathbf{R}_1{}^1\mathbf{R}_a$. En exprimant chacune des matrices de rotation intermédiaire, on obtient alors :

$$\begin{aligned}
 {}^f\mathbf{R}_a &= \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \\
 &= \begin{bmatrix} \cos \psi \cos \theta & -\sin \psi \cos \phi + \cos \psi \sin \theta \sin \phi & \sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi \\ \sin \psi \cos \theta & \cos \psi \cos \phi + \sin \psi \sin \theta \sin \phi & -\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix}
 \end{aligned} \tag{C.1}$$

Cette relation permet alors de passer de la forme matrice de rotation aux angles d'Euler (z, y, x) .

Deuxième interprétation : ${}^f\mathbf{R}_a = \mathbf{R}_{(z_f, \psi)} \mathbf{R}_{(y_f, \theta)} \mathbf{R}_{(x_f, \phi)}$
 Rotations successives dans le repère fixe \mathcal{F}_f (figure C.3) :

1. rotation d'un angle ϕ autour de l'axe x_f
2. rotation d'un angle θ autour de l'axe y_f
3. rotation d'un angle ψ autour de l'axe z_f

FIG. C.3 – Angles de roulis (ϕ), tangage (θ), lacet (ψ).

On peut alors décomposer ${}^f\mathbf{R}_a$ sous la forme ${}^f\mathbf{R}_a = \mathbf{R}_{(z_f, \psi)} \mathbf{R}_{(y_f, \theta)} \mathbf{R}_{(x_f, \phi)}$. En exprimant chacune des matrices de rotation intermédiaire, on obtient alors :

$${}^f\mathbf{R}_a = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \quad (\text{C.2})$$

On voit immédiatement qu'il s'agit de la même équation que (C.1).

Finalement, les angles de roulis-tangage-lacet sont équivalents aux angles d'Euler définis selon la convention (z, y, x) .

Remarque. On peut également montrer la relation suivante entre les composantes de la vitesse angulaire ($\boldsymbol{\omega}_a = (\omega_{a_x}, \omega_{a_y}, \omega_{a_z})$) du repère mobile \mathcal{F}_A par rapport au repère fixe \mathcal{F}_F , et les variations des angles de roulis-tangage-lacet ($\dot{\phi}, \dot{\theta}, \dot{\psi}$) [Nelson 98, Cook 97, Stevens 92] :

$$\begin{bmatrix} \omega_{a_x} \\ \omega_{a_y} \\ \omega_{a_z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & c\theta s\phi \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}. \quad (\text{C.3})$$

Annexe D

Paramètres du modèle d'avion

D.1 Valeurs numériques pour le modèle d'avion

Le modèle non-linéaire d'avion est donné par les équations (3.28) à (3.34). Les constantes intervenant dans ces équations prennent les valeurs suivantes :

$$\begin{aligned}v_{a_0} &= 55.28 && \text{m/s} \\ \alpha_0 &= 0.13218 && (= 7.57 \text{ degrés})\end{aligned}$$

avec la commande de poussée :

$$\delta_{T_0} = 0.1$$

et

$$\begin{aligned}n_{z_\alpha} &= 4.2 \\ n_{x_\alpha} &= -0.3 \\ n_{z_{00}} &= 0.99 \\ n_{x_{00}} &= -0.1175 \\ M_\alpha &= -10 \\ M_{q_0} &= -3.5 \\ M_{\delta_{m_0}} &= 0.3 \\ M_{FCS} &= 2.35 \\ L_{p_0} &= -3 \\ L_{\delta_{i_0}} &= 0.97 \\ \omega_T &= 0.25 \\ X_{\delta_T} &= 0.414 \\ L_{v-} &= -\infty \\ L_{v+} &= \infty.\end{aligned}$$

L'état de référence autour duquel les équations sont valables est le suivant :

$$\begin{aligned}
 \alpha &= 0.1326 && (= 7.6 \text{ degrés}) \\
 \omega_y &= \omega_{y0} = 0 && (\text{ou } -0.0013 \text{ pour mieux recouper les simulations}) \\
 \gamma &= \gamma_0 = -0.096 && (= -5.5 \text{ degrés}) \\
 p_v &= p_{v0} = 0 \\
 \mu &= \mu_0 = 0 \\
 v_a &= v_{a0} = 55.28 \\
 n_{x_T} &= n_{x_{T0}} = 0.0414 && (= \delta_{T_0} X_{\delta_T})
 \end{aligned}$$

avec les entrées de commande

$$\begin{aligned}
 \delta_m = \delta_{m0} &= 0 \\
 \delta_l = \delta_{l0} &= 0 \\
 \delta_T = \delta_{T0} &= 0.1.
 \end{aligned}$$

D.2 Valeurs numériques pendant la phase d'approche

On considère une pente nominale d'approche telle que $\gamma_0 = -5.5$ degrés et les valeurs nominales données précédemment (annexe D.1). On obtient alors pour les modèles donnés aux équations (3.61) et (3.68) :

Modèle latéral.

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\psi} \\ \dot{\psi} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -3.0 & 0 & 0 & 0 \\ 0.1773 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 55.026 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\psi} \\ \psi \\ Y \end{bmatrix} + \begin{bmatrix} 0.9615 \\ 0 \\ 0 \\ 0 \end{bmatrix} \delta_l \quad (\text{D.1})$$

Modèle longitudinal.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\gamma} \\ \dot{v}_a \\ \ddot{v}_a \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -10 & -3.5 & 0 & -0.08502 & 0 & 0 \\ 0.759 & 0 & -0.7526 & 0.00656 & 0.002405 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -5.881 & -2.993 & 3.398 & -0.06329 & -0.342 & 0 \\ 0 & 0 & -55.03 & 0.09585 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \gamma \\ v_a \\ \dot{v}_a \\ Z \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0.3 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1.005 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_m \\ \delta_T \end{bmatrix} \quad (\text{D.2})$$

Annexe E

Analyse de la loi de commande latérale

On considère le système bouclé par la loi de commande latérale (3.128) (voir figure 3.26). On considère ici que l'on reste dans le domaine linéaire, et on ne tient pas compte des saturations. On rappelle les valeurs numériques des gains ω_i ($i \in \{1, 2, 3\}$) : $\omega_1 = 0.08$ rad/s, $\omega_2 = 0.2$ rad/s, $\omega_3 = 0.5$ rad/s,

De manière classique, on utilise la transformée de Laplace et on considère les fonctions de transfert du système et du correcteur [Ostertag 04]. Le schéma-bloc du système bouclé devient alors celui de la figure E.1, où s est la variable symbolique $s = j\omega$.

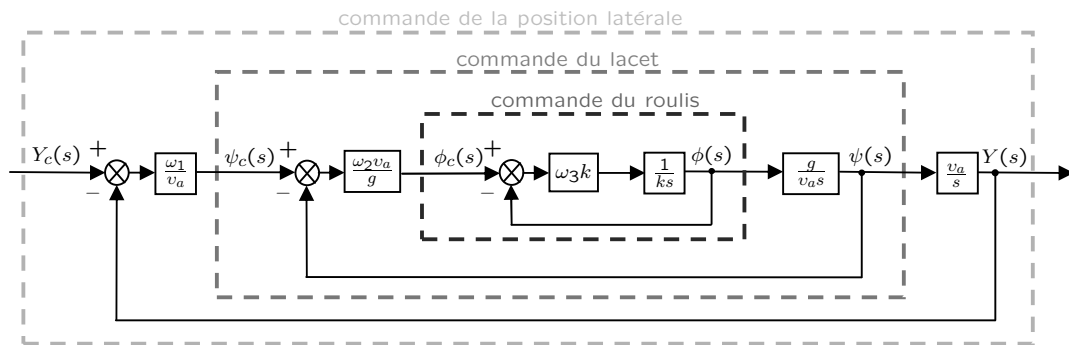


FIG. E.1 – Schéma de commande latérale.

On voit que l'on peut considérer trois "niveaux" de commande, chacun contenant un intégrateur (forme en $\frac{1}{s}$ de la fonction de transfert) : la boucle la plus interne régule l'angle de roulis, ensuite vient celle concernant l'angle de lacet, et enfin la position latérale.

Considérons tout d'abord la boucle de commande la plus interne. On peut écrire :

$$\phi(s) = \frac{\omega_3}{s}(\phi_c(s) - \phi(s))$$

d'où on déduit la fonction de transfert :

$$F_r(s) = \frac{\phi(s)}{\phi_c(s)} = \frac{\omega_3}{s + \omega_3}. \quad (\text{E.1})$$

Il agit d'un système du premier ordre, avec un pôle (racine du dénominateur de la fonction de transfert $F_r(s)$) :

$$s_r = -\omega_3 = -0.5. \quad (\text{E.2})$$

Notons qu'on peut exprimer $F_r(s)$ sous la forme classique

$$F_r(s) = \frac{1}{\tau s + 1},$$

avec $\tau = \frac{1}{\omega_3} = 2$ s la constante de temps.

Considérons à présent les deux boucles les plus internes : la commande de lacet. On peut cette fois écrire :

$$\begin{aligned} \psi(s) &= \frac{g}{v_a s} \phi(s) \\ \phi_c(s) &= \frac{\omega_2 v_a}{g} (\psi_c(s) - \psi(s)) \end{aligned}$$

d'où l'on déduit, en exprimant $\phi(s)$ en fonction de $\phi_c(s)$ avec (E.1), la fonction de transfert suivante :

$$F_l(s) = \frac{\psi(s)}{\psi_c(s)} = \frac{\omega_2 \omega_3}{s^2 + \omega_3 s + \omega_2 \omega_3}. \quad (\text{E.3})$$

On a cette fois un système du second ordre, avec deux pôles complexes conjugués à partie réelle négative (racines du polynôme $s^2 + \omega_3 s + \omega_2 \omega_3$, au dénominateur de la fonction de transfert $F_l(s)$) :

$$\begin{aligned} s_{l1} &= \frac{-\omega_3}{2} + j \frac{\sqrt{-\omega_3^2 + 4\omega_2 \omega_3}}{2} \\ &= -0.25 + j \frac{\sqrt{0.15}}{2} \\ &\simeq -0.25 + 0.19j \end{aligned} \quad (\text{E.4})$$

$$\begin{aligned} s_{l2} &= \frac{-\omega_3}{2} - j \frac{\sqrt{-\omega_3^2 + 4\omega_2 \omega_3}}{2} \\ &= -0.25 - j \frac{\sqrt{0.15}}{2} \\ &\simeq -0.25 - 0.19j. \end{aligned} \quad (\text{E.5})$$

Notons qu'on peut exprimer $F_l(s)$ sous la forme classique

$$F_l(s) = \frac{\omega_{n_l}^2}{s^2 + 2\zeta_l \omega_{n_l} s + \omega_{n_l}^2},$$

avec $\omega_{n_l} = \sqrt{\omega_2\omega_3} \simeq 0.32$ rad/s la pulsation propre et $\zeta_l = \frac{\omega_3}{2\sqrt{\omega_2\omega_3}} \simeq 0.79$ le coefficient d'amortissement. On a également $|s_{l_{1,2}}| = \omega_{n_l}$.

Enfin, considérons le système complet et la commande de la position latérale Y . On écrit, comme précédemment :

$$\begin{aligned} Y(s) &= \frac{v_a}{s}\psi(s) \\ \psi_c(s) &= \frac{\omega_1}{v_a}(Y_c(s) - Y(s)) \end{aligned}$$

et on déduit, en exprimant $\psi(s)$ en fonction de $\psi_c(s)$ avec (E.3) :

$$F_y(s) = \frac{Y(s)}{Y_c(s)} = \frac{\omega_1\omega_2\omega_3}{s^3 + \omega_3s^2 + \omega_2\omega_3s + \omega_1\omega_2\omega_3}. \quad (\text{E.6})$$

Le système entier est du troisième ordre. Il a un pôle réel négatif et deux pôles complexes conjugués à partie réelle négative (racines du polynôme $s^3 + \omega_3s^2 + \omega_2\omega_3s + \omega_1\omega_2\omega_3$, au dénominateur de la fonction de transfert $F_3(s)$:

$$\begin{aligned} s_{y_1} &= -\omega_2 \\ &= -0.2 \end{aligned} \quad (\text{E.7})$$

$$\begin{aligned} s_{y_2} &= \frac{-(\omega_3 - \omega_2)^2}{2} + j\frac{\sqrt{-(\omega_3 - \omega_2)^2 + 4\omega_1\omega_3}}{2} \\ &= -0.15 + j\frac{\sqrt{0.07}}{2} \\ &\simeq -0.15 + 0.13j \end{aligned} \quad (\text{E.8})$$

$$\begin{aligned} s_{y_3} &= \frac{-(\omega_3 - \omega_2)^2}{2} - j\frac{\sqrt{-(\omega_3 - \omega_2)^2 + 4\omega_1\omega_3}}{2} \\ &= -0.15 - j\frac{\sqrt{0.07}}{2} \\ &\simeq -0.15 - 0.13j. \end{aligned} \quad (\text{E.9})$$

On voit alors que l'on peut mettre la fonction de transfert $F_y(s)$ sous la forme :

$$F_y(s) = \frac{\omega_2}{(s + \omega_2)} \frac{\omega_1\omega_3}{(s^2 + (\omega_3 - \omega_2)s + \omega_1\omega_3)}. \quad (\text{E.10})$$

Et sous la forme classique

$$F_y(s) = \frac{\omega_2}{(s + \omega_2)} \frac{\omega_{n_y}^2}{(s^2 + 2\zeta_y\omega_{n_y}s + \omega_n^2)},$$

avec $\omega_{n_y} = \sqrt{\omega_1\omega_3} = 0.20$ rad/s la pulsation propre et $\zeta_y = \frac{\omega_3 - \omega_2}{2\sqrt{\omega_1\omega_3}} = 0.75$ le coefficient d'amortissement. Ici aussi, on a $|s_{y_{2,3}}| = \omega_{n_y}$.

Comme on peut le voir aisément sur le cas du premier ordre, plus les pôles sont loin de l'axe imaginaire, plus la constante de temps du système est faible, et donc plus

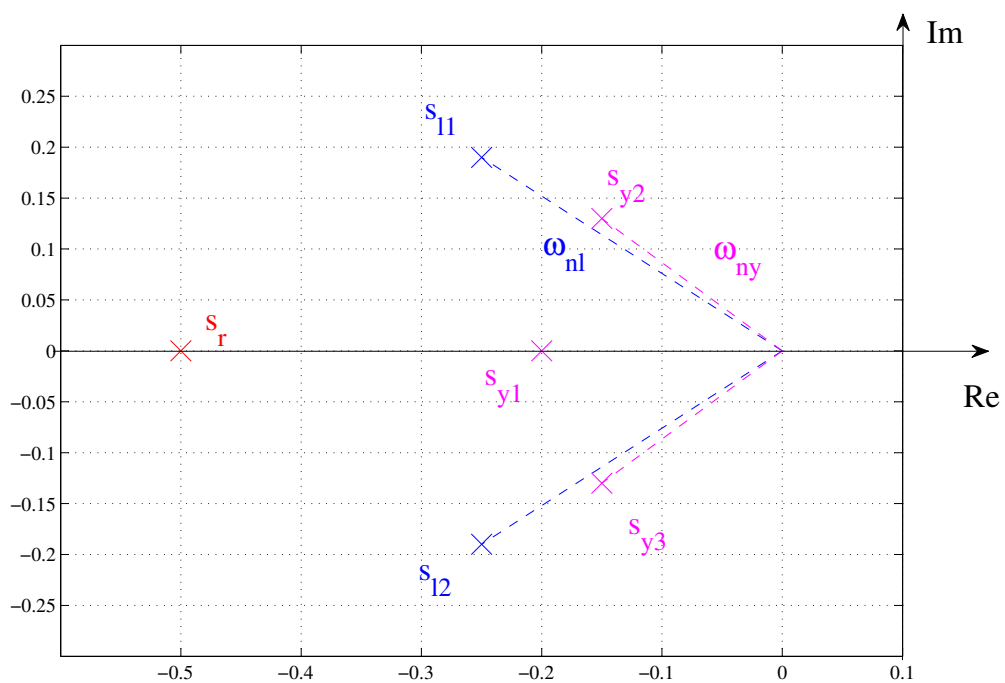


FIG. E.2 – Position des pôles pour chaque boucle de commande, dans le plan complexe : en rouge le pôle s_r de la boucle de commande du roulis, en bleu les pôles s_{l1} et s_{l2} de la boucle de commande de lacet, et en mauve les pôles s_{y1} , s_{y2} et s_{y3} de la boucle de commande de la position latérale.

la réponse du système est rapide [Ostertag 04]. Dans notre cas, on voit que la boucle la plus interne (commande du roulis) a le pôle le plus éloigné de l'axe imaginaire (s_r , voir (E.2) et figure E.2) : c'est donc la boucle qui a le temps de réponse le plus rapide. Ensuite, lorsque l'on considère la boucle de commande du lacet, ses pôles (s_{l_1} et s_{l_2}) s'approchent de l'axe imaginaire (voir (E.4) et figure E.2) : son temps de réponse est plus élevé. Enfin, les pôles de la boucle externe (s_{y_1} , s_{y_2} , s_{y_3}) sont les plus proches de l'axe imaginaire (voir (E.7) et figure E.2) : ce sont les plus lents.

Le choix des gains ω_i ($i \in \{1, 2, 3\}$) a été fait justement pour que ce soit la boucle la plus interne la plus rapide, et la boucle la plus externe la plus lente. Cela est intéressant pour découpler chaque étage. En effet, intuitivement, on comprend bien qu'il est intéressant que les parties bas-niveau d'un système réagissent rapidement, pour que la précédente consigne soit atteinte avant que la nouvelle n'arrive.

Ainsi, lorsque l'on souhaite modifier la position latérale Y_c de l'avion, cela induit un changement de cap désiré ψ_c et un nouvel angle de roulis désiré ϕ_c . L'angle de roulis réel de l'avion ϕ est alors régulé "plus vite" que l'angle de lacet ψ , qui est lui-même régulé "plus vite" que la position latérale Y .

Annexe F

Calculs des paramètres des trajectoires

F.1 Calcul des coefficients du polynôme $Y^*(t)$

On cherche à déterminer les coefficients a, b, c, d, e, f, g, h du polynôme

$$Y(t) = at^7 + bt^6 + ct^5 + dt^4 + et^3 + ft^2 + gt + h \quad (\text{F.1})$$

de manière à respecter les huit conditions suivantes :

$$\begin{aligned} Y(t_0) &= Y_i & \text{et} & Y(t_f) = 0 \\ \dot{Y}(t_0) &= a_5\psi_i & \text{et} & \dot{Y}(t_f) = 0 \\ \ddot{Y}(t_0) &= a_5a_3\phi_i & \text{et} & \ddot{Y}(t_f) = 0 \\ \dddot{Y}(t_0) &= 0 & \text{et} & \dddot{Y}(t_f) = 0 \end{aligned} \quad (\text{F.2})$$

En dérivant successivement (F.1), on obtient :

$$\dot{Y}(t) = 7at^6 + 6bt^5 + 5ct^4 + 4dt^3 + 3et^2 + 2ft + g \quad (\text{F.3})$$

$$\ddot{Y}(t) = 42at^5 + 30bt^4 + 20ct^3 + 12dt^2 + 6et + 2f \quad (\text{F.4})$$

$$\dddot{Y}(t) = 210at^4 + 120bt^3 + 60ct^2 + 24dt + 6e. \quad (\text{F.5})$$

En écrivant ces relations aux instants t_0 et t_f , on a alors, sous forme matricielle :

$$\begin{bmatrix} t_i^7 & t_i^6 & t_i^5 & t_i^4 & t_i^3 & t_i^2 & t_i & 1 \\ t_f^7 & t_f^6 & t_f^5 & t_f^4 & t_f^3 & t_f^2 & t_f & 1 \\ 7t_i^6 & 6t_i^5 & 5t_i^4 & 4t_i^3 & 3t_i^2 & 2t_i & g & 0 \\ 7t_f^6 & 6t_f^5 & 5t_f^4 & 4t_f^3 & 3t_f^2 & 2t_f & g & 0 \\ 42t_i^5 & 30t_i^4 & 20t_i^3 & 12t_i^2 & 6t_i & 2 & 0 & 0 \\ 42t_f^5 & 30t_f^4 & 20t_f^3 & 12t_f^2 & 6t_f & 2 & 0 & 0 \\ 210t_i^4 & 120t_i^3 & 60t_i^2 & 24t_i & 6 & 0 & 0 & 0 \\ 210t_f^4 & 120t_f^3 & 60t_f^2 & 24t_f & 6 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} Y_i \\ 0 \\ a_5\psi_i \\ 0 \\ a_3a_5\phi_i \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (\text{F.6})$$

On a donc une relation de la forme $\mathbf{Ax} = \mathbf{y}$ avec $\mathbf{x} = (a, b, c, f, e, g, h)$ et $\mathbf{y} = (Y_i, 0, a_5\psi_i, 0, a_3a_5\phi_i, 0, 0, 0)$. En inversant la matrice \mathbf{A} , on obtient alors $\mathbf{y} = \mathbf{A}^{-1}\mathbf{x}$ et on en déduit l'expression des coefficients en fonction de la position initiale Y_i , de l'orientation ϕ_i, ψ_i de l'avion, et des instants initiaux t_0 et finaux t_f :

$$a = \frac{2}{(-t_0 + t_f)^7} (a_3a_5\phi_it_f^2 + 5t_fa_5\psi_i - 2t_ft_0a_3a_5\phi_i + 10Y_i + t_0^2a_3a_5\phi_i - 5t_0a_5\psi_i)$$

$$b = \frac{-1}{2(-t_0 + t_f)^7} (15a_3a_5\phi_it_f^3 - 17t_f^2t_0a_3a_5\phi_i + 72t_f^2a_5\psi_i - 4t_ft_0a_5\psi_i + 140t_fY_i - 11t_ft_0^2a_3a_5\phi_i - 68t_0^2a_5\psi_i + 140Y_it_0 + 13t_0^3a_3a_5\phi_i)$$

$$c = \frac{1}{(-t_0 + t_f)^7} (10a_3a_5\phi_it_f^4 + 5t_f^3t_0a_3a_5\phi_i + 45a_5\psi_it_f^3 + 84t_f^2Y_i - 33t_f^2t_0^2a_3a_5\phi_i + 81t_f^2t_0a_5\psi_i + 252t_fY_it_0 - 87t_ft_0^2a_5\psi_i + 11t_ft_0^3a_3a_5\phi_i + 84Y_it_0^2 - 39t_0^3a_5\psi_i + 7t_0^4a_3a_5\phi_i)$$

$$d = \frac{-5}{2(-t_0 + t_f)^7} (2a_3a_5\phi_it_f^5 + 8t_f^4a_5\psi_i + 10t_f^4t_0a_3a_5\phi_i + 14Y_it_f^3 - 15t_0^2a_3a_5\phi_it_f^3 + 58t_0a_5\psi_it_f^3 - 6t_0^2a_5\psi_it_f^2 - 7t_0^3a_3a_5\phi_it_f^2 + 126Y_it_f^2t_0 - 54t_0^3a_5\psi_it_f + 9t_0^4a_3a_5\phi_it_f + 126Y_it_0^2t_f + t_0^5a_3a_5\phi_i - 6t_0^4a_5\psi_i + 14Y_it_0^3)$$

$$e = \frac{10t_ft_0}{(-t_0 + t_f)^7} (2a_3a_5\phi_it_f^4 + 8a_5\psi_it_f^3 + 14t_f^2Y_i - 5t_f^2t_0^2a_3a_5\phi_i + 13t_f^2t_0a_5\psi_i + 42t_fY_it_0 - 15t_ft_0^2a_5\psi_i + 2t_ft_0^3a_3a_5\phi_i + 14Y_it_0^2 + t_0^4a_3a_5\phi_i - 6t_0^3a_5\psi_i)$$

$$f = \frac{t_f^2}{2(-t_0 + t_f)^7} (a_3a_5\phi_it_f^5 - 7t_f^4t_0a_3a_5\phi_i - 39t_0^2a_3a_5\phi_it_f^3 + 65t_0^3a_3a_5\phi_it_f^2 - 240t_0^2a_5\psi_it_f^2 - 420t_0^2t_f + 10t_0^4a_3a_5\phi_it_f + 60t_0^3a_5\psi_it_f + 180t_0^4a_5\psi_i - 30t_0^5a_3a_5\phi_i - 420t_0^3)$$

$$g = \frac{-t_f^3}{(-t_0 + t_f)^7} (-t_f^4a_5\psi_i + t_f^4t_0a_3a_5\phi_i - 7t_0^2a_3a_5\phi_it_f^3 + 7t_0a_5\psi_it_f^3 - 21t_0^2a_5\psi_it_f^2 + t_0^3a_3a_5\phi_it_f^2 + 15t_0^4a_3a_5\phi_it_f - 45t_0^3a_5\psi_it_f - 140t_0^3 - 10t_0^5a_3a_5\phi_i + 60t_0^4a_5\psi_i)$$

$$\begin{aligned}
h = & \frac{t_f^4}{2(-t_0 + t_f)^7} (2yt_f^3 - 14yt_f^2t_0 + 42yt_0^2t_f - 70yt_0^3 + 14t_0^2a_5\psi_it_f^2 \\
& - 2t_0a_5\psi_it_f^3 - 42t_0^3a_5\psi_it_f + 30t_0^4a_5\psi_i - 7t_0^3a_3a_5\phi_it_f^2 \\
& + t_0^2a_3a_5\phi_it_f^3 + 11t_0^4a_3a_5\phi_it_f - 5t_0^5a_3a_5\phi_i)
\end{aligned}$$

F.2 Calcul des coefficients du polynôme $Z^*(X)$

Les paramètres a, b, c, d du polynôme

$$Z^*(X) = aX^3 + bX^2 + cX + d, \quad (\text{F.7})$$

tel que

$$Z^*(X_i) = Z_i \quad (\text{F.8})$$

$$Z^*(X_f) = Z_f \quad (\text{F.9})$$

$$\frac{\partial Z^*}{\partial X}(X_i) = \tan(-\gamma_i) \quad (\text{F.10})$$

$$\frac{\partial Z^*}{\partial X}(X_f) = \tan(-\gamma_0) \quad (\text{F.11})$$

sont obtenus selon la même méthode que pour $Y^*(t)$.

On a :

$$\begin{aligned}
a = & \frac{1}{\cos(\gamma_0) \cos(\gamma_i)(-X_i + X_f)^3} (2 Z_i \cos(\gamma_0) \cos(\gamma_i) \\
& + X_f \sin(\gamma_0) \cos(\gamma_i) + X_i \sin(\gamma_i) \cos(\gamma_0) \\
& - X_f \sin(\gamma_i) \cos(\gamma_0) + X_i \sin(\gamma_0) \cos(\gamma_i))
\end{aligned}$$

$$\begin{aligned}
b = & -\frac{1}{\cos(\gamma_0) \cos(\gamma_i)(-X_i + X_f)^3} (-2 X_f^2 \sin(\gamma_i) \cos(\gamma_0) \\
& + 2 X_f^2 \sin(\gamma_0) \cos(\gamma_i) + X_i X_f \sin(\gamma_i) \cos(\gamma_0) \\
& + 3 X_f Z_i \cos(\gamma_0) \cos(\gamma_i) + 2 X_f X_i \sin(\gamma_0) \cos(\gamma_i) \\
& + 3 Z_i X_i \cos(\gamma_0) \cos(\gamma_i) + 2 X_i^2 \sin(\gamma_0) \cos(\gamma_i) \\
& + X_i^2 \sin(\gamma_i) \cos(\gamma_0))
\end{aligned}$$

$$\begin{aligned}
c = & -\frac{1}{\cos(\gamma_0) \cos(\gamma_i)(-X_i + X_f)^3} (-6 X_f Z_i \cos(\gamma_0) \cos(\gamma_i) X_i \\
& - 4 X_f^2 X_i \sin(\gamma_0) \cos(\gamma_i) + X_f^3 \sin(\gamma_i) \cos(\gamma_0) \\
& + X_i X_f^2 \sin(\gamma_i) \cos(\gamma_0) - 2 X_i^2 X_f \sin(\gamma_i) \cos(\gamma_0) \\
& - X_i^2 \sin(\gamma_0) \cos(\gamma_i) X_f - X_i^3 \sin(\gamma_0) \cos(\gamma_i))
\end{aligned}$$

$$d = \frac{1}{\cos(\gamma_0) \cos(\gamma_i) (-X_i + X_f)^3} (X_f^2 (X_f Z_i \cos(\gamma_0) \cos(\gamma_i) - 3Z_i X_i \cos(\gamma_0) \cos(\gamma_i) - 2X_i^2 \sin(\gamma_0) \cos(\gamma_i) - X_i^2 \sin(\gamma_i) \cos(\gamma_0) + X_i X_f \sin(\gamma_i) \cos(\gamma_0)))$$

Bibliographie

- [Altuž 05] E. Altuž, J. P. Ostrowski, C. J. Taylor. – Control of a quadrotor helicopter using dual camera visual feedback. *International Journal of Robotics Research, IJRR'05*, 24(5) :329–341, 2005.
- [Amidi 99] O. Amidi, T. Kanade, K. Fujita. – A visual odometer for autonomous helicopter flight. *Journal of Robotics and Autonomous Systems*, 28 :185–193, Août 1999.
- [Andreff 00] N. Andreff, B. Espiau, R. Horaud. – Visual servoing from lines. – *IEEE International Conference on Robotics and Automation, ICRA'00*, vol. 3, pp. 2070–2075, San Francisco, CA, États-Unis, Avril 2000.
- [Andreff 02] N. Andreff, B. Espiau, R. Horaud. – Visual servoing from lines. *International Journal of Robotics Research*, 21(8) :679–700, Août 2002.
- [Arulampalam 02] S. Arulampalam, S. Maskell, N. J. Gordon, T. Clapp. – A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions of Signal Processing*, 50(2) :174–188, Février 2002.
- [Ava] Site web du projet AVATAR. – <http://www-robotics.usc.edu/~avatar/>.
- [Azinheira 02] J. R. Azinheira, P. Rives, J. R. H. Carvalho, G. F. Silveira, E. C. de Paiva, S. S. Bueno. – Visual servo control for the hovering of an outdoor robotic airship. – *IEEE International Conference on Robotics and Automation, ICRA'02*, vol. 3, pp. 2787–2792, Washington, DC, États-Unis, Mai 2002.
- [Ban] Site web du projet de robot d'inspection de lignes électriques de *Bangor University*. – http://www.informatics.bangor.ac.uk/%7Edewi/ci_grp/ripl_web.htm.
- [Bea] Site web du projet BEAR. – <http://robotics.eecs.berkeley.edu/bear/>.
- [Bonnet 01] S. Bonnet, J. Verrière. – *La mécanique du vol de l'avion léger*. – Cépaduès éditions, 2001.

- [Breheny 03] S.H. Breheny, R. D'Andrea, J.C. Miller. – Using airborne vehicle-based antenna arrays to improve communications with UAV clusters. – *IEEE Conference on Decision and Control CDC'03*, vol. 4, pp. 4158–4162, Hyatt Regency Maui, Hawaii, États Unis, Décembre 2003.
- [Buskey 03] G. Buskey, P. Corke, J. Roberts, P. Ridley, G. Wyeth. – The CSIRO autonomous helicopter project. *Experimntal Robotics VIII*, éd. par Bruno Siciliano, Paulo Dario. – Springer-Verlag, 2003.
- [Caballero 05] F. Caballero, L. Merino, J. Ferruz, A. Ollero. – A visual odometer without 3D reconstruction for aerial vehicles. applications to building inspection. – *IEEE International Conference on Robotics and Automation, ICRA'05*, Barcelone, Espagne, Avril 2005.
- [Car] Site web du projet d'hélicoptère de *Carnegie Mellon University*.
–
<http://www.cs.cmu.edu/afs/cs/project/chopper/www/goals.html>.
- [Cervera 03] E. Cervera, A. P. del Pobil, F. Berry, P. Martinet. – Improving image-based visual servoing with three-dimensional features. *International Journal of Robotics Research, IJRR'03*, 22(10-11) :821–839, 2003.
- [Chatterji 96] G. B. Chatterji, P. K. Menon, B. Sridhar. – Vision-based position and attitude determination for aircraft night landing. – *AIAA Guidance, Navigation and Control Conference, 1996*, San Diego, CA, États Unis, Juillet 1996.
- [Chaumette 90] F. Chaumette. – *La relation vision-commande : théorie et application à des tâches robotiques*. – Thèse de doctorat, Université de Rennes 1, IRISA, Juillet 1990.
- [Chaumette 98] F. Chaumette. – Potential problems of stability and convergence in image-based and position-based visual servoing. *The Confluence of Vision and Control*, éd. par D. Kriegman, G. Hager, A.S. Morse, pp. 66–78. – LNCIS Series, No 237, Springer-Verlag, 1998.
- [Chaumette 04] F. Chaumette. – Image moments : a general and useful set of features for visual servoing. *IEEE Transactions on Robotics*, 20(4) :713–723, Août 2004.
- [Chaumette 06] F. Chaumette, S. Hutchinson. – Visual servo control, part I : Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4) :82–90, Décembre 2006.
- [Chaumette 07] F. Chaumette, S. Hutchinson. – Visual servo control, part II : Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1) :109–118, Mars 2007.

- [Che 01] Jun Che, Degang Chen. – Automatic landing control using h_∞ control and stable inversion. – *IEEE Conference on Decision and Control, CDC'01*, vol. 1, pp. 241–246, Orlando, FL, États-Unis, Décembre 2001.
- [Chenavier 92] F. Chenavier, J.L. Crowley. – Position estimation for a mobile robot using vision and odometry. – *IEEE International Conference on Robotics and Automation, ICRA'92*, vol. 3, pp. 2588–2593, Nice, France, Mai 1992.
- [Col] Site web du projet COLIBRI. – <http://138.100.76.150/~colibri/>.
- [Com] Site web du projet COMETS. – <http://www.comets-uavs.org/>.
- [Cook 97] M. V. Cook. – *Flight Dynamics Principles*. – Butterworth Heinemann, 1997.
- [Corke 96] P. I. Corke, M. C. Good. – Dynamic effects in visual closed-loop systems. *IEEE Transactions on Robotics and Automation*, 12(5) :671–683, 1996.
- [Corke 01] P. Corke, S. A. Hutchinson. – A new partitioned approach to image-based visual servo control. *IEEE Transactions on Robotics and Automation*, 17(4) :507–515, Août 2001.
- [CoronadoVergara 05] J. Coronado-Vergara, G. Avina-Cervantes, M. Devy, C. Parra. – Towards landmine detection using artificial vision. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, pp. 659–664, Edmonton, AB, Canada, Août 2005.
- [Crétual 01] A. Crétual, F. Chaumette. – Visual servoing based on image motion. *International Journal of Robotics Research, IJRR'01*, 20(11) :857–877, Novembre 2001.
- [Csi] Site web du projet d'hélicoptère du CSIRO. – <http://www.ict.csiro.au/page.php?did=37>.
- [Deguchi 98] K. Deguchi. – Optimal motion control for image-based visual servoing bydecoupling translation and rotation. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'98*, vol. 2, pp. 705–711, Victoria, BC, Canada, Octobre 1998.
- [DeMenthon 95] D. DeMenthon, L. S. Davis. – Model-based object pose in 25 lines of code. *International Journal of Computer Vision, IJCV'95*, 15 :123–141, 1995.
- [Demonceaux 07] C. Demonceaux, P. Vasseur, C. Pégard. – UAV attitude computation by omnidirectional vision in urban environment. – *IEEE International Conference on Robotics and Automation, ICRA'07*, pp. 2017–2022, Rome, Italie, 10-14 Avril 2007.

- [Deng 05] L. Deng, F. Janabi-Sharifi, W. J. Wilson. – Hybrid motion control and planning strategies for visual servoing. *IEEE Transactions on Industrial Electronics*, 52 :1024–1040, Août 2005.
- [DeWagter 05] C. de De Wagter, J. A. Mulder. – Towards vision-based UAV situation awareness. – *AIAA Guidance, Navigation, and Control Conference*, Août 2005.
- [Doherty 00] P. Doherty, G. Granlund, K. Kuchcinski, E. Sandewall, K. Nordberg, E. Skarman, J. Wiklund. – The WITAS unmanned aerial vehicle project. – *European Conference on Artificial Intelligence, ECAI'00*, pp. 747–755, Berlin, Allemagne, Août 2000.
- [Doitsidis 04] L. Doitsidis, K.P. Valavanis, N.C. Tsourveloudis, M. Kontitsis. – A framework for fuzzy logic based UAV navigation and control. – *IEEE International Conference on Robotics and Automation, ICRA'04*, vol. 4, pp. 4041–4046, États-Unis, Avril 2004.
- [Dombre 88] E. Dombre, W. Khalil. – *Modélisation et commande des robots*. – Hermès, 1988.
- [Dunbabin 04] M. Dunbabin, P. Corke, G. Buskey. – Low-cost vision-based AUV guidance system for reef navigation. – *IEEE International Conference on Robotics and Automation, ICRA'04*, pp. 7–12, Nouvelle Orléans, LA, États-Unis, Avril 2004.
- [Dune 07] C. Dune, C. Leroux, E. Marchand. – Intuitive human interactive with an arm robot for severely handicapped people - A one click approach. – *IEEE International Conference on Rehabilitation Robotics, ICORR'2007*, pp. 582–589, Noordwijk, Pays-Bas, Juin 2007.
- [Earl 04] M.G. Earl, R. D'Andrea. – Real-time attitude estimation techniques applied to a four rotor helicopter. – *IEEE Conference on Decision and Control, CDC'04*, vol. 4, pp. 3956–3961, Atlantis, Paradise Island, Bahamas, Décembre 2004.
- [Elfes 98] A. Elfes, S. Siqueira Bueno, M. Bergerman, Jr. Ramos, J.G. – A semi-autonomous robotic airship for environmental monitoring missions. – *IEEE International Conference on Robotics and Automation, ICRA'98*, vol. 4, pp. 3449–3455, Louvain, Belgique, 16-20 Mai 1998.
- [Espiau 92] B. Espiau, F. Chaumette, P. Rives. – A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8(3) :313–326, Juin 1992.
- [Etkin 95] B. Etkin, L. D. Reid. – *Dynamics of flight : Stability and control*. – John Wiley & Sons Inc, Troisième édition, 1995.
- [Ettinger 02] S.M. Ettinger, M.C. Nechyba, P.G. Ifju, M. Waszak. – Vision-guided flight stability and control for micro air vehicles. –

- IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'02*, vol. 3, pp. 2134–2140, Lausanne, Suisse, Octobre 2002.
- [Eur] Site web du tutorial d'asservissement visuel présenté à IROS 2004 : P. Martinet, F. Chaumette, K. Hashimoto, E. Malis. – <http://www.robot.uji.es/EURON/visualservoing/tutorial/>.
- [Faugeras 93] O. Faugeras. – *Three-dimensional computer vision : a geometric viewpoint*. – MIT Press, Cambridge, MA, États Unis, 1993.
- [Ferreira 01] A. Ferreira, C. Cassier, Y. Haddab, P. Rougeot, N. Chaillet. – Development of a teleoperated micromanipulation system with visual and haptic feedback. – *SPIE, Microrobotics and Microassembly III*, vol. 4568, pp. 112–123, Newton, MA, États-Unis, Octobre 2001.
- [Fitzgerald 05] D. Fitzgerald, R. Walker, D. Campbell. – A vision based forced landing site selection system for an autonomous UAV. – *International Conference on Intelligent Sensors, Sensor Networks and Information Processing Conference, ISSNIP'05*, pp. 397–402, Melbourne, Australie, 5-8 Décembre 2005.
- [Flo] Site web du projet de drone de *University of Central Florida*. – <http://server.cs.ucf.edu/~vision/projects/UAV/UAV.html>.
- [Fravolini 03] M. Fravolini, A. Ficola, M. Napolitano, G. Campa, M. Perhinschi. – Development of modelling and control tools for aerial refueling for UAVs. – *AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA-2003-5798*, Austin, Texas, Août 2003.
- [Frew 04] E. Frew, T. McGee, Z. Kim, X. Xiao, S. Jackson, M. Morimoto, S. Rathinam, J. Padiyal, R. Sengupta. – Vision-based road following using a small autonomous aircraft. – *IEEE Aerospace Conference*, Big Sky, MT, Mars 2004.
- [Furst 99] S. Furst, E. D. Dickmanns. – A vision based navigation system for autonomous aircraft. *Robotics and Autonomous Systems*, 28(2-3) :173–184, Août 1999.
- [Gangloff 00] J. Gangloff, M. De Mathelin. – High speed visual servoing of a 6 DOF manipulator using MIMO predictive control. – *International Conference on Robotics and Automation, ICRA'00*, U.S.A., San Francisco, Californie, Avril 2000.
- [Gangloff 04] J. Gangloff. – *Asservissements visuels et chirurgie robotisée*. – Strasbourg, Habilitation à diriger les recherches, Université Louis Pasteur, Décembre 2004.
- [Gans 07] N. R. Gans, S. A. Hutchinson. – Stable visual servoing through hybrid switched-system control. *IEEE Transactions on Robotics, ITRO'07*, 23(3) :530–540, Juin 2007.

- [Ginhoux 04] R. Ginhoux, J. Gangloff, M. de Mathelin, L. Soler, M. A. Sanchez. – Beating heart tracking in robotic surgery using 500 Hz visual servoing, model predictive control and an adaptive observer. – *IEEE International Conference on Robotics and Automation, ICRA'04*, Nouvelle Orléans, LA, États-Unis, Avril 2004.
- [Green 04] W.E. Green, P.Y. Oh, G. Barrows. – Flying insect inspired vision for autonomous aerial robot maneuvers in near-earth environments. – *IEEE International Conference on Robotics and Automation, ICRA'04*, vol. 3, pp. 2347–2352, Nouvelle Orléans, LA, États-Unis, Avril 2004.
- [Guenard 05] N. Guenard, T. Hamel, V. Moreau. – Dynamic modeling and intuitive control strategy for an X4-flyer. – *International Conference on Control and Automation, ICCA'05*, Budapest, Hongrie, Juin 2005.
- [Guenard 06a] N. Guenard, T. Hamel, L. Eck. – Control law for the tele operation of an unmanned aerial vehicle known as an X4-flyer. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'06*, Pékin, Chine, Octobre 2006.
- [Guenard 06b] N. Guenard, T. Hamel, V. Moreau, R. Mahony. – Design of a controller allowed the intuitive control of an X4-flyer. – *International IFAC Symposium on Robot Control, University of Bologna*, Italie, Septembre 2006.
- [Guenard 07] N. Guenard, T. Hamel, R. Mahony. – A practical visual servo control for a unmanned aerial vehicle. – *IEEE International Conference on Robotics and Automation, ICRA'07*, pp. 1342–1348, Rome, Italie, 10-14 Avril 2007.
- [Hamel 02] T. Hamel, R. Mahony. – Visual servoing of an under-actuated dynamic rigid-body system : an image based approach. *IEEE Transactions on Robotics and Automation*, 18(2) :187–198, Avril 2002.
- [Hamel 06] T. Hamel, L. Eck, F. Chaumette, A. Chriette. – Rapport final du projet ROBVOLINT (ROBot VOLant d'INTérieur). – *Journées du Programme Interdisciplinaire ROBEA*, pp. 32–40, Paris, France, Avril 2006.
- [Han 01] J. W. Han, L. Guo. – An algorithm for automatic detection of runways in aerial images. *International Journal of Machine Graphics and Vision*, 10 :503–518, Septembre 2001.
- [Hartley 00] R. I. Hartley, A. Zisserman. – *Multiple View Geometry in Computer Vision*. – Cambridge University Press, 2000.
- [Hrabar 03] S. Hrabar, G. Sukhatme. – Omnidirectional vision for an autonomous helicopter. – *IEEE International Conference on Robotics*

- and Automation, ICRA '03*, pp. 3602–3609, Taipei, Taiwan, Septembre 2003.
- [Hrabar 05] S. Hrabar, G. Sukhatme, P. Corke, K. Usher, J. Roberts. – Combined optic-flow and stereo-based navigation of urban canyons for a UAV. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, Edmonton, AB, Canada, Août 2005.
- [Huang 05] J.-H. Huang, C.-M. and Jean, Y.-S. Cheng, L.-C. Fu. – Visual tracking and servoing system design for circling a target of an air vehicle simulated in virtual reality. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, pp. 2393–2398, Edmonton, AB, Canada, 2-6 Août 2005.
- [Huertas 90] A. Huertas, W. Cole, R. Nevatia. – Detecting runways in complex airport scenes. – *Computer Vision, Graphics, and Image Processing, CVGIP*, vol. 51, pp. 107–145, Août 1990.
- [Hum] Site web du projet HUMMINGBIRD. – <http://sun-valley.stanford.edu/~heli/>.
- [Hutchinson 96] S. Hutchinson, G. Hager, P. Corke. – A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5) :651–670, 1996.
- [Hygounen 04] E. Hygounen, I. K. Jung, P. Soueres, S. Lacroix. – The autonomous blimp project at LAAS/CNRS : achievements in flight control and terrain mapping. *International Journal of Robotics Research, IJRR'04*, 23(4-5) :473–511, Avril-Mai 2004.
- [Iiguni 98] Y. Iiguni, H. Akiyoshi, N. Adachi. – An intelligent landing system based on a human skill model. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3) :877–882, Juillet 1998.
- [Izadi 03] H. Izadi, M. Pakmehr, N. Sadati. – Optimal neuro-controller in longitudinal auto-landing of a commercial jet transport. – *IEEE Conference on Control Applications*, vol. 1, pp. 492–497, 23-25 Juin 2003.
- [Jorgensen 90] C. C. Jorgensen, C. Schley. – A neural network baseline problem for control of aircraft flare and touchdown. *Neural networks for control*, pp. 403–425, 1990.
- [Junkins 99] J. L. Junkins, D. C. Hughes, K. P. Wazni, V. Pariyapong. – Vision-based navigation for rendezvous, docking, and proximity operations. – *Annual AAS Guidance and Control Conference, AAS 99-021*, Février 1999.
- [Kemp 05] C. Kemp, T. Drummond. – Dynamic measurement clustering to aid real time tracking. – *IEEE International Conference on Computer Vision, ICCV'05*, pp. 1500–1507, Pékin, Chine, 17-20 Octobre 2005.

- [Kermode 01] A. C. Kermode. – *Mécanique du vol*. – Modulo, Deuxième édition, 2001.
- [Khalil 99] W. Khalil, E. Dombre. – *Modélisation, identification et commande des robots*. – Hermès, Paris, 2ème édition, 1999.
- [Krupa 07] A. Krupa, G. Fichtinger, G. D. Hager. – Real-time tissue tracking with B-mode ultrasound using speckle and visual servoing. – N. Ayache, S. Ourselin, A. Maeder (édité par), *International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI'07*, vol. 2 of *LNCS 4792*, pp. 1–8, Brisbane, Australie, Octobre 2007.
- [Lacroix 05] S. Lacroix. – Ballons dirigeables autonomes. – *Journées nationales de la recherche en robotique, JNRR'05*, Guidel, France, Octobre 2005.
- [Laga] Site web de l'équipe projet Lagadic de l'IRISA. – <http://www.irisa.fr/lagadic/>.
- [LeBras 06] F. Le Bras, R. Mahony, T. Hamel, P. Binetti. – Adaptive filtering and image based visual servo control of a ducted fan flying robot. – *IEEE Conference on Decision and Control, CDC'06*, pp. 1751–1757, 13-15 Décembre 2006.
- [Lexdas] Lexique bilingue en ligne, sur le site de Dassault Aviation. – <http://www.dassault-aviation.com/services/fr/transverse/lexique-bilingue.html>.
- [Li 04] Y. Li, N. Sundararajan, P. Saratchandran, W. Zhifeng. – Robust neuro-h/sub /spl infin controller design for aircraft auto-landing. *Transactions on Aerospace and Electronic Systems*, 40(1) :158–167, Janvier 2004.
- [Lowe 87] D. G. Lowe. – Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3) :355–395, Mars 1987.
- [Mahony 02a] R. Mahony, P. Corke, F. Chaumette. – Choice of image features for depth-axis control in image-based visual servo control. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'02*, vol. 1, pp. 390–395, Lausanne, Suisse, Octobre 2002.
- [Mahony 02b] R. Mahony, T. Hamel, F. Chaumette. – A decoupled image space approach to visual servoing for a robotic manipulator. – *IEEE International Conference on Robotics and Automation, ICRA'02*, Washington, DC, États-Unis, 2002.
- [Mahony 05a] R. Mahony, T. Hamel. – Image-based visual servo control of aerial robotic systems using linear image features. *IEEE Transactions on Robotics, ITRA'05*, 21 :227–239, 2005.

- [Mahony 05b] R. Mahony, T. Hamel, J.-M. Pflimlin. – Complementary filter design on the special orthogonal group $SO(3)$. – *IEEE Conference on Decision and Control et European Control Conference, CDC'05 et ECC'05*, pp. 1477–1484, Séville, Espagne, 2005.
- [Malis 98] E. Malis. – *Contributions à la modélisation et à la commande en asservissement visuel*. – PhD. Thesis, Université de Rennes 1, Télécommunications et Traitement du Signal, Novembre 1998.
- [Malis 99] E. Malis, F. Chaumette, S. Boudet. – 2 1/2 D visual servoing. *IEEE Transactions on Robotics and Automation*, 15(2) :238–250, Avril 1999.
- [Malis 02] E. Malis, F. Chaumette. – Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods. *IEEE Transactions on Robotics and Automation*, 18(2) :176–186, Avril 2002.
- [Malis 04] E. Malis. – Improving vision-based control using efficient second order minimization techniques. – *IEEE International Conference on Robotics and Automation, ICRA'04*, pp. 1843–1848, Nouvelle Orléans, Avril 2004.
- [Mansard 07] N. Mansard, O. Stasse, F. Chaumette, K. Yokoi. – Visually-guided grasping while walking on a humanoid robot. – *IEEE International Conference on Robotics and Automation, ICRA'07*, Rome, Italie, Avril 2007.
- [Marchand 02] E. Marchand, F. Chaumette. – Virtual visual servoing : a framework for real-time augmented reality. – G. Drettakis, H.-P. Seidel (édité par), *EUROGRAPHICS 2002 Conference Proceeding*, vol. 21(3) of *Computer Graphics Forum*, pp. 289–298, Saarebrücken, Allemagne, Septembre 2002.
- [Marchand 05] E. Marchand, F. Spindler, F. Chaumette. – ViSP for visual servoing : a generic software platform with a wide class of robot control skills. *IEEE Robotics and Automation Magazine*, 12(4) :40–52, Décembre 2005.
- [McLean 90] D. Mc Lean. – *Automatic Flight Control Systems*. – Prentice Hall International Series in Systems and Control Engineering, 1990.
- [Mejias 06a] L. Mejias, J. Roberts, K. Usher, P. Corke, P. Campoy. – Two seconds to touchdown. Vision-based controlled forced landing. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'06*, pp. 3527–3532, Pékin, Chine, Octobre 2006.
- [Mejias 06b] L. Mejias, S. Saripalli, P. Cervera, G. S. Sukhatme. – Visual servoing of an autonomous helicopter in urban areas using feature tracking. *In Journal of Field Robotics*, 3 :185–199, 2006.

- [Mezouar 02] Y. Mezouar, F. Chaumette. – Path planning for robust image-based control. *IEEE Transactions on Robotics and Automation*, 18(4) :534–549, Août 2002.
- [Mit] Site web du projet de drones du MIT. – <http://vertol.mit.edu/>.
- [Morel 99] G. Morel, J. Szewczyk, S. Boudet, J. Pot. – Explicit incorporation of 2d constraints in vision based control of robot manipulators. – P. Corke, J. Trevelyan (édité par), *ISER'99 : Experimental Robotics IV*, pp. 99–108, Australie, Sidney, Avril 1999. Springer Verlag.
- [Nar] Site web du projet NARVAL. – <http://damiao.isr.ist.utl.pt/vislab/NARVAL/index.htm>.
- [Nelson 98] R. C. Nelson. – *Flight Stability and Automatic Control*. – McGraw-Hill Education, 1998.
- [Ocv] Site web de la bibliothèque OpenCV. – <http://sourceforge.net/projects/opencvlibrary/>.
- [Ollero 04] A. Ollero, J. Ferruz, F. Caballero. – Motion compensation and object detection for autonomous helicopter visual navigation in the COMETS system. – *IEEE International Conference on Robotics and Automation, ICRA'04*, pp. 19–24, Nouvelle Orléans, LA, États-Unis, Avril 2004.
- [Ollero 05] A. Ollero, S. Lacroix, L. Merino, J. Gancet, J. and Wiklund, V. Remuss, L.G. Perez, I.V. and Gutierrez, D.X. Viegas, M.A.G. Benitez, A. Mallet, R. Alami, G. Chatila, R. and Hommel, F.J.C. Lechuga, B.C. Arrue, J. Ferruz, J.R. Martinez-De Dios, F. Caballero. – Multiple eyes in the skies : architecture and perception issues in the COMETS unmanned air vehicles project. *IEEE Robotics and Automation Magazine*, 12(2) :46–57, Juin 2005.
- [Ostertag 04] E. Ostertag. – *Systèmes et asservissements continus : Modélisation, analyse, synthèse des lois de commande*. – Ellipses, Coll. Technosup, Mai 2004.
- [Pallett 93] E. H. J. Pallett, S. Coyle. – *Automatic Flight Control*. – Blackwell, Quatrième édition, 1993.
- [Pega] Site web du projet européen PEGASE. – <http://dassault.ddd.net/pegase/>.
- [Pflimlin 06] J. M. Pflimlin. – *Commande d'un minidrone à hélice carénée : de la stabilisation dans le vent à la navigation autonome*. – PhD. Thesis, Supaero, 2006.
- [Proctor 04] A. A. Proctor, E. N. Johnson. – Vision-only aircraft flight control methods and test results. – *AIAA Guidance, Navigation and Control Conference, number AIAA-2004-5351*, Providence, RI, États-Unis, Août 2004.

- [Quigley 05a] M. Quigley, B. Barber, S. Griffiths, M. A. Goodrich. – Towards real-world searching with fixed-wing mini-UAVs. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, Edmonton, AB, Canada, 2-6 Août 2005.
- [Quigley 05b] M. Quigley, M. A. Goodrich, S. Griffiths, A. Eldredge, R. W. Beard. – Target acquisition, localization, and surveillance using a fixed-wing, mini-uav and gimbaled camera. – *IEEE International Conference on Robotics and Automation, ICRA'05*, Barcelone, Espagne, Avril 2005.
- [Rafi 06] F. Rafi, S. M. Khan, K. Shafiq, M. Shah. – Autonomous target following by unmanned aerial vehicles. – *SPIE Defense and Security Symposium*, vol. 6230, Orlando, FL, États-Unis, Mai 2006.
- [Rives 02] P. Rives, J. R. Azinheira. – *Visual Auto-landing of an Autonomous Aircraft*. – Rapport de Recherche n 4606, INRIA-Sophia Antipolis, Novembre 2002.
- [Rives 04] P. Rives, J. R. Azinheira. – Linear structures following by an airship using vanishing point and horizon line in visual servoing scheme. – *IEEE International Conference on Robotics and Automation, ICRA'04*, pp. 255–260, Nouvelle Orléans, Avril 2004.
- [Rob] Site web du projet ROBVOLINT. – <http://www.irccyn.ec-nantes.fr/hebergement/Robvolint/>.
- [Rock 98] S. M. Rock, E. W. Frew, H. L. Jones, E. LeMaster, B. Woodley. – Combined CDGPS and vision-based control of a small autonomous helicopter. – *American Control Conference, ACC'98*, pp. 694–698, Philadelphie, PA, États-Unis, Juin 1998.
- [Romero 06] H. Romero, R. Benosman, R. Lozano. – Stabilization and location of a four rotor helicopter applying vision. – *American Control Conference, ACC'06*, pp. 3930–3936, Minneapolis, MN, États-Unis, Juin 2006.
- [Royer 04] E. Royer, M. Lhuiller, M. Dhome, T. Chateau. – Towards an alternative GPS sensor in dense urban environment from visual memory. – *British Machine Vision Conference*, Londres, Angleterre, 2004.
- [Ruffier 04] F. Ruffier, N. Franceschini. – Visually guided micro-aerial vehicle : automatic take off, terrain following, landing and wind reaction. – *IEEE International Conference on Robotics and Automation, ICRA'04*, vol. 3, pp. 2339–2346, États-Unis, Avril 2004.
- [Saripalli 03] S. Saripalli, J. F. Montgomery, G. S. Sukhatme. – Visually-guided landing of an unmanned aerial vehicle. *IEEE Transac-*

- tions on Robotics and Automation, ITRA'03*, 19(3) :371–381, Juin 2003.
- [Saripalli 07] S. Saripalli, G. S. Sukhatme. – Landing a helicopter on a moving target. – *IEEE International Conference on Robotics and Automation, ICRA '07*, pp. 2030–2035, Rome, Italie, Avril 2007.
- [Serres 06] J. Serres, F. Ruffier, S. Viollet, N. Franceschini. – Toward optic flow regulation for wall-following and centring behaviours. *International Journal of Advanced Robotic Systems*, 3(2) :147–154, Juin 2006.
- [Shakernia 99] O. Shakernia, Y. Ma, T. Koo, S. Sastry. – Landing an unmanned air vehicle : Vision based motion estimation and nonlinear control. *Asian Journal of Control*, 1(3) :128–145, Septembre 1999.
- [Sharp 01] C. S. Sharp, O. Shakernia, S. S Sastry. – A vision system for landing an unmanned aerial vehicle. – *IEEE International Conference on Robotics and Automation, ICRA '01*, pp. 1720–1727, Séoul, Corée, Mai 2001.
- [Silveira 02] G. F. Silveira, J. R. H. Carvalho, P. Rives, J. R. Azinheira, S. S. Bueno, M. K. Madrid. – Optimal visual servoed guidance of outdoor autonomous robotic airships. – *American Control Conference, ACC'02*, pp. 779–784, Anchorage, AK, États-Unis, Mai 2002.
- [Silveira 03] G. F. Silveira, J. R. Azinheira, P. Rives, S. S. Bueno. – Line following visual servoing for aerial robots combined with complementary sensors. – *International Conference on Advanced Robotics, ICAR'03*, pp. 1160–1165, Coimbra, Portugal, Juillet 2003.
- [Soni 94] T. Soni, B. Sridhar. – Modelling issues in vision based aircraft navigation during landing. – *Workshop on Application of Computer Vision, WACV'94*, pp. 89–96, Sarasota, FL, États-Unis, décembre 1994.
- [Srinivasan 96] M. V. Srinivasan, S.W. Zhang, M. Lehrer, T.S. Collett. – Honeybee navigation en route to the goal : Visual flight control and odometry. *The Journal of Experimental Biology*, pp. 237–243, 1996.
- [Stevens 92] B. L. Stevens, F. L. Lewis. – *Aircraft Control and Simulation*. – John Wiley & Sons, 1992.
- [Tahri 05] O. Tahri, F. Chaumette. – Point-based and region-based image moments for visual servoing of planar objects. *IEEE Transactions on Robotics*, 21(6) :1116–1127, Décembre 2005.
- [Tischler 96] M. B. Tischler. – *Advances In Aircraft Flight Control*. – Taylor & Francis, 1996.

- [Tisse 05] C.-L. Tisse, H. Durrant-Whyte. – Hemispherical eye sensor in micro aerial vehicles using advanced pinhole imaging system. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, pp. 77–83, Edmonton, AB, Canada, Août 2005.
- [Tournier 06] G. Tournier, M. Valenti, J. How, E. Feron. – Estimation and control of a quadrotor vehicle using monocular vision and moiré patterns. – *AIAA Conference on Guidance, Navigation and Control*, Keystone, CO, États-Unis, Août 2006.
- [Tsai 87] R. Y. Tsai. – A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4) :323–344, Août 1987.
- [Tutcar] Site web du tutorial de commande avec Matlab proposé par l'université de Carnegie Mellon : exemple de la commande du tangage d'un avion. –
<http://www.engin.umich.edu/group/ctm/>.
- [Tutnas] Site web du guide du *Glenn Educational Programs Office* de la NASA. –
<http://www.grc.nasa.gov/WWW/K-12/airplane/short.html>.
- [Valasek 02] J. Valasek, J. Kimmett, D. Hughes, J. L. Junkins. – Vision based sensor and navigation system for autonomous aerial refueling. – *AIAA Unmanned Aerospace Vehicles, Systems, Technologies, and Operations Conference, AIAA-2002-3441*, Portsmouth, VA, États-Unis, Mai 2002.
- [Visp] Site web de la bibliothèque ViSP. –
<http://www.irisa.fr/lagadic/visp/visp.html>.
- [Wang 05] L.K. Wang, S.-C. Hsieh, E.C.-W. Hsueh, Fei-Bin H., Kou-Yuan H. – Complete pose determination for low altitude unmanned aerial vehicle using stereo vision. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, pp. 108–113, Edmonton, AB, Canada, 2-6 Août 2005.
- [Weiss 87] L. Weiss, A. C. Sanderson, C. P. Neuman. – Dynamic sensor-based control of robots with visual feedback. *IEEE Journal on Robotics and Automation*, RA-3(5), Octobre 1987.
- [Wikafcs] Article de Wikipedia au sujet des systèmes de commande de vol. –
http://en.wikipedia.org/wiki/Aircraft_flight_control_systems.
- [Wikavion] Article de Wikipedia au sujet des avions. –
<http://fr.wikipedia.org/wiki/Avion>.
- [Wikhelico] Article de Wikipedia au sujet des hélicoptères. –
<http://fr.wikipedia.org/wiki/Helicoptere>.

- [Wiksimu] Article de Wikipedia au sujet des simulateurs de vol. – http://fr.wikipedia.org/wiki/Simulateur_de_vol.
- [Williams 01] M. Williams, D.I. Jones, G.K. Earp. – Obstacle avoidance during aerial inspection of power lines. *Aircraft Engineering and Aerospace Technology*, 73 :472–479, 2001.
- [Wilson 96] W. J. Wilson, C.C. Williams Hulls, G.S. Bell. – Relative end-effector control using cartesian position based visual servoing. *IEEE Transactions on Robotics and Automation*, 12 :684–696, Octobre 1996.
- [Wit] Site web du projet WITAS. – <http://www.ida.liu.se/ext/witas/>.
- [Wu 05] A. D. Wu, E. N. Johnson, A. A. Proctor. – Vision-aided inertial navigation. for flight control. – *AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, CA, États-Unis, Août 2005.
- [Yoshihata 07] Y. Yoshihata, K. Watanabe, Y. Iwatani, K. Hashimoto. – Visual control of a micro helicopter under dynamic occlusions. – *International Conference on Advanced Robotics, ICAR'07*, pp. 785–790, Jeju, Corée, Août 2007.
- [Zhang 99] H. Zhang, J. P. Ostrowski. – Visual servoing with dynamics : control of an unmanned blimp. – *IEEE International Conference on Robotics and Automation, ICRA'99*, vol. 1999, pp. 618–623, Detroit, MI, États-Unis, Mai 1999.
- [Zufferey 06] J. C. Zufferey, A. Klaptocz, A. Beyeler, J. D. Nicoud, D. Floreano. – A 10-gram microflyer for vision-based indoor navigation. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'06*, pp. 3267–3272, Pékin, Chine, Octobre 2006.
- [Zwaan 00] S. van der Zwaan, A. Bernardino, J. Santos-Victor. – Vision based station keeping and docking for an aerial blimp. – *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'00*, vol. 1, pp. 614–619, Takamatsu, Japon, Octobre 2000.

Table des figures

1.1	Modélisation de la caméra	18
1.2	Schéma du plan image de la caméra	20
1.3	Schéma-bloc de l’asservissement visuel	22
1.4	Illustration du problème d’avance-retrait.	27
1.5	L’hélicoptère du projet AVATAR	30
1.6	L’hélicoptère du CSIRO à Brisbane	30
1.7	L’hélicoptère du projet WITAS	31
1.8	Le drone HoverEye de Bertin Technologies	32
1.9	Hélicoptère à hélices coaxiales	32
1.10	Hélicoptère à hélices coaxiales	33
1.11	Le quadrirotor du projet ROBVOLINT	33
1.12	Le dirigeable du LAAS à Toulouse	35
1.13	Le dirigeable du projet NARVAL	35
1.14	Un petit avion du C3UV à Berkeley	36
1.15	Le petit avion du LAAS à Toulouse	37
1.16	Le petit avion de <i>Brigham Young University</i>	37
1.17	Le planeur de l’institut de technologie de Géorgie	38
1.18	Le micro-drone de l’EPFL à Lausanne	39
1.19	Schéma-bloc de la stratégie de commande avec deux boucles imbriquées	42
1.20	Schéma-bloc de la commande PID	44
1.21	Schéma-bloc de la commande LQR avec retour de sortie.	45
2.1	Schéma de commande utilisé pour l’asservissement visuel de la position d’un drone	48
2.2	Schéma de fonctionnement d’un quadrirotor	49
2.3	Images initiale et finale correspondant aux simulations	50
2.4	Repères liés à la cible (\mathcal{F}_o), et à la caméra, à la position initiale (\mathcal{F}_c) et à la position finale (\mathcal{F}_c^*)	51
2.5	Résultats obtenus pour l’asservissement des rotations	51
2.6	Résultats obtenus avec la commande (2.14)	55
2.7	Résultats obtenus avec la commande (2.18)	56
2.8	Projections perspective et sphérique d’un point	58

2.9	Schéma de la projection d'une cible sur la sphère virtuelle, et du centre de gravité de l'image sphérique selon la distance cible-caméra	59
2.10	Résultats obtenus avec la commande proportionnelle	63
2.11	Résultats obtenus avec la commande classique	64
2.12	Résultats obtenus avec la commande reposant sur la linéarisation à la position désirée	65
2.13	Schéma représentant δ_{11} et δ_{12}	65
2.14	Commande partitionnée	67
2.15	Schéma représentant δ_{12}	68
2.16	Schéma de la relation entre la profondeur Z et l'information visuelle \mathbf{q} .	69
2.17	Courbe de la relation entre la profondeur Z et l'information visuelle $ \mathbf{q} $.	70
2.18	Schéma de la projection sphérique : variations de \mathbf{q} selon l'axe \mathbf{z}_c	71
2.19	Résultats obtenus avec la commande stable utilisant δ_f	73
2.20	Résultats obtenus avec la commande classique avec approximation de la matrice d'interaction	74
2.21	Résultats obtenus avec la commande proportionnelle avec δ_f	75
2.22	Courbes représentant les facteurs d'échelle $F(\mathbf{q})$ et $G(\mathbf{q})$	77
2.23	Résultats obtenus avec la commande stable basée sur δ_g	78
2.24	Implémentation de la loi de commande (2.75).	79
2.25	Le quadrirotor utilisé pour les expérimentations.	83
2.26	Caméra et cible utilisées	83
2.27	Résultats expérimentaux sur le quadrirotor : commande reposant sur les moments de l'image perspective	86
2.28	Résultats expérimentaux sur le quadrirotor : commande reposant sur les informations visuelles sphériques et la linéarisation à la position désirée .	88
2.29	Résultats expérimentaux sur le quadrirotor : commande reposant sur les informations visuelles sphériques et une loi de commande partitionnée .	89
2.30	Résultats expérimentaux sur le quadrirotor : commande proportionnelle utilisant les informations visuelles sphériques avec changement d'échelle approprié	90
2.31	Résultats expérimentaux sur le quadrirotor : commande stable utilisant les informations visuelles sphériques avec changement d'échelle approprié	92
3.1	Coordonnées géographiques et repère géocentrique.	97
3.2	Repère tangent à la surface de la Terre.	97
3.3	Coordonnées dans le repère fixe lié à la piste d'atterrissage.	98
3.4	Angles de roulis, tangage, lacet.	99
3.5	Repère lié à l'avion.	99
3.6	Incidence et dérapage.	100
3.7	Différents repères utilisés.	101
3.8	Repère cinématique \mathcal{F}_v	102
3.9	Forces agissant sur l'avion.	103
3.10	Dispositifs de commande d'un avion : gouvernes et moteurs.	104
3.11	Différents types de commande pilote.	104

3.12	Atterrissage vu d'en haut.	108
3.13	Phases d'atterrissage.	108
3.14	Phase d'arrondi.	110
3.15	Volets sortis pour l'atterrissage.	112
3.16	Décomposition du vecteur vitesse selon l'axe y_f	117
3.17	Paramètres de l'état lors du mouvement latéral de l'avion.	118
3.18	Réponse du simulateur à la commande de gauchissement.	119
3.19	Schéma bloc du modèle latéral.	119
3.20	Paramètres de l'état lors du mouvement longitudinal de l'avion.	121
3.21	Positionnement de la caméra sur l'avion.	121
3.22	Positionnement de la caméra par rapport à la piste, et projection des lignes de la piste.	124
3.23	Projection des lignes de la piste sur le plan image, à la position de désirée.	126
3.24	Projection des lignes de la piste sur le plan image : cas particulier où $(\theta_l, \theta_r) \rightarrow (k_1\pi, k_2\pi)$	130
3.25	Alignement de l'avion dans l'axe de la piste au cours de l'atterrissage. . .	132
3.26	Schéma de commande latéral.	135
3.27	Schéma de commande latérale par asservissement visuel.	136
3.28	Position initiale de l'avion par rapport à l'axe de la piste.	140
3.29	Mouvements longitudinaux de l'avion pendant l'atterrissage.	141
3.30	Position initiale de l'avion par rapport à la trajectoire d'approche. . . .	142
3.31	Schéma du simulateur utilisé.	146
3.32	Résultats de commande longitudinale avec mesures directes.	153
3.33	Résultats de commande longitudinale avec mesures visuelles.	154
3.34	Résultats de commande longitudinale avec mesures visuelles.	155
3.35	Vues de la caméra embarquée, aux étapes-clef de l'atterrissage.	156
3.36	Résultats de la commande longitudinale avec mesures visuelles et trajec- toire.	157
3.37	Résultats de la commande longitudinale avec mesures visuelles et trajec- toire.	158
3.38	Résultats de commande longitudinale avec mesures visuelles et trajectoire.	159
3.39	Résultats de commande longitudinale avec mesures visuelles et trajectoire.	160
3.40	Résultats de commande longitudinale avec mesures visuelles partielles et trajectoire.	161
3.41	Résultats de commande longitudinale avec mesures visuelles partielles et trajectoire.	162
3.42	Résultats avec la commande latérale (3.171) avec mesures directes. . . .	164
3.43	Résultats avec la commande latérale (3.172) avec mesures directes. . . .	165
3.44	Résultats avec la commande latérale (3.173) avec mesures visuelles. . . .	167
3.45	Résultats avec la commande latérale (3.173) avec mesures visuelles. . . .	170
3.46	Vues de la caméra embarquée, aux étapes-clef de l'atterrissage, lorsque la consigne est fixe.	171
3.47	Résultats avec la commande latérale (3.174) avec mesures visuelles. . . .	172
3.48	Résultats avec la commande latérale (3.174) avec mesures visuelles. . . .	173

3.49	Résultats avec la commande latérale (3.165) avec mesures visuelles et trajectoire.	174
3.50	Résultats avec la commande latérale (3.165) avec mesures visuelles et trajectoire.	175
3.51	Vues de la caméra embarquée, aux étapes-clef de l'atterrissage, lorsque la consigne est spécifiée sous forme de trajectoire.	176
3.52	Résultats avec la commande latérale (3.165) avec mesures visuelles et trajectoire.	177
3.53	Résultats avec la commande latérale (3.165) avec mesures visuelles et trajectoire.	178
3.54	Vues de la caméra embarquée, aux étapes-clef de l'atterrissage, lorsque la consigne est spécifiée sous forme de trajectoire.	179
3.55	Résultats avec la commande latérale (3.175) avec mesures visuelles partielles et trajectoire.	180
3.56	Résultats avec la commande latérale (3.175) avec mesures visuelles partielle et trajectoire.	181
3.57	Résultats avec les commandes (3.165) et (3.170).	185
3.58	Résultats avec les commandes (3.165) et (3.170).	186
3.59	Résultats avec les commandes (3.165) et (3.170).	187
3.60	Résultats avec les commandes (3.165) et (3.170).	188
3.61	Résultats avec les commandes (3.165) et (3.170).	189
3.62	Résultats avec les commandes (3.165) et (3.170).	190
3.63	Résultats avec les commandes (3.165) et (3.170).	191
3.64	Résultats avec les commandes (3.165) et (3.170).	192
3.65	Résultats avec les commandes (3.165) et (3.170), avec la caméra déplacée par rapport au centre de gravité de l'avion.	193
3.66	Résultats avec les commandes (3.165) et (3.170), avec la caméra déplacée par rapport au centre de gravité de l'avion.	194
3.67	Résultats avec les commandes (3.165) et (3.170), avec la caméra déplacée par rapport au centre de gravité de l'avion.	195
3.68	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise connaissance du modèle de la piste.	196
3.69	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise connaissance du modèle de la piste.	197
3.70	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise connaissance du modèle de la piste.	198
3.71	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra.	199
3.72	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra.	200
3.73	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra.	201
3.74	Résultats avec les commandes (3.165) et (3.170), et avec une mauvaise calibration de la caméra.	202

3.75	Résultats avec les commandes (3.165) et (3.170), et avec du vent de travers.	204
3.76	Alignement de l'avion dans l'axe de la piste au cours de l'atterrissage, en présence de vent de travers.	205
3.77	Alignement de l'avion avec la piste, avec du vent de travers et la loi de commande (3.171)	206
B.1	Repères successifs considérés pour passer du repère géocentrique \mathcal{F}_0 au repère \mathcal{F}_f , tangent à la surface de la Terre, orienté selon l'axe de la piste.	225
C.1	Orientation du repère \mathcal{F}_a par rapport au repère \mathcal{F}_f	227
C.2	Angles d'Euler (z, y, x)	228
C.3	Angles de roulis, tangage, lacet.	229
E.1	Schéma de commande latérale.	233
E.2	Position des pôles.	236

Résumé

L'asservissement visuel est une technique de commande reposant sur des mesures issues d'une caméra. Cette technique est utilisée en robotique pour commander divers systèmes dynamiques, du robot manipulateur à l'humanoïde, en passant par les robots médicaux, les voitures autonomes, les robots sous marins et les engins volants. Dans cette thèse, nous nous intéressons à la conception d'asservissements visuels pour deux types d'engins volants : un hélicoptère de type quadrirotor et un avion. Les schémas de commande proposés s'interfacent sur les boucles de commande existantes assurant la stabilisation bas-niveau des engins. Nous essayons autant que possible de spécifier le problème en terme de régulation dans l'image, en nous affranchissant d'une étape de reconstruction 3D. Pour cela, nous proposons des lois de commande d'asservissement visuel adaptées à la fois à la dynamique des engins considérés et reposant sur des informations visuelles issues de l'image de la scène observée. Afin de positionner le quadrirotor par rapport à une cible visuelle, nous proposons une série de lois de commande reposant sur des informations visuelles ayant des propriétés de découplage ou de passivité, issues de la projection perspective ou de la projection sphérique de la cible. Après en avoir étudié les performances, nous avons expérimenté avec succès les meilleures de ces lois de commande. Concernant l'atterrissage de l'avion, nous utilisons des informations visuelles issues des paramètres des lignes de la piste d'atterrissage dans le plan image. Nous proposons également une méthode de planification de la trajectoire prenant en compte le comportement de l'avion. Des lois de commande reposant sur un modèle linéarisé et découplé de l'avion intègrent les informations visuelles et les trajectoires. Les résultats obtenus sur un simulateur réaliste d'avion montrent que les schémas de commande proposés peuvent permettre à l'avion d'atterrir automatiquement.

Mots-clefs : Robotique, asservissement visuel, engins volants.

Abstract

Visual servoing is a control approach based on visual data. It can be used for controlling a large variety of robots, such as robot arms, humanoids, medical robots, car-like vehicles, underwater and aerial robots. In this thesis, visual servoing schemes are proposed to control a quadrotor during a positioning task, and to control an airplane during landing. A high level control is considered: for the quadrotor the proposed control laws are coupled with a high-gain inner control loop that regulates translational velocities and full attitude dynamics, and a fly-by-wire system is included in the considered model of the airplane. We choose not to deal with pose estimation, but to consider 2D visual servoing. Concerning the control of the quadrotor, several control schemes based on perspective and spherical image moments are designed. Their properties are presented and their performances are analysed and compared. The most promising control algorithms are successfully implemented on a quadrotor showing excellent performances. Concerning the control of the airplane, we use decoupled visual features and design control laws by taking into account a linearized and decoupled model of the airplane dynamics. A desired trajectory is also designed to improve the airplane behaviour. Simulation results are obtained with a quite realistic flight simulator. They show that the airplane is able to land automatically by using the proposed control scheme.

Keywords: Robotics, visual servoing, aircrafts.