

Titre: Contrôle d'attitude d'un lanceur en phase atmosphérique approche par applications gardiennes
Title: Contrôle d'attitude d'un lanceur en phase atmosphérique approche par applications gardiennes

Auteur: Vincent Dubanchet
Author:

Date: 2012

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Dubanchet, V. (2012). Contrôle d'attitude d'un lanceur en phase atmosphérique approche par applications gardiennes [Master's thesis, École Polytechnique de Montréal]. PolyPublie. <https://publications.polymtl.ca/931/>
Citation:

Document en libre accès dans PolyPublie Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/931/>
PolyPublie URL:

Directeurs de recherche: Richard Gourdeau, & Lahcen Saydy
Advisors:

Programme: Génie Électrique
Program:

UNIVERSITÉ DE MONTRÉAL

CONTRÔLE D'ATTITUDE D'UN LANCEUR EN PHASE ATMOSPHÉRIQUE
APPROCHE PAR APPLICATIONS GARDIENNES

VINCENT DUBANCHET
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE)
AOÛT 2012

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

CONTRÔLE D'ATTITUDE D'UN LANCEUR EN PHASE ATMOSPHÉRIQUE
APPROCHE PAR APPLICATIONS GARDIENNES

présenté par : DUBANCHET Vincent
en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées
a été dûment accepté par le jury d'examen constitué de :

M. SAUSSIÉ David, Ph.D., président

M. GOURDEAU Richard, Ph.D., membre et directeur de recherche

M. SAYDY Lahcen, Ph.D., membre et codirecteur de recherche

Mme BÉRARD Caroline, Ph.D., membre et codirectrice de recherche

M. DESBIENS André, Ph.D., membre

*À mon frère,
À mes parents,*

...

REMERCIEMENTS

Je tiens à remercier en premier lieu mes directeurs de recherche, M. Richard Gourdeau, M. Lahcen Saydy et Mme Caroline Bérard, pour leur aide et leur soutien permanents. Je leur suis extrêmement reconnaissant d'avoir géré les contraintes administratives, financières et humaines de cette collaboration entre deux écoles, pour me permettre de m'épanouir pleinement dans ma Maîtrise. À travers les cours qu'ils m'ont enseignés, ils m'ont passionné pour le domaine de l'automatique et m'ont poussé à m'engager dans ce projet. Grâce à leur expérience et leur générosité, ils ont su me motiver et me guider quand la recherche s'essoufflait. Ils ont fait de cette Maîtrise une expérience marquante dans mon parcours académique, tant d'un point de vue humain que professionnel, grâce aux conditions de travail qu'ils m'ont offertes et aux charges de laboratoire qu'ils m'ont confiées.

Je tiens aussi à remercier tout particulièrement M. David Saussié pour son aide permanente, son encadrement et sa motivation sans faille tout au long de ma Maîtrise. Il est pour beaucoup dans la publication des deux articles de conférence associés à ce projet, et il m'a donné le goût de la recherche, par sa passion, son engagement et son enthousiasme.

Je remercie ensuite tout le personnel administratif des deux écoles, qui ont permis le bon déroulement de cet échange. Je pense en particulier à Mme Françoise Loytier, Mme Suzanne Lebel et Mme Nathalie Lévesque, pour leur professionnalisme. De même, je remercie les techniciens du laboratoire d'automation, M. Richard Grenier et Mme Marie-Lyne Brisson pour toute leur aide, leur disponibilité et leur sympathie.

Je terminerai en remerciant mes collègues de laboratoire, avec qui j'ai pris énormément de plaisir à travailler : Amin, Anas, Giuseppe, Lamine, Marcos et Reza. Je tiens aussi à mentionner mes amis, qui m'ont supporté, encouragé et soutenu pendant ces deux ans : Benjamin, Cyrill, François, Laure, Mathilda, Sabrina, Thomas, Tiphaïne et Valentin.

Ma dernière pensée va à ma famille. Elle m'a toujours inspiré, encouragé et motivé. Tout ce que j'ai pu réaliser jusqu'ici n'est que le résultat de leur soutien et de tous leurs sacrifices. Je leur dédie ce mémoire, l'aboutissement de mon travail et le fruit de leur dévouement.

RÉSUMÉ

Dans un premier temps, la modélisation d'un lanceur spatial met en avant l'importance des variations des paramètres au cours de l'ascension. Ils sont en effet fortement dépendants de la variation de la masse avec la consommation des ergols. Le modèle utilisé prend ainsi en compte cette évolution en linéarisant les équations autour de 6 points de vol principaux, pour obtenir des modèles linéaires invariants dans le temps. Chacun de ces modèles devra être stabilisé par une unique loi de commande, tout en respectant les performances désirées. La considération des modes de flexion du lanceur rend alors la synthèse plus complexe.

Une autre conséquence de l'évolution temporelle est le séquencement des gains des contrôleurs obtenus. Il est en effet montré dans le mémoire qu'aucun réglage constant du correcteur adopté ne permet de respecter le cahier des charges sur toute la trajectoire.

Ensuite, la complexité de modélisation du lanceur complet amène à considérer les erreurs et les incertitudes de modèle. Elles sont là encore un enjeu majeur du projet puisqu'il faut pouvoir assurer les performances nominales précédentes de façon robuste. Les applications gardiennes, de par leur propriétés, se sont révélées les plus adaptées pour traiter un tel problème de séquencement avec respect de performances robustes.

Dans cette optique, un aspect essentiel de la Maîtrise porte sur le développement de méthodes de synthèse basées sur les applications gardiennes. Il est en effet apparu que les quelques travaux portant sur ce sujet développaient essentiellement la théorie, et que la mise en œuvre de ces méthodes pouvait être améliorée.

Deux approches sont ainsi développées à travers le projet. La première s'appuie sur une visualisation graphique des lieux d'annulation des applications gardiennes. Un programme basé sur l'analyse d'image permet de vérifier automatiquement les réglages des gains qui satisfont les contraintes. La seconde méthode s'appuie sur une optimisation portant sur les applications gardiennes dans des zones d'intérêt. Par itérations successives, il est alors possible de partir du système en boucle ouverte pour aboutir à un réglage des gains acceptable en boucle fermée.

Les méthodes élaborées ont été testées et éprouvées sur le cas du lanceur, avec le cahier des charges fourni par ASTRIUM-ST. Cette application pratique a été motivée par la complexité du système, les contraintes variées à considérer et l'importance de la robustesse pour ce type de commande. Autant de contraintes qui ont permis de valider l'intérêt et l'efficacité des applications gardiennes sur ce type de problème.

ABSTRACT

In a first phase, the modelling process underlines the presence of highly time varying parameters during the ascent, due to a fast mass variation along with propellant consumption. Linearizing the dynamical equations at six main flight instants yields linear time invariant models to be considered during control design. Each of them is to be stabilized by one control law, while respecting given specifications. The synthesis becomes even more complex when the bending modes are taken into account.

Moreover, scheduling appears necessary to deal with the time variations. Indeed it is shown that no single gain setting is able to respect all the specifications along the trajectory.

Furthermore, increasing complexity when modelling a whole launch vehicle pushes one to consider the model's errors and uncertainties. They represent a major issue in this study since it is asked to ensure the nominal performances in a robust fashion. Owing to their properties, guardian maps appear to be the most suitable tool to deal with such a problem of scheduling with robust performances.

In light of this, the development of synthesis methods based on guardian maps is the main contribution of the project. It appears that actual state of the art in this field is focused on theoretical issues, whereas practical ones could be improved.

Two approaches are presented in the memoire. The first one is based on a graphical approach consisting in drawing the vanishing locus of guardian maps. A program using image analysis techniques is devised to check automatically which gain settings satisfy the constraints. The second one is based on an optimisation procedure involving guardian maps. Starting with the open loop system, the iterative process proposed ends up with a satisfactory gain setting for the closed-loop.

These methods are tried and tested for the launch vehicle, with specifications from ASTRIUM-ST. Their practical application is motivated by the system complexity, the different kinds of constraints and the essential need for robustness. Many restrictions that finally bring about the interest and the efficiency of guardian maps for such a problem.

TABLE DES MATIÈRES

DÉDICACE	iii
REMERCIEMENTS	iv
RÉSUMÉ	v
ABSTRACT	vi
TABLE DES MATIÈRES	vii
LISTE DES TABLEAUX	x
LISTE DES FIGURES	xi
LISTE DES ANNEXES	xiv
LISTE DES SIGLES ET ABRÉVIATIONS	xv
CHAPITRE 1 INTRODUCTION	1
1.1 Mise en contexte	1
1.2 Objectifs de recherche	3
1.3 Plan du mémoire	3
CHAPITRE 2 REVUE DE LITTÉRATURE	5
CHAPITRE 3 MODELISATION DU LANCEUR	9
3.1 Introduction	9
3.2 Schéma du lanceur	10
3.3 Conditions de vol	10
3.4 Modélisation rigide	12
3.4.1 Principe Fondamental de la Dynamique	12
3.5 Modélisation flexible	16
3.5.1 Pulsations et modes propres	16
3.5.2 Modèle du lanceur souple	19
3.5.3 Corrections apportées par les modes flexibles	20
3.6 Modélisation de l'actionneur	21

3.6.1	Aperçu du modèle fourni	21
3.6.2	Actionneur électromécanique à retour d'effort	21
3.7	Evolution temporelle et modélisation des incertitudes	26
3.7.1	Evolutions temporelles des réponses fréquentielles	26
3.7.2	Représentation des incertitudes	29
3.8	Synthèse robuste et respect de performances	33
3.8.1	Structure de correction adoptée	33
3.8.2	Cahier des charges	34
3.9	Simulation et validation	35
CHAPITRE 4 APPLICATIONS GARDIENNES		38
4.1	Introduction	38
4.2	Outils Mathématiques	39
4.2.1	Produit de Kronecker	39
4.2.2	Produit Bialterné	40
4.3	Définitions et applications gardiennes usuelles	42
4.3.1	Définitions	42
4.3.2	Applications gardiennes usuelles	43
4.3.3	Construction d'une application (semi-)gardiennne	45
4.4	Principaux résultats	49
4.4.1	Cas général	49
4.4.2	Cas de familles monoparamétriques	50
4.5	Exemple d'analyse robuste	51
4.6	Exemple de synthèse	53
4.7	Application au cas du lanceur	55
4.7.1	Modèle et structure de correction	55
4.7.2	Applications gardiennes considérées	56
4.8	Visualisation des composantes	57
CHAPITRE 5 SYNTHÈSE PAR ANALYSE GRAPHIQUE		60
5.1	Introduction	60
5.2	Analyse d'image	61
5.2.1	Calcul des applications gardiennes	62
5.2.2	Calcul du lieu d'annulation	63
5.2.3	Construction d'une image Noir & Blanc	64
5.2.4	Pré-traitement de l'image	65
5.2.5	Détection des composantes dans l'image	66

5.2.6	Choix d'un point intérieur	67
5.2.7	Analyse de la Ω -stabilité	69
5.3	Séquencement des gains	70
5.4	Application au cas du lanceur	72
5.4.1	Cas du lanceur rigide	73
5.4.2	Cas du lanceur flexible	80
CHAPITRE 6 SYNTHÈSE PAR OPTIMISATION		86
6.1	Introduction	86
6.2	Rappels en optimisation	87
6.2.1	Algorithme général	87
6.2.2	Recherche linéaire	88
6.2.3	Méthodes de Newton	89
6.2.4	Méthodes de Quasi-Newton	89
6.2.5	Recherche linéaire modifiée	90
6.3	Algorithme de synthèse par applications gardiennes	91
6.3.1	Algorithme complet	92
6.3.2	Calcul des gradients	98
6.3.3	Prise en compte des applications gardiennes	100
6.4	Application au cas du lanceur	105
6.4.1	Synthèse sur le lanceur flexible	106
6.4.2	Problèmes rencontrés	111
CHAPITRE 7 CONCLUSION		113
RÉFÉRENCES		117
ANNEXES		121

LISTE DES TABLEAUX

Tableau 3.1	Valeurs des gains identifiées avec le modèle de l'actionneur	25
Tableau 4.1	Valeurs des points de test et valeurs propres associées	52
Tableau 5.1	Valeurs séquencées des gains pour le correcteur rigide PID	77
Tableau 5.2	Valeurs séquencées des gains du correcteur PIDFRo	83
Tableau 6.1	Algorithme de Synthèse par Optimisation – Phase de Stabilisation .	96
Tableau 6.2	Algorithme de Synthèse par Optimisation – Phase d'Optimisation .	97
Tableau 6.3	Valeurs séquencées des gains du correcteur PIDFRo	109

LISTE DES FIGURES

Figure 2.1	Schéma du système d'orientation de la poussée, tiré de Maillebouis (1988)	6
Figure 3.1	Schéma du lanceur complet	10
Figure 3.2	Evolution temporelle des paramètres	11
Figure 3.3	Profil de vent utilisé en simulation	12
Figure 3.4	Influence du vent sur l'incidence	13
Figure 3.5	Modélisation adoptée pour représenter les mouvements de la tuyère	14
Figure 3.6	Modélisation discrète d'un lanceur avec prise en compte des ergols et de la charge utile, tirée de Marty (1986)	17
Figure 3.7	Déformées latérales pour les deux premiers modes de flexion, tirées de Marty (1986)	18
Figure 3.8	Lieu des pôles et zéros de la fonction de transfert actionneur	21
Figure 3.9	Schéma cinématique de l'actionneur EMA tiré de Renault (2005)	22
Figure 3.10	Boucle de contrôle avec retour en position et en effort	24
Figure 3.11	Lieu des racines du système complet	26
Figure 3.12	Evolution du lieu des racines avec le temps	27
Figure 3.13	Evolution du diagramme de Bode avec le temps - Modèle rigide	28
Figure 3.14	Evolution du diagramme de Bode avec le temps - Modèle flexible	28
Figure 3.15	Evolution de la poussée et des masses avec les modèles incertains	29
Figure 3.16	Evolution des bras de levier avec les modèles incertains	30
Figure 3.17	Evolution des paramètres aérodynamiques avec les modèles incertains	31
Figure 3.18	Evolution du lieu des racines avec les modèles incertains	32
Figure 3.19	Evolution du diagramme de Bode avec les modèles incertains	32
Figure 3.20	Structure de correction du lanceur flexible avec son actionneur	33
Figure 3.21	Exemple de simulation à une rafale de vent	36
Figure 3.22	Exemple de marges obtenues pour le système corrigé	37
Figure 4.1	Zone de confinement pour la stabilité Hurwitz et la α -stabilité	44
Figure 4.2	Zones de confinement pour la stabilité Schur et la ξ -stabilité	45
Figure 4.3	Frontière polynomiale de l'ensemble Ω	46
Figure 4.4	Décomposition d'un ensemble Ω en sous-ensembles basiques	48
Figure 4.5	Choix des points à l'intérieur de chaque composante	51
Figure 4.6	Visualisation des composantes stables au sens de Hurwitz	52
Figure 4.7	Moteur électrique couplé avec un pendule	53

Figure 4.8	Composantes de l'application gardienne et réponses temporelles pour le moteur+pendule	54
Figure 4.9	Structure de contrôle du lanceur avec la mesure sur θ	55
Figure 4.10	Zones de confinement pour la stabilité des pôles rigides et flexibles . . .	56
Figure 4.11	Lieux d'annulation des applications gardiennes ν_α , ν_β et ν_ξ	57
Figure 4.12	Zone de stabilité commune à ν_α , ν_ζ et ν_β dans l'espace des paramètres .	58
Figure 4.13	Zone de stabilité de ν_{hf} et ν_{lf}	59
Figure 4.14	Zone de stabilité commune à ν_{nom} , ν_{hf} et ν_{lf}	59
Figure 5.1	Construction d'une courbe de niveau par la fonction <code>contour</code> au niveau d'une maille	64
Figure 5.2	Construction d'une image Noir & Blanc représentant un contour	65
Figure 5.3	Image Noir & Blanc représentant les composantes d'une application gardienne	65
Figure 5.4	Image Noir & Blanc représentant les composantes après le pré-traitement	66
Figure 5.5	Composantes identifiées à partir du tracé du lieu d'annulation	66
Figure 5.6	Construction d'un ensemble convexe par parcours de la frontière	68
Figure 5.7	Construction d'un ensemble convexe inclus dans la composante	68
Figure 5.8	Construction de segments inclus dans la composante	69
Figure 5.9	Lieu des racines en boucle fermée pour une correction d'ordre 5	70
Figure 5.10	Composante stable commune aux trois modèles <code>nom</code> , <code>hf</code> et <code>lf</code>	71
Figure 5.11	Enveloppe de stabilité pour le choix du gain K_1 sur le cas du lanceur .	71
Figure 5.12	Enveloppe de stabilité pour les gains K_1 et K_2 sur le cas du lanceur .	72
Figure 5.13	Structure de la boucle de contrôle	74
Figure 5.14	Légendes utilisées dans la Fig. 5.16	75
Figure 5.15	Enveloppe de stabilité des gains K_1 et K_2 pour le lanceur rigide	75
Figure 5.16	Intersections des composantes stables sur les trois modèles pour chaque temps	76
Figure 5.17	Simulations temporelles pour le lanceur rigide	78
Figure 5.18	Evolution des marges au cours du vol pour le lanceur rigide	79
Figure 5.19	Lieu des racines du lanceur flexible avec correcteur PID	80
Figure 5.20	Lieu des racines du filtre elliptique complet	81
Figure 5.21	Effet du filtre d'ordre 2 et du roll-off sur le lieu des racines	82
Figure 5.22	Enveloppe de stabilité des gains K_1 et K_2 pour le lanceur flexible . . .	83
Figure 5.23	Simulations temporelles pour le lanceur flexible	84
Figure 5.24	Evolution des marges au cours du vol pour le lanceur flexible	85
Figure 6.1	Illustration du rôle de la condition d'Armijo	88

Figure 6.2	Illustration du rôle des conditions de Wolfe	91
Figure 6.3	Chemin suivi par l'algorithme de stabilisation	93
Figure 6.4	Evolution des pôles le long du chemin d'optimisation	93
Figure 6.5	Chemin suivi par l'algorithme de stabilisation	95
Figure 6.6	Evolution des pôles le long du chemin d'optimisation	95
Figure 6.7	Valeurs d'une application gardienne à travers une composante	101
Figure 6.8	Recherche d'un point satisfaisant les conditions de Wolfe	103
Figure 6.9	Valeurs d'une application gardienne au niveau d'une intersection	104
Figure 6.10	Construction d'une direction de descente pointant à l'intérieur de la composante	105
Figure 6.11	Influence des modes flexibles sur l'intersection des composantes stables	106
Figure 6.12	Chemin d'optimisation suivi pour respecter les performances	107
Figure 6.13	Intervalles de stabilité obtenus sur les réglages issus de l'optimisation .	108
Figure 6.14	Séquencement des gains à partir des réglages obtenus par optimisation	109
Figure 6.15	Simulations temporelles pour le lanceur flexible	110
Figure 6.16	Evolution des marges au cours du vol pour le lanceur flexible	111
Figure 6.17	Changements d'échelle sur les valeurs de l'application gardienne.	112
Figure 7.1	Premier vol du lanceur européen VEGA, le 13 Février 2012	116
Figure A.1	Intersections des composantes stables sur les trois modèles pour chaque temps	
	Contrainte en partie réelle sur les pôles rigides . . .	121
Figure B.1	Intersections des composantes stables sur les trois modèles pour chaque temps	
	Contrainte en amortissement sur les pôles rigides . .	122
Figure C.1	Intersections des composantes stables sur les trois modèles pour chaque temps	
	Contrainte en stabilité sur les pôles flexibles	123
Figure D.1	Intersections des composantes stables sur les trois modèles pour chaque temps	124

LISTE DES ANNEXES

Annexe A	Composantes des applications gardiennes ν_α	121
Annexe B	Composantes des applications gardiennes ν_β	122
Annexe C	Composantes des applications gardiennes ν_ξ	123
Annexe D	Superposition des composantes des applications gardiennes ν_α , ν_β et ν_ξ	124

LISTE DES SIGLES ET ABRÉVIATIONS

– Modélisation du lanceur

- **Repères :**

- (GX, GY) repère inertiel lié à la surface terrestre
- (Gx, Gy) repère local dans les axes du lanceur

- **Points caractéristiques :**

- C foyer aérodynamique, point d'application des forces aérodynamiques
- G centre de gravité du lanceur complet
- G_n centre de gravité de la tuyère (nozzle)
- N point d'application de la poussée, centre de rotation de la tuyère

- **Grandeurs et bras de levier :**

- h hauteur du lanceur
- d diamètre du lanceur
- M masse du lanceur
- m masse de la tuyère
- I moment d'inertie du lanceur complet par rapport à G
- I_n moment d'inertie de la tuyère par rapport à G_n
- $I_{n,N}$ moment d'inertie de la tuyère par rapport à N
- Q pression dynamique ($Q = \frac{1}{2}\rho V^2$)
- S_{ref} surface de référence des efforts aérodynamiques (surface en coupe du lanceur)
- l_{GC} bras de levier des forces aérodynamiques
- l_{GN} bras de levier de la poussée
- l_n distance entre G_n et le centre de rotation N de la tuyère
- V vitesse du lanceur
- W vitesse du vent latéral

- **Forces appliquées au lanceur :**

- Cz_α dérivée de stabilité du coefficient de portance
- Cx coefficient de traînée
- L portance générée par le lanceur
- D traînée générée par le lanceur
- T poussée du lanceur

- **Angles caractéristiques :**
 - α incidence du lanceur
 - γ pente du vecteur vitesse
 - θ assiette du lanceur par rapport à la verticale
 - β déflexion de la poussée
- **Moments appliquées au lanceur, ramenés en G :**
 - $\mathcal{M}_{G,L}$ moment engendré par la portance
 - $\mathcal{M}_{G,D}$ moment engendré par la traînée
 - $\mathcal{M}_{G,T}$ moment engendré par la poussée
 - $\mathcal{M}_{G,N}$ moment engendré par l'inertie de la tuyère en mouvement
- **Modèles d'état**
 - (A_r, B_r, C_r, D_r) modèle d'état du lanceur rigide
 - (A_f, B_f) dynamique des modes flexibles du lanceur
 - $(\tilde{A}_F, \tilde{B}_F, \tilde{C}_F, \tilde{D}_F)$ modèle d'état du lanceur flexible
 - (A_F, B_F, C_F, D_F) modèle d'état du lanceur flexible avec actionneur
- Applications gardiennes
- **Notations matricielles**
 - $\mathbb{R}^{m \times p}$ matrices réelles de taille $m \times p$
 - \mathcal{M}_n matrices carrées de taille n
 - p_r pôles de la partie rigide du lanceur
 - p_f pôles de la partie flexible du lanceur
 - $Re(p)$ partie réelle du pôle p
 - $\xi(p)$ amortissement du pôle p
 - $\lambda_i(A)$ i^{eme} valeur propre de la matrice A
 - $\sigma(A)$ spectre de la matrice A
 - A_{BF} matrice en boucle fermée
 - K_p gain proportionnel
 - K_d gain dérivé
 - K_i gain intégral

- **Applications gardiennes**

Ω sous-ensemble ouvert du plan complexe, symétrique par rapport à l'axe réel

$\overline{\Omega}$ adhérence de l'ensemble Ω

$\partial\Omega$ frontière de l'ensemble Ω

$\overset{\circ}{\Omega}$ intérieur de l'ensemble Ω

ν_α application gardienne liée à la partie réelle des pôles rigides

ν_ξ application gardienne liée à l'amortissement des pôles rigides

ν_β application gardienne liée à la partie réelle des pôles flexibles

CHAPITRE 1

INTRODUCTION

1.1 Mise en contexte

Après une large domination du domaine spatial par les puissances américaines et russes depuis la guerre froide, de nombreux pays ont suivi leurs traces pour s'imposer aujourd’hui sur la scène internationale. Pour ne citer que les plus grandes agences actuelles, l'espace est aujourd’hui à la portée de la France, de l'Allemagne, du Japon, de l'Inde et bien sûr de la Chine (Rahman *et al.*, 2010). Cette dernière est même devenue la troisième puissance à envoyer des hommes dans l'espace avec son vaisseau Shenzhou en 2003.

Néanmoins, la majorité des lancements actuels sont destinés aux satellites, que ce soit pour le domaine militaire, scientifique ou commercial. Un véritable marché de lanceurs s'ouvre donc aux industries désirant mettre un satellite en orbite. Dans cette optique, il est primordial d'offrir une fiabilité excellente pour espérer être compétitif, tout en proposant une large gamme de lanceurs suivant le poids des charges utiles à envoyer. On peut citer le développement du lanceur VEGA de l'agence spatiale européenne ESA¹, qui vient compléter la gamme des fusées Ariane, en offrant ses services pour les satellites de petite taille.

Ainsi, le développement de tels lanceurs doit rencontrer des spécifications très élevées en terme de fiabilité et de performance pour garantir la livraison du satellite sur une orbite précise. Ces dernières sont exprimées au travers d'un cahier des charges portant sur toutes les disciplines impliquées dans une telle réalisation : depuis le système de propulsion jusqu'à la structure, en passant par les commandes de vol. Pour ces dernières, le pilotage du lanceur est rendu difficile par la quantité de données incertaines à considérer, avec, par exemple, un environnement atmosphérique capricieux ou un comportement irrégulier du moteur.

La modélisation prend dès lors une importance capitale puisqu'elle doit traduire le plus fidèlement possible le comportement du lanceur, l'évolution des conditions atmosphériques avec l'altitude et l'intensité du vent rencontré lors de la montée. Des modèles complets sont réalisés en prenant en compte chaque sous-partie du lanceur. Différents niveaux de précision sont possibles, en ajoutant par exemple la dynamique de la tuyère ou le ballotement des ergols

1. European Space Agency

au cours de l'ascension. De même, les données statistiques récupérées par des ballon-sondes permettent d'avoir une idée plus précise des conditions extrêmes de vent que peut rencontrer le lanceur lors de sa montée.

Pour pallier à ces imperfections de modèle, il serait possible d'ajouter des capteurs pour disposer du plus grand nombre de données possibles sur l'état du système. Mais bien souvent ces derniers sont coûteux ou trop imprécis pour certaines grandeurs. On cherche donc à contourner le problème en gardant peu de capteurs, et en réalisant une loi de pilotage la plus « robuste » possible à ces incertitudes sur les grandeurs. Elle doit assurer des performances à la fois sur le modèle nominal, et sur des modèles obtenus pour les valeurs extrêmes des paramètres incertains. Le contrôle d'un lanceur repose alors sur un jeu de modèles représentant un cas nominal et des pires cas traduisant les imprécisions.

Les algorithmes de guidage et de contrôle du lanceur VEGA ont ainsi été réalisés par l'entreprise française ASTRIUM-ST d'après l'expertise qu'elle avait développée avec les projets Ariane. En se basant sur ses projets passés, elle a bâti un modèle de lanceur fictif pour l'étude de diverses méthodes de contrôle (Bérard *et al.*, 2008; Hernandez, 2008). Ce type d'étude peut avoir un impact économique fort sur la fabrication du lanceur, comme l'illustrent les travaux de Sophie Mauffrey au sein de EADS (Mauffrey et Schoeller, 1999). Ils ont permis de gagner 600kg sur le poids du lanceur Ariane V, permettant l'emport de satellites plus lourds, ou l'envoi sur des orbites plus élevées (Desessard, 2002).

Le projet de Maîtrise recherche présenté dans ce mémoire a été effectué dans le cadre d'un accord de double diplôme, entre l'école d'ingénieurs française ISAE²-SUPAÉRO et l'Ecole Polytechnique de Montréal. Il s'est déroulé en majorité à l'École Polytechnique, au sein du Département de Génie Électrique, en Section Automation & Systèmes, tandis qu'un stage de 4 mois a été effectué à SUPAÉRO à l'été 2011, au département DMIA³.

Il reprend les modèles fictifs et le cahier des charges fournis par ASTRIUM-ST. Une loi de commande doit alors être réalisée pour contrôler ce lanceur spatial sur sa trajectoire atmosphérique. Elle doit conserver la stabilité de la structure et réduire au maximum les efforts latéraux qui pourraient mener à sa dislocation, tout en respectant certaines performances. Pour ce faire, la méthode étudiée et développée au cours du projet utilise les applications gardiennes (Saydy *et al.*, 1990; Barmish, 1994).

2. Institut Supérieur de l'Aéronautique et de l'Espace

3. Département de Mathématiques, Informatique et Automatique

1.2 Objectifs de recherche

Deux objectifs principaux se dégagent de ce travail de recherche : le design d'un contrôleur et le développement des outils liés aux applications gardiennes.

Dans un premier temps, un correcteur, d'une structure aussi simple que possible, doit être réalisé pour stabiliser de façon robuste le modèle du lanceur fourni. Pour ce faire, il doit stabiliser simultanément trois modèles différents, correspondant aux cas extrêmes de variation des incertitudes. De plus, des performances doivent être respectées selon le cahier des charges, et ce pour les trois modèles à la fois. On parlera alors d'un travail de synthèse avec performances robustes.

Dans un second temps, des outils de synthèse basés sur les applications gardiennes devront être développés. Même si ces dernières font parfois appels à de lourds calculs numériques, les résultats qu'elles amènent sont puissants. Leur polyvalence permet de les utiliser pour faire aussi bien une étude de robustesse qu'une synthèse de correcteur. Un objectif du projet est donc de rendre plus efficaces les algorithmes de calculs utilisant les applications gardiennes.

1.3 Plan du mémoire

Le travail de recherche présenté à travers ce mémoire se découpe en 7 chapitres principaux.

L'introduction permet de situer rapidement le projet au niveau du domaine spatial, en insistant sur ses tenants et ses aboutissants. La revue de littérature permet de faire un état de l'art sur les différents points abordés à travers cette recherche : depuis un travail de modélisation de lanceur spatial, jusqu'au séquencement d'un correcteur, en passant par les méthodes de synthèse robuste.

Ensuite, la modélisation du lanceur et la prise en compte des incertitudes sont détaillées dans le troisième chapitre. Les modèles fournis sont détaillés à travers la dynamique du lanceur rigide, de la prise en compte des ses modes flexibles, et de la dynamique de l'actionneur de la tuyère. La modélisation des incertitudes est explicitée et mise en avant par l'évolution des paramètres du lanceur et des conditions atmosphériques au cours du vol. Une méthode de validation du contrôleur est finalement détaillée, en observant sa réponse à un profil de vent donné pendant l'ascension.

Le chapitre 4 fait l'objet d'une introduction aux applications gardiennes. Il présente cet outil de façon théorique et énonce les principaux résultats qui y sont liés. Il illustre leur utilisation par des exemples simples sur une étude de robustesse, et sur une méthode de réglage de correcteur à deux paramètres.

Enfin, les chapitres 5 et 6 présentent deux méthodes de synthèse robuste développées à partir des applications gardiennes : une par approche graphique et une par optimisation. La première permet d'appliquer les résultats du chapitre 4 sur le cas du lanceur flexible, avec la prise en compte de nombreuses contraintes. La seconde étend la méthode précédente limitée à deux paramètres variables, en utilisant l'optimisation pour créer une synthèse par itérations. En minimisant progressivement les applications gardiennes adéquates, un correcteur est obtenu, qui respecte l'ensemble des contraintes fixées.

Enfin, le septième chapitre tient lieu de conclusion, en établissant un bilan sur les objectifs poursuivis, et en insistant sur les perspectives possibles de ce projet.

CHAPITRE 2

REVUE DE LITTÉRATURE

Le problème posé passe par cinq étapes principales :

- Modélisation du lanceur rigide/flexible, avec dynamique de l'actionneur ;
- Linéarisation du modèle obtenu autour de points de vol spécifiques ;
- Synthèse d'un contrôleur par point de vol ;
- Séquencement du contrôleur sur l'ensemble du vol considéré ;
- Validation de la loi de commande avec le système complet.

Chacune de ces étapes va être détaillée et illustrée dans la suite, à partir de la littérature actuelle.

Modélisation du lanceur

La dynamique du lanceur comme corps rigide à six degrés de liberté a été largement traitée. De nombreuses configurations sont ainsi disponibles pour le contrôle : avec des ailettes (Siouris, 2004)), avec des jets latéraux (Ping, 2010), ou enfin avec l'orientation de la tuyère (*Thrust vectoring control, TVC*) (Blakelock, 1965; Vanthuyne, 2009).

L'utilisation de surfaces de contrôle pour orienter le lanceur est souvent couplée avec un système d'orientation du moteur complet pour diriger la poussée. Ce genre de système, utilisé sur Saturn V ou Soyouz par exemple, était rendu possible en basculant la chambre de combustion et la tuyère pour des ergols liquides (Teren *et al.*, 1968). Cependant, il est impossible de considérer le même procédé avec des ergols solides car la chambre de stockage et le moteur sont solidaires. Le mouvement de la tuyère devient alors le seul moyen d'orienter la poussée.

Il a fallu attendre les progrès réalisés avec les actionneurs électromécaniques pour atteindre des puissances suffisantes, ainsi que des progrès sur les batteries pour permettre d'embarquer un système complet d'actionneurs électromécaniques à bord d'un lanceur. Ces progrès sont visibles avec les deux boosters d'appoint de la fusée Ariane V ou avec le lanceur VEGA, complètement dépourvu de surfaces de contrôle (Maillebouis, 1988; Carnevale et Resta, 2007).

Linéarisation autour de points de vol

Avec de tels actionneurs, le mouvement de la tuyère introduit des comportements non-linéaires. De plus, la consommation des ergols et l'évolution de l'atmosphère avec l'altitude rendent le modèle variant dans le temps.

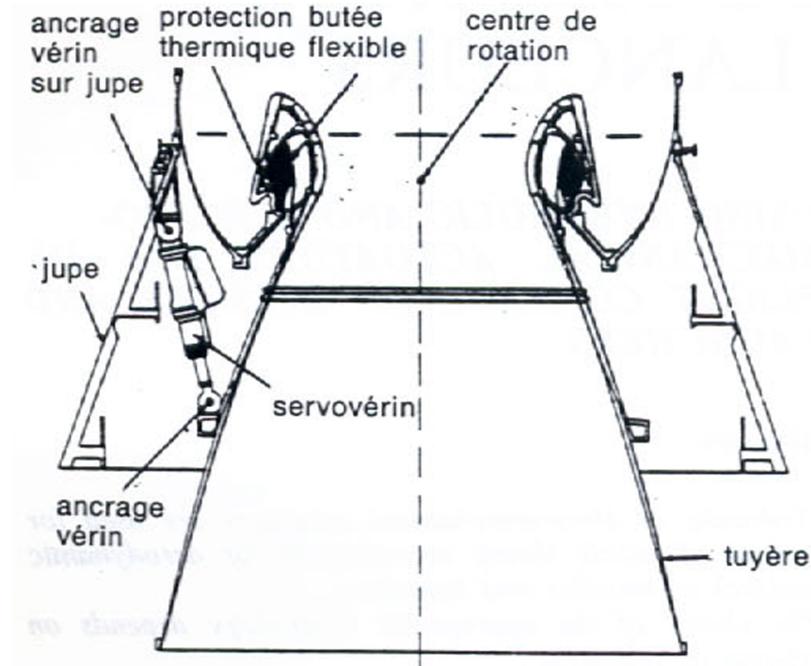


Figure 2.1 Schéma du système d'orientation de la poussée, tiré de Maillebouis (1988)

La symétrie de révolution du lanceur apporte une première simplification, en considérant un mouvement planaire. Elle est évidente pour la structure, mais les actionneurs sont aussi placés symétriquement dans les plans transverses en tangage et lacet, et l'aérodynamique est admise invariante suivant l'angle de roulis. Les équations obtenues sont fournies dans un cadre général dans Greensite (1970), incluant la dynamique du lanceur rigide, des modes flexibles, du ballotement des ergols, des efforts aérodynamiques, de la poussée et l'influence de l'inertie du moteur en mouvement. La modélisation des modes flexibles est réalisée dans Greensite (1970) et Marty (1986) en prenant en compte les liens élastiques entre les différents étages du lanceur et entre le satellite embarqué et son support.

La trajectoire verticale et la minimisation de l'incidence aérodynamique, pour éviter toute sollicitation structurelle, amènent à considérer un modèle petits signaux en linéarisant le modèle non-linéaire précédent. De plus, l'évolution de la masse avec la consommation des ergols et l'ascension atmosphérique à des vitesses supersoniques imposent de considérer les grandeurs variables dans le temps (Turner, 2009). Pour les profils cylindriques considérés le coefficient de portance varie même avec le nombre de Mach, comme l'illustre United States Army (1990).

Le modèle linéaire est alors évalué en différents points de vol pour fournir un jeu de modèles à contrôler.

Synthèse d'un contrôleur

Sur chaque modèle ainsi obtenu, linéaire et invariant dans le temps (LTI), une synthèse de contrôleur est effectuée suivant les performances et les incertitudes considérées. De nombreuses approches sont ici possibles, comme la synthèse H_∞ permettant d'assurer les performances en présence d'incertitudes (Mauffrey *et al.*, 1997). Cependant une telle méthode aboutit à un contrôleur d'ordre élevé. De même dans Voinot *et al.* (2002), une méthode similaire est utilisée mais les modes flexibles ne sont pas pris en compte dans la synthèse, ils interviennent a posteriori pour la validation de la loi de commande. Ce dernier article s'appuyait sur une synthèse dérivée de la méthode H_∞ : la *Forme Croisée Standard* (Cross Standard Form), présentée dans Alazard et Apkarian (1999) et Delmond *et al.* (2006). Une autre synthèse par placement de structure propre est mise en place dans Saussié *et al.* (2008). Enfin, des méthodes plus originales basées sur l'intelligence artificielle sont présentées dans Mehrabian *et al.* (2006).

L'intérêt de la méthode présentée dans ce projet est la prise en compte des modes flexibles directement dans la synthèse, ainsi que des performances exprimées sur le placement des pôles (temps de réponse, amortissement). Cette méthode est basée sur les applications gardiennes présentées dans Saydy *et al.* (1990, 1988) qui viennent compléter les premiers résultats de commande robuste obtenus par Kharitonov (1981), et qui sont résumés dans Barmish (1994). Les méthodes utilisées dans ces travaux concernent l'analyse robuste. Elles sont étendues avec Saussié *et al.* (2010) pour la synthèse par applications gardiennes dans le cas d'un contrôleur ayant une structure fixée. Elles permettent de satisfaire des performances en boucle fermée malgré des incertitudes sur les modèles. Il est ainsi possible de partir de structures de correction simples pour aboutir à contrôleur d'ordre plus faible que les résultats obtenus par l'approche H_∞ par exemple.

Le projet de maîtrise présenté ici portera essentiellement sur une méthode de choix des gains en utilisant les applications gardiennes. Après la présentation d'une approche graphique, une méthode d'optimisation sera utilisée pour améliorer les algorithmes déjà présentés dans Saussié *et al.* (2010), en se basant sur la méthode *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) présentée par exemple dans Nocedal et Wright (2006).

Séquencement des contrôleurs

Une fois la synthèse réalisée pour chaque point de vol, les contrôleurs doivent être interpolés pour obtenir un unique correcteur variant dans le temps qui remplit le cahier des charges. Un bilan des méthodes disponibles pour remplir cette tâche est réalisé au travers de Rugh et Shamma (2000) et Leith et Leithead (2000). Un exemple de séquencement est donné dans

Voinot *et al.* (2002) avec une interpolation linéaire entre chaque instant. Une approche avec observateur permet de rendre les transitions plus lisses entre les contrôleurs avec Voinot *et al.* (2003). Une autre approche basée sur la représentation LFT est illustrée par Rotunno *et al.* (2007); Bérard *et al.* (2012). Dans notre cas, une unique interpolation linéaire est considérée entre les instants de début et de fin de la trajectoire étudiée.

Validation

Enfin, la validation de la structure de correction est faite sur un modèle LFT du lanceur, basé sur les résultats de Bérard *et al.* (2012), avec une interpolation d'ordre 6 pour traduire l'évolution temporelle des paramètres. Cette méthode diffère de Voinot *et al.* (2002) pour lesquels la validation se faisait avec une interpolation linéaire entre les modèles obtenus à chaque instant de vol. Une validation sur un système non linéaire est aussi possible, comme le montre Mehrabian *et al.* (2006).

CHAPITRE 3

MODELISATION DU LANCEUR

3.1 Introduction

Les données fournies par ASTRIUM-ST couvrent les modèles d'état du lanceur, le modèle de l'actionneur pour l'orientation de la poussée et les conditions de vol au cours de l'ascension. Il est en effet indispensable de considérer les variations spatiales et temporelles de ces modèles car le lanceur consomme la majorité de ses ergols au cours des premières minutes de la montée. Ceci entraîne une diminution rapide de sa masse et un changement de sa dynamique avec le déplacement de son centre de gravité et la variation de la poussée. De même, en traversant les différentes couches de l'atmosphère, les conditions de pression et de température influencent les efforts aérodynamiques, venant ainsi dégrader les performances du système.

Il apparaît donc clairement que les modèles considérés sont variants dans le temps et incertains sur plusieurs paramètres. Il est en effet difficile de quantifier, par exemple, la consommation des ergols. Le choix fait par ASTRIUM-ST pour représenter ces aléas est de fournir deux modèles supplémentaires représentant les variations extrêmes des paramètres incertains tels la masse du lanceur ou la position du centre de gravité. Les variations des conditions atmosphériques seront aussi prises en compte dans ces modèles.

De plus, un lanceur étant bien plus élancé qu'un simple missile, le modèle prendra en compte les déformations de la structure dues à la poussée d'un côté, et au poids et efforts aérodynamiques de l'autre. Ces forces créent une résultante latérale qui vient en effet exciter les modes flexibles. Sachant qu'une fusée est principalement conçue pour endurer des efforts en compression, ces sollicitations latérales peuvent engendrer une flexion importante, conduisant à la destruction du lanceur sans une loi de commande adaptée. Dans le cas présent, deux modes flexibles seront considérés.

Par soucis de précision et d'intégration système, un modèle de l'actionneur de la tuyère est fourni. Il permet le contrôle du lanceur par orientation de la tuyère d'éjection, technique utilisée pour les contrôles à faible vitesse car une commande par surface de contrôle, comme des ailerons, n'est pas suffisamment efficace au décollage. Ce système est par exemple présent sur le nouveau lanceur VEGA de l'ESA.

Enfin, le but de la structure de correction étant de stabiliser le lanceur sur la phase de vol considérée, la validation du correcteur sera faite pour un pire cas de rafale de vent. Le profil en vitesse de ce vent latéral est aussi fourni et sera détaillé plus loin.

3.2 Schéma du lanceur

La modélisation du lanceur est détaillée dans le schéma de la Fig. 3.1, en faisant un inventaire des forces considérées. On note (GX, GY) le référentiel terrestre attaché à G , et (Gx, Gy) le référentiel des axes principaux du lanceur.

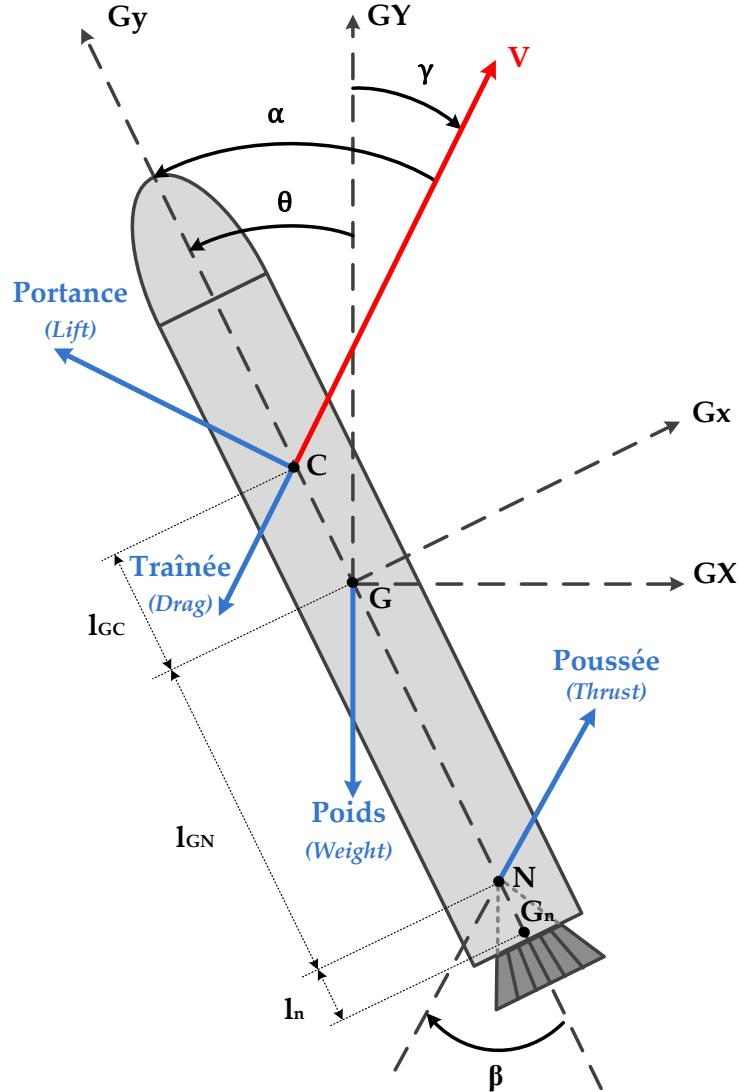


Figure 3.1 Schéma du lanceur complet

3.3 Conditions de vol

La phase de vol considérée débute à 25s après le décollage et se termine à 60s. C'est seulement après les 10 à 20 premières secondes que le lanceur quitte complètement l'aire de

lancement pour entamer l'ascension. Dans ce créneau, la trajectoire est encore atmosphérique et les manœuvres de basculement n'ont pas encore eu lieu. Le but de la loi de commande et donc de maintenir la verticalité. De plus, pour prendre en compte les évolutions temporelles importantes des paramètres, les modèles utilisés pour le lanceur et son environnement sont linéarisés en 6 points de vol particuliers, sélectionnés par ASTRIUM-ST.

Les données de vol peuvent ainsi être tracées au cours de l'ascension. Le graphe Fig. 3.2 présente l'évolution de l'altitude, des conditions atmosphériques (température, pression, masse volumique), de la vitesse du lanceur, de la pression dynamique, de la masse, de la poussée et enfin du déplacement du centre de gravité .

La pression dynamique est particulièrement importante, car elle présente un maximum, connu sous le nom de « *zone Max Q* ». A cet instant, les efforts aérodynamiques externes atteignent leur intensité maximale et la structure endure un chargement extrême.

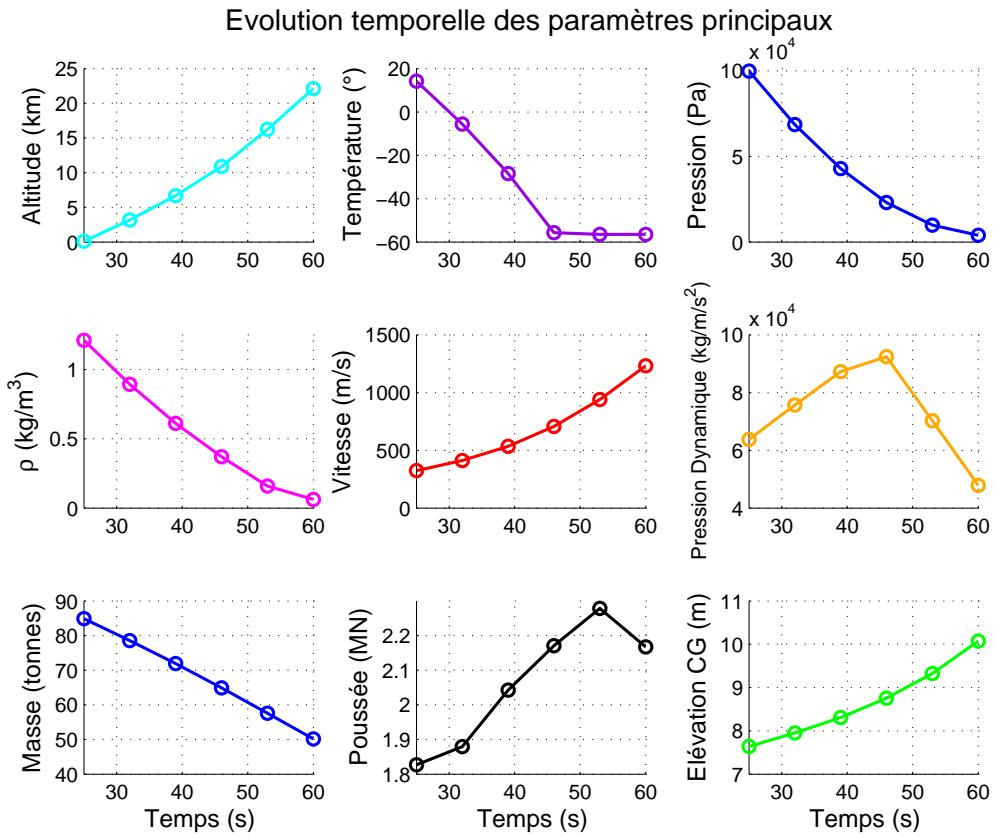


Figure 3.2 Evolution temporelle des paramètres

Enfin, le profil de vent représentant le pire cas à endurer pour le système est représenté en Fig. 3.3. La vitesse considérée se rapproche d'un échelon de l'ordre de $-20m/s$, avec un dépassement supplémentaire d'environ 25%.

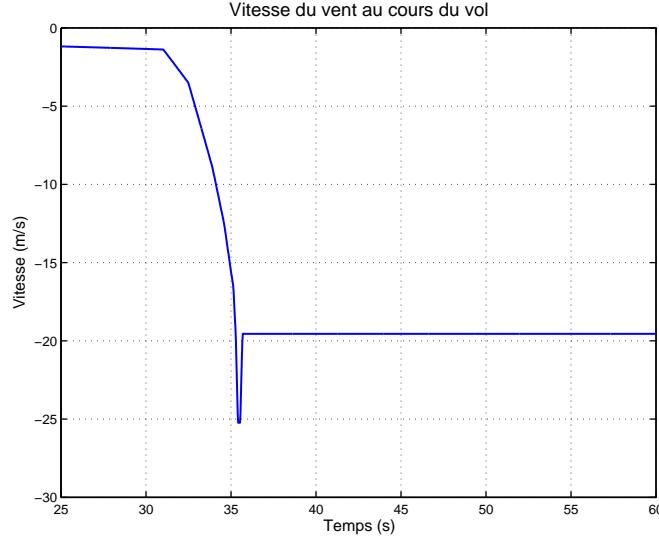


Figure 3.3 Profil de vent utilisé en simulation

3.4 Modélisation rigide

Le mouvement du lanceur est considéré planaire. Les efforts auxquels il est soumis sont son poids, sa poussée, les efforts aérodynamiques (portance et traînée) et les efforts d'inertie dus au mouvement de la tuyère. Chacune de ces forces sera considérée à travers les moments qu'elle induit au niveau du centre de gravité G .

Pour les conditions de vol précédentes, les équations du mouvement d'un corps rigide dans l'atmosphère vont être linéarisées autour des 6 points de vol régulièrement espacés : 25s, 32s, ..., 60s. Pour ce faire, la dynamique longitudinale (étant donné l'axisymétrie du problème) peut être obtenue dans un cas très général d'après Blakelock (1965). Il est aussi simple de repartir du principe fondamental de la dynamique pour l'obtenir dans notre cas particulier.

3.4.1 Principe Fondamental de la Dynamique

S'appuyant sur le référentiel terrestre attaché au lanceur, son attitude est décrite par l'assiette θ , et est dictée par les moments extérieurs appliqués au lanceur rigide dans un premier temps. On obtient alors l'équation (3.1) appliquée au centre de gravité G .

$$I\ddot{\theta} = \mathcal{M}_{G,ext} \quad (3.1)$$

Ce qui donne (3.2) avec les moments présents (le poids ne crée aucun moment en G).

$$I\ddot{\theta} = \mathcal{M}_{G,L} + \mathcal{M}_{G,D} + \mathcal{M}_{G,T} + \mathcal{M}_{G,N} \quad (3.2)$$

Couples extérieurs appliqués

Les couples appliqués sur le lanceur vont être explicités en fonction des différents paramètres.

Forces aérodynamiques Les moments aérodynamiques sont engendrés par les composantes latérales de la portance et de la traînée. La projection sur l'axe Gx de ces efforts fournit $\mathcal{M}_{G,L} = L \cos \alpha l_{GC}$ et $\mathcal{M}_{G,D} = D \sin \alpha l_{GC}$. Les forces de portance et de traînée s'expriment comme (3.3) et (3.4) où $Q = \frac{1}{2}\rho V^2$ est la pression dynamique.

$$L = QS_{ref}C_{z_\alpha}\alpha \quad (3.3)$$

$$D = QS_{ref}C_x \quad (3.4)$$

Il reste à prendre en compte l'effet du vent sur le lanceur. Sachant qu'il vient modifier l'incidence locale du lanceur, il sera considéré comme un couple perturbateur qui modifie le couple aérodynamique de la portance sans vent. Le schéma Fig. 3.4 présente la correction à apporter en présence de vent : $\alpha_W = \alpha - \Delta\alpha$.

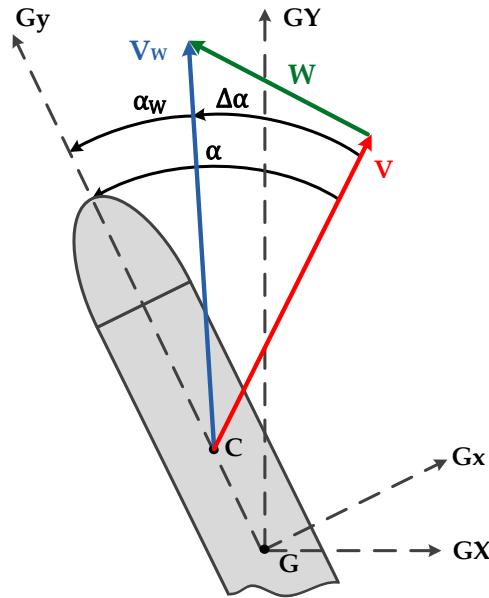


Figure 3.4 Influence du vent sur l'incidence

Le moment aérodynamique de portance peut donc s'écrire comme (3.5), avec $\alpha = \theta$ sans vent car le vecteur vitesse de référence est vertical (i.e. $\gamma = \theta - \alpha = 0$).

Le moment aérodynamique de traînée est donné en (3.6).

$$\mathcal{M}_{G,L} = QS_{ref}C_{z_\alpha}(\theta - \Delta\alpha) l_{GC} \quad (3.5)$$

$$\mathcal{M}_{G,D} = QS_{ref}C_x l_{GC} \quad (3.6)$$

De plus, en considérant le triangle formé par les vitesses, il est possible de relier $\Delta\alpha$ à la vitesse du vent : $\tan \Delta\alpha = \frac{W}{V}$.

Force de poussée Le moment engendré par la poussée du moteur, appliquée en N s'exprime en fonction de sa composante latérale par (3.7).

$$\mathcal{M}_{G,T} = T \sin \beta l_{GN} \quad (3.7)$$

Forces d'inertie de la tuyère Le mouvement de la tuyère devant se faire rapidement pour rétablir la verticalité, les accélérations et décélérations mises en jeu sont importantes et doivent être prises en compte dans le bilan des moments. Il peut en effet arriver que ces moments viennent perturber l'effet de la tuyère. Ce phénomène est appelé le « Tail-Wags-Dog » effect (Blakelock, 1965). Ainsi, le moment dû à l'accélération angulaire $\ddot{\beta}$ s'exprime par $-I_{n,N}\ddot{\beta}$. Le théorème de Huygens permet finalement d'obtenir $-I_{n,N}\ddot{\beta} = -(I_n + ml_n^2)\ddot{\beta}$.

De plus, étant donné la masse non négligeable de la tuyère et le bras de levier dont elle bénéficie, les efforts d'inertie dûs à l'accélération de G_n sont pris en compte. Ils sont présents lors du départ et de l'arrêt de la tuyère, au moment où son accélération est la plus élevée.

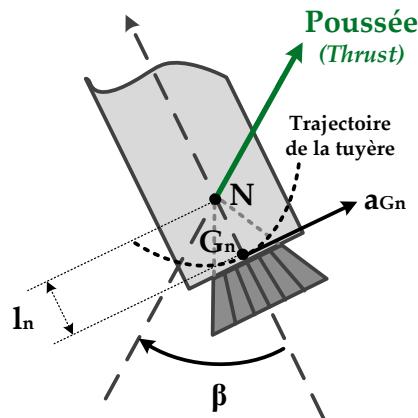


Figure 3.5 Modélisation adoptée pour représenter les mouvements de la tuyère

Considérant le schéma de la Fig. 3.5, cette force d'inertie peut s'écrire comme $-ma_{G_n}$, avec a_{G_n} l'accélération latérale. Elle s'applique au point N , profitant du bras de levier l_{GN} . Se plaçant alors dans le cas de petites déflexions ($\beta \ll 1$), ce qui se vérifie en réalité, on aboutit à $a_{G_n} = l_n \ddot{\beta}$.

Finalement le moment complet dû au mouvement de la tuyère est résumé en (3.8).

$$\mathcal{M}_{G,N} = -(I_n + ml_n(l_n + l_{GN})) \ddot{\beta} \quad (3.8)$$

Linéarisation du modèle non-linéaire

Pour simplifier l'étude, une linéarisation des équations est réalisée autour du pseudo-équilibre en montée pour lequel $\alpha_0 = \theta_0 = 0^\circ$, $\gamma_0 = 0^\circ$, $W_0 = 0m/s$ et $\beta_0 = 0^\circ$. Ainsi, le modèle petits signaux s'exprime en considérant $\theta \ll 1$, $\beta \ll 1$, et $\gamma = 0$ pour une trajectoire verticale sans vent, soit $\alpha = \theta$. De plus, la force de traînée est négligée devant les efforts latéraux dus à la portance ou à la poussée.

Par conséquent :

- $\mathcal{M}_{G,L} \simeq QS_{ref}C_{z_\alpha}(\theta - \frac{W}{V}) l_{GC}$
- $\mathcal{M}_{G,D} \simeq 0$
- $\mathcal{M}_{G,T} \simeq T\beta l_{GN}$
- $\mathcal{M}_{G,N} = -(I_n + ml_n(l_n + l_{GN})) \ddot{\beta}$
- $\alpha_W = \theta - \Delta\alpha \simeq \theta - \frac{W}{V}$

L'équation linéarisée de la dynamique du lanceur rigide en tangage est donnée en (3.9). Les variables d'état sont en rouge, les entrées en bleu.

$I \ddot{\theta} = QS_{ref}C_{z_\alpha}l_{GC} \theta - QS_{ref}C_{z_\alpha} \frac{l_{GC}}{V} W + Tl_{GN} \beta - (I_n + ml_n(l_n + l_{GN})) \ddot{\beta}$

(3.9)

Il est important de souligner ici que les paramètres du modèle obtenu, tels que l'inertie du lanceur, la pression dynamique, le coefficient de portance ou même les bras de levier relatifs au centre de gravité, sont des fonctions du temps. Pour considérer ces évolutions, le modèle est évalué en 6 instants de vol : toutes les 7s de 25s à 60s après le décollage. Les jeux de données sont tous fournis par ASTRIUM-ST.

Modèle d'état du lanceur rigide

Le modèle d'état s'obtient à partir de l'équation précédente, en considérant les entrées $U = \begin{bmatrix} W & \beta & \ddot{\beta} \end{bmatrix}^T$, et les sorties $Y = \begin{bmatrix} \theta & \alpha \end{bmatrix}^T$. Les variables d'état utilisées sont $X = \begin{bmatrix} \dot{\theta} & \theta \end{bmatrix}^T$. On aboutit au modèle fourni en (3.10) et (3.11).

$$\dot{X} = \begin{bmatrix} 0 & \frac{QS_{ref}l_{GC}C_{z\alpha}}{I} \\ 1 & 0 \end{bmatrix} X + \begin{bmatrix} -\frac{1}{V} \frac{QS_{ref}l_{GC}C_{z\alpha}}{I} & \frac{Tl_{GN}}{I} & -\frac{I_{tuy}+ml_n(l_n+l_{GN})}{I} \\ 0 & 0 & 0 \end{bmatrix} U \quad (3.10)$$

$$Y = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} X + \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{V} & 0 & 0 \end{bmatrix} U \quad (3.11)$$

Ce modèle sera noté (A_r, B_r, C_r, D_r) avec la notation habituelle :

$$\dot{X} = A_r X + B_r U$$

$$Y = C_r X + D_r U$$

3.5 Modélisation flexible

La prise en compte des modes flexibles dans les modèles fournis par ASTRIUM-ST est faite pour les deux premiers modes. Ils apportent une correction sur le bras de levier de la poussée dans la formule précédente, ainsi que sur l'angle d'attitude, qui est légèrement modifiée pour un lanceur souple. Pour ce faire, nous allons voir une méthode pour obtenir les pulsations et les modes propres d'un lanceur, pour ensuite évaluer les déformées latérales sous l'influence de la poussée (qui créé un moment fléchissant) et leur influence sur l'incidence du lanceur flexible.

3.5.1 Pulsations et modes propres

Pour l'étude des pulsations propres et des modes propres associés, il faut dans un premier temps mettre le comportement dynamique du système sous forme matricielle, c'est-à-dire discréteriser le lanceur. Pour ce faire, il est possible de le considérer comme un ensemble de masses le long d'une poutre, reliées entre elles par des ressorts et des amortisseurs. Chaque étage est ainsi modélisé, avec sa masse d'ergols en mouvement et sa structure, et le satellite est lié au dernier étage. Une modélisation adoptée dans Marty (1986) est donnée en Fig. 3.6.

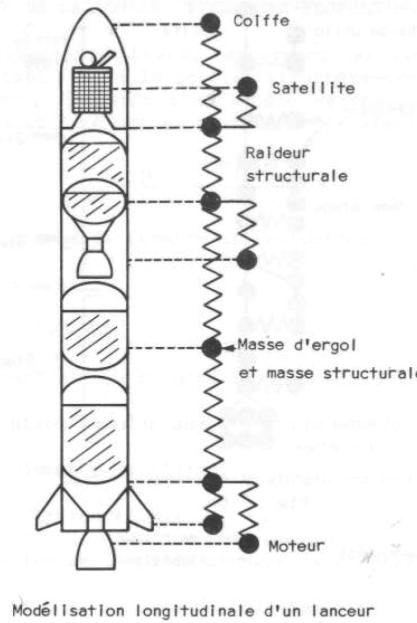


Figure 3.6 Modélisation discrète d'un lanceur avec prise en compte des ergols et de la charge utile, tirée de Marty (1986)

Une équation matricielle permet de décrire la dynamique du système complet :

$$[M]\ddot{X} + [C]\dot{X} + [K]X = F$$

en notant :

- X l'ensemble des coordonnées de chaque masse du modèle discret du lanceur
- $[M]$ la matrice des masses
- $[C]$ la matrice des amortissements
- $[K]$ la matrice des raideurs
- F le vecteur des efforts appliqués sur le lanceur

A partir de cette équation, on obtient les pulsations et les modes propres en prenant le "système conservatif associé", c'est-à-dire en cherchant les valeurs propres et les vecteurs propres associés pour les matrices $[M]$ et $[K]$ qui sont simultanément diagonalisables d'après la théorie des structures. On obtient alors les pulsations ω_i de chaque mode par les valeurs propres, et les modes associés ϕ_i avec les vecteurs propres correspondants.

Un changement de variable sur l'équation du système est opéré pour diagonaliser ce système d'équations, en considérant que les termes extra-diagonaux de la matrice des amortissements ne modifieront que faiblement les pulsations propres et les modes obtenus. La matrice de masses obtenue est $[\mu]$ et celle de raideurs $[k]$. La matrice du changement de base sera notée $[\phi]$ et elle contient les modes propres normés. L'équation se transforme en (3.12) :

$$[\mu].\ddot{q} + [C'].\dot{q} + [k].q = [\phi]^{-1}.F \quad (3.12)$$

avec
$$\begin{cases} q &= [\phi]^{-1}.X \\ [\mu] &= [\phi]^{-1}.[M].[\phi] \\ [C'] &= [\phi]^{-1}.[C].[\phi] \\ [k] &= [\phi]^{-1}.[K].[\phi] \end{cases}$$

Dans la pratique, on considère aussi que la matrice des amortissements $[C']$ est diagonale en négligeant ses termes extra-diagonaux, dans le but de découpler les équations sur les coordonnées généralisées q . Enfin, les coordonnées réelles X sont obtenues par le changement de base inverse avec $q = [\phi]^{-1}.X$.

Ce raisonnement permet par exemple d'obtenir les deux premiers modes propres présents sur la Fig. 3.7 pour la déformation en flexion du lanceur, d'après les résultats tirés de Marty (1986).

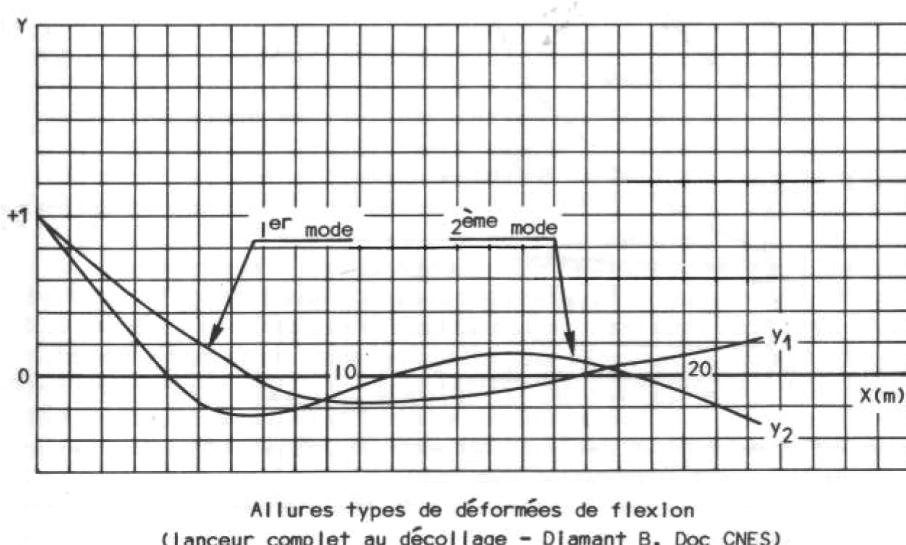


Figure 3.7 Déformées latérales pour les deux premiers modes de flexion, tirées de Marty (1986)

3.5.2 Modèle du lanceur souple

Il est ici important de remarquer que les pulsations propres et les modes propres sont tous dépendants du temps, car la masse du lanceur varie continument le long de la trajectoire. Deux couples (pulsation, mode propre) sont donc calculés à chacun des 6 instants de linéarisation. Ensuite, sachant que la poussée varie aussi à chaque phase, en raison des conditions atmosphériques et de la vitesse du lanceur, l'amplitude des déformées des modes sont aussi réévaluées.

Le modèle d'état du lanceur rigide sera donc augmenté avec ces deux dynamiques, pour avoir un modèle du lanceur souple. Les nouvelles variables d'état considérées sont q_1 et q_2 , les deux premières coordonnées généralisées, ainsi que leur dérivées. Ces variables satisfont l'équation en (3.13), pour laquelle ω_i change mais l'amortissement $\xi = 0.01$ reste le même pour les deux modes.

$$\ddot{q}_i + 2\xi\omega_i\dot{q}_i + \omega^2 q_i = F_i \quad (3.13)$$

Sachant que les efforts qui excitent ces déformées sont dûs à la poussée, on doit considérer d'une part le couple engendré par la composante latérale de la poussée, et d'autre part les couples dus au mouvement du centre de gravité de la tuyère. Ceci nous amène au modèle d'état en (3.14), ayant deux entrées manipulables β et $\ddot{\beta}$, une perturbation W et un vecteur d'état $X = \begin{bmatrix} \dot{q}_1 & \omega_1 q_1 & \dot{q}_2 & \omega_2 q_2 \end{bmatrix}^T$.

$$\begin{bmatrix} \ddot{q}_1 \\ \omega_1 \dot{q}_1 \\ \ddot{q}_2 \\ \omega_2 \dot{q}_2 \end{bmatrix} = \begin{bmatrix} -2\xi\omega_1 & -\omega_1 & 0 & 0 \\ \omega_1 & 0 & 0 & 0 \\ 0 & 0 & -2\xi\omega_2 & -\omega_2 \\ 0 & 0 & \omega_2 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \omega_1 q_1 \\ \dot{q}_2 \\ \omega_2 q_2 \end{bmatrix} + \begin{bmatrix} 0 & B32 & B33 \\ 0 & 0 & 0 \\ 0 & B52 & B53 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} W \\ \beta \\ \ddot{\beta} \end{bmatrix} \quad (3.14)$$

Cette dynamique sera notée $\dot{X} = A_f X + B_f U$.

Les termes $B32$, $B33$, $B52$ et $B53$ sont fournis par ASTRIUM-ST. Ils correspondent aux efforts généralisés, qui sont obtenus en considérant comme efforts de cisaillement le couple engendré par la poussée latérale, et celui engendré par le mouvement du centre de gravité de la tuyère.

3.5.3 Corrections apportées par les modes flexibles

Enfin, la dynamique de ces modes de flexion est incorporé dans le modèle du lanceur rigide (3.9). La principale correction se fait sur le bras de levier de la poussée, car le centre de rotation de la tuyère, point d'application de la poussée, bouge légèrement avec la structure flexible. La modification du couple dû à l'inertie de la tuyère est négligée devant celui dû à la poussée latérale.

Le couple correcteur prend la forme : $\mathcal{C}_{cor} = T.\Delta L_{GP} = T.\zeta q_1 + T.\eta q_2$ où ζ et η sont des coefficients reliés à l'amplitude des déformées des modes propres.

Finalement, l'équation en (3.15) traduit la dynamique du lanceur flexible, avec la prise en compte des deux premiers modes de flexion. (variables d'état en rouge, entrées en bleu)

$$I\ddot{\theta} = QS_{ref}l_{GC}C_{z_\alpha}\theta + T\zeta q_1 + T\eta q_2 + \frac{QS_{ref}l_{GC}C_{z_\alpha}}{V}W + Tl_{GN}\beta - (I_{noz} + ml_n(l_n + l_{GN}))\ddot{\beta} \quad (3.15)$$

Enfin, les déformées du lanceur changent sa forme au niveau des centrales inertielles qui mesurent son attitude. La correction à apporter est donc liée au fait que ces mesures seront légèrement faussées pour le lanceur flexible.

$$\theta_{flexible} = \theta_{rigide} + \epsilon q_1 + \mu q_2 \quad (3.16)$$

Modèle d'état du lanceur flexible

Le modèle d'état flexible s'obtient à partir des équations (3.10-3.14-3.15-3.16), en considérant les mêmes entrées $U = [W \ \beta \ \ddot{\beta}]^T$ et sorties $Y = [\theta \ \alpha]^T$. Le vecteur d'état est par contre augmenté avec les déplacements et leur vitesse : $X_F = [\dot{\theta} \ \theta \ \dot{q}_1 \ \omega_1 q_1 \ \dot{q}_2 \ \omega_2 q_2]^T$. On aboutit au modèle fourni en (3.17-3.18).

$$\dot{X}_F = \left[\begin{array}{c|cccc} A_r & 0 & T.\zeta & 0 & T.\eta \\ 0 & 0 & 0 & 0 & 0 \\ \hline 0 & & & & A_f \end{array} \right] X_F + \left[\begin{array}{c} B_r \\ B_f \end{array} \right] U \quad (3.17)$$

$$Y = \left[\begin{array}{c|ccccc} C_r & 0 & \epsilon & 0 & \mu \\ 0 & 0 & 0 & 0 & 0 \end{array} \right] X_F + D_r U \quad (3.18)$$

Ce modèle sera noté $(\tilde{A}_F, \tilde{B}_F, \tilde{C}_F, \tilde{D}_F)$ pour "Full", sachant qu'il contient la dynamique complète de la structure flexible. Le modèle final (A_F, B_F, C_F, D_F) incorporera la dynamique de l'actionneur.

3.6 Modélisation de l'actionneur

Le modèle fourni par ASTRIUM-ST pour décrire la dynamique de l'actionneur est d'ordre 4. Il prend en compte le mouvement de la tuyère selon son axe de rotulage, car il peut exister des cas d'instabilité par rapport aux perturbations extérieures si la rigidité des structures est négligée (Renault, 2005). Partant d'un modèle numérique sous forme de fonction de transfert, un modèle théorique a ainsi été extrait de la littérature pour expliquer le fonctionnement de l'actionneur.

3.6.1 Aperçu du modèle fourni

Le modèle fourni est un système d'ordre 4, dont la fonction de transfert possède 2 paires de pôles complexes conjugués et une paire de zéros complexes conjugués, comme le montre le lieu des racines de la Fig. 3.8.

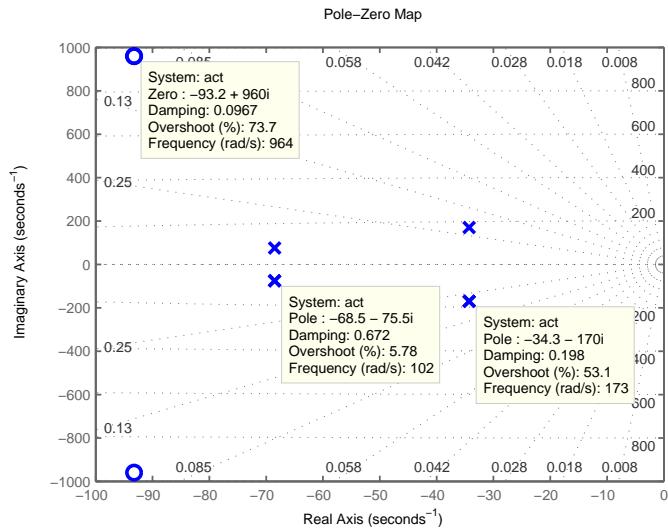


Figure 3.8 Lieu des pôles et zéros de la fonction de transfert actionneur

La fonction de transfert associée est donnée sous forme factorisée en (3.19).

$$F(s) = \frac{359.7(s^2 + 186.5s + 9.3 \cdot 10^5)}{(s^2 + 137s + 1.04 \cdot 10^4)(s^2 + 68.6s + 3.01 \cdot 10^4)} \quad (3.19)$$

3.6.2 Actionneur électromécanique à retour d'effort

Au vu de la bande passante obtenue pour la boucle ouverte, il s'agit probablement d'un actionneur électromécanique (EMA) plutôt qu'hydraulique (bande passante plus faible (Maillebouis, 1988)). Un tel actionneur est constitué d'un moteur électrique, d'un réducteur, d'une

vis à rouleaux qui transforme le mouvement angulaire en translation, et finalement d'une charge. Dans notre cas la charge est la tuyère.

En s'appuyant sur la modélisation des actionneurs électromécaniques et sur la structure de contrôle présents dans Renault (2005) et Carnevale et Resta (2007), un modèle théorique cohérent avec les valeurs numériques peut être obtenu.

Dans un premier temps, l'asservissement de la position angulaire β de la tuyère se fait sur la position linéaire y du piston de l'actionneur en raison des capteurs disponibles. Ainsi, un capteur de position linéaire attaché à la position du piston (LVDT sensor), ainsi qu'un capteur de vitesse sur la rotation du moteur (resolver sensor), renvoient la vitesse de déplacement et la position. Cette considération est confirmée dans Carnevale et Resta (2007) sur l'actionneur d'orientation de poussée du lanceur européen VEGA.

Ensuite, il est d'usage d'incorporer un capteur de force (load sensor) sur l'actionneur électromécanique pour permettre de mieux rejeter les perturbations sur la tuyère. Ces dernières rendent la boucle de contrôle instable si un simple filtre réjecteur est utilisé pour amortir le système (Renault, 2005).

On peut donc considérer que l'on dispose maintenant de 4 retours pour réaliser la boucle de contrôle du moteur et de la tuyère : un retour en position et en vitesse sur la position de l'actionneur, et un retour proportionnel et dérivé sur la force exercée par l'actionneur.

La représentation de l'actionneur est donnée en Fig. 3.9 sous forme de schéma cinématique.

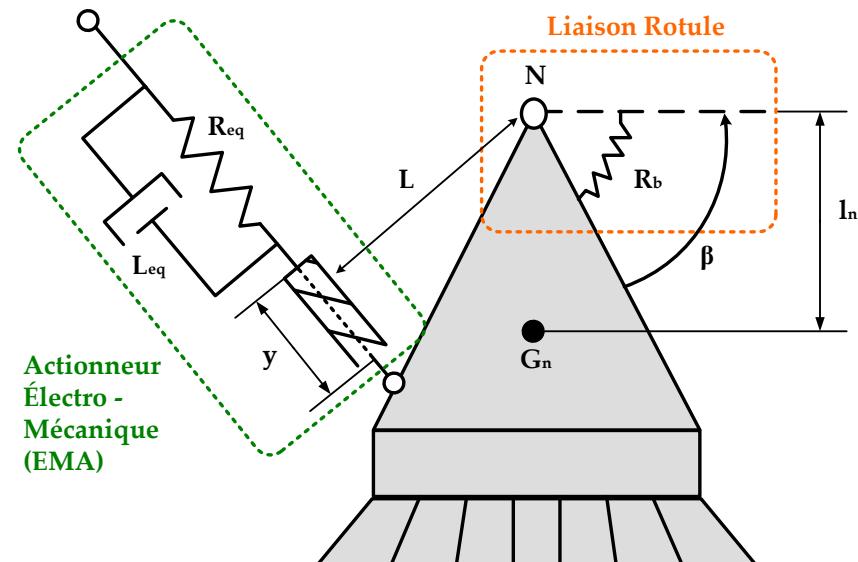


Figure 3.9 Schéma cinématique de l'actionneur EMA tiré de Renault (2005)

Les 3 équations dynamiques de l'actionneur représentent la dynamique du moteur avec la vis à rouleaux (3.20), puis la force appliquée par l'actionneur sur la tuyère (3.21), et enfin la dynamique de la tuyère (3.22). On suppose que le moteur possède déjà une boucle de contrôle interne rapide, de sorte que le gain est unitaire : $T/u = 1$ avec T le couple produit et u la consigne donnée au moteur rebouclé.

$$I_m N_{red} s^2 y = u - \frac{F_a}{N_{red}} \quad (3.20)$$

$$F_a = -R_{eq} (L\beta - y) - s L_{eq} (L\beta - y) \quad (3.21)$$

$$(I_n + m l_n^2) s^2 \beta = L F_a - R_b \beta \quad (3.22)$$

avec

- y : déplacement longitudinal du piston de l'actionneur
- β : l'angle de déflection de la tuyère
- u : la consigne envoyée au moteur
- F_a : force exercée par l'actionneur sur la tuyère
- I_m : inertie du moteur selon son axe de rotation
- N_{red} : rapport de réduction introduit par la vis à rouleaux
- L : bras de levier de l'actionneur par rapport au point de rotulage de la tuyère
- R_{eq} : rigidité équivalente de l'actionneur et de sa liaison avec la tuyère
- L_{eq} : amortissement équivalent de l'actionneur et de sa liaison avec la tuyère
- I_n : inertie de la tuyère selon son axe de rotulage
- m : masse de la tuyère
- l_n : distance entre le centre de gravité et le point de rotulage de la tuyère
- R_b : rigidité de la liaison rotule de la tuyère

Partant de ces équations, les fonctions de transfert entre y et u , puis entre F_a et u sont données en (3.23-3.24). Elles permettent d'exprimer plus facilement la fonction de transfert de l'actionneur EMA complet, couplé avec la tuyère. Dans ce modèle, on négligera la rigidité de la tuyère en rotation R_b , car elle sera très faible devant la rigidité en déplacement de l'actionneur $R_{eq}L^2$.

$$\frac{y}{u} = \frac{s^2 + 2\xi_0\omega_0 s + \omega_0^2}{I_m N_{red} s^2 (s^2 + 2\xi_n\omega_n s + \omega_n^2)} \quad (3.23)$$

$$\frac{F_a}{u} = \frac{L_{eq} s + R_{eq}}{I_m N_{red} s^2 (s^2 + 2\xi_n\omega_n s + \omega_n^2)} \quad (3.24)$$

Les pulsations ω_0 et ω_n correspondent respectivement aux dynamiques de la tuyère, et de l'actionneur en présence de la charge. Leurs expressions et celles des amortissements en fonction des paramètres du problème sont les suivantes :

$$\begin{aligned}\omega_0^2 &= \frac{R_{eq}L^2}{I_n + ml_n^2} \\ 2\xi_0\omega_0 &= \frac{L_{eq}L^2}{I_n + ml_n^2}\end{aligned}$$

$$\begin{aligned}\omega_n^2 &= R_{eq} \left(\frac{L^2}{I_n + ml_n^2} + \frac{1}{I_m N_{red}^2} \right) \\ 2\xi_n\omega_n &= L_{eq} \left(\frac{L^2}{I_n + ml_n^2} + \frac{1}{I_m N_{red}^2} \right)\end{aligned}$$

Si l'on s'intéresse maintenant à une boucle de contrôle avec un retour proportionnel sur l'écart en position, un retour dérivé sur la position seulement, un proportionnel et un dérivé sur l'effort, la consigne envoyée à l'actionneur peut se réécrire comme (3.25). Ces retours sont choisis en fonction des capteurs disponibles. La structure de correction est présentée sur la Fig. 3.10.

$$u = K_p(L\beta_{ref} - y) - K_d s y - (K_{FFp} - K_{FFd} s)F_a \quad (3.25)$$

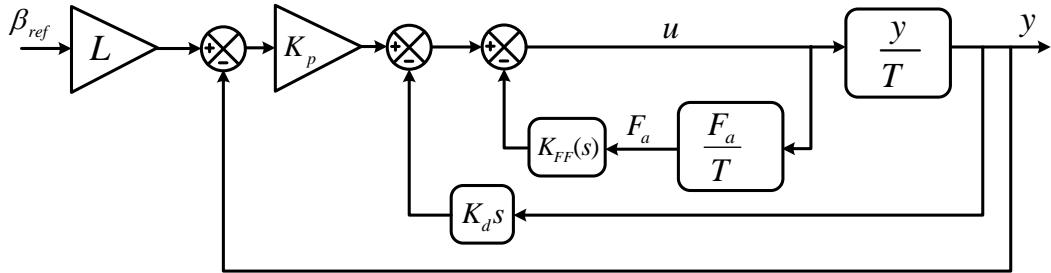


Figure 3.10 Boucle de contrôle avec retour en position et en effort

Tout calcul fait, la fonction de transfert en boucle fermée (3.27) possède deux zéros et quatre pôles comme le modèle recherché. Elle possède au numérateur les deux zéros très mal amortis tandis que le dénominateur permet un placement de pôles avec les retours choisis sur la position et l'effort.

$$\frac{y}{\beta_{ref}} = L \frac{K_p (s^2 + 2\xi_0\omega_0 s + \omega_0^2)}{I_m N_{red} s^2 (s^2 + 2\xi_n\omega_n s + \omega_n^2) + R_{eq} (K_{FFd} s + K_{FFp}) s^2 + (K_d s + K_p) (s^2 + 2\xi_0\omega_0 s + \omega_0^2)} \quad (3.26)$$

Le développement du dénominateur amène à (3.27).

$$\frac{y}{\beta_{ref}} = L \frac{\bar{K}_p s^2 + 2\xi_0\omega_0\bar{K}_p s + \omega_0^2\bar{K}_p}{A_4 s^4 + A_3 s^3 + A_2 s^2 + A_1 s + A_0} \quad (3.27)$$

avec

$$\begin{aligned} A_4 &= 1 + \bar{K}_{FFd} L_{eq} \\ A_3 &= \bar{K}_{FFd} R_{eq} + \bar{K}_{FFp} L_{eq} + \bar{K}_d + 2\xi_n \omega_n \\ A_2 &= \bar{K}_{FFp} R_{eq} + \bar{K}_p + \omega_n^2 + 2\xi_0 \omega_0 \bar{K}_d \\ A_1 &= \omega_0^2 \bar{K}_d + 2\xi_0 \omega_0 \bar{K}_p \\ A_0 &= \omega_0^2 \bar{K}_p \end{aligned}$$

Enfin, l'identification avec les valeurs numériques disponibles dans Liscouët (2010) sur les actionneurs du lanceur VEGA, avec les modèles du lanceur permettent de tirer les valeurs des gains, résumées dans la table 3.1.

Tableau 3.1 Valeurs des gains identifiées avec le modèle de l'actionneur

Grandeurs fournies	Grandeurs identifiées
$L = 1.07 \text{ m}$	$\bar{K}_p = 335.8$
$N_{red} = 1081$	$K_d = 5.2$
$I_m = 9.24 \cdot 10^{-4} \text{ kg.m}^2$	$K_{FFp} = -2.04 \cdot 10^{-3}$
$I_n = 920.1 \text{ kg.m}^2$	$K_{FFd} = 6.62 \cdot 10^{-7}$
$l_n = 0.2 \text{ m}$	
$R_{eq} = 7.88 \cdot 10^8 \text{ Nm/rad}$	
$L_{eq} = 1.58 \cdot 10^5 \text{ Nm/(rad/s)}$	
$\omega_0 = 964 \text{ rad/s}$	
$\xi_0 = 3 \cdot 10^{-4}$	
$\omega_n = 1220 \text{ rad/s}$	
$\xi_n = 3 \cdot 10^{-4}$	

On supposera enfin que la relation entre le déplacement longitudinal y de l'actionneur et le déplacement angulaire β de la tuyère vaut $y = L\beta$, car cette dynamique est rapide devant celle du système bouclé. On obtient alors la fonction de transfert attendue, qui relie une référence β_{ref} et la vraie valeur de β :

$$\frac{\beta}{\beta_{ref}} = \frac{1}{L} \frac{y}{\beta_{ref}}$$

3.7 Evolution temporelle et modélisation des incertitudes

Il a déjà été mentionné que le modèle du lanceur était variant dans le temps car la trajectoire considérée de 25s à 60s ne permet pas de considérer la masse ou les inerties constantes par exemple. Les variations temporelles des paramètres ayant déjà été détaillées dans la partie 3.3, on s'attardera ici à leur influence sur les pôles du système ainsi que sur le diagramme de Bode du lanceur flexible.

De plus, la prédiction de la courbe de poussée du moteur est extrêmement difficile du fait de l'utilisation fréquente de propergols solides pour le premier étage ou même pour les propulseurs d'appoint (cf. lanceurs Ariane). Différents profils de poussée sont alors envisagés pour représenter cette incertitude, modifiant l'évolution temporelle des paramètres.

3.7.1 Evolutions temporelles des réponses fréquentielles

La variation temporelle de la masse du lanceur, des conditions atmosphériques ainsi que du régime de l'écoulement avec le passage supersonique au cours de la trajectoire, entraînent une modification significative des caractéristiques fréquentielles du système en boucle ouverte.

L'étude du lieu des pôles pour le système flexible complet avec l'actionneur en série reprend les matrices $(\tilde{A}_F, \tilde{B}_F, \tilde{C}_F, \tilde{D}_F)$ de (3.17-3.18) en incorporant la dynamique de (3.27).

La prise en compte de l'évolution temporelle sera faite par l'évaluation du modèle d'état complet en six instants de la trajectoire, aux temps 25s, 32s, ... 60s. Les différentes dynamiques sont présentées en Fig. 3.11, sur laquelle on aperçoit d'une part la partie instable de la partie rigide, symétrique par rapport à l'origine, la dynamique oscillatoire des deux modes de flexion, et enfin les pulsations extrêmement rapides de l'actionneur.

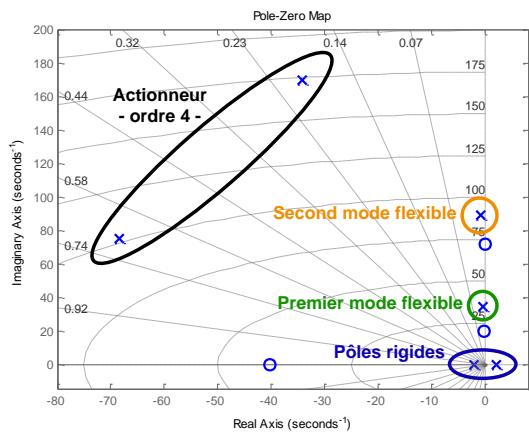


Figure 3.11 Lieu des racines du système complet

Pour chacun des modèles obtenus pour les six instants, les pôles et zéros de la partie lanceur flexible ont été tracés en Fig. 3.12, sachant que la dynamique de l'actionneur reste constante dans le temps.

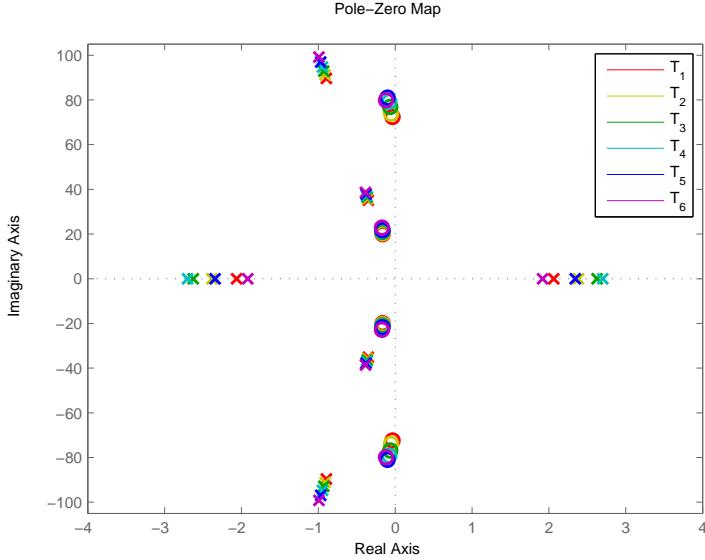


Figure 3.12 Evolution du lieu des racines avec le temps

La dynamique des modes flexibles évolue peu au vu de la position regroupée des pôles flexibles. Avec la diminution de la masse du lanceur, la pulsation des deux modes a tendance à augmenter, tandis que leur amortissement reste égal à 0.1.

La dynamique rigide quant à elle subit une variation liée directement à la pression dynamique. Il faut ici rappeler que la dynamique rigide est définie par le bloc matriciel A_r donné en (3.10) dont les pôles sont situés en $\pm\sqrt{\frac{QS_{ref}l_{GCC}C_{za}}{I}}$. La grandeur la plus variable relativement à sa valeur initiale est la pression dynamique Q . Ceci se traduit par une accélération des pôles rigides avec Q , jusqu'à la zone "Max-Q" au temps T_4 , avant un ralentissement progressif.

Ensuite, l'analyse du diagramme de Bode présenté en Fig. 3.13 permet de voir l'évolution de la réponse fréquentielle du système rigide avec le temps, basé sur le modèle (A_r, B_r, C_r, D_r) . Ce diagramme montre que seul le gain du système augmente au cours du temps, tandis que la phase est identique. La décroissance en basse fréquence est déclenchée plus ou moins tôt suivant la position des pôles rigides correspondants. La dynamique haute fréquence est due à l'actionneur et influence peu le gain mais dégrade la phase du système en introduisant un fort retard entre 100 et 1000 rad/s.

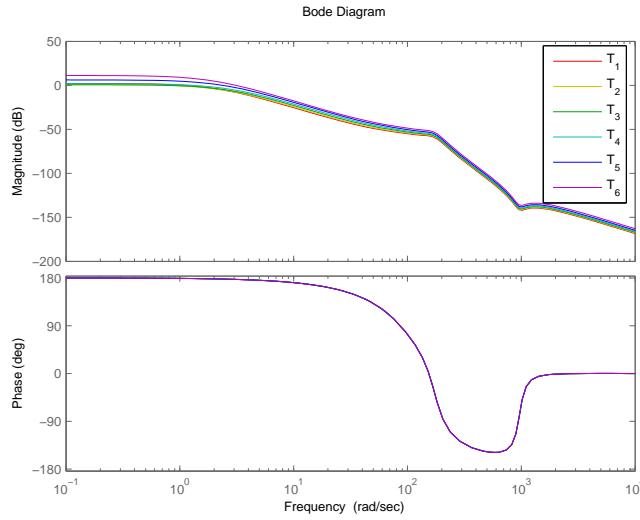


Figure 3.13 Evolution du diagramme de Bode avec le temps - Modèle rigide

La prise en compte des modes flexibles crée une dynamique intermédiaire entre celle du mode rigide et celle de l'actionneur comme l'illustre la Fig. 3.14. L'évolution de la réponse temporelle est cohérente avec la position des pôles : les pics de résonance et d'atténuation correspondant respectivement aux pôles et zéros flexibles sont translatés vers des fréquences plus élevées. Au niveau de la phase, deux avances de phase apparaissent entre 10 et 100 rad/s correspondants aux couples pôles/zéros précédents.

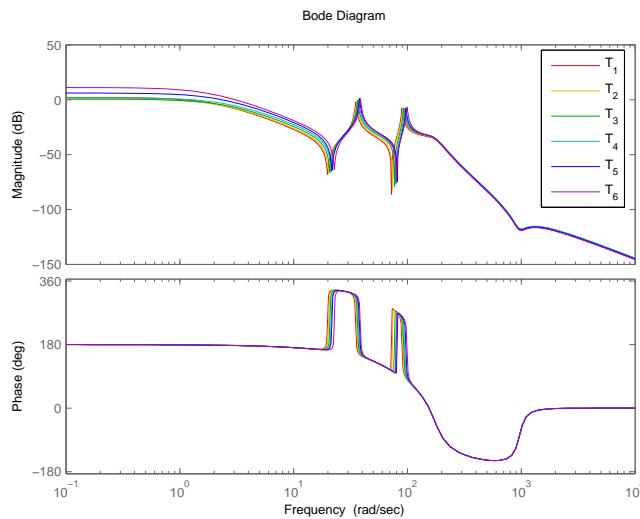


Figure 3.14 Evolution du diagramme de Bode avec le temps - Modèle flexible

3.7.2 Représentation des incertitudes

Description des modèles incertains

En plus de la dépendance temporelle des paramètres, il est important de considérer aussi des incertitudes de modèle. Il n'est en effet pas aisé de prévoir la courbe de poussée du moteur depuis le décollage, surtout quand des moteurs à propergols solides sont utilisés. Ces derniers ne permettent pas de contrôler la combustion une fois la réaction entamée, il est donc difficile d'imposer une poussée constante et régulière au cours du temps. Pour cette raison, ASTRIUM-ST considère deux modèles incertains par rapport au modèle nominal, pour traduire deux profils de poussée distincts.

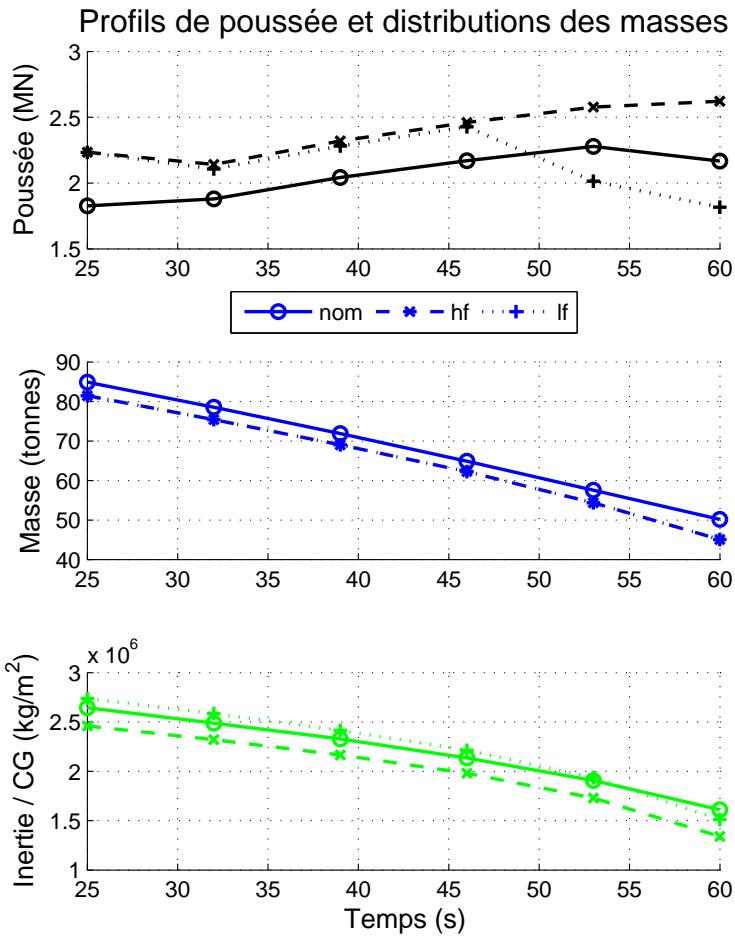


Figure 3.15 Evolution de la poussée et des masses avec les modèles incertains

Les trois modèles obtenus seront appelés **nom** pour le modèle nominal, **hf** pour le modèle avec un profil de poussée plus élevé et plus long (i.e. "high frequency"), et **lf** pour le dernier

modèle obtenu pour un profil de poussée plus élevé mais plus court (i.e. "low frequency"). Il est important de noter que les variations de la poussée modifient fortement la consommation et la répartition des ergols, et par conséquent l'inertie totale du lanceur. Ces résultats sont résumés en Fig. 3.15.

La répartition changeante des masses vient aussi modifier la position du centre de gravité du lanceur, et par conséquent les bras de leviers mis en jeu dans l'équation des moments (3.15). L'évolution temporelle des bras de levier des forces aérodynamiques et de poussée sont rassemblés en Fig. 3.16.

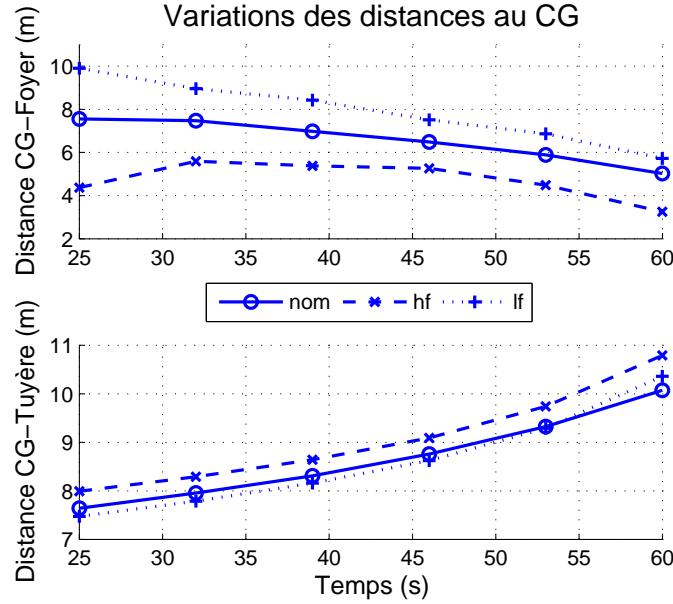


Figure 3.16 Evolution des bras de levier avec les modèles incertains

Le but du moteur étant d'accélérer le lanceur, les courbes de vitesses vont aussi se retrouver modifiées pour les modèles **hf** et **lf**. De plus, les conditions atmosphériques combinées avec ces vitesses peuvent accentuer fortement la pression dynamique Q , et enfin l'influence du Mach sur le coefficient de portance $C_{z\alpha}$ vient changer les propriétés aérodynamiques du lanceur. Toutes ces données sont résumées sur la Fig. 3.17.

Modifications des réponses fréquentielles

Finalement ces considérations sur les incertitudes paramétriques des modèles vont aussi affecter la réponse fréquentielle du modèle. Le lieu des pôles pour les trois modèles à l'instant initial a été tracé en Fig. 3.18. Les pôles flexibles ne sont pas affectés car les incertitudes

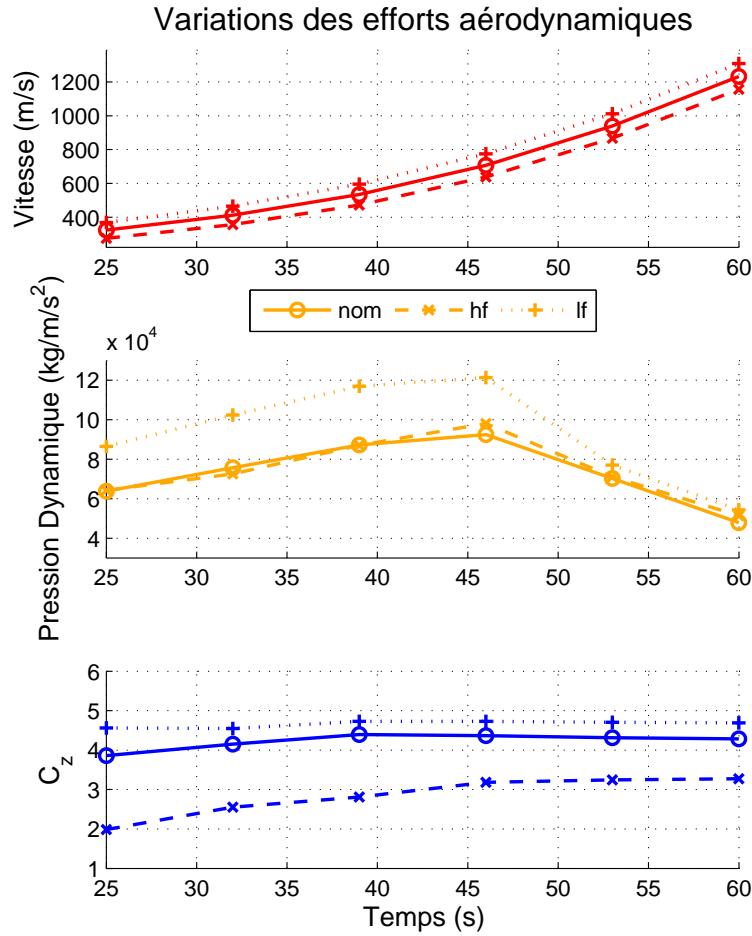


Figure 3.17 Evolution des paramètres aérodynamiques avec les modèles incertains

ne portent pas sur la structure du lanceur et donc sur ses modes de vibration. Le principal changement apparaît pour les modes rigides et pour les zéros proches du second mode flexible. Leurs évolutions sont principalement dues aux efforts aérodynamiques variables, comme décrit plus haut. Il en ressort un encadrement des pôles et zéros du modèle nominal **nom** par ceux des modèles **hf** et **lf**.

Pour finir, le diagramme de Bode des trois modèles est fourni en Fig. 3.19. Les principales modifications sont liées au changement du gain de chaque modèle et à la position des zéros proches des modes flexibles, qui entraînent une dérive de la fréquence de coupure. La phase, quant à elle, reste peu affectée par les modèles incertains.

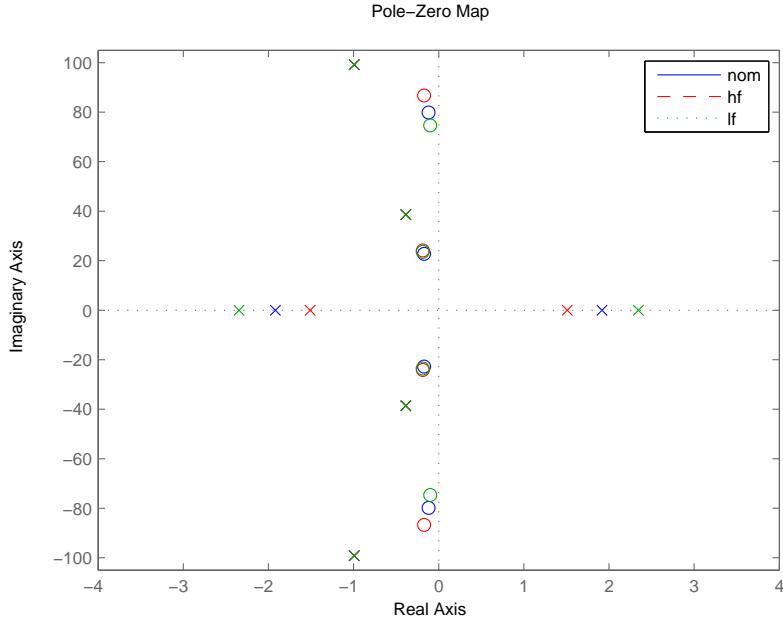


Figure 3.18 Evolution du lieu des racines avec les modèles incertains

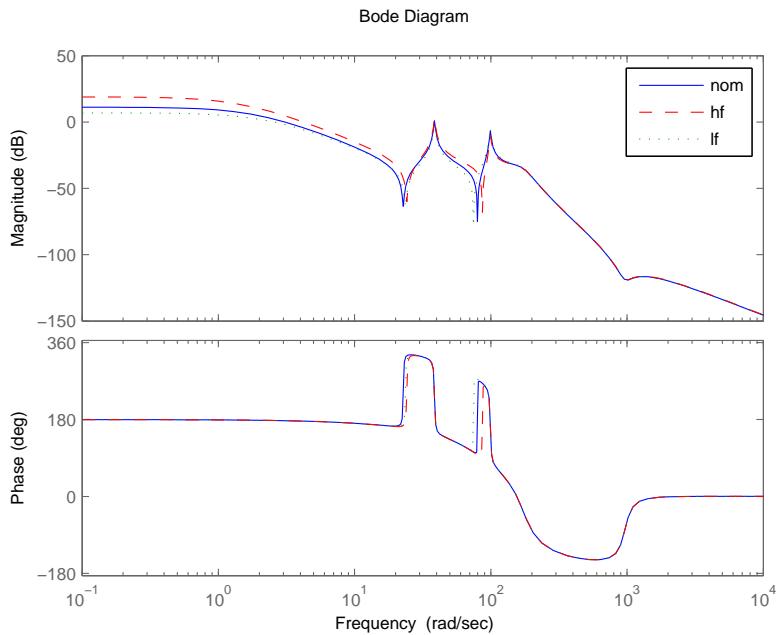


Figure 3.19 Evolution du diagramme de Bode avec les modèles incertains

3.8 Synthèse robuste et respect de performances

Le problème posé consiste désormais à réaliser une loi de commande qui puisse premièrement stabiliser le modèle nominal en présence d'un profil de vent donné. Ensuite cette stabilité doit être conservée pour les modèles incertains *hf* et *lf*; on qualifiera alors le correcteur de "robuste" aux incertitudes de modèle. Enfin des performances fixées par ASTRIUM-ST devront être respectées pendant et après la synthèse.

3.8.1 Structure de correction adoptée

Les contraintes de design se placent au niveau des entrées et sorties disponibles pour contrôler le lanceur. D'après le modèle d'état développé en (3.17-3.18) et mis en série avec l'actionneur, on extrait le modèle utilisé pour le contrôle, possédant une seule entrée β_c et une seule sortie θ . On obtient le modèle d'état présenté en (3.28-3.29) pour le vecteur d'état augmenté $X = \begin{bmatrix} X_F & X_{act} \end{bmatrix}^T$.

$$\dot{X} = A_F X + B_F \beta_c \quad (3.28)$$

$$\theta = C_F X + D_F \beta_c \quad (3.29)$$

Le correcteur synthétisé est ensuite placé en série avec le système complet, pour obtenir la structure de correction donnée en Fig. 3.20. La difficulté va être de contrôler l'angle α en ayant uniquement accès à θ par la centrale inertie du lanceur, et en agissant sur la commande β_c envoyée à l'actionneur de la tuyère.

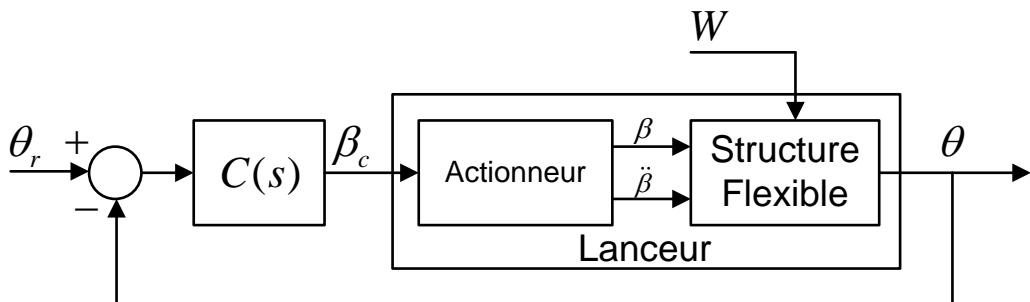


Figure 3.20 Structure de correction du lanceur flexible avec son actionneur

3.8.2 Cahier des charges

La contrainte critique dans le cas d'un lanceur correspond aux efforts latéraux. Un tel système est en effet bâti pour supporter les efforts en compression entre la force de poussée et le poids pendant le vol, mais il reste fragile en flexion. Il est donc primordial de maintenir les efforts latéraux, comme la portance, les plus faibles possibles. La limite fournie par ASTRIUM-ST est ici de $|\alpha| < 3^\circ$.

En plus de cette exigence fondamentale, le cahier des charges fourni peut se décomposer entre les contraintes dites de design, qui seront prises en compte pour synthétiser le correcteur, et les contraintes de certification qui seront validées *a posteriori*.

Contraintes de Design

Ces exigences sont exprimées sur la position des pôles du système en boucle fermée. Elles s'attachent à limiter les oscillations sur l'attitude du lanceur (limite en amortissement), à obtenir un temps de réponse suffisamment rapide (limite sur la partie réelle) et enfin à être robuste aux incertitudes de modèle.

Ces contraintes se déclinent comme suit :

- **Pôles de la partie rigide p_r**
 - Partie réelle maximale $Re(p_r) \leq -0.5$
 - Amortissement minimal $\xi(p_r) \geq 0.5$ sur le cas **nom**, et $\xi(p_r) \geq 0.15$ sur les cas **hf** et **1f**

- **Pôles de la partie flexible p_f**
 - Partie réelle maximale $Re(p_f) \leq -0.2$ pour assurer la stabilité

Contraintes de certification

Le reste du cahier des charges concerne des données qui seront vérifiées une fois le correcteur synthétisé. Il est ainsi demandé de :

- Limiter l'incidence avec $|\alpha| \leq 3^\circ$ pour un profil de vent extrême de 20 m/s fourni en Fig. 3.3
- Respecter une marge de gain de $M_G \geq 3dB$ pour les cas **nom** et **hf**, et de $M_G \geq 1dB$ pour le cas **1f**
- Assurer une marge de retard de $M_R \geq 40ms$

La dernière contrainte doit permettre de réaliser un échantillonnage du système à $T_s = 27ms$ pour une implantation numérique. Cette marge traduit le retard maximal que peut tolérer le système bouclé discréétisé avant de se déstabiliser.

Ces contraintes-là seront vérifiées par des simulations fournies par ASTRIUM-ST.

3.9 Simulation et validation

Les contraintes de design seront assurées directement par la méthode de synthèse utilisée dans ce projet : les applications gardiennes. Les contraintes de certification sont quant à elles évaluées sur une simulation du lanceur complet avec la structure de correction en réponse à un profil de vent donné.

Cette simulation reprend les six modèles linéaires et constants correspondant à l'état du système pour six instants de la trajectoire. Une interpolation permet alors de trouver des polynômes fonction du temps qui interpolent exactement chacun des coefficients variants du modèle matriciel (A_F, B_F, C_F, D_F). Il est alors possible de mettre le système sous forme LFT en considérant le temps comme une variable incertaine (Bérard *et al.*, 2012). Le bloc Δ de la forme standard habituelle est donc fonction du temps, et peut ainsi être mis à jour par une horloge pour faire varier les paramètres du système. L'intérêt d'une telle représentation vient des outils déjà développés sous Matlab (Magni, 2002; Biannic *et al.*, 2006), qui permettent une simulation rapide malgré un système d'ordre 20 une fois la structure de correction implantée.

Un exemple de courbe obtenue pour la réponse à un échelon de vent est fourni en Fig. 3.21. Les sorties de simulation comprennent l'incidence α , l'attitude θ et la commande β_c .

Enfin, les contraintes en marges sont vérifiées par la fonction `allmargin` de Matlab appliquée à la chaîne directe constituée du correcteur et du lanceur complet. Un exemple de courbes obtenues est donné en Fig. 3.22. Les marges du cahier des charges sont tracées suivant les modèles considérés.

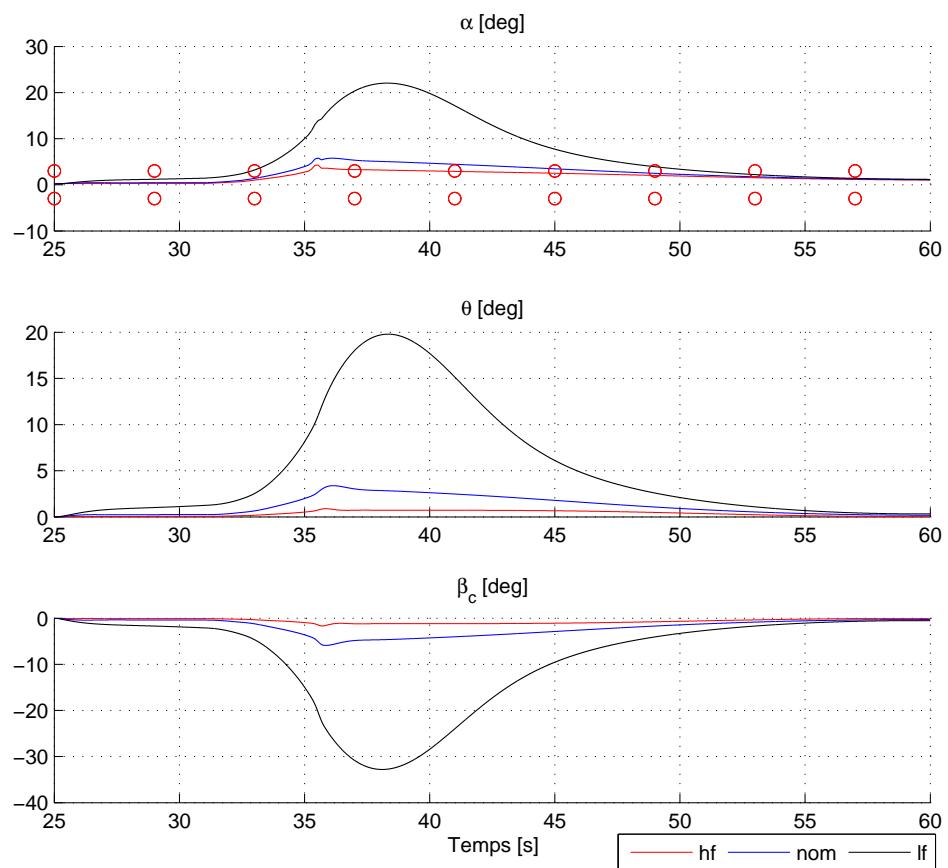


Figure 3.21 Exemple de simulation à une rafale de vent

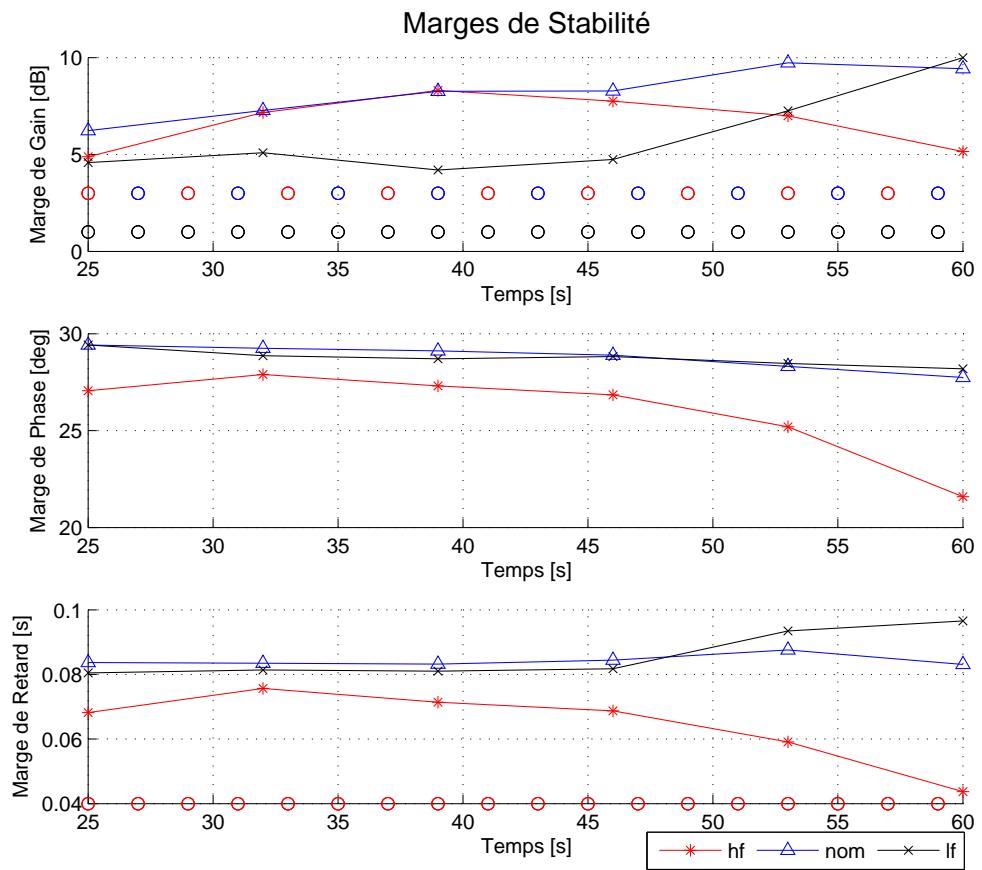


Figure 3.22 Exemple de marges obtenues pour le système corrigé

CHAPITRE 4

APPLICATIONS GARDIENNES

4.1 Introduction

Les applications gardiennes constituent un outil pour l'étude de la stabilité de matrices paramétrées (resp. polynômes paramétrés). Le terme stabilité est ici à prendre au sens général en considérant le confinement des valeurs propres (resp. racines) dans un sous-ensemble *ouvert* Ω du plan complexe : on parle alors de Ω -stabilité. Le demi-plan gauche correspond ainsi à rechercher la stabilité Hurwitz, et le disque unitaire centré à l'origine à rechercher la stabilité Schur. Avec des ensembles plus élaborés il est possible de confiner les pôles avec des limites en amortissement ou en pulsation. Dès lors, les applications gardiennes interviennent comme un outil d'analyse de performance robuste si les paramètres sont incertains, ou de synthèse de correcteur si les paramètres sont des gains à régler.

Avec les travaux initiés par Kharitonov (1981), un premier pas a été réalisé dans l'analyse de la stabilité de familles de matrices (resp. polynômes) paramétrées. Ces résultats émergeront quelques années plus tard dans les centres de recherche occidentaux, pour aboutir au *Edge Theorem* avec Bartlett *et al.* (1988). Enfin, les applications gardiennes viennent généraliser cette approche avec les travaux de Saydy *et al.* (1990) pour l'étude de la Ω -stabilité de familles paramétrées de matrices (resp. polynômes). Si on note \mathcal{M}_n l'ensemble des matrices carrées réelles de taille n , une application gardienne est en fait une fonction scalaire $\nu_\Omega : \mathcal{M}_n \rightarrow \mathbb{R}$ qui s'annule quand une des valeurs propres de la matrice touche la frontière de l'ensemble Ω . En visualisant les lieux d'annulation de cette fonction, il est alors possible de déterminer pour quelles valeurs des paramètres la famille de matrices (resp. polynômes) reste stable relativement à Ω .

Un résumé des méthodes de commande robuste est réalisé dans le livre Barmish (1994) à travers de nombreux exemples d'application. Une introduction aux applications gardiennes y est ainsi disponible.

Avant de pousser plus loin la présentation des applications gardiennes, nous allons poser la définition du produit bialterné, outil indispensable pour les exprimer de façon concise. Les principaux résultats théoriques seront énoncés avant d'être appliqués à un cas d'analyse robuste, puis à un cas de synthèse de correcteur. Enfin, les applications gardiennes intervenant dans la loi de commande du lanceur seront exprimées, afin de respecter le cahier des charges.

4.2 Outils Mathématiques

La construction des applications gardiennes repose sur des produits et sommes de matrices à partir des produits de Kronecker, de Schläflian (Aitken, 1928), ou encore du produit bialterné. Chacun de ces produits possède des propriétés remarquables quant aux valeurs propres des matrices résultantes. Pour des questions de simplicité, seuls le produit de Kronecker et le produit bialterné seront présentés en raison de leur similarité.

4.2.1 Produit de Kronecker

Le produit de Kronecker de deux matrices $A \in \mathbb{R}^{m \times n}$ et $B \in \mathbb{R}^{p \times q}$, noté $A \otimes B$, est défini par la matrice de taille $mp \times nq$ définie par bloc. Le ij -ème bloc matriciel est donné par $a_{ij}B$. On obtient la forme suivante :

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix}$$

La somme de Kronecker, notée $A \oplus B$, est alors définie pour des matrices carrées $A \in \mathbb{R}^{n \times n}$ et $B \in \mathbb{R}^{m \times m}$ par la matrice carrée de taille nm suivante : $A \oplus B = A \otimes I_n + I_m \otimes B$.

Ce produit et cette somme présentent des propriétés remarquables au niveau de leurs valeurs propres $\lambda_i(A \otimes B)$ et $\lambda_i(A \oplus B)$. Il est en effet bien connu que les valeurs propres d'un produit matriciel ne sont pas toujours les produits des valeurs propres, à moins que les deux matrices ne soient simultanément diagonalisables : $\lambda_i(AB) \neq \lambda_i(A)\lambda_i(B)$. Il en va de même pour la somme de matrice. L'intérêt du produit et de la somme de Kronecker réside dans le lemme suivant, présenté par Stephanos (1900) :

Lemme 4.2.1

Soit p le polynôme matriciel à coefficients complexes défini pour $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times m}$ par

$$p(A, B) = \sum_{i,j=0}^{i+j=N} p_{ij} A^i \otimes B^j$$

Alors les nm valeurs propres de $p(A, B)$ sont données par

$$\lambda_{ij}(p(A, B)) = p(\lambda_i(A), \lambda_j(B)) \quad \forall i = 1, \dots, n, j = 1, \dots, m$$

Concrètement ce lemme donne le résultat suivant pour le produit et la somme de Kronecker : les valeurs propres $\lambda_{ij}(A \otimes B)$ sont les nm produits $\lambda_i(A)\lambda_j(B)$, et les valeurs propres $\lambda_{ij}(A \oplus B)$ sont les nm sommes $\lambda_i(A) + \lambda_j(B)$, $i = 1, \dots, n$, $j = 1, \dots, m$.

4.2.2 Produit Bialterné

Un désavantage du produit de Kronecker est la répétition des valeurs propres pour le produit $A \otimes A$ avec par exemple la présence double de la valeur propre $\lambda_i(A)\lambda_j(A)$ et $\lambda_j(A)\lambda_i(A)$. Ce problème va être réglé par l'utilisation du produit bialterné qui conserve des propriétés semblables aux dépens d'une définition plus élaborée.

Définitions

Le produit bialterné est défini pour deux matrices carrées A et B de taille n par une matrice carrée $A \odot B$ de taille $\frac{n(n-1)}{2}$. Sa définition repose sur un vecteur particulier, que nous noterons V , et sur une fonction f de $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$.

V représente le vecteur des couples (p, q) classés dans l'ordre lexicographique, avec $p = 2, \dots, n$ et $q = 1, \dots, p - 1$. Ce vecteur de taille $2 \times \frac{n(n-1)}{2}$ prend la forme suivante :

$$V^n = \left[\begin{pmatrix} 2 \\ 1 \end{pmatrix} \begin{pmatrix} 3 \\ 1 \end{pmatrix} \begin{pmatrix} 3 \\ 2 \end{pmatrix} \cdots \begin{pmatrix} n \\ n-1 \end{pmatrix} \right]$$

La fonction f est définie par l'expression suivante :

$$f_{A,B}((p, q), (r, s)) = \frac{1}{2} \left(\det \begin{bmatrix} a_{pr} & a_{ps} \\ b_{qr} & b_{qs} \end{bmatrix} + \det \begin{bmatrix} b_{pr} & b_{ps} \\ a_{qr} & a_{qs} \end{bmatrix} \right)$$

A l'aide du vecteur V et de la fonction f , le produit bialterné se définit comme suit.

Définition 4.2.1

$A \odot B$ représente la matrice carrée de taille $\frac{n(n-1)}{2}$ dont le terme général s'exprime par :

$$(A \odot B)_{(i,j)} = f_{A,B}(V_i^n, V_j^n) \quad i, j = 1, \dots, \frac{n(n-1)}{2}$$

Propriétés

Ce produit, semblable au produit de Kronecker, possède des propriétés spécifiques quant aux valeurs propres de la matrice produit. Il permet d'éviter les répétitions de valeurs propres

quand on effectue par exemple le produit de Kronecker d'une matrice avec elle-même ou avec l'identité. En s'appuyant sur les résultats présentés dans Stephanos (1900), un résultat puissant peut être énoncé sur les valeurs propres du produit bialterné, à mettre en parallèle avec le lemme 4.2.1.

Théorème 4.1

Soit Ψ le polynôme matriciel défini pour deux matrices carrées A, B de taille n

$$\Psi(A, B) = \sum_{k,l=0}^{k+l=N} \psi_{kl} A^k \odot B^l$$

Alors les valeurs propres de $\Psi(A, A)$ sont les $\frac{n(n-1)}{2}$ valeurs :

$$\lambda_{ij}(\Psi(A, A)) = \frac{1}{2} \sum_{k,l=1}^{k+l=N} \psi_{kl} (\lambda_i^k \lambda_j^l + \lambda_j^k \lambda_i^l) \quad i = 2, \dots, n, j = 1, \dots, i-1$$

On en déduit par exemple les résultats fondamentaux suivants :

Lemme 4.2.2

Soit A une matrice carrée de taille n , le spectre de $A \odot A$ est donné par les $\frac{n(n-1)}{2}$ valeurs :

$$\lambda_{ij}(A \odot A) = \lambda_i(A) \lambda_j(A) \quad i = 2, \dots, n, j = 1, \dots, i-1$$

Lemme 4.2.3

Soit A une matrice carrée de taille n , le spectre de $2A \odot I$ est donné par les $\frac{n(n-1)}{2}$ valeurs :

$$\lambda_{ij}(2A \odot I) = \lambda_i(A) + \lambda_j(A) \quad i = 2, \dots, n, j = 1, \dots, i-1$$

Ensuite, quelques relations utiles pour la suite sont énoncées à partir de Stephanos (1900). Elles présentent quelques similarités avec le produit de Kronecker. Cependant, il est à noter que le produit bialterné est commutatif, contrairement au produit de Kronecker.

Résultat 4.2.1

Soit A, B et C des matrices carrées de $\mathbb{R}^{n \times n}$, et $k \in \mathbb{R}$:

$$\begin{aligned} A \odot B &= B \odot A \\ A \odot (kB) &= k(A \odot B) \\ A \odot (B + C) &= A \odot B + A \odot C \\ (A \odot A)(B \odot B) &= (AB) \odot (AB) \end{aligned}$$

De même, avec P une matrice inversible de taille n :

$$\begin{aligned} (P \odot P)^{-1} &= P^{-1} \odot P^{-1} \\ (P^{-1}AP) \odot (P^{-1}BP) &= (P^{-1} \odot P^{-1})(A \odot B)(P \odot P) \\ 2(P^{-1}AP) \odot I_n &= (P^{-1} \odot P^{-1})(2A \odot I_n)(P \odot P) \end{aligned}$$

Les principaux résultats liés au produit bialterné ont été donnés dans ce chapitre afin d'aborder la définition puis la construction des applications gardiennes et semi-gardiennes.

4.3 Définitions et applications gardiennes usuelles

Tous les résultats et définitions énoncés ci-après vont se concentrer sur les applications gardiennes appliquées à des matrices, mais peuvent aisément être étendus à l'étude des polynômes. Pour plus de précisions, voir les travaux Saydy *et al.* (1990, 1988).

4.3.1 Définitions

Une application gardienne se définit toujours par rapport à un ensemble ouvert Ω du plan complexe, dans lequel les valeurs propres d'un système dynamique doivent être confinées. $\mathcal{S}(\Omega)$ désignera l'ensemble des matrices qui respectent ce critère, appelé ensemble de stabilité généralisée (Saussié, 2010). En notant σ le spectre de la matrice, la définition suivante de $\mathcal{S}(\Omega)$ devient :

Définition 4.3.1

$$\mathcal{S}(\Omega) = \{A \in \mathbb{R}^{n \times n} \mid \sigma(A) \subset \Omega\}$$

Avec cette notation, la définition d'une application gardienne peut être donnée, telle qu'elle apparaît dans Saydy *et al.* (1990). Dans la suite, $\overline{\Omega}$, $\partial\Omega$ et $\overset{\circ}{\Omega}$ représentent respectivement l'adhérence, la frontière et l'intérieur d'un ensemble Ω , tels que $\overline{\Omega} = \partial\Omega \cup \overset{\circ}{\Omega}$.

Définition 4.3.2

Soit $\mathbb{R}^{n \times n}$ l'ensemble des matrices carrées réelles de taille n , et soit \mathcal{S} un sous-ensemble ouvert de $\mathbb{R}^{n \times n}$. Soit ν une application de $\mathbb{R}^{n \times n} \rightarrow \mathbb{C}$. ν garde \mathcal{S} si :

$$\forall x \in \overline{\mathcal{S}}, \quad \nu(x) = 0 \iff x \in \partial\mathcal{S}$$

ν est alors appelée *application gardienne* pour \mathcal{S} .

La définition d'une application semi-gardienne diffère légèrement pour traiter un classe plus générale de fonctions.

Définition 4.3.3

Soit \mathcal{S} et ν définis précédemment. ν est une application semi-gardienne pour \mathcal{S} si :

$$\forall x \in \overline{\mathcal{S}} \quad x \in \partial\mathcal{S} \implies \nu(x) = 0$$

Un élément $x \in \overset{\circ}{\mathcal{S}} = \mathcal{S}$, pour lequel $\nu(x) = 0$ est appelé *blind spot* pour (ν, \mathcal{S}) .

4.3.2 Applications gardiennes usuelles

Les ensembles de confinement fondamentaux vont être passés en revue dans cette partie. Pour chacun d'entre eux, une application gardienne sera donnée.

Stabilité Hurwitz La stabilité Hurwitz reste la première des contraintes à satisfaire lors de la synthèse d'une loi de commande dans le domaine continu. Il est primordial de placer les pôles du système rebouclé dans le demi-plan gauche ouvert. Pour ce faire, l'application (4.1) garde l'ensemble $\mathcal{S}(\Omega)$ pour $\Omega = \overset{\circ}{\mathbb{C}}_-$, présenté en Fig. 4.1.

$$\nu_H(A) = \det(A) \det(A \odot I) \tag{4.1}$$

Cette application gardienne s'assure qu'aucune valeur propre de A ne se situe sur l'axe imaginaire, signe d'instabilité. Elle s'annule si un des deux termes s'annule. Le premier déterminant vaut 0 si un pôle réel est au niveau de l'origine. Le second s'annule uniquement en présence de pôles imaginaires purs.

A titre d'exemple, une autre application gardienne pour $\mathcal{S}(\Omega)$ s'obtient en utilisant le produit de Kronecker : $\nu_H(A) = \det(A \oplus A)$.

α -stabilité Il est souvent utile d'introduire une marge de stabilité α par rapport à l'axe imaginaire, que ce soit pour assurer la stabilité du système en présence d'incertitudes ou pour garantir une performance sur le temps de réponse lors de la synthèse. Une application gardienne pour un tel ensemble est obtenu à partir de l'application précédente en ajoutant une *translation* sur les pôles. L'application (4.2) garde $\mathcal{S}(\Omega)$ pour $\Omega = \{z \in \mathbb{C}; \operatorname{Re}(z) < \alpha\}$ présenté en Fig. 4.1.

$$\nu_\alpha(A) = \det(A - \alpha I) \det((A - \alpha I) \odot I) \quad (4.2)$$

Le rôle de chaque déterminant est le même que précédemment appliqué cette fois-ci avec l'axe $\operatorname{Re}(z) = \alpha$ au lieu de l'axe imaginaire. Une application semi-gardienne basée sur le produit de Kronecker est donnée par : $\nu_\alpha(A) = \det(\frac{1}{2}A \oplus A - \alpha I)$.

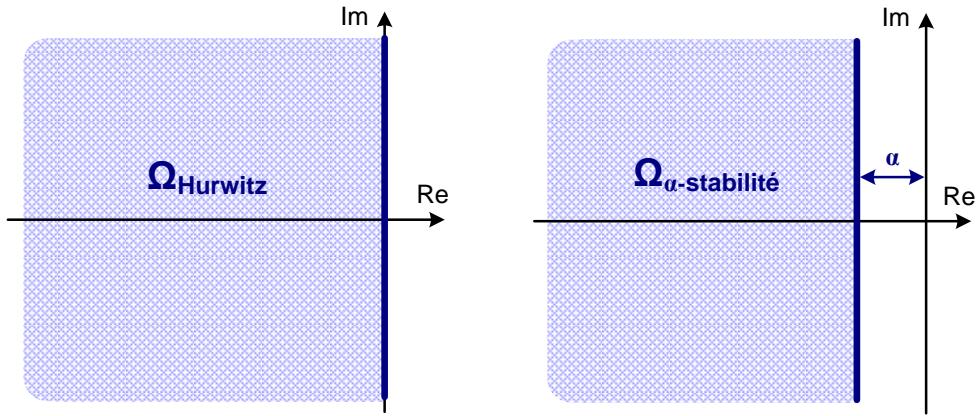


Figure 4.1 Zone de confinement pour la stabilité Hurwitz et la α -stabilité

Stabilité Schur Pour le cas discret, les pôles du système rebouclé doivent être situés dans le disque unité ouvert pour assurer la stabilité. Considérant alors le domaine $\mathcal{S}(\Omega)$ pour $\Omega = \{z \in \mathbb{C}; |z| < \omega\}$ donné pour un cas plus général de disque de rayon ω , une application gardienne est donnée par (4.3). Ω est illustré par Fig. 4.2.

$$\nu_S(A) = \det(A + \omega I) \det(A - \omega I) \det(A \odot A - \omega^2 I \odot I) \quad (4.3)$$

Cette application gardienne s'annule si une valeur propre de A vient annuler un des trois déterminants. Les deux premiers valent 0 respectivement si un pôle se situe sur une intersection de $\partial\Omega$ avec l'axe réel : $(+\omega, 0)$ ou $(-\omega, 0)$. Le dernier déterminant s'annule si des pôles complexes conjugués sont sur le cercle de rayon ω .

Le produit de Kronecker fournit une autre application gardienne pour $\mathcal{S}(\Omega)$, donnée par

$$\nu_S(A) = \det(A \otimes A - \omega^2 I)$$

ξ -stabilité Il est aussi commun de considérer une limite en amortissement sur les pôles d'un système afin d'atténuer les oscillations de la réponse temporelle. Dans ce cas, l'ensemble considéré est $\Omega = \{z \in \mathbb{C}; |\pi - \angle(z)| < \theta\}$ pour confiner dans le cône d'angle 2θ illustré en Fig. 4.2. L'amortissement d'un tel cône est $\xi = \cos(\theta)$. Une application gardienne pour $\mathcal{S}(\Omega)$ est donnée par (4.4).

$$\nu_\xi(A) = \det(A) \det (A^2 \odot I + (1 - 2\xi^2)A \odot A) \quad (4.4)$$

Cette application s'annule quand une valeur propre se place à l'origine, avec le premier déterminant, ou quand une paire de pôles complexes conjugués se situe sur les droites qui délimitent le cône d'amortissement ξ .

Le produit de Kronecker fournit l'application semi-gardienne

$$\nu_\xi(A) = \det(\frac{1}{2}A \oplus A + (1 - 2\xi^2)A \otimes A)$$

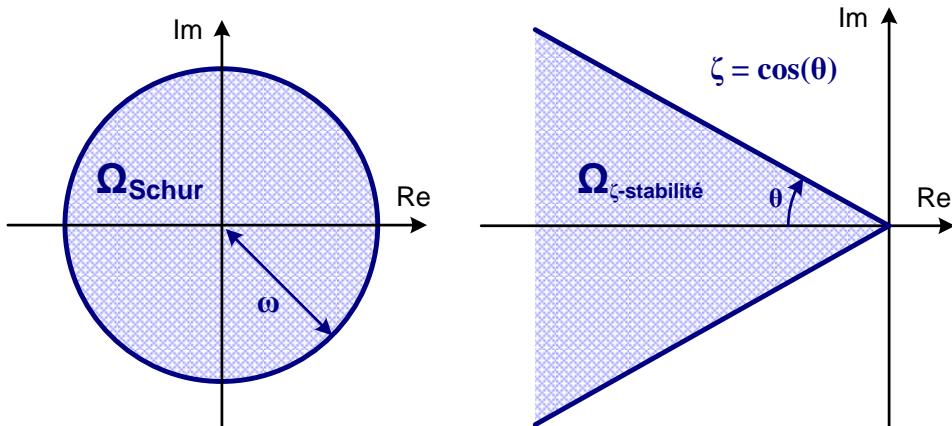


Figure 4.2 Zones de confinement pour la stabilité Schur et la ξ -stabilité

4.3.3 Construction d'une application (semi-)gardienne

Une procédure de construction systématique d'une application semi-gardienne est donnée dans Saydy *et al.* (1990). Il existe de plus des conditions supplémentaires à vérifier sur l'ensemble considéré pour s'assurer que l'application est gardienne. Par souci de clarté, la procédure sera décrite pour la construction des applications semi-gardiennes uniquement ; le lecteur curieux pourra parcourir Saydy *et al.* (1990) pour plus de détails.

La procédure s'applique pour tout ensemble dont la frontière est décrite par un polynôme des variables x et y du plan complexe, comme illustré en Fig. 4.3.

$$\Omega = \{x + iy \in \mathbb{C}; p(x, y) < 0\}$$

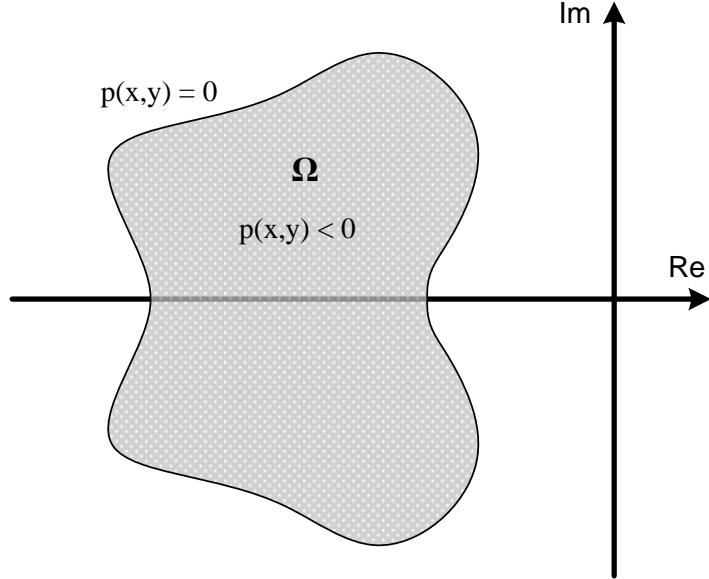


Figure 4.3 Frontière polynomiale de l'ensemble Ω

Dès lors, on s'attardera sur des polynômes de la forme (4.5) pour s'assurer que l'on traite des matrices réelles, donc des ensembles Ω symétriques par rapport à l'axe réel. Cette considération est assurée par des puissances paires sur la partie imaginaire y .

$$p(x, y) = \sum_{k,l=0}^{k+2l=N} p_{kl} x^k y^{2l} \quad (4.5)$$

Le polynôme q suivant est ensuite considéré :

$$q(\lambda, \lambda^*) = p\left(\frac{\lambda + \lambda^*}{2}, \frac{\lambda - \lambda^*}{2i}\right)$$

Il peut se réécrire sous la forme (4.6) avec des coefficients réels grâce aux puissances paires dans (4.5).

$$q(\lambda, \lambda^*) = \sum_{k,l=0}^{k+l=N} q_{kl} \lambda^k (\lambda^*)^l \quad (4.6)$$

Ce polynôme q permet d'exprimer l'ensemble Ω de façon alternative comme :

$$\Omega = \{\lambda \in \mathbb{C}; q(\lambda, \lambda^*) < 0\}$$

Il est à la base de la formulation d'une application semi-gardienne pour $\mathcal{S}(\Omega)$. Les deux résultats suivants sont prouvés dans Saydy *et al.* (1990) et permettent par exemple de retrouver toutes les applications gardiennes et semi-gardiennes énoncées plus haut, pour le produit de Kronecker ou pour le produit bialterné.

Proposition 4.3.1

En admettant qu'elle n'est pas identiquement nulle, l'application :

$$\nu : A \mapsto \det \left(\sum_{k,l=0}^{k+l=N} q_{kl} A^k \otimes A^l \right)$$

est semi-gardienne pour $\mathcal{S}(\Omega)$.

La condition de *transformabilité* Ω , énoncée dans Saydy *et al.* (1990), permet de vérifier s'il s'agit effectivement d'une application gardienne.

Le second résultat est énoncé pour les applications gardiennes basées sur le produit bialterné, qui conduisent à des matrices plus petites de taille $\frac{n(n-1)}{2}$ au lieu de n^2 avec le produit de Kronecker. Le gain en temps de calcul pour le calcul du déterminant peut s'avérer décisif.

Cette proposition s'appuie sur l'intersection de la frontière de l'ensemble avec l'axe réel : $\partial\Omega \cap \mathbb{R} = \prod_{i=1}^m [\alpha_i, \beta_i]$. On supposera de plus que $\alpha_1 \leq \beta_1 < \alpha_2 \leq \dots < \alpha_m \leq \beta_m$. On note alors l'application matricielle $\mathcal{P}(A) = \prod_{i=1}^m (A - \alpha_i I)(A - \beta_i I)$ pour lequel les termes $(A - \alpha_1 I)$ et $(A - \beta_m I)$ sont omis si $\alpha_1 = -\infty$ et $\beta_m = +\infty$. De plus, si cette intersection est vide, on prendra $\mathcal{P}(A) = 1$.

Proposition 4.3.2

En admettant qu'elle n'est pas identiquement nulle, l'application :

$$\nu : A \mapsto \det(\mathcal{P}(A)) \det \left(\sum_{k,l=0}^{k+l=N} q_{kl} A^k \odot A^l \right)$$

est semi-gardienne pour $\mathcal{S}(\Omega)$.

Un dernier résultat énoncé dans Saydy *et al.* (1990) sera utilisé dans la synthèse de la structure de correction du lanceur. Il s'agit de combiner les applications gardiennes usuelles présentées plus haut, pour obtenir une application gardienne sur un ensemble Ω plus complexe.

Supposons que Ω se décompose comme suit $\Omega = \Omega_1 \cap \Omega_2$, alors on obtient dans l'espace des matrices $\mathcal{S}(\Omega) = \mathcal{S}(\Omega_1) \cap \mathcal{S}(\Omega_2)$. Il est alors admis que :

Lemme 4.3.1

Si $\mathcal{S}(\Omega_1)$ ***est gardé par*** ν_1 ***et*** $\mathcal{S}(\Omega_2)$ ***par*** ν_2 ,

Alors $\mathcal{S}(\Omega)$ ***est gardé par*** $\nu : A \mapsto \nu_1(A)\nu_2(A)$.

Ce lemme s'applique sur l'ensemble Ω présenté en Fig. 4.4 comme l'intersection de deux sous-ensembles Ω_1 et Ω_2 . D'après les résultats précédents, $\nu_1(A) = \det(A - \alpha I) \det((A - \alpha I) \odot I)$ est une application gardienne pour \mathcal{S}_{Ω_1} , et $\nu_2(A) = \det(A) \det(A^2 \odot I + (1 - 2\xi^2)A \odot A)$ avec $\xi = \cos(\theta)$ en est une pour \mathcal{S}_{Ω_2} . On en déduit donc que \mathcal{S}_Ω est gardé par

$$\nu(A) = \det(A - \alpha I) \det((A - \alpha I) \odot I) \times \det(A) \det(A^2 \odot I + (1 - 2\xi^2)A \odot A)$$

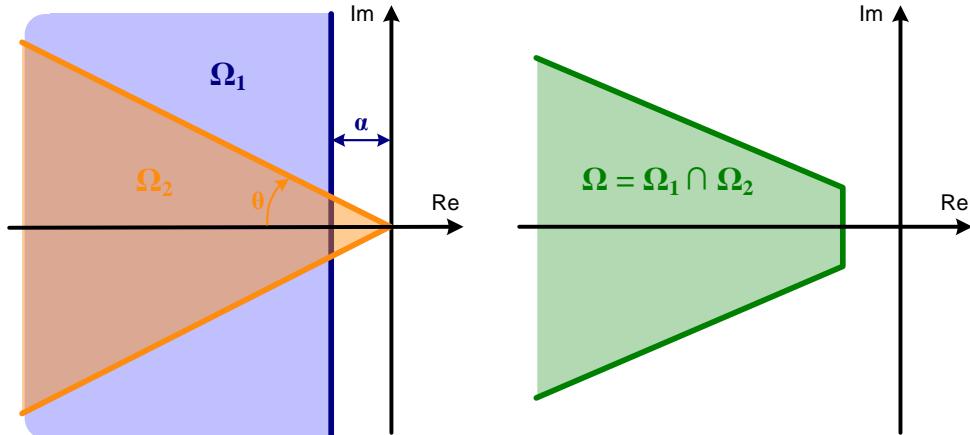


Figure 4.4 Décomposition d'un ensemble Ω en sous-ensembles basiques

4.4 Principaux résultats

Ces applications sont destinées à l'étude de la stabilité généralisée de familles de matrices (resp. de polynômes) paramétrées. Elles permettent de vérifier si les valeurs propres d'une matrice restent dans un ensemble Ω du plan complexe tandis que des paramètres de la matrice varient. On parle de Ω -stabilité pour une famille $\mathcal{A} = \{A(r) \in \mathbb{R}^{n \times n}; r \in \mathcal{U}\}$.

4.4.1 Cas général

La matrice $A(r)$ est paramétrée de façon **continue** par un vecteur $r = (r_1, r_2, \dots, r_k) \in \mathcal{U} \subset \mathbb{R}^k$, avec \mathcal{U} un ensemble **connexe**. Le but est de savoir si la matrice $A(r)$ reste confinée dans l'ensemble ouvert $\mathcal{S} \subset \mathcal{M}_n$ quand r parcourt \mathcal{U} .

Lemme 4.4.1

Soit \mathcal{S} un sous ensemble ouvert de $\mathbb{R}^{n \times n}$. soit ν une application gardienne de \mathcal{S} . On suppose que $\exists r_0 \in \mathcal{U}$ tel que $A(r_0) \in \mathcal{S}$. Alors :

$$\forall r \in \mathcal{U}, A(r) \in \mathcal{S} \iff \forall r \in \mathcal{U}, \nu(A(r)) \neq 0$$

De la même façon, ce lemme s'étend au cas des applications semi-gardiennes pour obtenir la version suivante :

Lemme 4.4.2

Soit \mathcal{S} et ν définis précédemment. On suppose que $\exists r_0 \in \mathcal{U}$ tel que $A(r_0) \in \mathcal{S}$. En notant $\mathcal{U}_{cr} = \{r \in \mathcal{U} \mid \nu(A(r)) = 0\}$ l'ensemble des points d'annulation de la famille de matrices, on obtient :

$$\forall r \in \mathcal{U}, A(r) \in \mathcal{S} \iff \forall r \in \mathcal{U}_{cr}, A(r) \in \mathcal{S}$$

Dans cette seconde version, il s'agit de vérifier que les points d'annulation de l'application gardienne sont en fait des *blind spots* et non pas des points à la frontière de $\mathcal{S}(\Omega)$, auquel cas la matrice $A(r)$ perd la Ω -stabilité.

Enfin, un dernier résultat vient compléter ces lemmes. Il s'agit cette fois de considérer le domaine \mathcal{U} dans lequel évolue le paramètre de la famille \mathcal{A} . Avec les mêmes hypothèses sur l'ensemble \mathcal{U} connexe et sur la continuité de $A(r)$, on obtient le lemme suivant grâce à Saussié *et al.* (2010) :

Lemme 4.4.3

Soit $\mathcal{S}(\Omega)$ gardé par l'application ν_Ω et la famille $\{A(r); r \in \mathcal{U}\}$. Alors l'ensemble \mathcal{C} défini par :

$$\mathcal{C} = \{r \in \mathcal{U}; \nu_\Omega(A(r)) = 0\}$$

divise l'espace des paramètres en composantes \mathcal{C}_i qui sont stable ou instable par rapport à Ω .

De plus, la stabilité d'une seule matrice $A(r_i)$ avec $r_i \in \mathcal{C}_i$ suffit à conclure sur la stabilité de toute la composante \mathcal{C}_i par rapport à Ω .

Ce dernier lemme permet de tirer un résultat puissant sur les familles monoparamétriques, exposé dans la partie suivante. Le point important est le fait que la stabilité d'une composante toute entière est déterminée par un seul point à l'intérieur de celle-ci. De plus, le nombre de pôles qui respectent ou violent la Ω -stabilité reste le même dans toute la composante.

4.4.2 Cas de familles monoparamétriques

La stabilité d'une famille de matrices dépendant d'un seul paramètre r et paramétrées de façon polynomiale est explorée à travers les résultats présentés dans Saussié *et al.* (2010). Il est donc supposé que :

$$\exists A_0, A_1, \dots, A_n \in \mathbb{R}^{n \times n}; \quad A(r) = A_0 + r A_1 + r^2 A_2 + \cdots + r^k A_k$$

Dans ce cas, d'après les propriétés du déterminant, on obtient que l'application gardienne pour un domaine Ω à frontière polynomiale est aussi polynomiale en r . Partant d'un point r_0 pour lequel la matrice $A(r_0)$ est stable par rapport à Ω , le but est de trouver le plus grand intervalle ouvert I pour r tel que $A(r)$ reste stable. On définit alors :

$$\begin{aligned} r^- &= \sup \{r < r_0 \mid \nu_\Omega(A(r)) = 0\} \quad (\text{ou } -\infty \text{ si non}) \\ r^+ &= \inf \{r > r_0 \mid \nu_\Omega(A(r)) = 0\} \quad (\text{ou } +\infty \text{ si non}) \end{aligned}$$

qui donnent les bornes de cet intervalle contenant r_0 . L'application gardienne ne change pas de signe entre r^- et r_0 , ou entre r_0 et r^+ , sans quoi elle s'annulerait. Ceci permet de conserver la stabilité assurée en r_0 sur tout $I = [r^-, r^+]$, car r évolue dans la même composante.

Le lemme suivant résume ce résultat de façon plus formelle :

Lemme 4.4.4

Soit $A(r) = A_0 + r A_1 + \cdots + r^k A_k$ une famille de matrices dépendant du paramètre incertain réel r avec A_i des matrices constantes fixées, et soit r_0 tel que $A(r_0)$ soit stable par rapport à Ω . Soit ν_Ω une application gardienne pour $\mathcal{S}(\Omega)$. Avec la définition précédente, on détermine r^- et r^+ .

Alors $\forall r \in]r^-, r^+[$, $A(r)$ est stable par rapport à Ω .

C'est même le plus grand intervalle contenant r_0 .

4.5 Exemple d'analyse robuste

Considérons une famille de matrices biparamétrique dont la stabilité Hurwitz est étudiée :

$$\mathcal{A} = \{A(r_1, r_2); (r_1, r_2) \in [0, 1] \times [0, 1]\}$$

La famille \mathcal{A} est définie par

$$A(r_1, r_2) = \begin{bmatrix} -3 - r_2 + 3r_2^2 & -1 + r_2 + 4r_1r_2 \\ -1 + 2r_1 & -2 + 3r_1 + r_2 - r_2^2 \end{bmatrix}$$

La stabilité nominale est assurée pour la matrice $A(0, 0) = \begin{bmatrix} -3 & -1 \\ -1 & -2 \end{bmatrix}$ qui a pour valeurs propres -3.62 et -1.38.

L'étude de la stabilité robuste par les applications gardiennes passe par le calcul de $\nu_{Hurwitz}(A(r_1, r_2))$. A l'aide d'un logiciel de calcul formel tel que **Matlab** il est possible de calculer l'expression exacte de cette fonction. Elle est donnée en (4.7).

$$\nu(A(r_1, r_2)) = -(r_2^2 + \frac{3}{2}r_1 - \frac{5}{2}) \times (8r_1^2r_2 - 9r_1r_2^2 + r_1r_2 + 7r_1 + 3r_2^4 - 4r_2^3 + 4r_2^2 - 5) \quad (4.7)$$

La visualisation des composantes est rendue possible par le calcul du lieu d'annulation de ν . La fonction **contour** de **Matlab** permet de réaliser ce découpage. Un point (r_1, r_2) doit ensuite être sélectionné dans chaque zone pour tester la stabilité de cette dernière, tel qu'illustré en Fig. 4.5.

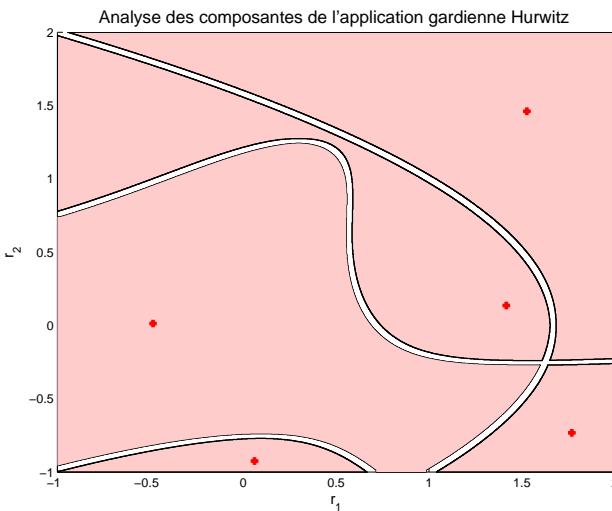


Figure 4.5 Choix des points à l'intérieur de chaque composante

Un test de stabilité pour chaque point (r_1, r_2) est effectué par le calcul des valeurs propres de $A(r_1, r_2)$. Il permet de vérifier quelles composantes conservent ou perdent la stabilité Hurwitz. Avec cet exemple, on obtient les résultats suivants :

Tableau 4.1 Valeurs des points de test et valeurs propres associées

(r_1, r_2)	$\lambda_i(A(r_1, r_2))$
(1.17, 0.30)	(-3.22, +1.91)
(-0.49, 0)	(-1.81, -4.65)
(0.18, -0.90)	(+0.76, -3.59)
(0.54, 1.69)	(+3.99, -1.61)
(1.78, -0.73)	(0.69 \pm 3.95i)

D'après le lemme 4.4.3, il est possible de conclure quant à la stabilité de toute la composante. Les zones stables et instables obtenues sont présentées en Fig. 4.6. Il est possible de conclure que si $(r_1, r_2) \in [0, 1] \times [0, 1]$ alors la famille complète \mathcal{A} ne sera pas stable Hurwitz. Cette conclusion peut se raffiner en donnant par exemple un ensemble de variation pour lequel la stabilité est conservée : $(r_1, r_2) \in [0, 0.5] \times [0, 1]$.

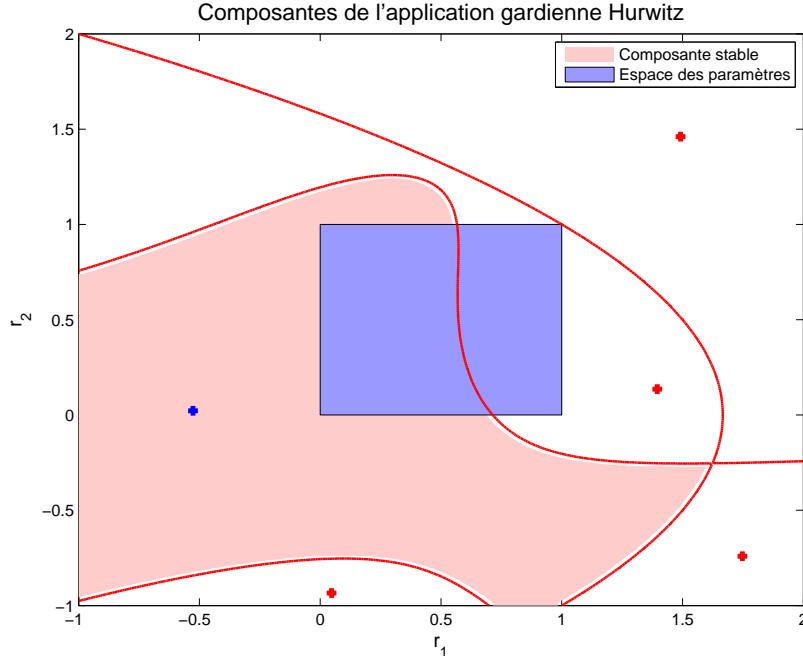


Figure 4.6 Visualisation des composantes stables au sens de Hurwitz

4.6 Exemple de synthèse

Les applications gardiennes peuvent aussi servir comme outil pour la synthèse de correcteur. Considérant un système dynamique avec une structure de correction fixée et des gains de réglage variables pour cette correction, elles vont donner les zones de réglage permettant le respect de la stabilité ou même de certaines performances.

Les gains sont ainsi considérés comme les variables des applications gardiennes, et les ensembles Ω considérés sont désormais liés aux performances qui doivent être atteintes en boucle fermée.

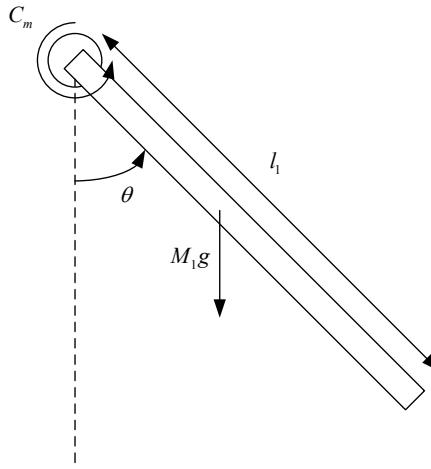


Figure 4.7 Moteur électrique couplé avec un pendule

Prenons l'exemple d'un moteur couplé avec un pendule, décrit en Fig. 4.7, les équations de la dynamique fournissent le modèle d'ordre 3 donné par (4.8-4.9). Les données R, L, K_ω, K_c, J_1 et B_1 sont relatives aux caractéristiques du moteur, et M_1, l_1, J_1 et B_1 sont liées au pendule.

Le vecteur d'état est $X = \begin{bmatrix} i & \theta & \dot{\theta} \end{bmatrix}^T$, avec les valeurs petits signaux linéarisées autour du point d'équilibre $\theta_e = \frac{\pi}{6}$. L'entrée considérée est la tension envoyée au moteur u . Les sorties seront la position θ du pendule ainsi que sa dérivée pour réaliser une loi de commande proportionnelle-dérivée (PD) avec les deux gains K_p et K_d à régler par les applications gardiennes.

$$\dot{X} = \begin{bmatrix} -\frac{R}{L} & 0 & -\frac{K_\omega}{L} \\ 0 & 0 & 1 \\ \frac{K_c}{J_m+J_1} & -\frac{M_1 g l_1 \cos(\frac{\pi}{6})}{2(J_m+J_1)} & -\frac{B_m+B_1}{J_m+J_1} \end{bmatrix} X + \begin{bmatrix} \frac{1}{L} \\ 0 \\ 0 \end{bmatrix} u \quad (4.8)$$

$$\begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} X + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \quad (4.9)$$

Le but de cette synthèse est d'améliorer la rapidité du système en boucle fermée. Les pôles du système en boucle ouverte sont situés en $(-997, -2 \pm 3.5i)$, permettant un temps de réponse de $2s$ environ. La boucle fermée devra rendre le système cinq fois plus rapide qu'il ne l'est déjà, en plaçant tous les pôles au-delà de $\alpha = -10$. Pour ce faire, l'application gardienne suivante est considérée :

$$\nu_\alpha(A) = \det(A - \alpha I) \det((A - \alpha I) \odot I) \quad (4.10)$$

Les composantes de ν_α vont être étudiées pour les variables $K = (K_p, K_d)$, quand on considère la matrice en boucle fermée $A_{BF}(K) = A - BKC$. Ces paramètres influencent la position des pôles de A_{BF} et l'étude de l'application gardienne ν_α indiquera quels réglages choisir pour respecter le critère de performance fixé.

Le tracé de son lieu d'annulation et l'analyse des zones rendent un graphe similaire au cas précédent pour l'analyse robuste, présenté en Fig. 4.8. L'interprétation en est cependant différente : ici les zones indiquent les réglages des gains (K_p, K_d) qui donnent un système bouclé suffisamment rapide.

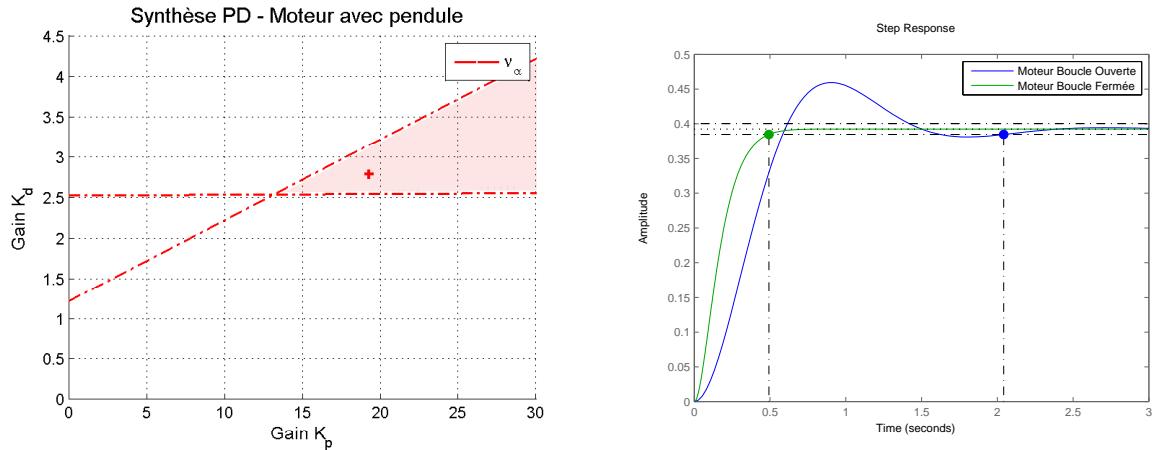


Figure 4.8 Composantes de l'application gardienne et réponses temporelles pour le moteur+pendule

Le choix d'un réglage a été effectué dans la composante stable, menant à la réponse à l'échelon présentée en Fig. 4.8, sur laquelle le système bouclé est clairement plus rapide que le système original.

4.7 Application au cas du lanceur

Partant de la méthode de synthèse précédente, le cadre pour la synthèse d'une loi de commande respectant le cahier des charges du lanceur va être présenté. La loi de commande en tant que telle sera réalisée dans les deux prochains chapitres par une méthode graphique puis par une méthode d'optimisation.

Les applications gardiennes liées au problème seront détaillées, puis un exemple de synthèse pour le modèle nominal au temps T_1 sera donné pour montrer comment satisfaire toutes les performances.

4.7.1 Modèle et structure de correction

Le modèle utilisé est celui du lanceur flexible avec la dynamique de l'actionneur. Les matrices obtenues sont données en (4.11,4.12) sous la forme standard (A_F, B_F, C_F, D_F). Les sorties considérées ici sont l'attitude θ et sa dérivée, dans le but de synthétiser un contrôleur ayant deux paramètres variables. Le but est d'illustrer comment prendre en considération les performances réclamées par le cahier des charges. La matrice A en boucle fermée s'écrit : $A(K_1, K_2) = A_F - B_F [K_1 \ K_2] C_F + D_F$, avec les gains K_1 et K_2 qui seront choisis grâce aux applications gardiennes.

Un rappel de la structure de correction adoptée est fait avec la Fig. 4.9.

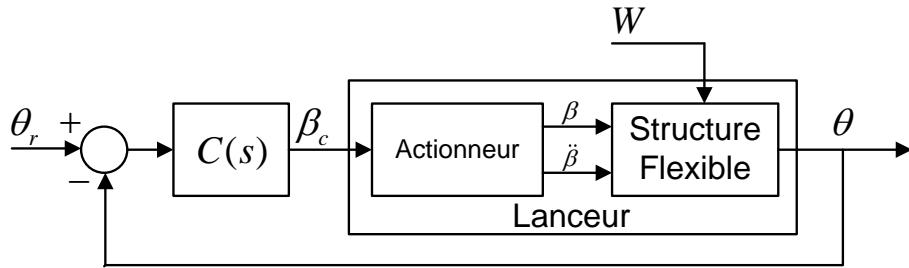


Figure 4.9 Structure de contrôle du lanceur avec la mesure sur θ

$$\dot{X} = \left[\begin{array}{c|ccccc} 0 & 4.26 & 0 & 1.53 \cdot 10^{-5} & 0 & 6.28 \cdot 10^{-6} \\ 1 & 0 & 0 & 0 & 0 & 0 \end{array} \right] \left[\begin{array}{cccc} 0.014 & -0.64 & 0.03 & 0.43 \\ 0 & 0 & 0 & 0 \\ -25.53 & 1156 & -40.71 & 624.7 \\ 0 & 0 & 0 & 0 \end{array} \right] X + \left[\begin{array}{c} -0.38 \\ 0 \\ 685.6 \\ 0 \\ -760.1 \\ 0 \end{array} \right] \beta_c \quad (4.11)$$

$$\left[\begin{array}{c} \theta \\ \dot{\theta} \end{array} \right] = \left[\begin{array}{c|ccccc} 0 & 1 & 0 & 7.74 \cdot 10^{-5} & 0 & 1.01 \cdot 10^{-4} \\ 1 & 0 & 2.7 \cdot 10^{-3} & 0 & 9.1 \cdot 10^{-3} & 0 \end{array} \right] \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right] X + \left[\begin{array}{c} 0 \\ 0 \end{array} \right] \beta_c \quad (4.12)$$

4.7.2 Applications gardiennes considérées

Trois contraintes principales doivent être respectées sur les pôles de $A(K_1, K_2)$: l'amortissement, la rapidité sur les modes rigides et la stabilité sur les modes flexibles. A chaque contrainte est associée une application gardienne. Voici les trois applications à considérer, en notant $A(K) = A(K_1, K_2)$ pour simplifier les notations.

- limite de rapidité à $\alpha = -0.5$ sur les modes rigides :

$$\nu_\alpha(A(K)) = \det(A(K) - \alpha I) \det((A(K) - \alpha I) \odot I) \quad (4.13)$$

- limite d'amortissement à $\xi = 0.5$ sur les modes rigides :

$$\nu_\xi(A(K)) = \det(A(K)) \det(A(K)^2 \odot I + (1 - 2\xi^2)A(K) \odot A(K)) \quad (4.14)$$

- limite de stabilité à $\beta = -0.2$ sur les modes flexibles :

$$\nu_\beta(A(K)) = \det(A(K) - \beta I) \det((A(K) - \beta I) \odot I) \quad (4.15)$$

Le but est de confiner les pôles associés au mode rigide dans Ω_{rigide} et ceux des modes flexibles dans $\Omega_{flexible}$, pour les ensembles décrits en Fig. 4.10.

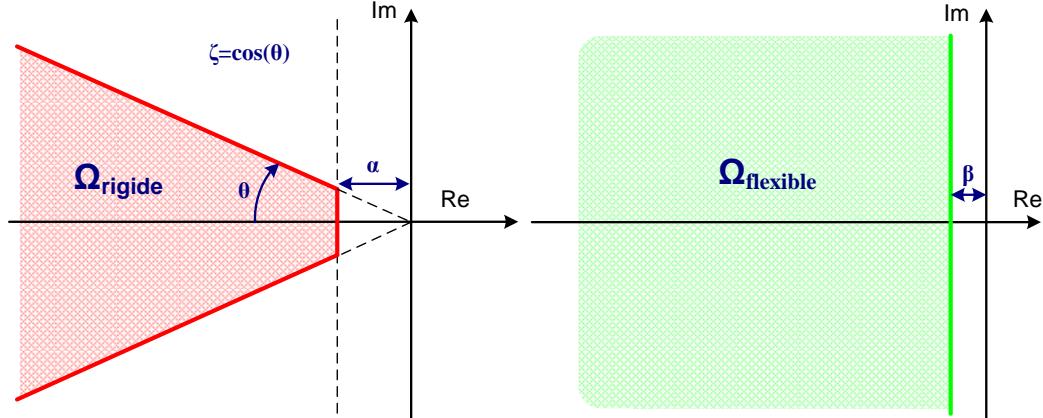


Figure 4.10 Zones de confinement pour la stabilité des pôles rigides et flexibles

Il est important de noter que les applications gardiennes s'appliqueront à la même matrice, même si les contraintes étudiées s'appliquent sur différents pôles. Chaque application est désormais fonction des gains de la structure de contrôle (ici réduits à K_1 et K_2 pour simplifier) : $\nu(A(K)) = \nu(A_F - B_F K C_F + D_F) \triangleq \nu(K_1, K_2)$.

4.8 Visualisation des composantes

Quand deux gains seulement doivent être réglés, il est possible de visualiser les composantes de chaque application gardienne en 2D et de tester pour chacune d'entre elles la stabilité par rapport au Ω concerné. Dès que le nombre de paramètres augmente, il n'y a plus de visualisation possible, c'est pourquoi des algorithmes d'optimisation seront développés. Dans les deux cas, le but de la méthode est de fournir un réglage (K_1, K_2) qui assure le placement des pôles dans les ensembles Ω considérés.

Les lieux d'annulation des trois applications gardiennes précédentes sont fournies en Fig. 4.11, en calculant $\nu_\alpha(K_1, K_2) = 0$, $\nu_\beta(K_1, K_2) = 0$ et $\nu_\xi(K_1, K_2) = 0$. La stabilité des composantes est ensuite évaluée en classifiant les pôles obtenus en boucle fermée, pour vérifier les contraintes sur les pôles rigides ou flexibles uniquement.

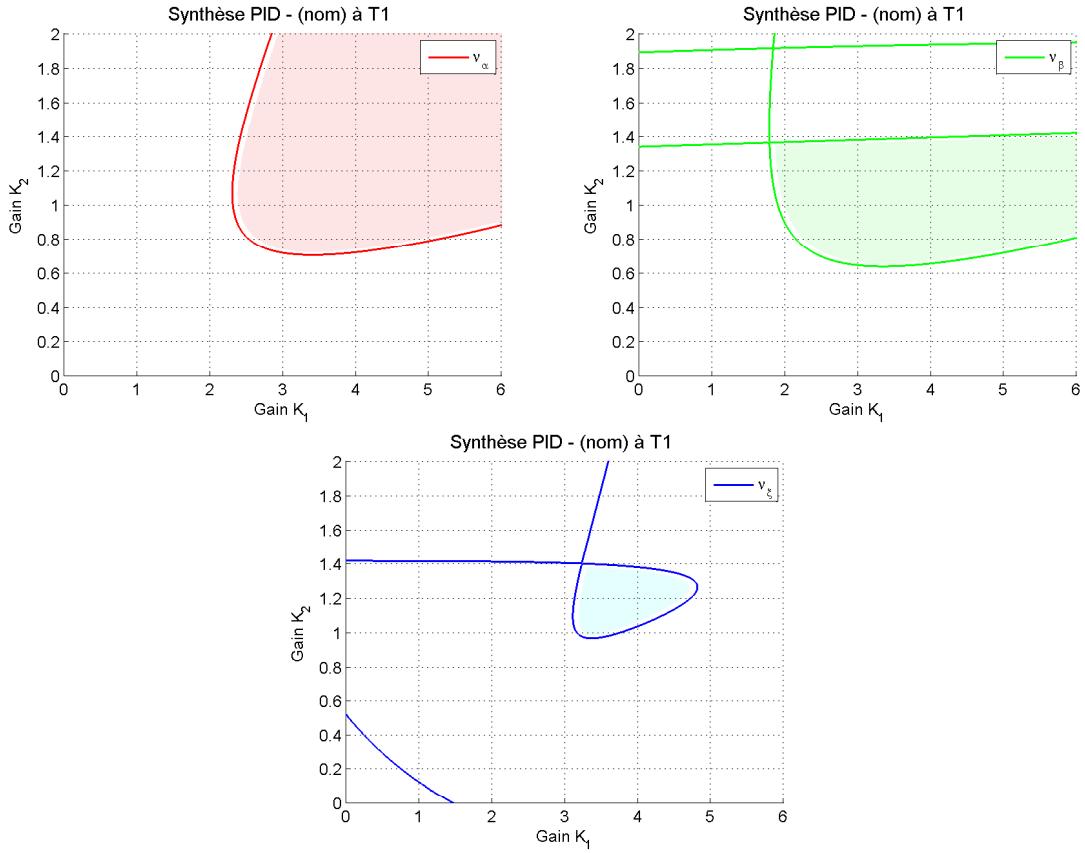


Figure 4.11 Lieux d'annulation des applications gardiennes ν_α , ν_β et ν_ξ

Les composantes de chaque application ont été tracées puis un test de Ω -stabilité a été réalisé pour fournir directement les composantes d'intérêt. La satisfaction de tous les critères nécessite de trouver un seul réglage situé dans les trois composantes stables identifiées. Une superposition des graphiques permet alors d'observer l'existence ou non d'une composante stable pour l'application gardienne

$$\nu(K) = \nu_\alpha(K) \times \nu_\beta(K) \times \nu_\xi(K)$$

car on a simplement décomposé ce produit pour analyser l'annulation de chaque terme séparément, soit l'annulation de chaque application gardienne. Cette analyse est présentée en Fig. 4.12.

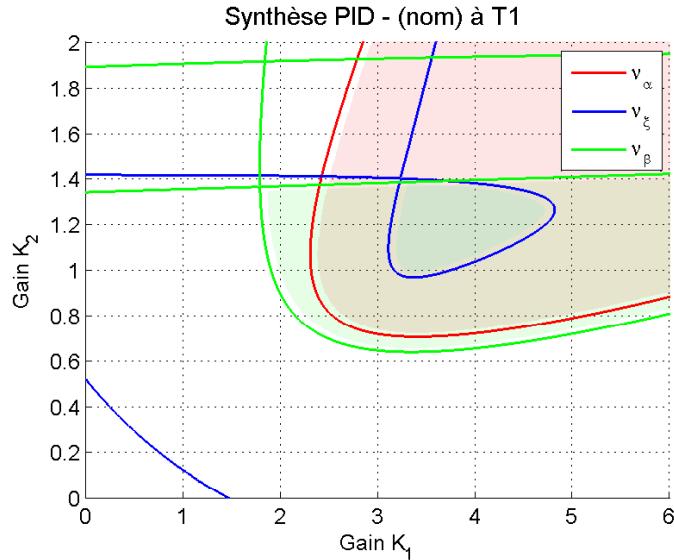


Figure 4.12 Zone de stabilité commune à ν_α , ν_ζ et ν_β dans l'espace des paramètres

Cette méthode de synthèse doit maintenant s'appliquer avec les deux autres modèles incertains **hf** et **lf** pour permettre un contrôle robuste du lanceur à un instant donné.

La superposition des zones stables pour les trois applications gardiennes suivantes permet de satisfaire finalement les trois performances du cahier des charges sur les trois modèles à la fois, comme illustré en Fig. 4.14. Il est donc possible de réaliser une loi de commande robuste avec respect de performances grâce aux applications gardiennes.

Pour ce faire, on considère l'application gardienne :

$$\nu(A) = \nu_{\text{nom}}(A) \times \nu_{\text{hf}}(A) \times \nu_{\text{lf}}(A)$$

basée sur les applications suivantes :

$$\begin{aligned}\nu_{\text{nom}}(K) &= \nu_{\alpha,\text{nom}}(K) \times \nu_{\beta,\text{nom}}(K) \times \nu_{\xi,\text{nom}}(K) \\ \nu_{\text{hf}}(K) &= \nu_{\alpha,\text{hf}}(K) \times \nu_{\beta,\text{hf}}(K) \times \nu_{\xi,\text{hf}}(K) \\ \nu_{\text{lf}}(K) &= \nu_{\alpha,\text{lf}}(K) \times \nu_{\beta,\text{lf}}(K) \times \nu_{\xi,\text{lf}}(K)\end{aligned}$$

Au final, cela revient à superposer les composantes stables des 9 applications gardiennes basiques qui interviennent, pour constater la présence ou non d'une intersection. Si elle existe, alors un réglage des gains K_1 et K_2 permet de satisfaire les performances de façon robuste sur les trois modèles, sinon il faut revoir la structure de correction et/ou les performances demandées.

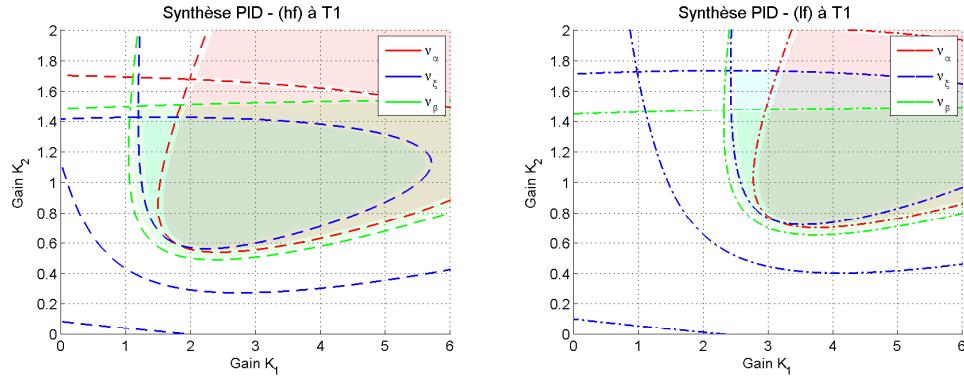


Figure 4.13 Zone de stabilité de ν_{hf} et ν_{lf}

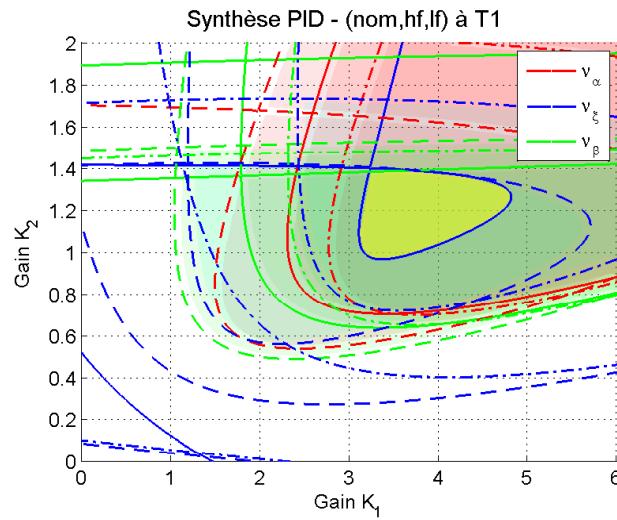


Figure 4.14 Zone de stabilité commune à ν_{nom} , ν_{hf} et ν_{lf}

CHAPITRE 5

SYNTHÈSE PAR ANALYSE GRAPHIQUE

5.1 Introduction

La méthode de synthèse basée sur les applications gardiennes exige le calcul de toutes les composantes avant de vérifier la stabilité de chacune par rapport à la contrainte considérée. Une fois les composantes stables obtenues, l'existence d'une intersection entre elles est nécessaire à l'obtention d'un contrôleur respectant l'ensemble des contraintes. Cette approche graphique peut se faire automatiquement en utilisant les fonctions d'analyse d'image du logiciel Matlab. Elles vont permettre d'obtenir directement les composantes stables, si celles-ci existent, pour ensuite vérifier la présence d'une intersection non-vide.

Ce chapitre détaille la mise en pratique et la théorie utilisée pour analyser les composantes d'une application gardienne. Un premier point sera fait sur la reconstruction d'une image à partir des données sur les frontières des composantes, puis un pré-traitement avec les bibliothèques de traitement d'images sera effectué pour permettre l'analyse. Cette dernière consiste à détecter les zones présentes dans l'image et à donner les coordonnées de leur frontière. Enfin deux méthodes seront exposées pour obtenir un point à l'intérieur d'un ensemble quelconque. Ce point intérieur permet les tests de stabilité sur chaque composante.

A la fin du chapitre, ces méthodes seront appliquées au cas du lanceur pour synthétiser un contrôleur qui respecte le cahier des charges complet sur le placement des pôles sur les six instants de vol considérés. Pour ce faire, le choix de la structure du correcteur sera détaillé. Un séquencement sera aussi envisagé pour permettre d'améliorer sa robustesse et d'atténuer les dépassements sur l'incidence en présence de vent. Les résultats obtenus seront enfin validés par les simulations temporelles et par le calcul des marges.

5.2 Analyse d'image

La méthode de synthèse présentée se base sur l'analyse d'images provenant des graphes des applications gardiennes. Le calcul de leurs lieux d'annulation permet d'obtenir un « pavage » de l'espace des paramètres en composantes. D'après le lemme des composantes, il est possible de vérifier la Ω -stabilité d'un seul point de chaque composante pour conclure sur la Ω -stabilité des gains de toute la composante. Voici les étapes suivies pour réaliser cette analyse :

- Choix d'un maillage de l'espace des gains : $[K_{1,i}, K_{2,j}]$
- Calcul de l'application gardienne sur ce maillage $\nu_{ij} = \nu(K_{1,i}, K_{2,j})$
- Calcul du lieu d'annulation de ν pour obtenir le contour C
- Construction d'une image noir et blanc représentant C
- Pré-traitement de l'image
- Détection des zones présentes dans l'image
- Choix d'un point (K_1^*, K_2^*) intérieur à chaque zone trouvée
- Analyse de la Ω -stabilité pour chaque point (K_1^*, K_2^*)

Le code Matlab générique associé à cette approche est donné ci-dessous :

```

global A B C D;

% State-Space model definition
A = ... ;
B = ... ;
C = ... ;
D = ... ;
sys = ss(A,B,C,D);

% Gains space meshing
[K1, K2] = meshgrid( x0:dx:x1, y0:dy:y1 );

% Guardian Maps computation
% based on the function 'GM' defining the considered guardian map for
% the variables K1 & K2
GM_Values = GM( K1, K2 );

% Guardian Maps analysis
GM_Contour = contour( K1, K2, GM_Values, [0 0] );

% Graphical analysis of each component
GM_Contour_Picture = image_processing(GM_contour);
GM_Components      = component_analysis(GM_Contour_Picture);
[K1_star, K2_star] = interior_point(GM_Components);
component_stability_analysis(K1_star,K2_star);

```

Chacune de ces étapes va être détaillée dans la suite du chapitre.

5.2.1 Calcul des applications gardiennes

Les applications gardiennes font intervenir des calculs de déterminants sur des matrices issues du produit bialterné. Ces dernières peuvent atteindre des tailles de 120×120 quand le système étudié est d'ordre 15. Cette explosion dans la complexité oblige à considérer le moins d'opérations possibles pour leur calcul car il faut les répéter à chaque point du maillage avec $\nu_{ij} = \nu(K_{1,i}, K_{2,j})$. Un travail de réécriture sur les matrices des applications gardiennes permet d'obtenir la forme donnée en (5.1) pour laquelle les matrices $M_{k,l}$ sont constantes par rapport aux variables K_1 et K_2 .

$$\nu(K_1, K_2) = \det \left(\sum_{k,l=0}^{k+l=N} M_{k,l} K_1^k K_2^l \right) \quad (5.1)$$

Cas particulier pour $\det(A - BKC)$

Cette écriture donne lieu à une simplification considérable pour le calcul de cette application gardienne quand le système est mono-entrée (*Single Input*) ou mono-sortie (*Single Output*). En notant $C_i = C(i, :)$ dans le cas mono-entrée, comme pour l'exemple du lanceur, les matrices BC_1 et BC_2 sont de rang 1 et simultanément diagonalisables car liées au même vecteur propre d'après (5.2). Pour leur unique valeur propre non nulle, le couple valeur/vecteur propre associé est $(C_i B, B)$.

$$BC_i B = (C_i B) B = \lambda_i B \quad (5.2)$$

La diagonalisation simultanée de ces deux matrices dans le calcul $\det(A - BC_1 K_1 - BC_2 K_2)$ permet d'obtenir la formule suivante, avec Q la matrice des vecteurs propres communs de BC_i :

$$\nu(K_1, K_2) = \det \left(Q^{-1} \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} Q - \begin{bmatrix} (C_1 B)K_1 + (C_2 B)K_2 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \right)$$

D'après les propriétés de développement du déterminant, la dépendance de ν est linéaire par rapport à K_1 et K_2 . De plus, le gradient du déterminant $\det(A_0 + A_1 x)$ s'écrit $\nabla_x = \text{trace}(\text{adj}(A_0)A_1)$ avec $\text{adj}(\cdot)$ désignant l'adjointe d'une matrice. Avec ces résultats, (5.3) donne l'expression littérale linéaire de cette application gardienne.

$$\det(A - BKC) = \det(A) - \text{trace}(\text{adj}(A)BC_1)K_1 - \text{trace}(\text{adj}(A)BC_2)K_2$$

Soit finalement :

$$\det(A - BKC) = \det(A) - (C_1\text{adj}(A)B)K_1 - (C_2\text{adj}(A)B)K_2 \quad (5.3)$$

5.2.2 Calcul du lieu d'annulation

Une fois les applications gardiennes calculées sur un maillage donné $[K_{1,i}, K_{2,j}]$, leur lieu d'annulation doit être évalué pour trouver leurs composantes. Cette étape repose sur l'utilisation de la fonction `contour` de Matlab, qui permet de tracer les isovaleurs d'une fonction de deux variables. L'algorithme de cette fonction pour tracer une courbe de niveau repose sur une matrice booléenne $\text{Checked}_{i,j}$ de la taille du maillage, dont les valeurs sont 1 si la case considérée a déjà été visitée et 0 sinon.

L'appel à cette fonction se fait à partir d'un maillage $[X_i, Y_j]$ fourni par la fonction `mehs-grid`, de la matrice Z donnant les valeurs de la fonction pour tous les points du maillage, et enfin des lignes de niveau *levels* à étudier.

```
C = contour(X, Y, Z, levels)
```

La fonction part d'une valeur c parmi les niveaux donnés dans *levels*. En se basant ensuite sur la matrice Z , elle parcourt le maillage pour trouver deux valeurs consécutives de Z qui encadrent c , par exemple $Z_{ij} < c < Z_{i+1,j}$. Alors la ligne de niveau coupe le segment $[(X_i, Y_j); (X_{i+1}, Y_j)]$, et leur point d'intersection est obtenu par interpolation linéaire des coordonnées des sommets. Le k^{ime} point du contour est obtenu : $c (X_{c,k}, Y_{c,k})$, comme illustré à la Fig. 5.1.

La ligne de niveau se retrouve alors dans une maille élémentaire dans laquelle elle est entrée par le bord $[(X_i, Y_j); (X_{i+1}, Y_j)]$, et de laquelle elle doit sortir. En vérifiant les 2 autres sommets de la maille élémentaire le bord de sortie et le point d'intersection sont obtenus par le même principe : $(X_{k+1,x}, Y_{k+1,c})$.

Une fois les bords d'entrée et de sortie trouvés, la maille élémentaire est marquée comme lue dans la matrice *Checked*, en mettant la valeur $\text{Checked}_{i,j}$ à 1.

L'opération est ainsi répétée jusqu'à ce que la maille élémentaire suivante repasse par la maille de départ pour obtenir un contour fermé, ou qu'elle arrive sur un des bords du maillage. Dans les deux cas, la courbe obtenue est ajoutée à la matrice de sortie C . Le processus est répété jusqu'à ce que toutes les mailles élémentaires aient été parcourues (i.e. la matrice *Checked* est remplie de 1).

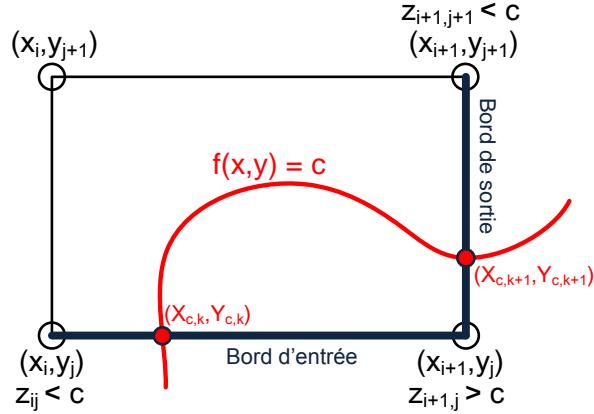


Figure 5.1 Construction d'une courbe de niveau par la fonction `contour` au niveau d'une maille

La matrice de sortie C contient alors toutes les lignes de niveau qui peuvent apparaître dans le maillage. Certaines peuvent faire référence au même niveau si jamais les courbes décrites ne sont pas fermées. Elle se présente sous la forme générale suivante :

$$C = \left[\begin{array}{c|cccc} c_1 & x_{c_1,1} & x_{c_1,2} & \cdots & x_{c_1,n_1} \\ \hline n_1 & y_{c_1,1} & y_{c_1,2} & \cdots & y_{c_1,n_1} \end{array} \right] \left[\begin{array}{c|cc} c_2 & x_{c_2,1} & \cdots \\ \hline n_2 & y_{c_2,1} & \cdots \end{array} \right]$$

Dans le cas des applications gardiennes, le seul niveau évalué est 0 car les composantes à étudier sont obtenues en visualisant le lieu d'annulation de la fonction.

5.2.3 Construction d'une image Noir & Blanc

Une image Noir & Blanc est ensuite construite sur la base de la matrice des contours C . Une image en `Matlab` est représentée par une matrice remplie de 0 pour la couleur noire, et de 1 pour la couleur blanche. La $(i, j)^{me}$ valeur de la matrice donne la couleur du pixel en (i, j) . Il y a donc simplement un changement de coordonnées à faire entre l'espace des gains et l'espace des pixels. Les coordonnées des contours sont données dans le premier, tandis qu'il faut les avoir dans le second pour entamer une analyse d'image. Il est à noter que les directions des axes des pixels sont obtenues par rotation de -90° du repère des gains.

En se basant sur la Fig. 5.2 les coordonnées pixels sont obtenues par les formules de passage (5.4,5.5), pour des pas de maillage dx et dy et des bornes $[x_0, x_1] \times [y_0, y_1]$.

$$i = \left(\lfloor \frac{y_1 - y_0}{dy} \rfloor + 1 \right) - \lfloor \frac{y - y_0}{dy} \rfloor \quad (5.4)$$

$$j = \lfloor \frac{x - x_0}{dx} \rfloor + 1 \quad (5.5)$$

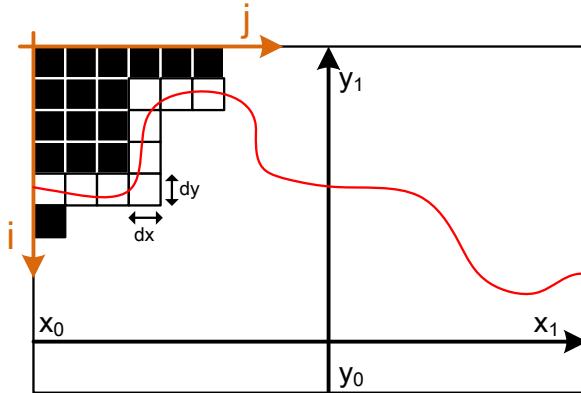


Figure 5.2 Construction d'une image Noir & Blanc représentant un contour

Un exemple d'image obtenue est donnée en Fig. 5.3, quand les méthodes précédentes sont appliquées sur le lieu d'annulation d'une application gardienne.

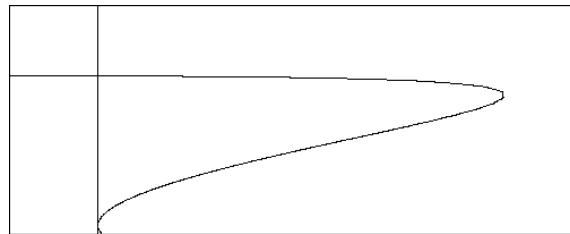


Figure 5.3 Image Noir & Blanc représentant les composantes d'une application gardienne

5.2.4 Pré-traitement de l'image

Le calcul du contour donne parfois lieu à des erreurs numériques lorsque le pas de maillage n'est pas suffisamment faible. Des contours de quelques pixels seulement peuvent apparaître et doivent être retirés. De plus, l'image obtenue étant basée sur les coordonnées d'une courbe de niveau, les lignes obtenues peuvent être trop fines pour permettre une séparation franche des composantes.

Le premier point est solutionné par l'utilisation de la fonction `bwareaopen` dédiée au traitement d'images. Elle permet d'effacer les zones dont l'aire est inférieure à N pixels. Elle est appelée par la commande : `Image_retouchee = bwareaopen(Image, N)`.

Le second point est résolu par l'utilisation de filtres pour lisser et élargir les contours. Le filtre utilisé dans ce pré-traitement attribue à un pixel la valeur moyenne des pixels contenus dans le rectangle de taille $p \times p$, centré sur lui. L'action du filtre sur l'image est réalisée par la commande : `Image_floutee = imfilter(Image_retouchee, fspecial('average', p))`. Les zones grises sont ensuite converties en Noir & Blanc par la fonction `im2bw`.

En appliquant ce traitement à l'exemple précédent en Fig. 5.3, l'image Fig. 5.4 est obtenue.

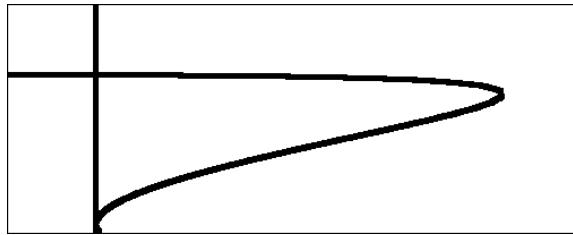


Figure 5.4 Image Noir & Blanc représentant les composantes après le pré-traitement

5.2.5 Détection des composantes dans l'image

Une fois le pré-traitement effectué, l'analyse des composantes peut être réalisée. Elle repose sur la fonction `bwboundaries` qui implémente l'algorithme de tracé basé sur le voisinage de Moore (Gonzalez *et al.*, 2004). Ce dernier consiste à parcourir les voisins d'un pixel de la frontière dans le sens horaire pour trouver le point suivant sur la frontière. En parcourant ainsi le bord d'un objet noir sur un fond blanc par exemple, on obtient au terme de l'algorithme les pixels de la frontière complète.

La fonction `bwboundaries` répète cet algorithme autant de fois qu'elle trouve un nouveau pixel noir en parcourant le reste de l'image. À terme, la frontière de chaque objet est obtenue et stockée dans un ensemble de cellules appelé `Frontières`. L'appel à cette fonction se fait par la ligne de commande : `[Frontieres, Image_numerotee] = bwboundaries(Image_floutee)`. Le second argument en sortie `Image_numerotee` contient une image où les pixels de chaque objet identifié ont été mis à une valeur arbitraire pour les dissocier.

Le résultat de cette commande sur l'exemple Fig. 5.4 est fourni en Fig. 5.5 avec une mise en couleur de chaque zone.

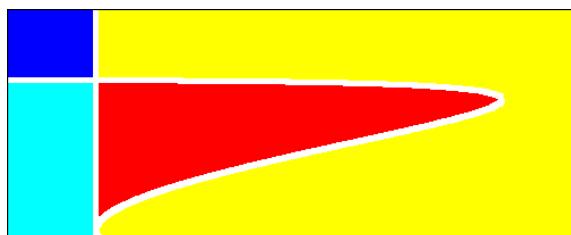


Figure 5.5 Composantes identifiées à partir du tracé du lieu d'annulation

5.2.6 Choix d'un point intérieur

Une fois les composantes identifiées, un point intérieur à chacune d'entre elles doit être choisi pour effectuer le test de stabilité. Le problème majeur qui se pose est de trouver un point intérieur quand la composante n'est pas convexe. Un simple barycentre ne peut pas être utilisé car il peut se situer à l'extérieur de l'ensemble. Aussi, deux méthodes ont été développée pour trouver un point intérieur quelque soit la forme de la composante.

Construction d'un sous-ensemble convexe

Cette méthode parcourt la frontière de la composante en ne gardant que certains sommets dans le but d'obtenir un ensemble convexe à l'intérieur. Il est ainsi possible de prendre le barycentre de ce sous-ensemble pour avoir un point strictement à l'intérieur de la composante. L'algorithme suivi est le suivant :

0 - Sélection du premier, second et troisième sommet, respectivement V_1 , V_2 et V_3

1 - Calcul des vecteurs $v_1 = V_2 - V_1$ et $v_2 = V_3 - V_2$

2 - Tant que tous les sommets n'ont pas été visités

2.1 - Si le sommet V_3 conserve la convexité du domaine

$$(\text{i.e. } \det(v_1, v_2) = \sin(\theta_1)|v_1||v_2| \leq 0)$$

- Mettre en mémoire le sommet V_3
- Poser $V_1 = V_2$, $V_2 = V_3$
- Poser V_3 égal au sommet suivant

2.2 - Sinon

- Poser V_3 égal au sommet suivant

2.3 - Mettre à jour $v_1 = V_2 - V_1$ et $v_2 = V_3 - V_2$

3 - Prendre le barycentre de tous les sommets mis en mémoire.

Ce processus est illustré par la Fig. 5.6 où les sommets V_1 , V_2 , V_3 et V_4 seront conservés tandis que V_5 sera rejeté. Le résultat de cette analyse est donné en Fig. 5.7 illustrant l'ensemble convexe obtenu dans la composante.

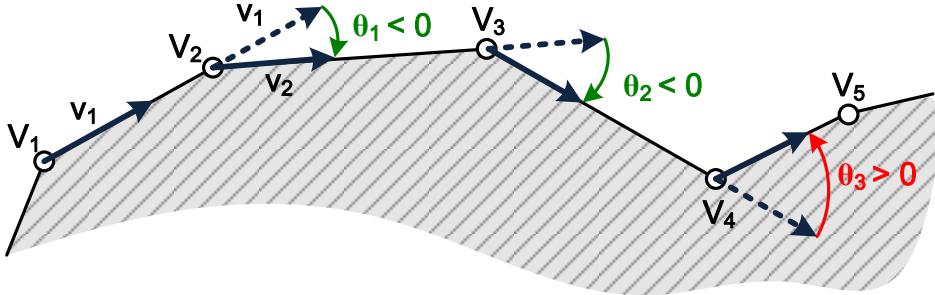


Figure 5.6 Construction d'un ensemble convexe par parcours de la frontière

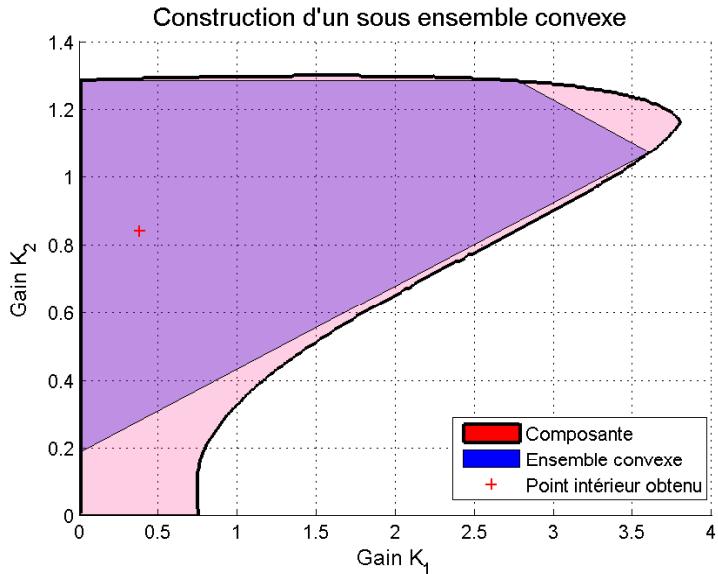


Figure 5.7 Construction d'un ensemble convexe inclus dans la composante

Recherche de segments

La méthode précédente étant gourmande en temps de calcul une autre approche a été envisagée pour accélérer la recherche du point intérieur. Cette seconde méthode repose sur une construction d'ensemble convexe encore plus simple : le segment. Les intersections entre la frontière et des droites passant par son centre sont calculées pour obtenir des segments qui sont inclus ou exclus de la composante. Le centre du plus grand de ces segments donne alors un point strictement inclus dans la composante. Ce raisonnement est illustré en Fig. 5.8 où les droites verticales et horizontales ont été utilisées.

L'algorithme utilisé part du constat qu'un nombre pair d'intersections doit être trouvé puisque la composante est fermée. Ainsi en numérotant par i_1, i_2, \dots, i_{2p} , les intersections trouvées avec la droite, les segments $[i_{2k-1}, i_{2k}], \forall k > 1$ sont inclus dans la composante. Il

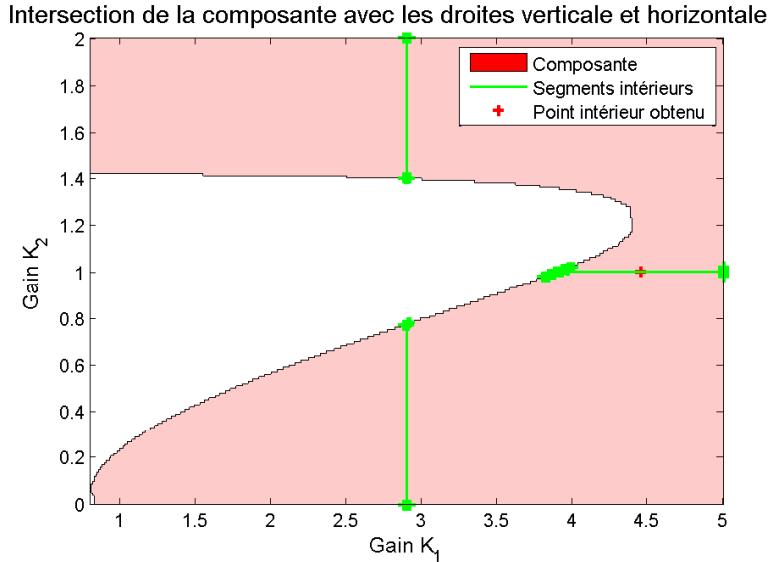


Figure 5.8 Construction de segments inclus dans la composante

reste à choisir le plus grand et à prendre son milieu.

5.2.7 Analyse de la Ω -stabilité

Enfin, à partir du point intérieur obtenu, un test est effectué pour vérifier si les réglages de chaque composante sont Ω -stables. Pour ce faire, il suffit de calculer les pôles du système en boucle fermée avec la matrice $A - BKC$ pour le réglage $K = [K_1, K_2]$ correspondant au point intérieur.

Dans le cas du lanceur, la seule difficulté est de séparer les pôles suivant les différentes dynamiques présentes. Les spécifications touchent en effet à la partie rigide et à la partie flexible. Partant de l'ensemble des pôles obtenus avec la structure de correction, illustrés en Fig. 5.9, l'actionneur, la partie flexible et la partie rigide corrigée s'obtiennent comme suit, en retirant à chaque fois les pôles cités de la liste :

- les 4 pôles ayant la plus grande pulsation correspondent à l'actionneur
- les 4 pôles ayant le plus faible amortissement correspondent à la partie flexible
- les pôles restant constituent la partie rigide et la structure de correction

En isolant ainsi les pôles, il est possible de vérifier les contraintes spécifiques aux à la partie rigide d'une part (partie réelle et amortissement) , et à la partie flexible d'autre part (stabilité).

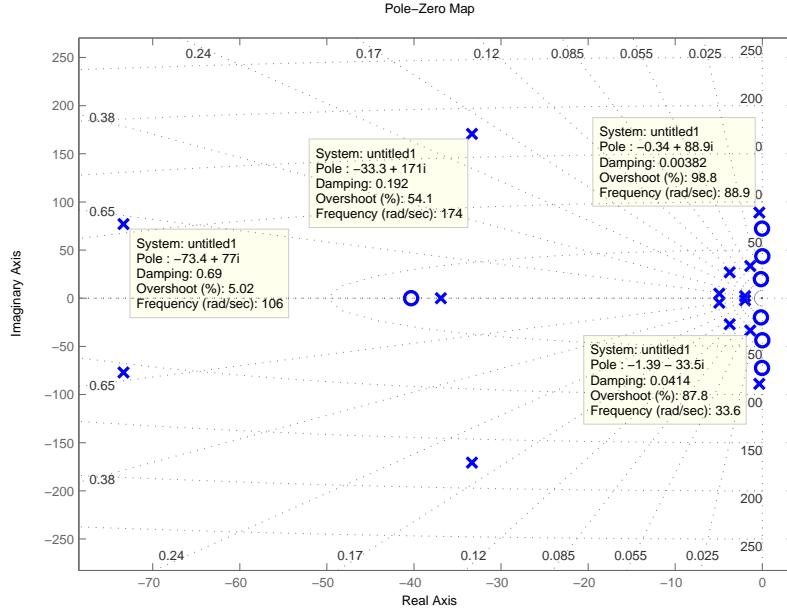


Figure 5.9 Lieu des racines en boucle fermée pour une correction d'ordre 5

5.3 Séquencement des gains

Une fois les zones de stabilité identifiées, le réglage des gains doit se situer dans cette composante-là. Sachant que la représentation adoptée pour le lanceur est variante dans le temps, le travail doit être répété sur chaque instant. Cette procédure va permettre de séquencer les gains au cours du temps de façon linéaire. Le séquencement apparaît nécessaire pour satisfaire les contraintes *a posteriori* telles que l'amplitude maximale de l'incidence ou les marges.

Pour ce faire, l'instant étant fixé, l'analyse automatique des zones est répétée sur les trois modèles de façon à trouver une intersection. Dans la Fig. 5.10, la composante stable pour les trois modèles à la fois est colorée en jaune.

Les gains devant être choisis dans cette composante, les bornes de la composante sont évaluées selon le premier gain. Pour la Fig. 5.10, l'intervalle obtenu est $K_1 \in [3.10, 4.80]$. Ce travail répété sur les cinq instants restants fournit une enveloppe de variation pour le gain K_1 . Une interpolation linéaire est alors réalisée entre le premier et le dernier instant, de manière à simplifier au maximum l'implantation du contrôleur dans la pratique.

L'enveloppe obtenue ainsi que l'interpolation réalisée sont illustrées en Fig. 5.11.

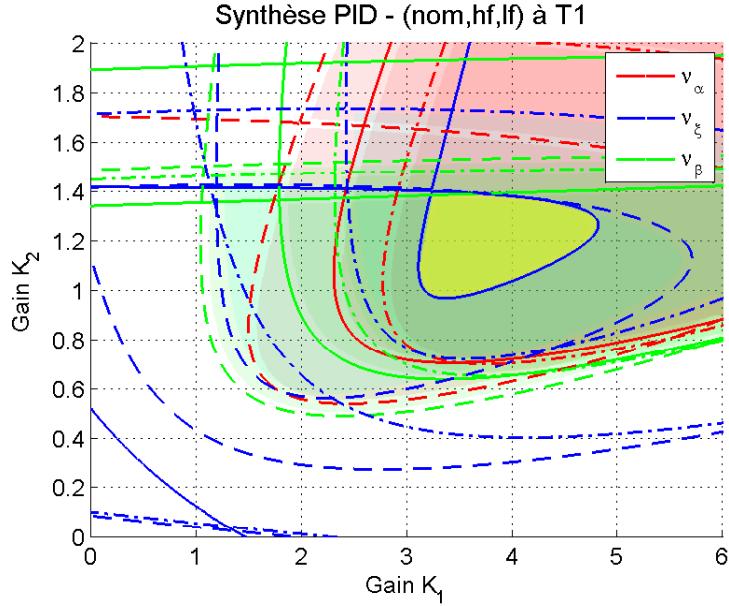


Figure 5.10 Composante stable commune aux trois modèles nom, hf et lf

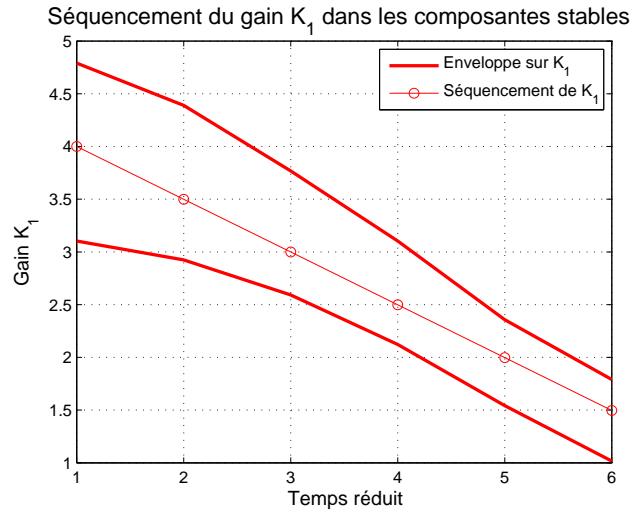


Figure 5.11 Enveloppe de stabilité pour le choix du gain K_1 sur le cas du lanceur

Le choix du séquencement sur le gain K_1 oblige à reprendre les composantes stables pour évaluer l'intervalle de stabilité sur K_2 . Sur la Fig. 5.10, l'intervalle obtenu pour la valeur $K_1(T1) = 4$ est $K_2 \in [1.04, 1.38]$.

En répétant encore une fois la même procédure sur les autres instants, l'enveloppe sur le gain K_2 est obtenue et l'interpolation linéaire peut se faire à l'intérieur de celle-ci. Il est à noter que le choix de séquencement réalisé sur K_1 est arbitraire, il a été choisi de se placer le

plus possible à l'intérieur des composantes pour une question de robustesse. Ce choix influence directement le séquencement sur le gain K_2 puisque les composantes stables ne sont pas des rectangles.

Le résultat du séquencement est illustré en Fig. 5.12. le gain K_1 a été interpolé en premier, puis, sur la base de ces valeurs, le gain K_2 a été interpolé à son tour.

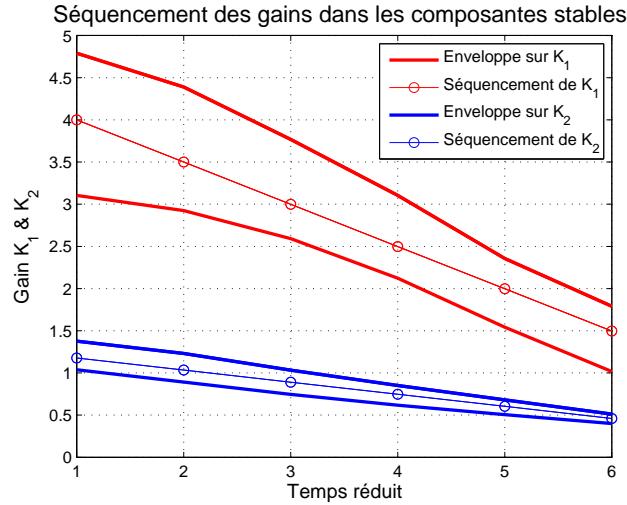


Figure 5.12 Enveloppe de stabilité pour les gains K_1 et K_2 sur le cas du lanceur

5.4 Application au cas du lanceur

Les méthodes exposées plus tôt sont maintenant appliquées sur le cas du lanceur. Il est d'usage de faire la synthèse d'un correcteur de lanceur en deux étapes : la stabilisation de la partie rigide est réalisée avec le respect de performances, puis la dynamique des modes flexibles est ajoutée au modèle pour vérifier la conservation de la stabilité. Le contrôle n'est pas repris pour le modèle souple du lanceur, mais un filtre est plutôt conçu pour atténuer l'effet des modes flexibles et les maintenir stables (Clément *et al.*, 2005; Baldesi, 2008; Hernandez, 2008).

La méthode de synthèse par applications gardiennes permet d'envisager une approche différente. Un premier correcteur basé sur le modèle rigide du lanceur sera considéré, puis les modes flexibles seront ajoutés et pris en compte dans une nouvelle synthèse sur le modèle complet cette fois. L'avantage des applications gardiennes est de pouvoir visualiser les réglages des gains qui conserveront la stabilité de ces modes tout en assurant les performances de la partie rigide.

5.4.1 Cas du lanceur rigide

Le modèle du lanceur rigide se base sur la dynamique de la structure rigide d'ordre 2 et de l'actionneur d'ordre 4. Cependant une analyse des valeurs singulières du modèle montre que la structure rigide est largement dominante, de sorte que le modèle se ramène à un second ordre de la forme (5.6).

$$F_{rig}(s) = -5.5 \cdot 10^{-4} \frac{(s + 202.8)(s - 49.3)}{(s - 2.063)(s + 2.063)} \quad (5.6)$$

Structure de correction

Le contrôle d'un tel système s'apparente à celui d'un pendule inverse. Une avance de phase est nécessaire pour apporter la stabilité d'après un simple lieu des racines. Il est donc approprié d'envisager premièrement un correcteur proportionnel-dérivé (PD) sur la mesure d'attitude θ du lanceur. Cette structure est adoptée dans un cas général dans Blakelock (1965), avec parfois un retour supplémentaire sur l'incidence comme dans Greensite (1970). Dans les deux cas, un retour en position et en vitesse par l'intermédiaire respectivement de gyroscopes et de gyromètres permet de stabiliser un lanceur rigide.

De plus, sachant que l'incidence doit être minimisée, une action intégrale est nécessaire pour amener la précision requise face aux rafales de vent. La structure retenue est donc un simple correcteur proportionnel-intégral-dérivée (PID). Le réglage du pôle et du gain intégral se fait à partir des applications gardiennes. L'expression du contrôleur est donnée en (5.7), avec quatre variables de synthèse : les gains K_p , K_d , K_i et le pôle p_1 . Il s'inscrit dans la boucle de contrôle présentée auparavant et rappelée en Fig. 5.13.

$$C(s) = K_p + \frac{K_d s}{s/p_1 + 1} + \frac{K_i}{s} = \frac{K_i + K_1 s + K_2 s^2}{s(s/p_1 + 1)} \quad (5.7)$$

Le but principal dans la synthèse par applications gardiennes étant d'obtenir des intersections entre les différentes composantes stables, les paramètres doivent être réglés de manière à maximiser l'aire de ces intersections. Ainsi, considérant que les deux gains les plus influents sur le contrôle sont ici le gain proportionnel K_p et le gain dérivé K_d , le gain intégral K_i et la position du pôle p seront fixés *a priori*. En cherchant à maximiser l'aire de l'intersection des composantes stables les valeurs retenues sont $p_1 = 30$ et $K_i(T) = \frac{2.5-6.5}{5}(T-1) + 6.5$. Le gain intégral a été interpolé linéairement depuis le temps T_1 jusqu'au temps T_6 pour conserver les intersections à chaque temps de vol. L'interpolation sur la position du pôle n'était pas nécessaire pour assurer l'existence de ces intersections.

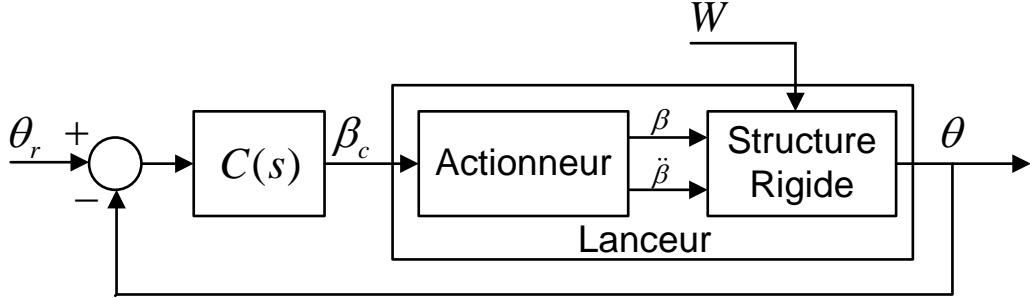


Figure 5.13 Structure de la boucle de contrôle

Réglage des gains

Le réglage des deux gains restants peut se faire de façon graphique avec la méthode de synthèse présentée plus haut. Pour ce faire, les applications gardiennes considérées sont données en (5.8,5.9).

$$\nu_\alpha(A) = \det(A - \alpha I) \det((A - \alpha I) \odot I) \quad (5.8)$$

$$\nu_\xi(A) = \det(A) \det(A^2 \odot I + (1 - 2\xi^2)A \odot A) \quad (5.9)$$

Elles sont évaluées sur la matrice en boucle fermée

$$A_{BF} = A - BKC = A - BC_1 K_1 - BC_2 K_2 - BC_3 K_i$$

pour le retour $K = [K_1, K_2, K_i]$ où $K_1 = K_p + K_i/p$ et $K_2 = K_d + K_p/p$.

Le modèle d'état est alors augmenté avec les deux pôles du correcteur. Les variables des applications sont les deux gains K_1 et K_2 . Les expressions obtenues pour ces variables sont données (5.10,5.11).

$$\begin{aligned} \nu_\alpha(K_1, K_2) &= \det(A_{\alpha,i} - BC_1 K_1 - BC_2 K_2) \\ &\times \det(A_{\alpha,i} \odot I - BC_1 \odot I K_1 - BC_2 \odot I K_2) \end{aligned} \quad (5.10)$$

$$\begin{aligned} \nu_\xi(K_1, K_2) &= \det(A_i - BC_1 K_1 - BC_2 K_2) \\ &\times \det(M_{00} + M_{10} K_1 + M_{01} K_2 + M_{11} K_1^2 + M_{12} K_1 K_2 + M_{22} K_2^2) \end{aligned} \quad (5.11)$$

en notant

$$\begin{aligned}
 \mu &= 1 - 2\xi^2 \\
 A_i &= A - BC_3 K_i \\
 A_{\alpha,i} &= A - BC_3 K_i - \alpha I \\
 M_{00} &= A_i^2 \odot I + \mu(A_i \odot A_i) \\
 M_{10} &= -(A_{\alpha,i}BC_1 + BC_1A_{\alpha,i}) \odot I - 2\mu(BC_1 \odot A_i) \\
 M_{01} &= -(A_{\alpha,i}BC_2 + BC_2A_{\alpha,i}) \odot I - 2\mu(BC_2 \odot A_i) \\
 M_{11} &= (C_1B)BC_1 \odot I + \mu(BC_1 \odot BC_1) \\
 M_{12} &= ((C_1B)BC_2 + (C_2B)BC_1) \odot I + 2\mu(BC_1 \odot BC_2) \\
 M_{22} &= (C_2B)BC_2 \odot I + \mu(BC_2 \odot BC_2)
 \end{aligned}$$

Pour un maillage donné de l'espace $[K_{1,i}, K_{2,j}]$, ces applications gardiennes sont calculées en chaque point. Les intersections des composantes stables de chaque modèle `nom`, `hf` et `lf` sont ensuite obtenues puis l'intersection entre elles est colorée en jaune. Le résultat est donné en Fig. 5.16 avec la légende complète donnée en 5.14. Pour des raisons de lisibilité seules les légendes principales ont été laissées sur les graphes.

Le séquencement se fait ensuite à travers la méthode présentée plus haut, en commençant par le gain K_1 puis en terminant par le gain K_2 . Le résultat obtenu est présenté en Fig. 5.15. Les valeurs séquencées des gains K_p , K_d et K_i sont résumées dans le Tableau 5.1.



Figure 5.14 Légendes utilisées dans la Fig. 5.16

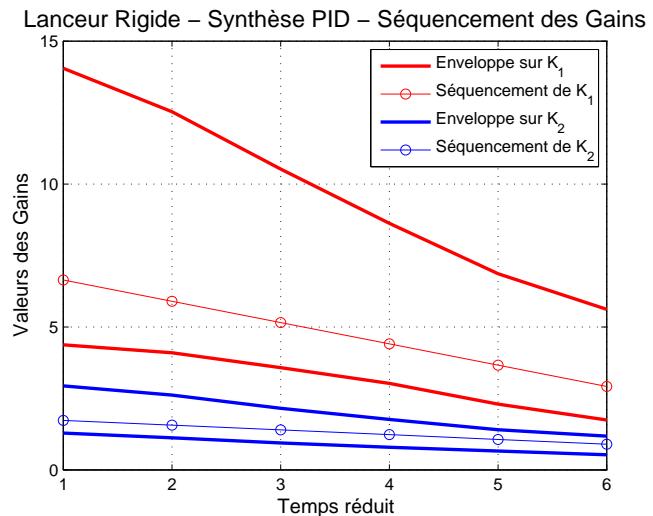


Figure 5.15 Enveloppe de stabilité des gains K_1 et K_2 pour le lanceur rigide

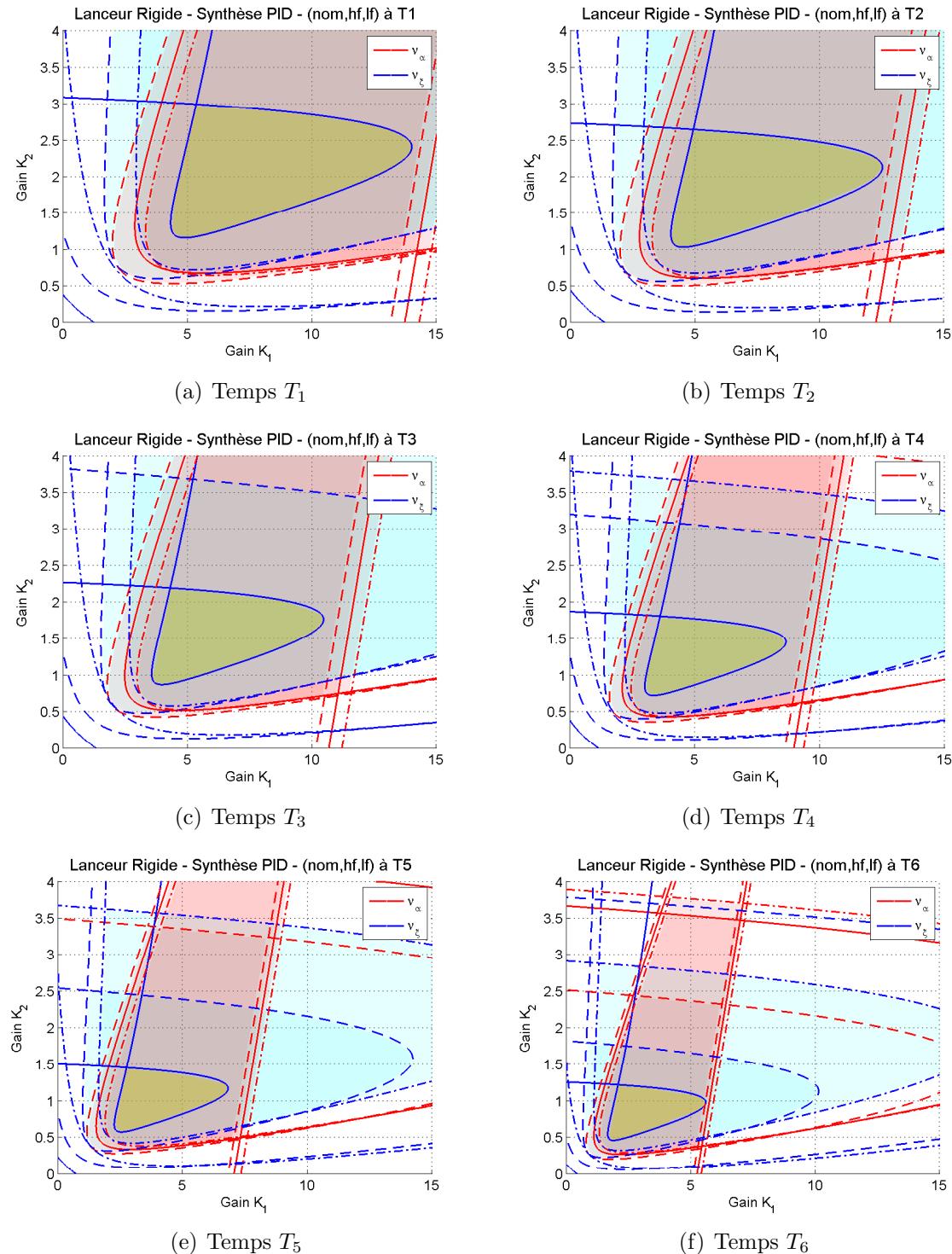


Figure 5.16 Intersections des composantes stables sur les trois modèles pour chaque temps

Tableau 5.1 Valeurs séquencées des gains pour le correcteur rigide PID

Gains \ Temps	T_1	T_2	T_3	T_4	T_5	T_6
K_p	6.69	5.81	4.93	4.06	3.18	2.30
K_d	1.50	1.34	1.18	1.02	0.86	0.70
K_i	6.50	5.70	4.90	4.10	3.30	2.50

Validation du correcteur

La validation du correcteur séquencé précédent passe par une simulation temporelle basé sur un modèle LFT interpolant les six modèles d'état. Elle présente les réponses du lanceur à une rafale de vent d'environ $20m/s$ apparaissant à $36s$. L'incidence, l'attitude et la commande d'orientation de la poussée sont présentées.

L'incidence doit rester dans l'intervalle $\pm 3^\circ$ selon les contraintes fixées par ASTRIUM-ST. Aucune spécification n'est donnée sur l'attitude ou sur la commande angulaire. Il faut cependant noter que l'attitude θ est égale à l'incidence α en l'absence de vent, ce qui motive à garder cette grandeur la plus faible possible. Pour donner un ordre d'idée sur la commande de la tuyère β_c , sur le lanceur européen VEGA la consigne maximale acceptable est de $\pm 6^\circ$.

Les réponses sont présentées en Fig. 5.17.

L'incidence respecte la contrainte fixée sur l'ensemble du vol excepté au niveau de l'apparition de la rafale de vent. Le dépassement maximal sur les trois modèles est de 3.6° . Il dure moins d'une seconde et intervient sur le modèle incertain `hf`. Il faut aussi considérer que ce réglage des gains est le meilleur compromis obtenu avec cette structure de correction pour respecter à la fois toutes les contraintes sur le placement des pôles et sur la limite de l'incidence.

L'attitude est maintenue très faible avec un dépassement maximal inférieur à 0.5° ce qui confirme que le lanceur maintient sa verticale. Pour la commande sur la tuyère, l'angle maximal de consigne est de 5° , ce qui est acceptable.

Enfin, les marges obtenues au cours du vol sont résumées en Fig. 5.18. Elles sont toutes respectées largement, y compris pour les deux modèles incertains `hf` et `1f` ce qui confirme la robustesse de notre loi de commande.

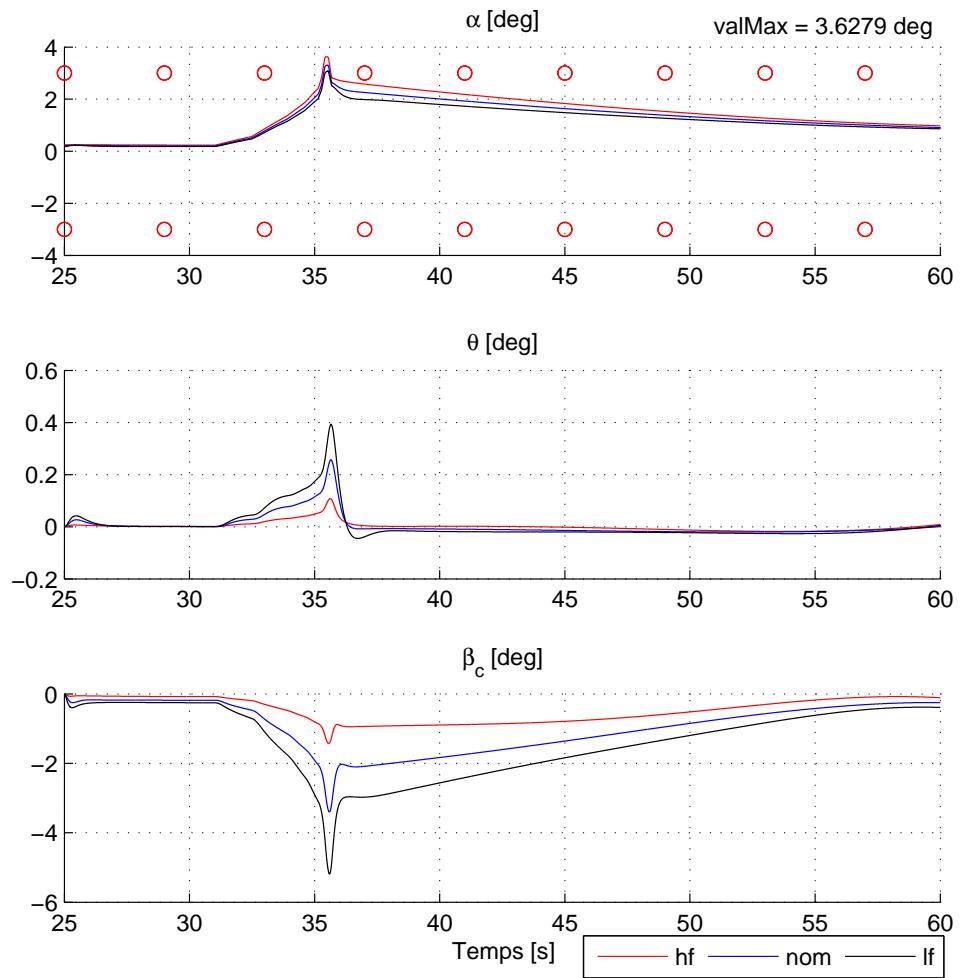


Figure 5.17 Simulations temporelles pour le lanceur rigide

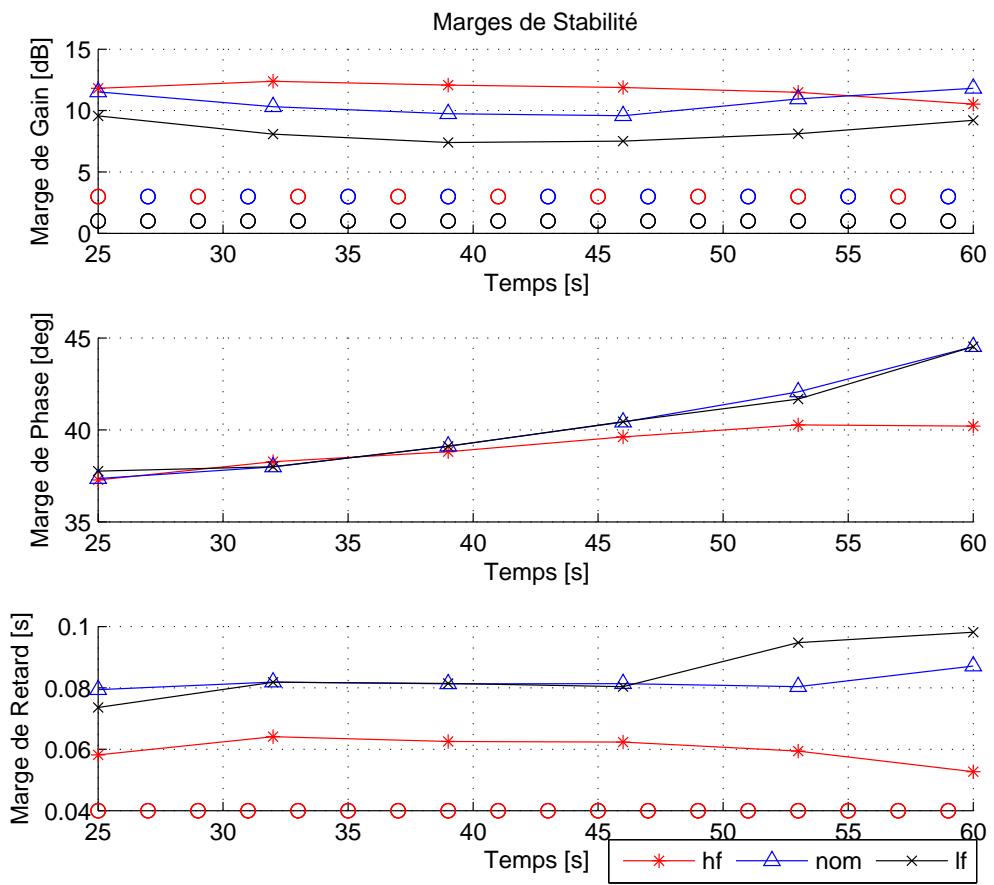


Figure 5.18 Evolution des marges au cours du vol pour le lanceur rigide

5.4.2 Cas du lanceur flexible

La prise en compte de la structure souple est nécessaire pour augmenter la fidélité du modèle. Les modes introduits ont cependant une tendance à se déstabiliser si on conserve la correction réalisée sur un lanceur supposé rigide. La déformation du lanceur vient modifier les bras de levier des efforts de portance et de poussée, ainsi que la mesure d'attitude. Les gyroscopes étant solidaires de la structure du lanceur leur mesure s'effectue évidemment sur la structure flexible.

Une preuve de la déstabilisation apportée par la structure souple est donnée en Fig. 5.19. Ce lieu des racines montre que les modes flexibles introduits, originellement stables et mal amortis, se sont complètement déstabilisés avec la structure de correction précédente. Il y a donc lieu de la revoir pour l'améliorer et atténuer l'effet de ces pôles.

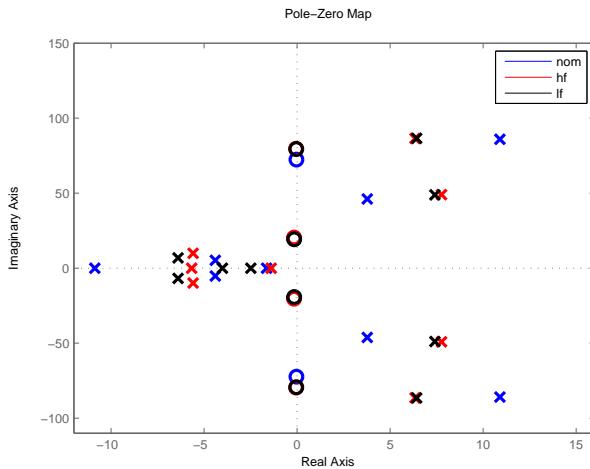


Figure 5.19 Lieu des racines du lanceur flexible avec correcteur PID

Structure de correction

Il est commun de contrôler « en phase » le premier mode flexible (Greensite, 1970; Clément *et al.*, 2005). Ce dernier possède une pulsation suffisamment faible pour se retrouver dans la bande passante du contrôleur. Le contrôle en phase consiste alors à utiliser un filtre pour ajouter de la phase au niveau de ce mode et atténuer ainsi son effet déstabilisant. Le filtre utilisé dans cette étude s'inspire du filtre elliptique présenté dans Hernandez (2008); Bérard *et al.* (2008), caractérisé par une coupure plus franche que les autres filtres. Son expression est donnée en (5.12) et son lieu des racines en Fig. 5.20.

$$F_{ell}(s) = \frac{0.056(s^2 + 1456)(s^2 + 5860)}{(s^2 + 29.09s + 535.5)(s^2 + 5.55s + 949.1)} \quad (5.12)$$

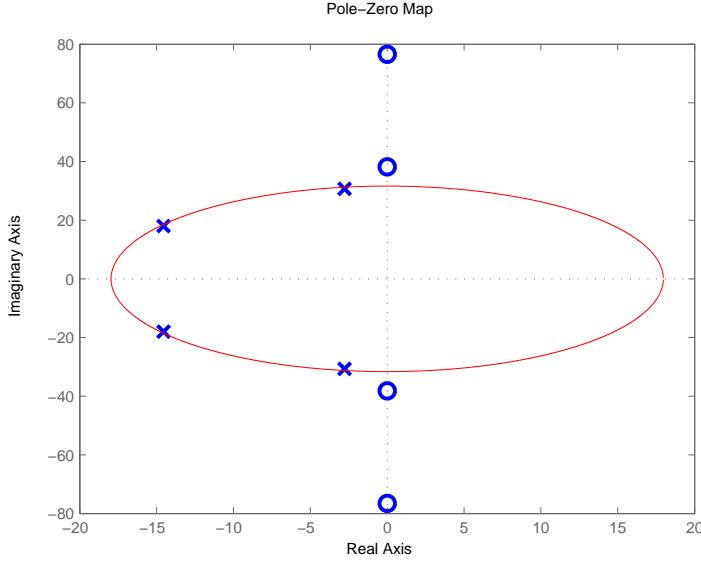


Figure 5.20 Lieu des racines du filtre elliptique complet

Cependant une étude du filtre sur le problème du lanceur a permis de mettre en évidence qu'un filtre plus simple d'ordre 2 couplé avec du roll-off avait le même effet. Il permet de conserver les intersections entre les composantes stables sans réduire leurs surfaces. Le choix a donc été fait de prendre un filtre d'ordre 2 seulement dont l'effet se concentre sur le premier mode flexible, tandis qu'un pôle simple est ajouté pour filtrer les hautes fréquences.

Ce pôle supplémentaire permet d'amener du « roll-off » d'ordre 1 pour agir sur le second mode flexible. Celui-ci est fortement atténué en amplitude par ce pôle supplémentaire et son effet déstabilisant disparaît.

La structure de correction est donc augmentée pour aboutir à l'expression (5.13). Le filtre d'ordre 2 a été synthétisé par les applications gardiennes, par essais-erreurs pour obtenir les intersections les plus grandes possibles entre les composantes stables. Son expression est fournie en (5.14). Le même procédé a permis de placer le pôle supplémentaire en $p_2 = 30$.

$$C(s) = F(s) \frac{K_i + K_1 s + K_2 s^2}{s(s/p_1 + 1)(s/p_2 + 1)} \quad (5.13)$$

$$F(s) = \frac{580}{1900} \frac{s^2 + 1900}{s^2 + 7s + 580} \quad (5.14)$$

L'effet du filtre puis du roll-off sont présentés en Fig. 5.21.

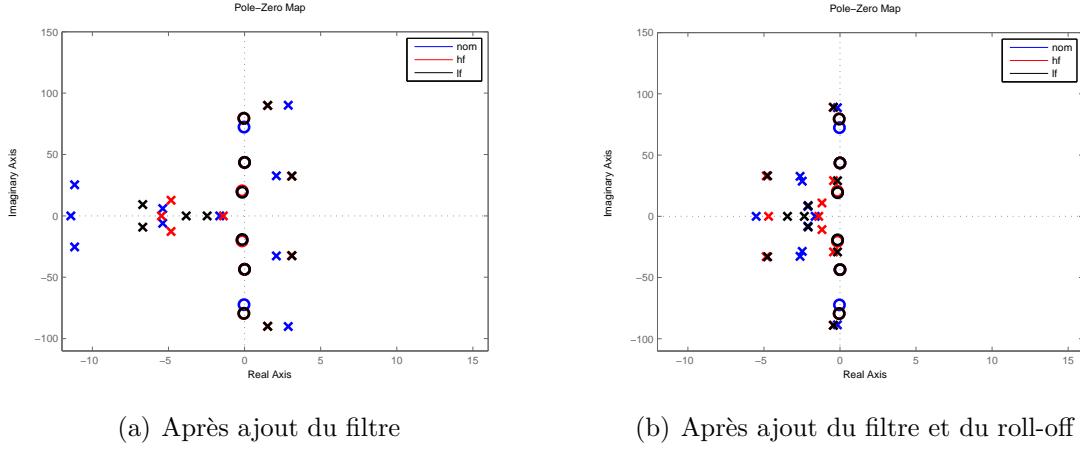


Figure 5.21 Effet du filtre d'ordre 2 et du roll-off sur le lieu des racines

Les réglages précédents ont permis de fixer les valeurs du filtre, des deux pôles du correcteur en $p_1 = -30$ et $p_2 = -30$, puis de séquencer la valeur du gain intégral K_i pour obtenir $K_i(T) = \frac{2.5-6.5}{5}(T-1) + 6.5$. Il reste uniquement les gains K_1 et K_2 à régler par les applications gardiennes. Ce correcteur sera abrégé « PIDFRo » dans les figures qui suivent, pour rappeler qu'il contient une partie PID, un Filtre et un pôle pour le Roll-off.

Réglage des gains

La méthode de synthèse par approche graphique s'appuie cette fois-ci sur les trois applications gardiennes suivantes :

$$\nu_\alpha(A) = \det(A - \alpha I) \det((A - \alpha I) \odot I) \quad (5.15)$$

$$\nu_\xi(A) = \det(A) \det(A^2 \odot I + (1 - 2\xi^2)A \odot A) \quad (5.16)$$

$$\nu_\beta(A) = \det(A - \beta I) \det((A - \beta I) \odot I) \quad (5.17)$$

Elles sont évaluées sur le modèle d'état flexible augmenté, contenant les trois pôles du correcteur et les sorties en position, dérivée et intégrale. Les expressions analytiques obtenues en fonction des gains K_1 et K_2 sont identiques à (5.10, 5.11). Il suffit de prendre le modèle d'état du lanceur flexible, et de remplacer α par β pour obtenir la troisième application gardienne.

La synthèse sur les six temps de vol avec les trois modèles renvoie des graphes très denses et difficiles à analyser au premier coup d'œil. Les composantes des applications gardiennes ont donc été tracées séparément, tout en conservant la superposition sur les trois modèles.

L'annexe A présente les graphes pour ν_α , l'annexe B présente les graphes pour ν_β , l'annexe C présente les graphes pour ν_ξ et enfin l'annexe D superpose tous ces graphes pour obtenir les zones de réglages des gains.

Le séquencement fournit les interpolations de la Fig. 5.22.

Les valeurs obtenues pour les gains K_p , K_d et K_i sont réunies dans le Tableau 5.2.

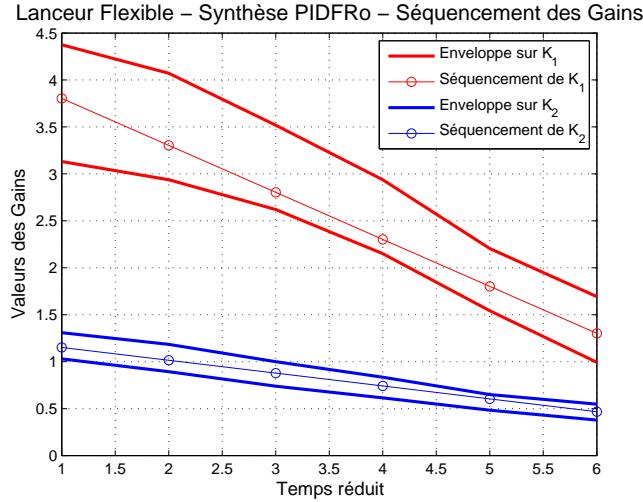


Figure 5.22 Enveloppe de stabilité des gains K_1 et K_2 pour le lanceur flexible

Tableau 5.2 Valeurs séquencées des gains du correcteur PIDFRo

Gains \ Temps	T_1	T_2	T_3	T_4	T_5	T_6
K_p	3.71	3.24	2.76	2.29	1.81	1.34
K_d	1.10	0.97	0.84	0.71	0.58	0.45
K_i	3.70	3.16	2.62	2.08	1.54	1.00

Validation du correcteur

La simulation de validation du correcteur flexible est exactement la même que pour le correcteur rigide : la même rafale de vent est considérée et les mêmes limites doivent être respectées pour l'incidence et les marges.

Les réponses temporelles sont présentées en Fig. 6.15. Le même constat s'impose sur l'incidence du lanceur, qui reste dans la limite $\pm 3^\circ$ sur quasiment tout le vol, avec un dépassement de 3.7° cette fois-ci. Ce résultat reste une performance en soi puisqu'il a été possible d'ajouter la dynamique des modes flexibles et de réaliser un correcteur qui atteigne le même résultat

que pour le modèle rigide. L'attitude et la consigne sur l'orientation de la tuyère restent aussi acceptables.

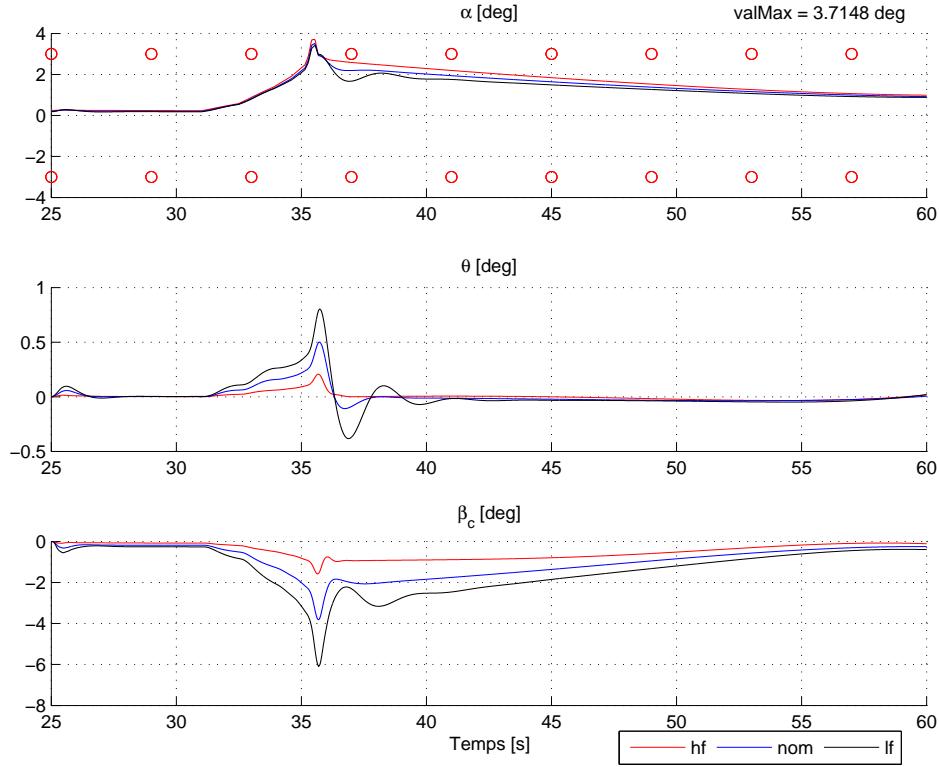


Figure 5.23 Simulations temporelles pour le lanceur flexible

Pour finir, les marges ont été calculées sur le modèle flexible et sont présentées en Fig. 6.16. Elles sont toujours respectées mais se sont dégradées par rapport au cas rigide avec des marges de gains très proches des limites fixées sur certains temps. De même les marges de phase ont légèrement diminué tandis que les marges de retard n'ont quasiment pas changées, ce qui indique que la fréquence de coupure a légèrement diminuée.

Ces changements sur les marges témoignent directement de l'effet déstabilisant des modes flexibles. Il a pu être atténué grâce à la structure de correction utilisée.

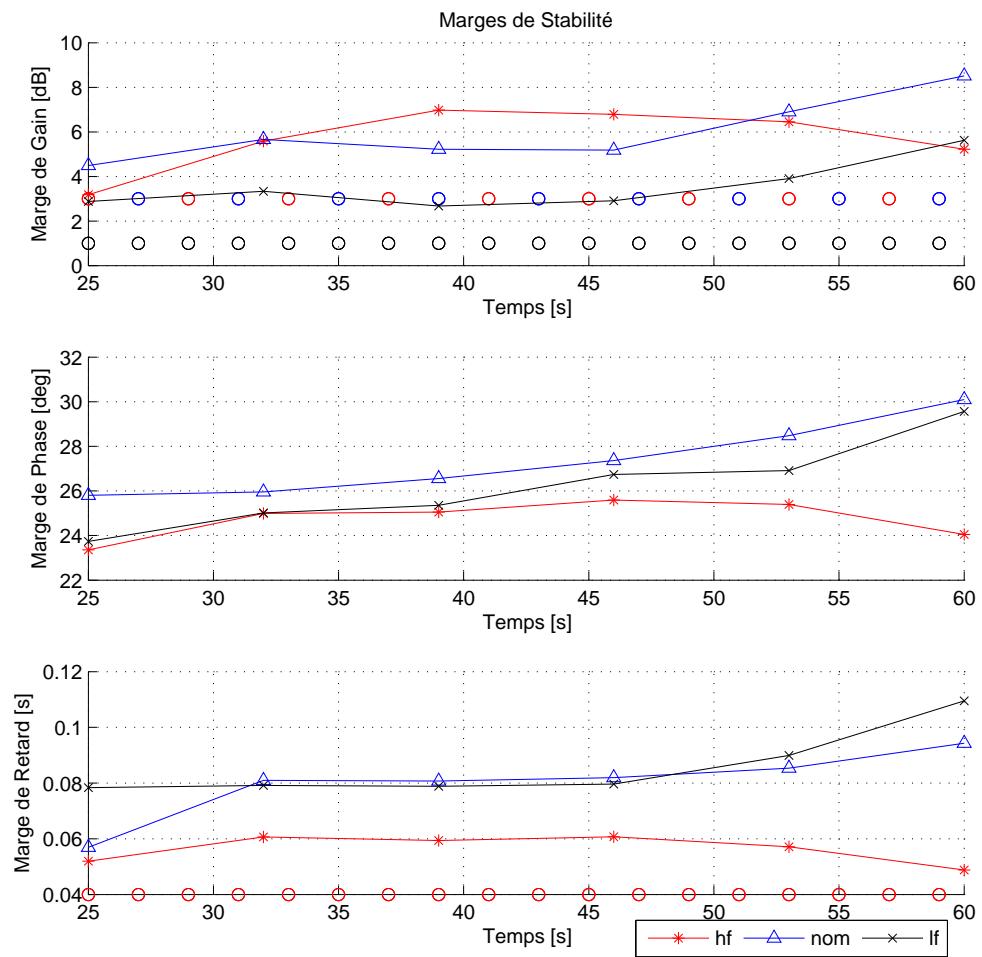


Figure 5.24 Evolution des marges au cours du vol pour le lanceur flexible

CHAPITRE 6

SYNTHÈSE PAR OPTIMISATION

6.1 Introduction

La méthode de synthèse par approche graphique présente un grand intérêt pour appréhender le concept des composantes des applications gardiennes. Elle permet aussi de réaliser une loi séquencée de correction en se limitant cependant à deux paramètres variables. Cette approche est donc restreinte dès que le nombre de paramètres du contrôleur augmente. Il faut régler ces variables supplémentaires *a priori* en s'appuyant sur les intersections des composantes stables. Il a donc été envisagé une approche beaucoup plus générale à travers Saussié *et al.* (2010), basée sur la recherche du centre « géométrique » de la composante stable.

Cette nouvelle approche de synthèse peut prendre en compte un plus grand nombre de paramètres. Dans le cas du lanceur, il pourrait être judicieux de considérer les positions des pôles du contrôleur ou les zéros et pôles du filtre directement dans la phase de synthèse. La méthode décrite dans Saussié *et al.* (2010) présente une recherche d'un point intérieur à la composante stable, en se déplaçant dans la direction des gains, et amenant une progression souvent lente vers le centre géométrique. Ce chapitre a donc pour but d'appliquer des méthodes d'optimisation adaptées, pour obtenir cette fois le minimum de l'application gardienne dans la composante stable.

Dans cette optique, un algorithme de descente basé sur la méthode de quasi-Newton BFGS¹ a été développé (Nocedal et Wright, 2006). Le programme permettant cette optimisation a été entièrement réalisé sur le logiciel Matlab en reprogrammant toutes les étapes pour permettre d'y incorporer les spécificités des applications gardiennes. L'algorithme complet sera exposé après un bref rappel en optimisation. Il sera ensuite appliqué sur le cas du lanceur, afin d'en comparer les performances avec les résultats de la synthèse par approche graphique.

1. Broyden-Fletcher-Goldfarb-Shanno

6.2 Rappels en optimisation

L'optimisation joue un rôle crucial en commande des systèmes. Elle apparaît par exemple pour la commande LQG en cherchant à minimiser un critère sur les efforts de commande ou les erreurs de suivi. Le fait de minimiser une fonction a pour but de récupérer la valeur optimale et les coordonnées du minimum. Dans le cas des applications gardiennes, l'optimisation sera utilisée pour permettre de rentrer à l'intérieur des composantes stables. Un bref rappel sur les méthodes d'optimisation classique sera fait en insistant sur les méthodes de Newton et sur les différentes stratégies pour la recherche linéaire.

6.2.1 Algorithme général

Le problème d'optimisation le plus général s'écrit comme suit :

$$\left\{ \begin{array}{l} \min_{X \in \Omega} f(X) \\ \text{s.c. } h_i(X) = 0, \forall i = 1, \dots, m \\ c_j(X) \geq 0, \forall j = 1, \dots, p \end{array} \right. \quad (6.1)$$

Dans le cas présent, aucune contrainte ne sera considérée directement dans le problème de minimisation, de sorte qu'il reste à minimiser une fonction sur un ensemble Ω .

Dès lors, l'algorithme général pour résoudre un tel problème s'écrit :

- **Itération 0**

Choix d'un point de départ X_0 , et d'une tolérance d'arrêt tol , initialisation $k = 0$.

- **Itération k**

– **Si** $\|\nabla f_k\| \leq tol$

$X^* \leftarrow X_k$ et **STOP**

– **Sinon**

Trouver un « meilleur » X_{k+1} (i.e. $f(X_{k+1}) \leq f(X_k)$)

Mises à jour : $X_k \leftarrow X_{k+1}$, $\nabla f_k \leftarrow \nabla f(X_{k+1})$, $k \leftarrow k + 1$

- **Retour à Itération k**

De nombreux raffinements sont possibles sur la tolérance d'arrêt, en prenant une tolérance absolue, relative ou même une combinaison des deux. Cette considération permet d'adapter la précision pour effectuer une recherche grossière quand le point est loin du minimum et pour la raffiner quand il s'en rapproche.

Ensuite, le choix du « meilleur » point se fait le plus souvent en choisissant une direction d_k dite de descente, avant de choisir la taille du pas t_k à faire dans cette direction :

$X_{k+1} = X_k + t_k d_k$. La direction de descente est admissible si elle permet de faire décroître la fonction objectif (i.e. $\nabla f_k^T \cdot d_k < 0$).

6.2.2 Recherche linéaire

Le choix du pas t_k se fait selon une recherche linéaire dans la direction de descente d_k . En notant $\phi(t) = f(X_k + t \cdot d_k)$, on obtient une fonction d'une variable décroissante au voisinage de 0^+ . Il faut alors trouver une valeur de t qui minimise cette fonction ϕ .

Backtracking Une méthode élémentaire consiste à partir d'un point t_{max} , à vérifier si $\phi(t_{max}) \leq \phi(0)$, puis à diviser t_{max} par 2 si cette condition n'est pas satisfaite. Cette étape est répétée jusqu'à obtenir un point dont la valeur de ϕ est plus faible. D'après le choix de la direction de descente et par continuité de f , l'existence d'un tel point est assurée.

Une telle méthode peut entraîner une oscillation autour d'un minimum si jamais le pas est trop grand. Comme illustré en Fig. 6.1, la méthode d'Armijo permet de trouver plus efficacement un point dans la direction d_k . La condition associée permet de considérer des points éloignés du point courant X_k si ces derniers permettent une décroissance suffisante de la fonction. Cette condition est donnée en (6.2) pour $0 < \alpha < 1$.

$$f(X_k + t d_k) \leq f(X_k) + \alpha t \nabla f^T(X_k) \cdot d_k \quad (6.2)$$

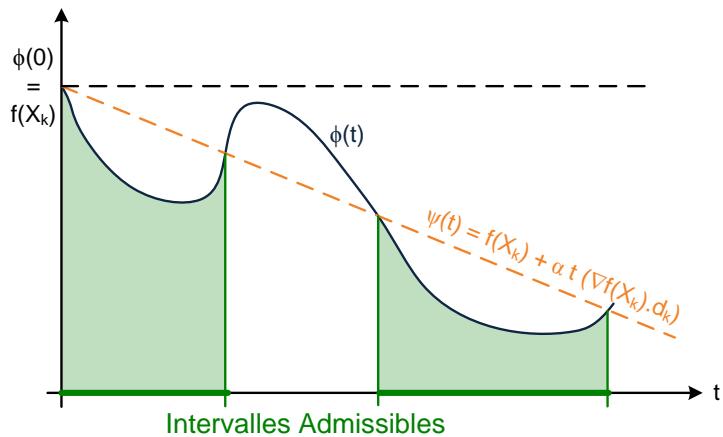


Figure 6.1 Illustration du rôle de la condition d'Armijo

En associant cette condition avec le backtracking, un point de départ t_{max} est d'abord

sélectionné. S'il vérifie la condition, il est choisi, sinon il est divisé par 2 puis la fonction est à nouveau évaluée. Typiquement, le paramètre α est pris égal à 1/1000 pour permettre d'aller chercher des points suffisamment éloignés (Nocedal et Wright, 2006).

6.2.3 Méthodes de Newton

Le choix de la direction de descente d_k admet de nombreuses variantes. Les plus simples utilisent l'opposé du gradient pour choisir la direction de « plus forte pente » (« steepest descent direction »). D'autres, plus sophistiquées, utilisent les anciennes directions pour obtenir un ensemble de directions *conjuguées* : la méthode des gradients conjugués. Cette méthode permet d'obtenir le minimum d'une fonction quadratique de type $f(X) = -b^T \cdot X + \frac{1}{2} X^T \cdot A \cdot X$ dans \mathbb{R}^n en au plus n itérations. Elle est souvent utilisée pour obtenir la direction de descente dans ce que l'on appelle les méthodes de Newton.

Ces dernières sont basées sur une approximation locale de la fonction par une fonction quadratique. Un développement de Taylor autour du point courant X_k donne l'équation (6.3).

$$f(X) = f(X_k) + \nabla f(X_k)^T \cdot (X - X_k) + \frac{1}{2} (X - X_k)^T \cdot \nabla^2 f(X_k) \cdot (X - X_k) + \dots \quad (6.3)$$

Le minimum du développement tronqué se situe au point X^* où la dérivée s'annule, soit en $X^* = X_k - (\nabla^2 f(X_k))^{-1} \cdot \nabla f(X_k)$. Ce point fournit donc une direction de descente en considérant $d_k = X^* - X_k$. Cette direction permet effectivement de faire décroître la fonction objectif si la hessienne en X_k est définie positive. Ceci implique que le point de départ de l'algorithme global d'optimisation soit assez proche du minimum, où la hessienne est bien définie positive par définition.

On appelle alors méthode de Newton toute méthode qui cherche la direction de descente en résolvant le système $\nabla^2 f(X_k) d_k = -\nabla f(X_k)$. De nombreuses variantes existent là encore, avec la méthode des gradients conjugués ou, plus généralement, les méthodes de Krylov pour des systèmes de grande dimension (Nocedal et Wright, 2006).

6.2.4 Méthodes de Quasi-Newton

Si la hessienne de la fonction objectif n'est pas disponible ou si son évaluation réclame des coûts trop élevés en terme de calculs, il est préférable de se tourner vers des méthodes de type « quasi-Newton ». Elles vont reprendre les méthodes de Newton en approximant la hessienne par l'équation de la sécante, donnée en (6.4). Dans cette équation, l'inconnue est la matrice B_k qui approxime la hessienne.

$$B_k (X_{k+1} - X_k) = \nabla f(X_{k+1}) - \nabla f(X_k) \quad (6.4)$$

Pour pouvoir conserver l'information accumulée dans les approximations précédentes, la résolution de l'équation (6.4) passe par la formulation d'un sous-problème d'optimisation. La solution du problème donné en (6.5) peut s'obtenir sous forme analytique pour obtenir la formule de Broyden-Fletcher-Goldfarb-Shanno (BFGS) en (6.6).

$$\left\{ \begin{array}{ll} \min_{B \in \mathbb{R}^{n \times n}} & \|B - B_k\| \\ \text{s.c.} & B(X_{k+1} - X_k) = \nabla f_{k+1} - \nabla f_k \\ & B = B^T \end{array} \right. \quad (6.5)$$

En notant $s_k = X_{k+1} - X_k$ et $y_k = \nabla f_{k+1} - \nabla f_k$ on obtient la solution suivante.

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k} \quad (6.6)$$

En s'appuyant alors sur la formule de Sherman-Morrison pour inverser une somme de matrices inversibles, la formule « BFGS inverse » est obtenue en (6.7), avec $\rho_k = \frac{1}{y_k^T s_k}$ et H_k l'inverse de B_k .

$$H_{k+1} = (I - \rho_k s_k y_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T \quad (6.7)$$

Cette dernière formule permet de trouver directement la direction de Newton associée en calculant $d_k = -H_k \nabla f(X_k)$. Cet algorithme sera repris pour bâtir l'algorithme présenté plus loin pour la synthèse par optimisation des applications gardiennes.

6.2.5 Recherche linéaire modifiée

Il reste un dernier détail à régler si une méthode de quasi-Newton est utilisée. La solution donnée en (6.7) donne l'inverse de la matrice B_k supposée définie positive qui approxime la hessienne. Cette condition de positivité est héritée pour B_{k+1} si la matrice B_k est définie positive elle-même et si le produit $s_k^T y_k$ est strictement positif.

Une condition de courbure est alors ajoutée à la recherche linéaire d'Armijo pour assurer $s_k^T y_k > 0$. Associée avec la condition d'Armijo, elles constituent « les conditions de Wolfe fortes ». Elles sont données en (6.8) et illustrées en Fig. 6.2. La condition supplémentaire assure que la pente au point recherché est plus faible que celle du point courant X_k . Elle garantit ainsi que X_{k+1} est plus proche du minimum local recherché. Les paramètres doivent respecter $0 < \alpha < \gamma < 1$, avec des valeurs typiques de $\alpha = 1/1000$ et $\gamma = 0.9$ (Nocedal et Wright, 2006).

$$\begin{cases} f(X_k + t d_k) \leq f(X_k) + \alpha t \nabla f^T(X_k) \cdot d_k \\ |\nabla f(X_k + t d_k)^T d_k| \leq \gamma |\nabla f(X_k)^T d_k| \end{cases} \quad (6.8)$$

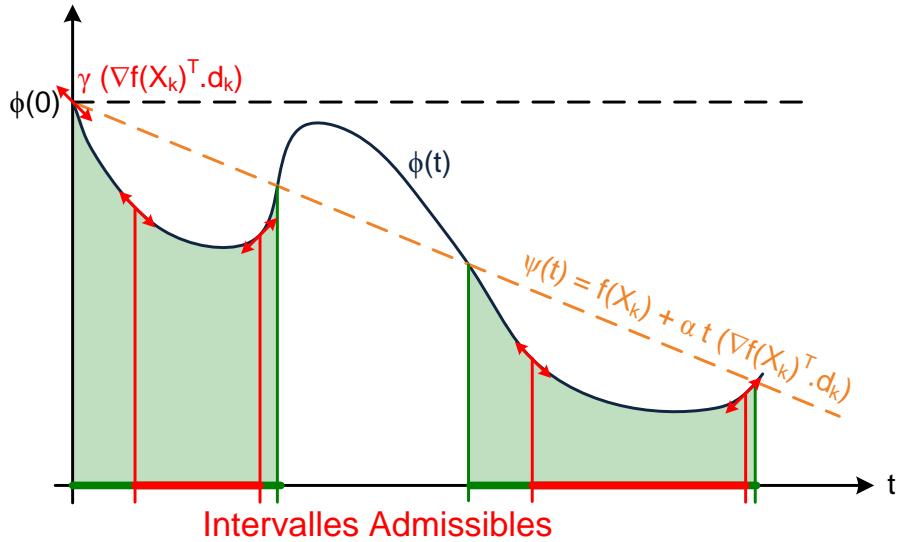


Figure 6.2 Illustration du rôle des conditions de Wolfe

6.3 Algorithme de synthèse par applications gardiennes

La synthèse par les applications gardiennes reprend un algorithme développé par Saussié *et al.* (2010). En partant du système en boucle ouverte avec une structure de correction fixée, il utilise les applications gardiennes pour trouver un réglage dans leurs composantes stables. L'inconvénient de cette précédente approche réside dans la recherche du réglage de gains à l'itération suivante. Elle repose sur une méthode *ad hoc* qui se déplace uniquement dans les directions des différents gains sans prendre avantage des applications gardiennes.

Le travail exposé dans la suite améliore cette approche en utilisant la méthode « BFGS inverse » dans la recherche de la direction de descente. Le calcul de la hessienne n'est pas aisément calculable pour les applications gardiennes, ce qui justifie l'utilisation de cette méthode de quasi-Newton. De plus, un point fondamental pour mettre en place une approche d'optimisation est de pouvoir calculer le gradient de la fonction objectif, ce qui sera détaillé pour les applications gardiennes. La recherche linéaire avec les conditions de Wolfe sera quant à elle privilégiée avec la méthode BFGS (Nocedal et Wright, 2006).

L'algorithme complet est détaillé dans un premier temps. Le calcul du gradient pour les applications gardiennes utilisées dans le cas du lanceur est développé ensuite. Pour terminer, les détails de programmation et cas particuliers relatifs aux applications gardiennes sont exposés dans la dernière partie.

6.3.1 Algorithme complet

L'algorithme général de synthèse va être donné dans cette partie. Il reprend la procédure développée par Saussié *et al.* (2010) en y ajoutant une méthode de Newton pour la recherche des réglages dans les composantes stables. Cette procédure guide, petit à petit, les pôles du système en boucle ouverte vers une zone désirée en boucle fermée.

L'algorithme se découpe en deux grandes phases :

- la stabilisation du système
- le respect des performances

La première phase utilise uniquement l'application gardienne relative à la partie réelle des pôles. Le choix de gains dans sa composante permet d'assurer dans un premier temps la stabilité du système. Il repose sur une alternance entre minimisation et mise à jour de l'application gardienne ν_α sur la stabilité. Elle est d'abord initialisée avec l'abscisse spectrale $\alpha > 0$ du système ouvert (supposé instable). Une phase d'optimisation nous amène au minimum K^* de ν_α dans la composante stable. Le réglage obtenu au minimum améliore l'abscisse spectrale α du système rebouclé car il est strictement à l'intérieur de la composante. Une mise à jour de l'application gardienne pour ce K^* avec $\alpha = \alpha(A - BK^*C)$ amène une nouvelle phase de minimisation, ...etc. L'algorithme est exposé dans le Tableau 6.1, et les notations sont données plus bas.

Cette phase termine quand le réglage obtenu stabilise le système (si la structure de correction le permet). La phase de minimisation est illustrée en Fig. 6.3, et le résultat obtenu sur le placement des pôles est donnée en Fig. 6.4.

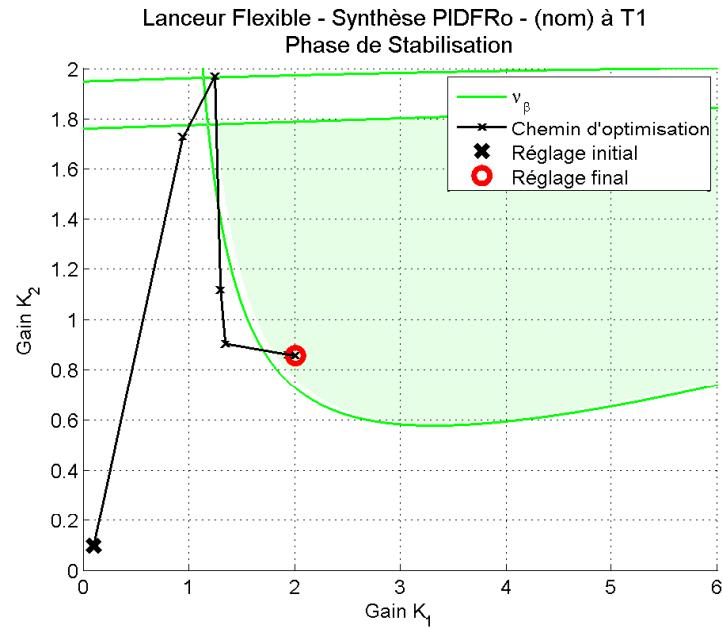


Figure 6.3 Chemin suivi par l'algorithme de stabilisation

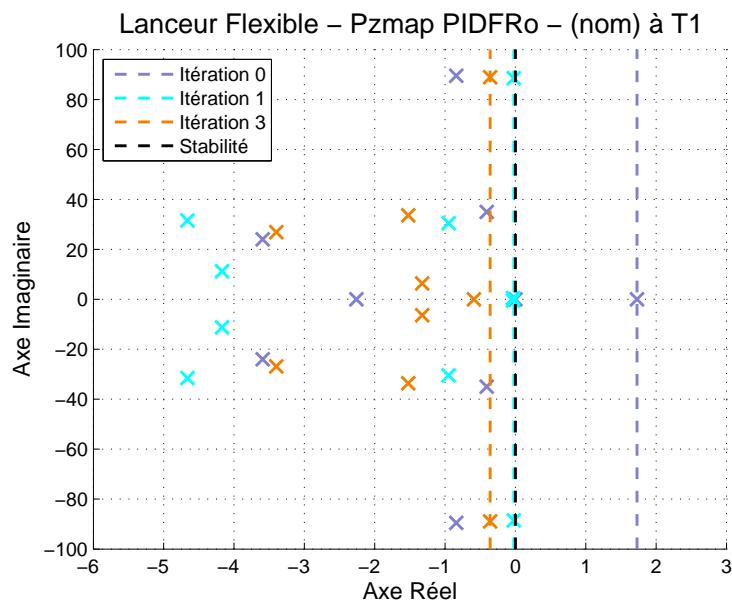


Figure 6.4 Evolution des pôles le long du chemin d'optimisation

Une fois la stabilité assurée, la considération des applications liées aux performances permet de terminer la synthèse. L'algorithme complet est donné dans le Tableau 6.2 pour le cas du lanceur. Il permet le confinement des pôles dans un ensemble Ω_d caractérisé par des limites (α_d, ξ_d) sur la partie réelle et l'amortissement des modes rigides, et une limite β_d sur la partie réelle des modes flexibles. La différence avec l'algorithme précédent se situe au niveau de l'application gardienne à considérer et au niveau des tests d'arrêt. Pour respecter toutes les performances, la fonction à minimiser est le produit des applications gardiennes liées aux contraintes sur le placement des pôles. Pour les conditions d'arrêt, elles correspondent simplement à vérifier que le réglage obtenu satisfait ces performances.

Cette phase se termine donc quand le réglage obtenu confine les pôles dans Ω_d (si la structure de correction le permet). Le respect graduel des performances est illustrée en Fig. 6.5, et l'évolution correspondante au niveau des pôles est donnée en Fig. 6.6.

Les notations utilisées dans la description des algorithmes sont détaillées ci-dessous.

Boucle principale

l	compteur
N_1	nombre maximum d'itérations
\mathbf{K}_l	valeur courante des gains
$\varepsilon_{\mathbf{K}}$	tolérance sur la taille du pas effectué
$\alpha(A)$	abscisse spectrale de la matrice A
$\zeta(A)$	amortissement minimal de la matrice A (supposée Hurwitz)

Boucle de minimisation

i	compteur
N_2	nombre maximum d'itérations
X_i	valeur courante des variables
ε	tolérance sur la norme du gradient
ν_i	valeur de la fonction objectif en X_i
$\nabla \nu_i$	gradient de la fonction objectif en X_i
H_i	inverse de l'approximation de la hessienne
d_i	direction de descente obtenue en X_i
δ	paramètre de pente dans la condition d'Armijo
γ	paramètre de courbure dans les conditions de Wolfe

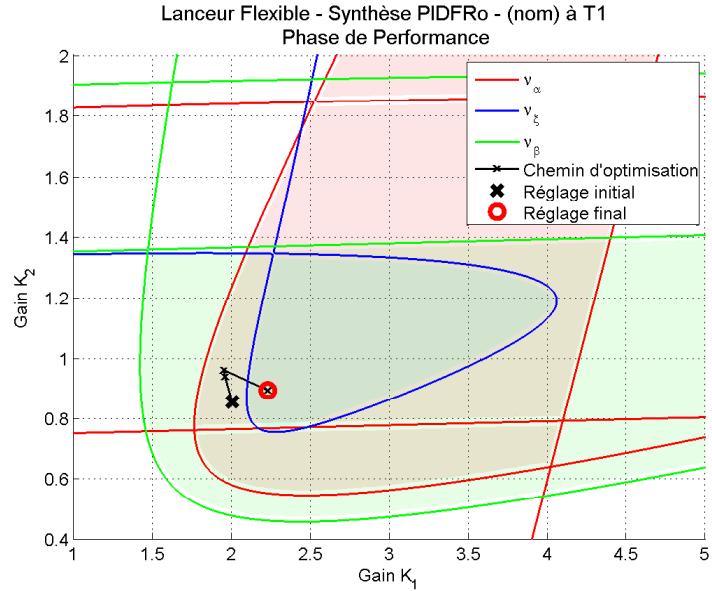


Figure 6.5 Chemin suivi par l'algorithme de stabilisation

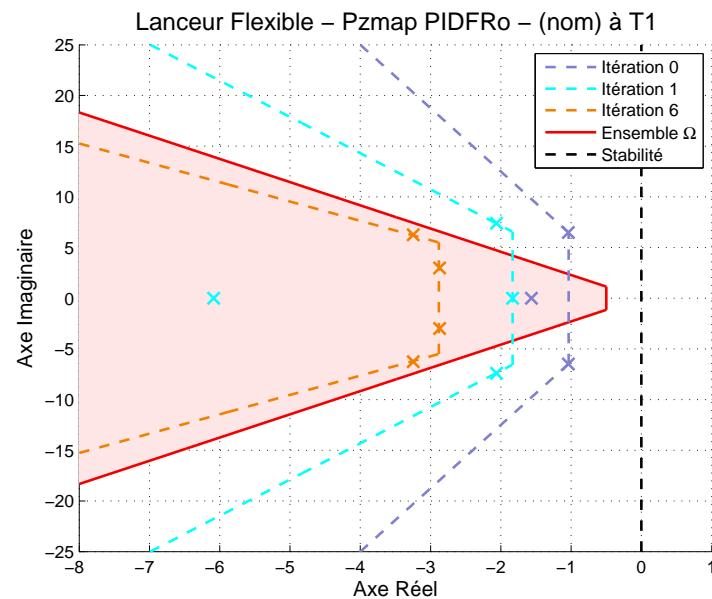


Figure 6.6 Evolution des pôles le long du chemin d'optimisation

Tableau 6.1 Algorithme de Synthèse par Optimisation – Phase de Stabilisation

-
- Choisir un réglage initial : $\mathbf{K}_0 \in \mathbb{R}^k$ (0 pour débuter en boucle ouverte)
 - Choisir les paramètres d'optimisation : $\epsilon, \epsilon_K, N_1, N_2, 0 < \delta < \gamma < 1$
 - $l \leftarrow 0$
 - **Boucle Principale** (indice l)

1. Calculer l'abscisse spectrale $\alpha_l = \alpha(A_{CL}(\mathbf{K}_l))$

Si ($\alpha_l \leq 0$) ou ($l > N_1$)

$\mathbf{K}^* \leftarrow \mathbf{K}_l$ et **STOP**

Sinon

Définition de $\nu = \nu_{\alpha_l}$

2. Poser $X_0 = \mathbf{K}_l, \nabla\nu_0 = \nabla\nu(X_0), H_0 = \frac{1}{\|\nabla\nu_0\|} I$ et $i \leftarrow 0$

3. **Boucle de Minimisation** (indice i)

3.a **Si** $\|\nabla\nu_i\| < \epsilon$ ou ($i > N_2$)

$X^* \leftarrow X_i$ et **STOP** (Aller en 4.)

3.b Obtenir la direction de descente par la méthode de Newton

$$d_i = -H_i \nabla\nu_i$$

3.c Effectuer la recherche linéaire sur t_i pour les conditions de Wolfe
(conditions d'Armijo et condition de courbure)

$$\nu(X_i + t_i d_i) \leq \nu_i + \delta t_i \nabla\nu_i^T d_i$$

$$|\nabla\nu(X_i + t_i d_i)^T d_i| \leq \gamma |\nabla\nu_i^T d_i|$$

3.d Poser $X_{i+1} \leftarrow X_i + t_i d_i$

Mises à jour	$\begin{cases} \nu_{i+1} \leftarrow \nu(X_{i+1}) \\ \nabla\nu_{i+1} \leftarrow \nabla\nu(X_{i+1}) \\ H_{i+1} = BFGS(H_i, X_{i+1} - X_i, \nabla\nu_{i+1} - \nabla\nu_i) \end{cases}$
--------------	---

$i \leftarrow i + 1$

FIN Boucle Minimisation

4. Poser $\mathbf{K}_{l+1} \leftarrow X^*$

Si $\|\mathbf{K}_{l+1} - \mathbf{K}_l\| < \epsilon_K$

$\mathbf{K}^* \leftarrow \mathbf{K}_l$ et **STOP**

$l \leftarrow l + 1$

- **FIN Boucle Principale**
-

Tableau 6.2 Algorithme de Synthèse par Optimisation – **Phase d’Optimisation**

- Choisir les paramètres de synthèse : $\Omega_d = \Omega_{\alpha_d, \xi_d, \beta_d}$, $\mathbf{K}_0 = K^*$ (issu de la stabilisation)
 - Choisir les paramètres d’optimisation : $\epsilon, \epsilon_K, N_1, N_2, 0 < \delta < \gamma < 1$
 - $l \leftarrow 0$
 - **Boucle Principale** (indice l)
 1. Calculer $\alpha_l = \alpha(A_{CL}(\mathbf{K}_l))$, $\xi_l = \xi(A_{CL}(\mathbf{K}_l))$ et $\beta_l = \beta(A_{CL}(\mathbf{K}_l))$
Si ($\alpha_l \leq \alpha_d$ et $\xi_l \geq \xi_d$ et $\beta_l \leq \beta_d$) ou ($l > N_1$)
 $\mathbf{K}^* \leftarrow \mathbf{K}_l$ et **STOP**
Sinon
 $\alpha_l = \max(\alpha_l, \alpha_d)$, $\zeta_l = \min(\zeta_l, \zeta_d)$, $\beta_l = \max(\beta_l, \beta_d)$
Définition de $\nu = \nu_{\alpha_l} \nu_{\xi_l} \nu_{\beta_l}$
 2. Poser $X_0 = \mathbf{K}_l$, $\nabla \nu_0 = \nabla \nu(X_0)$ et $H_0 = \frac{1}{\|\nabla \nu_0\|} I$
 $i \leftarrow 0$
 3. **Boucle de Minimisation**
Minimisation du ν pour obtenir X^*
 4. Poser $\mathbf{K}_{l+1} \leftarrow X^*$
Si $\|\mathbf{K}_{l+1} - \mathbf{K}_l\| < \epsilon_K$
 $K^* \leftarrow K_l$ et **STOP**
 $l \leftarrow l + 1$
 - **FIN Boucle Principale**
-

6.3.2 Calcul des gradients

Les deux phases de l'algorithme de synthèse font appel au calcul du gradient des applications gardiennes. Ces derniers reposent sur la formule de Jacobi. Elle est énoncée en (6.9) en considérant une application matricielle $F(K)$ dépendant des gains K_i . Un exemple est donné pour le cas usuel où la dépendance est linéaire en fonction des paramètres.

$$F(K) = F_0 + \sum_{i=1}^N K_i F_i$$

$$\frac{\partial \det F}{\partial K_i} = \text{tr} \left(\text{adj}(F(K)) \frac{\partial F}{\partial K_i} \right) = \text{tr} (\text{adj}(F(K)) F_i) \quad (6.9)$$

où $\text{adj}(F)$ désigne l'adjointe de la matrice F , caractérisée par la relation $\text{adj}(F)F = F\text{adj}(F) = \det F I$, et tr désigne la trace de la matrice. On prendra dans la suite la notation F^A pour désigner l'adjointe de la matrice F .

Calcul de l'adjointe L'algorithme de calcul de l'adjointe repose sur la décomposition en valeurs singulières (SVD). Il faut en effet noter que pour une matrice inversible U , l'adjointe s'obtient par l'inverse $U^A = \det(U)U^{-1}$. Partant de la décomposition SVD d'une matrice quelconque, on obtient (6.10) pour le cas réel.

$$\exists U, V \text{ unitaires (i.e. } UU^T = I, VV^T = I\text{)} \text{ et } \Sigma \text{ diagonale } |A = U\Sigma V^T \quad (6.10)$$

Dès lors, on en déduit (6.11) sachant que l'adjointe vérifie $(AB)^A = B^A A^A$, et que U et V sont inversibles avec $U^{-1} = U^T$ et $V^{-1} = V^T$.

$$A^A = (V^T)^A \Sigma^A U^A = \det(V) V \Sigma^A \det(U) U^T \quad (6.11)$$

Le calcul de Σ^A se fait à partir des valeurs singulières. $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$ donc $\Sigma^A = \text{diag}(\gamma_1, \gamma_2, \dots, \gamma_n) \triangleq \Gamma$ en notant $\gamma_i = \prod_{k=1, k \neq i}^n \sigma_k$. Au bilan, on obtient la formule (6.12) pour calculer l'adjointe d'une matrice.

$$A^A = \det(U) \det(V) V \Gamma U^T \quad (6.12)$$

Gradient de ν_α et ν_β Ces deux applications gardiennes font appel à la même formule. Elle vérifient que la partie réelle maximale (aussi appelée abscisse spectrale) du système ne dépasse pas une certaine valeur. Le travail est effectué sur ν_α et s'applique de manière identique sur ν_β .

L'application gardienne va être découpée selon les deux déterminants qui interviennent : $\nu_\alpha(A) = \nu_1(A) \nu_2(A)$ avec :

$$\begin{aligned}\nu_1(A) &= \det(A - \alpha I) \\ \nu_2(A) &= \det((A - \alpha I) \odot I)\end{aligned}$$

La matrice A considérée ici sera la matrice en boucle fermée, fonction des gains du correcteur : $A_{CL} = A - BKC$. On obtient alors, selon le formalisme adopté plus haut, les fonctions matricielles données en (6.13,6.14) en considérant un cas mono-entrée comme le lanceur.

$$F_1(K) = A_{CL,\alpha} = (A - \alpha I) - \sum_{i=1}^p K_i BC_i \quad (6.13)$$

$$F_2(K) = A_{CL,\alpha} \odot I = (A - \alpha I) \odot I - \sum_{i=1}^p K_i BC_i \odot I \quad (6.14)$$

où C_i désigne la i^{me} ligne de la matrice C, et $A_{CL,\alpha} = A - BKC - \alpha I$. Le calcul des gradients est obtenu directement par application de la formule de Jacobi (6.9) pour le cas linéaire. On aboutit aux expressions (6.15-6.16).

$$\forall i, \frac{\partial \nu_1}{\partial K_i} = -\text{tr}(A_{CL,\alpha}^A BC_i) \quad (6.15)$$

$$\forall i, \frac{\partial \nu_2}{\partial K_i} = -\text{tr}((A_{CL,\alpha} \odot I)^A BC_i \odot I) \quad (6.16)$$

Le gradient de ν_α découle alors du produit de ν_1 et ν_2 pour obtenir (6.17).

$$\forall i, \frac{\partial \nu_\alpha}{\partial K_i} = -\nu_2(K) \text{tr}(A_{CL,\alpha}^A BC_i) - \nu_1(K) \text{tr}((A_{CL,\alpha} \odot I)^A BC_i \odot I) \quad (6.17)$$

Gradient de ν_ξ Le calcul du gradient va maintenant être explicité pour l'application ν_ξ vérifiant l'amortissement minimal d'un système. Elle se découpe elle aussi selon deux applications différentes $\nu_\xi(A) = \nu_3(A) \nu_4(A)$ avec $\mu = 1 - 2\xi^2$:

$$\begin{aligned}\nu_3(A) &= \det(A) \\ \nu_4(A) &= \det(A^2 \odot I + \mu A \odot A)\end{aligned}$$

En considérant la même matrice en boucle fermée A_{CL} que précédemment, on aboutit

aux applications matricielles en (6.18,6.19).

$$F_3(K) = A_{CL} = A - \sum_{i=1}^p K_i BC_i \quad (6.18)$$

$$F_4(K) = A_{CL}^2 \odot I + \mu A_{CL} \odot A_{CL} = M_0 + \sum_{i=1}^p K_i M_i + \sum_{i,j=1}^p K_i K_j M_{ij} \quad (6.19)$$

en notant

$$\begin{aligned} M_0 &= A^2 \odot I + \mu A \odot A \\ M_i &= -(ABC_i + BC_i A) \odot I - 2\mu(A \odot BC_i) \\ M_{ij} &= (BC_i BC_j + BC_j BC_i) \odot I + 2\mu(BC_i \odot BC_j) \end{aligned}$$

La formule de Jacobi linéaire fournit le gradient de ν_3 en (6.20). Pour ν_4 la présence de termes quadratiques oblige à utiliser la formule de Jacobi générale. Le résultat est donné en (6.21).

$$\forall i, \frac{\partial \nu_3}{\partial K_i} = -\text{tr}(A_{CL}^A BC_i) \quad (6.20)$$

$$\forall i, \frac{\partial \nu_4}{\partial K_i} = -\text{tr}(F_4(K)^A ((BC_i A_{CL} + A_{CL} BC_i) \odot I + 2\mu BC_i \odot A_{CL})) \quad (6.21)$$

Pour finir, le gradient de ν_ξ s'obtient en (6.22) par le produit de ν_3 et ν_4 .

$$\forall i, \frac{\partial \nu_\xi}{\partial K_i} = -\nu_4(K) \text{tr}(A_{CL}^A BC_i) - \nu_3(K) \text{tr}(F_4(K)^A ((BC_i A_{CL} + A_{CL} BC_i) \odot I + 2\mu BC_i \odot A_{CL})) \quad (6.22)$$

6.3.3 Prise en compte des applications gardiennes

La structure du problème d'optimisation avec les applications gardiennes amène à considérer un algorithme particulier pour la recherche linéaire. Les conditions de Wolfe sont ainsi légèrement modifiées, ainsi que le choix de la direction de descente au niveau des points selles rencontrés.

Recherche linéaire

Sachant que l'on se place toujours dans une composante stable d'une application gardienne, l'ensemble d'étude est délimité par son lieu d'annulation. Ceci permet d'affirmer que si l'application s'annule sur la frontière alors elle admet par continuité un extrémum à l'intérieur de la composante quand celle-ci est fermée². Ainsi la recherche de Wolfe s'effectuera pour un pas t tel que l'on ne sorte pas de la composante. D'après le lemme (4.4.3) il existe un intervalle dans lequel le pas t doit être choisi : $t \in [t_{min}, t_{max}]$. L'application gardienne considérée pour ce calcul devient une fonction de t , en notant $K = K_k + t.D_k$ avec D_k la direction de descente à l'itération k :

$$\nu(t) = \nu(A - BK_k C - t BD_k C)$$

Le graphe d'une application gardienne dans une direction donnée D_k est fourni en Fig. 6.7. Les bornes de la composante sont délimitées par les lieux d'annulation de l'application gardienne.

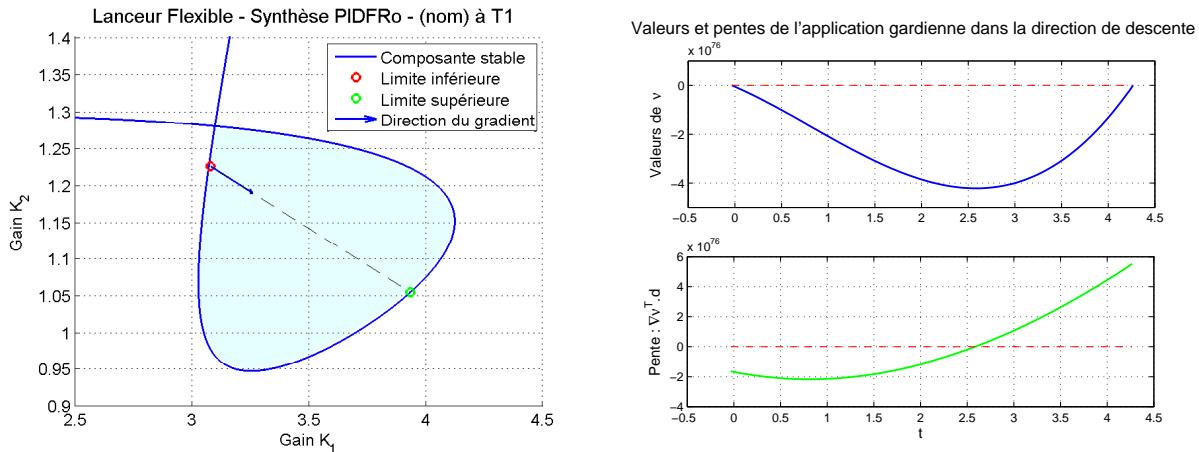


Figure 6.7 Valeurs d'une application gardienne à travers une composante

D'après l'existence d'un extrémum dans la composante, la présence de points satisfaisant les conditions de Wolfe est assurée. Pour l'obtenir, un algorithme basé sur Nocedal et Wright (2006) permet d'obtenir le pas t adéquat dans cet ensemble.

Il repose sur un encadrement du minimum par deux pas t_{min} et t_{max} issus des bornes de stabilité dans la direction D_k . La valeur de la fonction et son gradient sont évalués pour un pas t dans cet intervalle. Suivant si les conditions de Wolfe sont satisfaites ou violées, ce point fournit une nouvelle borne t_{inf} ou t_{sup} jusqu'à ce qu'un nouveau point respecte ces conditions.

2. Si la composante est ouverte, la valeur des gains sera limitée en pratique par une borne fixée par l'utilisateur

L'algorithme utilisé est le suivant :

Algorithme de Recherche linéaire

- Initialiser $t_0 = t_{min}$, $t_1 > 0$ et t_{max}
- $i \leftarrow 1$
- **Répéter**
 - Evaluer $\nu(t_i)$
 - **Si** $\nu(t_i) > \nu(0) + \alpha t_i \nu'(0)$ ou $[\nu(t_i) \geq \nu(t_{i-1}) \text{ et } i > 1]$
 $t^* \leftarrow find_step(t_{i-1}, t_i)$ et **STOP**
 - Evaluer $\nu'(t_i)$
 - **Si** $|\nu'(t_i)| \leq -\gamma \nu'(0)$
 $t^* \leftarrow t_i$ et **STOP**
 - **Si** $|\nu'(t_i)| \geq 0$
 $t^* \leftarrow find_step(t_i, t_{i-1})$ et **STOP**
 - Choisir $t_{i+1} \in [t_i, t_{max}]$
 - $i \leftarrow i + 1$
- **Fin (Répéter)**

La fonction $find_step(t_{inf}, t_{sup})$ permet de trouver un pas dans l'intervalle (t_{inf}, t_{sup}) qui satisfait les conditions de Wolfe.

Pour ce faire,

- les bornes (t_{inf}, t_{sup}) doivent toujours encadrer une valeur qui respecte ces conditions,
- la borne t_{inf} doit donner la plus faible valeur de la fonction parmi les points déjà visités qui respectaient la condition d'Armijo,
- la pente en t_{inf} doit être négative pour assurer l'encadrement du minimum.

L'algorithme utilisé pour cette fonction $find_step$ est tiré de Nocedal et Wright (2006).

Voici le processus qui est suivi :

Fonction $find_step(t_{inf}, t_{sup})$

- **Boucle**

- Choisir un pas $t_j \in [t_{inf}, t_{sup}]$ et évaluer $\nu(t_j)$
- Si $\nu(t_j) > \nu(0) + \alpha t_j \nu'(0)$ ou $\nu(t_j) \geq \nu(t_{inf})$

$$t_{sup} \leftarrow t_j$$

- **Sinon**

Evaluer $\nu'(t_j)$

Si $|\nu'(t_j)| \leq -\gamma \nu'(0)$

$$t^* \leftarrow t_j \text{ et STOP}$$

Si $\nu'(t_j)(t_{sup} - t_{inf}) \geq 0$

$$t_{sup} \leftarrow t_{inf}$$

$$t_{inf} \leftarrow t_j$$

- **Fin Boucle**

Ces algorithmes sont illustrés sur la Fig. 6.8. On y voit l'encadrement du minimum de la fonction par les deux bornes t_{min} et t_{max} , qui évoluent à chaque itération. A terme, on obtient le point en rouge, qui respecte effectivement les conditions de Wolfe.

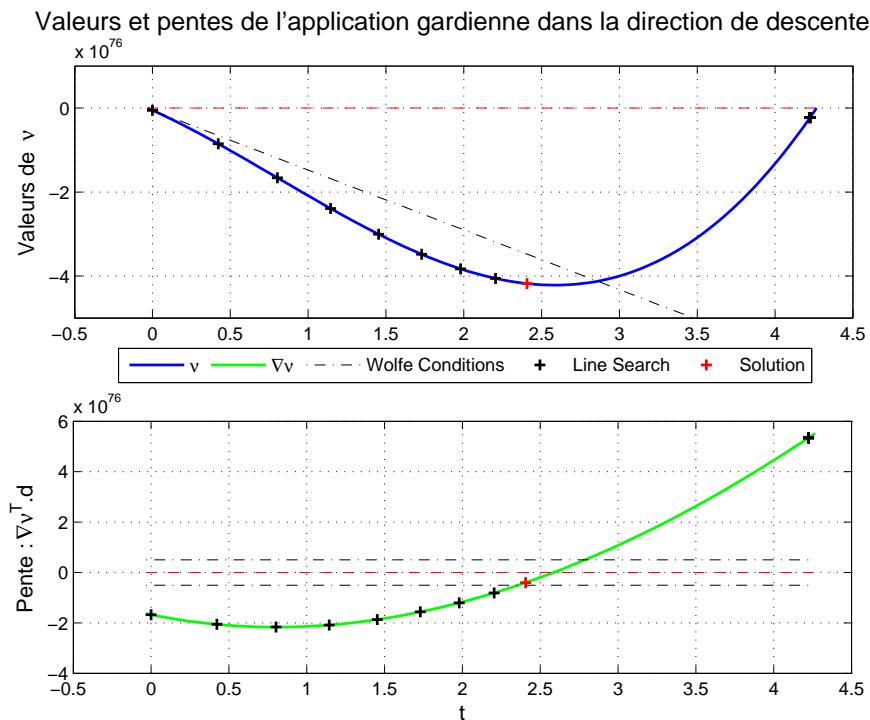


Figure 6.8 Recherche d'un point satisfaisant les conditions de Wolfe

Rencontre de points selle

Une autre caractéristique des applications gardiennes est la présence de nombreux points selle. Ces derniers apparaissent au niveau des intersections des lieux d'annulation, quand plusieurs sont considérées dans la synthèse. Ces points sont caractérisés par une annulation du gradient avec une matrice hessienne non définie positive ou négative.

En s'appuyant sur une application $\nu(t) = \nu_1(t)\nu_2(t)$, le calcul du gradient au niveau d'une intersection des lieux d'annulation donne : $\nu'(t) = \nu'_1(t)\nu_2(t) + \nu_1(t)\nu'_2(t)$. Or à une intersection, $\nu_1(t) = 0$ et $\nu_2(t) = 0$, donc $\nu'(t) = 0$. Malgré cette annulation le point d'intersection ne constitue pas un minimum, il suffit de vérifier le signe de chaque application ν_1 et ν_2 dans les cadrants ainsi découpés pour s'apercevoir que ν alterne les valeurs positives et négatives d'un cadran à l'autre comme illustré par Fig. 6.9.

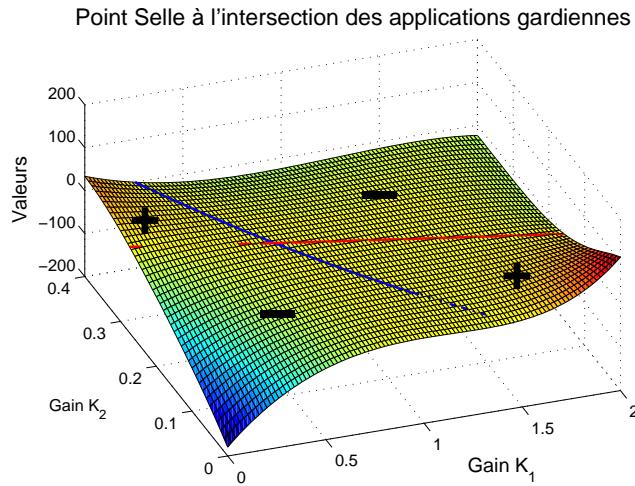


Figure 6.9 Valeurs d'une application gardienne au niveau d'une intersection

Ce cas est problématique dans la mise en place de l'algorithme de descente puisqu'au voisinage de tels points le gradient peut devenir arbitrairement petit. Une direction de descente particulière est calculée en ces points-là, pour ramener le point courant plus au centre de la composante et obtenir des gradients exploitables.

Pour ce faire, il faut s'intéresser aux deux applications gardiennes qui s'annulent. Ensuite, une normalisation des gradients, notée $\nabla_n\nu_1$ et $\nabla_n\nu_2$, permet d'obtenir une direction pointant strictement à l'intérieur de la composante. Cette direction est donnée en (6.23) et le principe est illustré en Fig. 6.10.

$$d = \text{signe}(\nu_1) \frac{\nabla \nu_1}{\|\nabla \nu_1\|} + \text{signe}(\nu_2) \frac{\nabla \nu_2}{\|\nabla \nu_2\|} \quad (6.23)$$

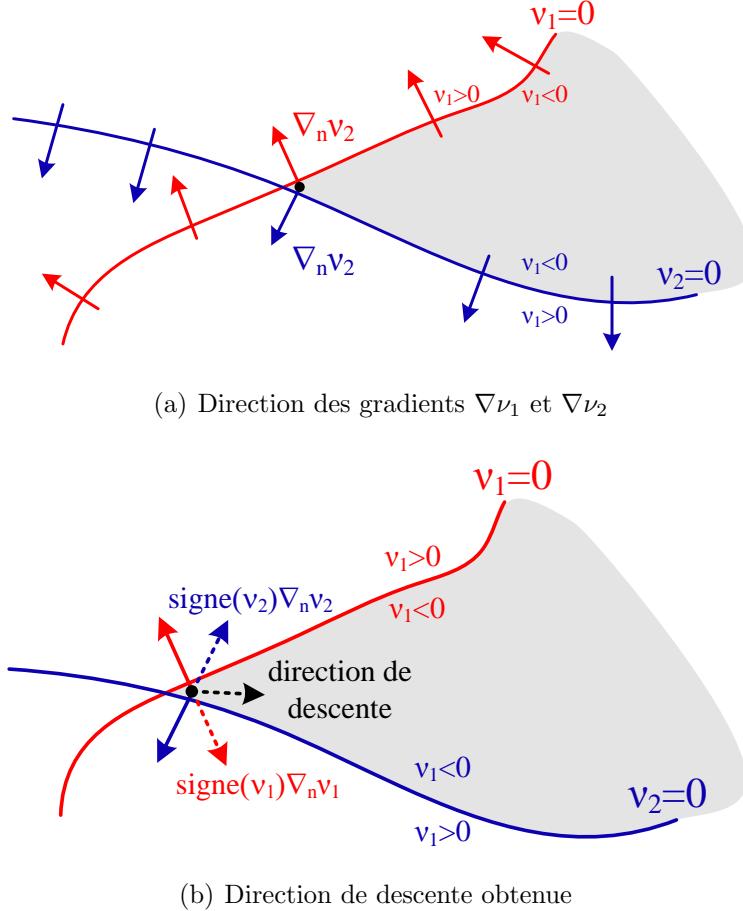


Figure 6.10 Construction d'une direction de descente pointant à l'intérieur de la composante

6.4 Application au cas du lanceur

L'algorithme peut maintenant être appliqué sur le cas du lanceur. La structure de correction reste la même avec un correcteur issu du contrôle de la partie rigide avec du roll-off, et un filtre d'ordre 2. L'expression est rappelée en (6.24). Les paramètres laissés libres sont cette-fois ci les trois gains K_1 , K_2 et K_i . La synthèse du correcteur est envisagée directement sur le cas flexible. Le réglage obtenu est ensuite comparé à celui obtenu par la synthèse graphique.

$$C(s) = F(s) \frac{K_i + K_1 s + K_2 s^2}{s (s/p_1 + 1) (s/p_2 + 1)} \quad (6.24)$$

6.4.1 Synthèse sur le lanceur flexible

Simplifications de modèle

Pour des questions algorithmiques, les applications gardiennes considérées ont été appliquées sur des modèles réduits du lanceur. Les applications gardiennes ν_α et ν_ξ relatives aux pôles rigides ont ainsi été calculées sur le modèle rigide du lanceur avec actionneur. Cette approximation se justifie par le fait que les modes flexibles modifient légèrement les formes des composantes stables. Un exemple est donné en Fig. 6.11. L'influence des modes flexibles est ici bénéfique puisqu'elle vient élargir la zone stable pour le choix des gains. Sur d'autres modèles, les modes flexibles peuvent au contraire la réduire. Dans tous les cas, cette approximation sera validée *a posteriori* avec le calcul des pôles sur le système flexible. L'abscisse spectrale et l'amortissement de la partie rigide seront vérifiés pour attester que les modes flexibles n'ont pas fait perdre les performances.

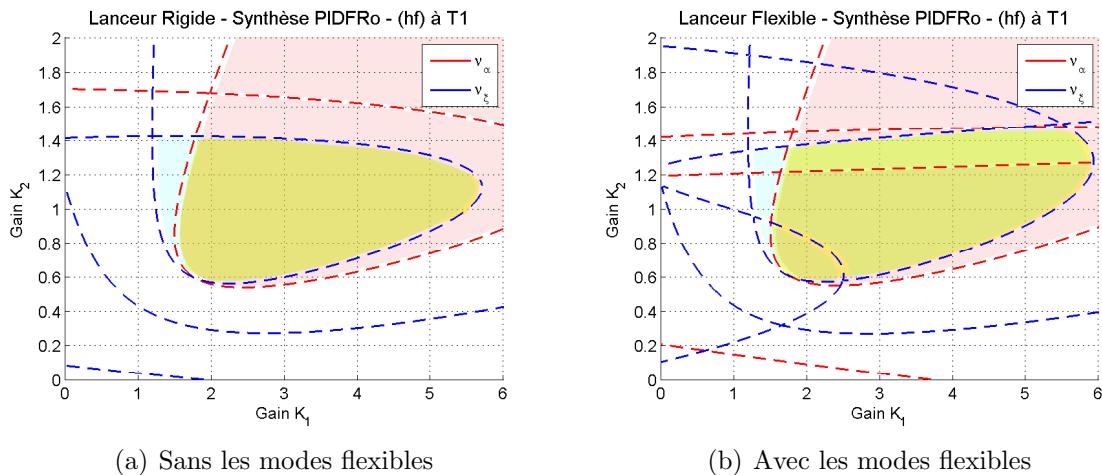


Figure 6.11 Influence des modes flexibles sur l'intersection des composantes stables

Ensuite, l'application ν_β s'appliquera sur le système flexible complet sur lequel une réduction de modèle a été réalisée. Cette approximation est nécessaire pour éviter les erreurs numériques qui apparaissent avec le système complet d'ordre 15. L'algorithme est parfaitement fonctionnel d'un point de vue théorique, il s'agit seulement d'un problème numérique apparaissant avec les systèmes de grande taille. Une analyse des valeurs singulières du système nous donne le résultat suivant :

$$\sigma = \{Inf, Inf, 1.75, 1.71, 0.84, 0.48, 0.47, 0.24, 0.23, 0.16, 0.037, 0.016, 0.005, 0.0014, 0.0003\}$$

Les deux paires de pôles de l'actionneur possèdent les valeurs les plus faibles en raison de

leur dynamique très haute fréquence. La paire de pôles la plus rapide des deux a ainsi été négligée pour aboutir à un système complet d'ordre 13 pour décrire le lanceur flexible. Là encore une vérification *a posteriori* des pôles du système permettra de valider la légitimité de cette simplification.

Réglage des gains

L'algorithme lancé sur un modèle du lanceur (`nom`, `hf` ou `lf`) à un instant donné permet d'obtenir un réglage de gains qui satisfait les performances. La procédure est répétée sur tous les autres modèles et sur tous les instants pour obtenir un jeu de 18 réglages de gains. Une illustration de l'algorithme complet est donnée en Fig. 6.12.

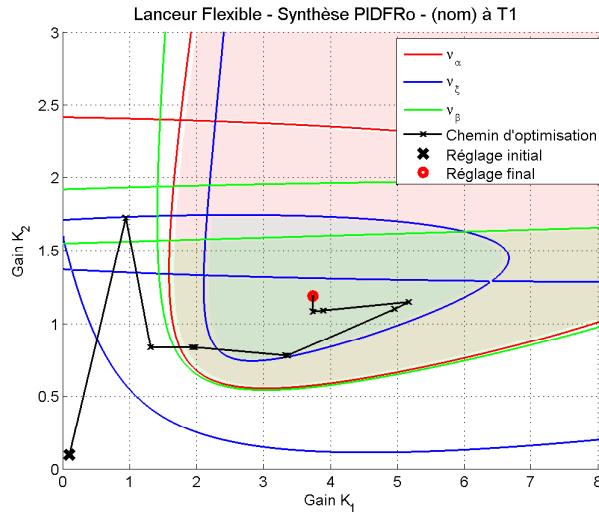


Figure 6.12 Chemin d'optimisation suivi pour respecter les performances

La robustesse du correcteur par rapport aux modèles incertains sera réalisée en considérant les intervalles de stabilité autour de chaque réglage pour les modèles (`nom`, `hf` et `lf`). Le but étant d'avoir une intersection entre ces segments, l'algorithme est relancé sur les cas litigieux pour obtenir un autre réglage possiblement plus robuste. Ces intervalles de stabilité sont calculés dans la direction des gains K_1, K_2 et K_i puisqu'il n'est pas possible de connaître la forme de la composante stable avec plus de 2 paramètres.

Un exemple d'intervalle obtenu est donné en Fig. 6.13. A chaque instant, les intervalles de stabilité des trois modèles sont superposés pour observer la présence ou non d'intersection et les marquages noirs représentent les gains obtenus par l'optimisation.

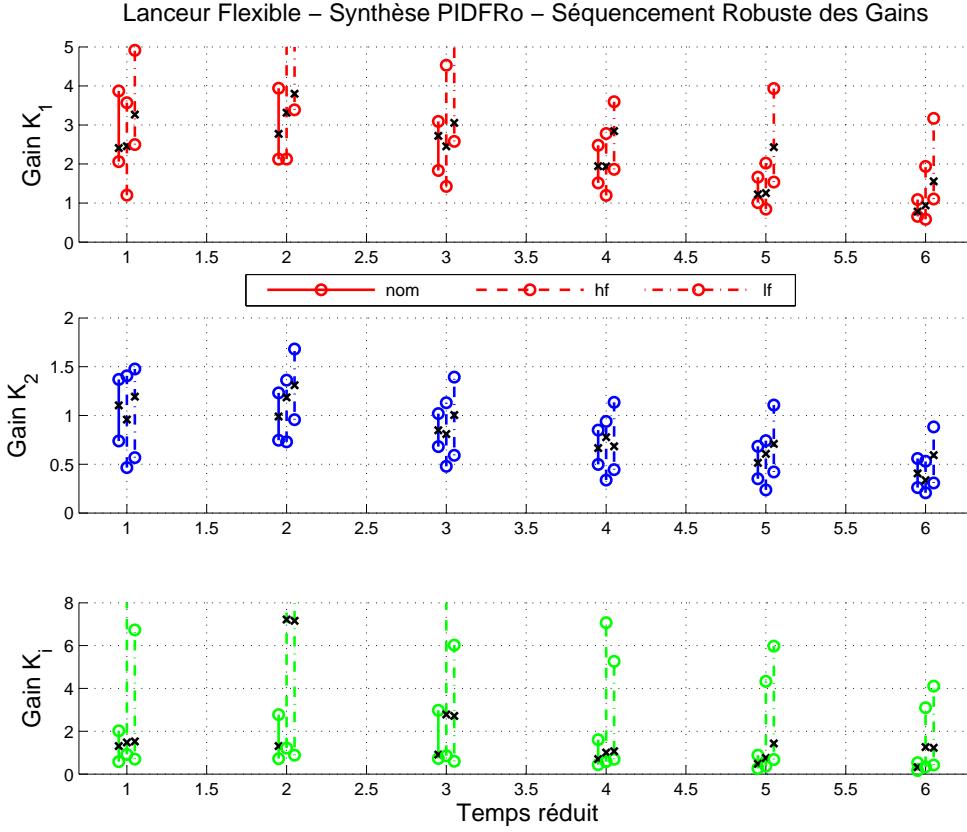


Figure 6.13 Intervalles de stabilité obtenus sur les réglages issus de l'optimisation

Séquencement des gains

Pour procéder au séquencement des gains une approche similaire à la synthèse graphique sera adoptée. Un premier gain K à séquencer est choisi. Un séquencement linéaire est possible si une droite peut passer à travers les 3 intervalles de stabilité à chaque instant. Sachant que ce séquencement du gain K a été choisi dans les intervalles de stabilité, les performances sont conservées. Cependant, les intervalles de stabilité sur les autres gains s'en trouvent modifiés car les valeurs de K ont changé. Il faut donc remettre à jour les intervalles avant de recommencer l'opération pour les deux autres gains jusqu'à obtenir un séquencement complet. En répétant le même travail sur les deux autres gains, la Fig. 6.14 est obtenue.

Une difficulté majeure dans cette façon de procéder vient du fait que les intervalles de stabilité changent parfois énormément après le séquencement d'un gain. L'intersection des intervalles nécessaire à la robustesse est alors plus difficile à atteindre. De nouveaux points doivent être obtenus à partir de l'optimisation pour obtenir des réglages alternatifs.

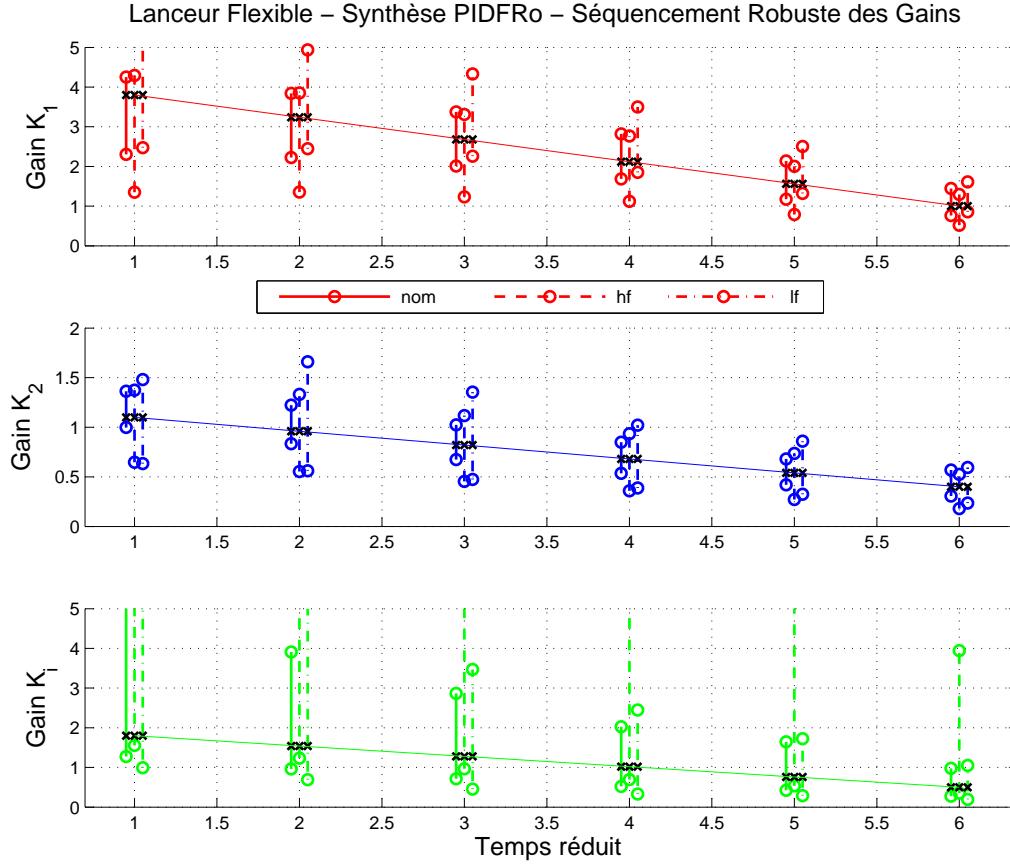


Figure 6.14 Séquencement des gains à partir des réglages obtenus par optimisation

Finalement le correcteur obtenu par cette synthèse par optimisation est donné dans le Tableau 6.3. Il va maintenant être validé à travers le calcul des marges et la simulation temporelle à la rafale de vent.

Tableau 6.3 Valeurs séquencées des gains du correcteur PIDFRo

Gains \ Temps	T_1	T_2	T_3	T_4	T_5	T_6
K_p	3.80	3.24	2.68	2.12	1.56	1.00
K_d	1.10	0.96	0.82	0.68	0.54	0.40
K_i	1.80	1.54	1.28	1.02	0.76	0.50

Validation du correcteur

Considérant la difficulté à obtenir un réglage des gains qui satisfasse les contraintes sur les marges et le dépassement maximal en incidence, le séquencement précédent se rapproche beaucoup de celui obtenu dans la synthèse graphique. Les résultats obtenus en Fig. 6.15 sont donc aussi proches sur la simulation temporelle à la rafale de vent. Un dépassement du même ordre est observé (environ 20%) sur l'incidence, et l'attitude θ et la consigne β_c restent aussi raisonnables.

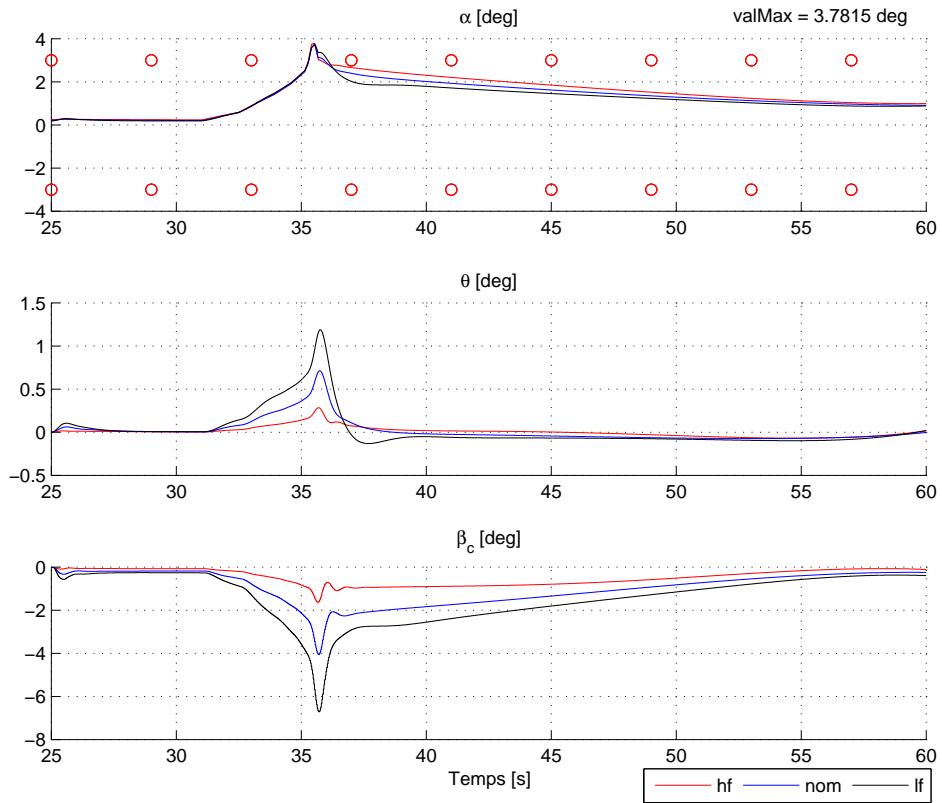


Figure 6.15 Simulations temporelles pour le lanceur flexible

Au niveau des marges présentées en Fig. 6.16, les marges de gain sont assez largement satisfaites. Les marges de retard deviennent proches de la limite fixée à 40ms sur la fin du vol avec le cas **hf**. Le contrôleur obtenu est donc validé une nouvelle fois.

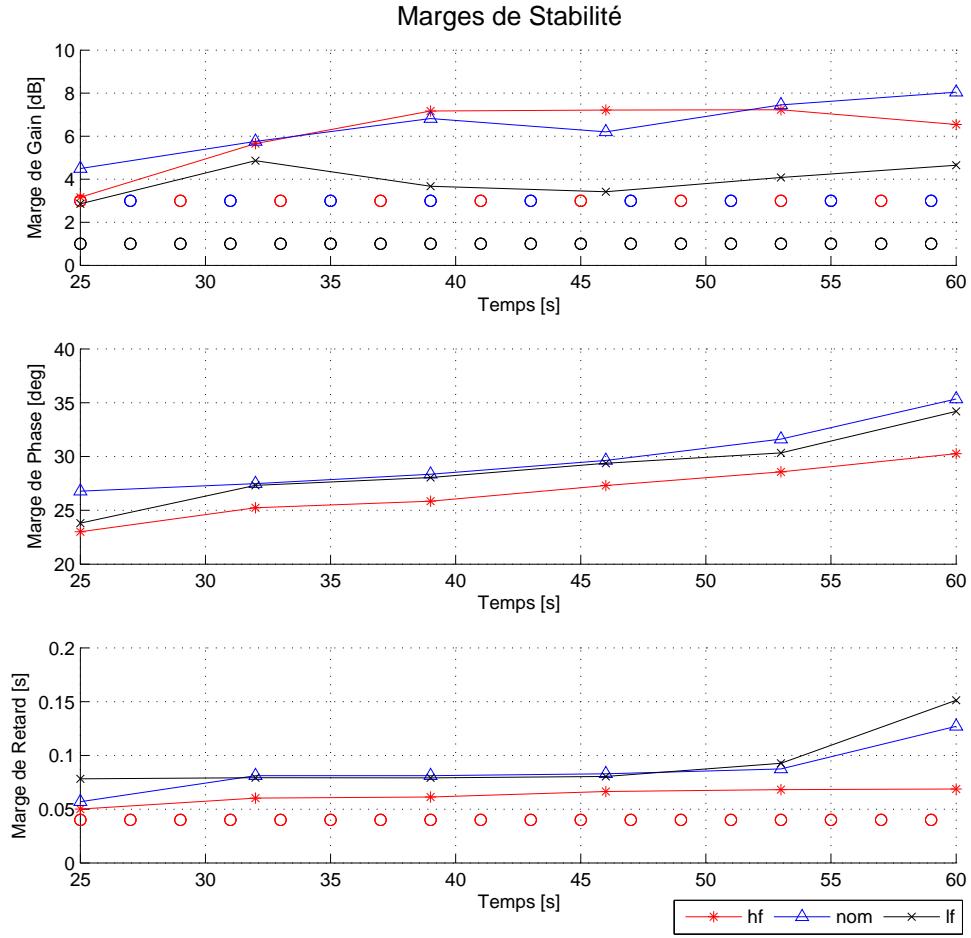


Figure 6.16 Evolution des marges au cours du vol pour le lanceur flexible

Un point intéressant est la similarité obtenue sur le réglage du correcteur. Il était en effet clair d'après l'analyse graphique que les zones stables pour la synthèse étaient uniques (pour les intervalles des gains considérés) et que leur taille ne permettait pas de trouver un réglage complètement différent. L'optimisation a donc convergé vers cette même zone stable pour chaque instant et chaque mode, confirmant son bon fonctionnement. Ensuite, le séquencement s'est aussi fait de façon à satisfaire les conditions sur l'incidence et les marges donc il est normal de retrouver, à terme, un réglage similaire.

6.4.2 Problèmes rencontrés

Une difficulté majeure dans l'optimisation des applications gardiennes est la complexité numérique et les changements d'échelle. Il s'agit d'évaluer des déterminants de matrices pouvant atteindre la taille 15 pour le lanceur complet, soit la taille 105 quand il s'agit des produits bialternés. De telles matrices amènent des déterminants gigantesques de l'ordre de

10^{120} . Lorsqu'il s'agit de combiner les applications gardiennes pour différentes performances, les valeurs atteignent très rapidement les limites de la machine. Il est donc important de chercher à changer l'échelle de valeurs des applications gardiennes.

Ceci peut se faire en multipliant les matrices par un facteur quelconque sensé atténuer la valeur du déterminant. De même, une multiplication de la valeur des applications gardiennes par un autre facteur d'échelle ne suffit pas à obtenir des résultats satisfaisants. L'idéal serait d'utiliser une fonction qui puisse simplement atténuer les hautes valeurs de l'application dans le centre des composantes. Il serait ainsi plus aisément de visualiser les détails au niveau des lieux d'annulation.

Des exemples sont donnés en Fig. 6.17 pour illustrer cette idée. Le graphe d'une application gardienne est donné avec et sans traitement. Les deux exemples pour le changement d'échelle sont la fonction logarithme (évaluée sur la valeur absolue des applications gardiennes) et la fonction puissance $1/8$. Cette dernière fournit un bon compromis, mais le problème devrait être investi plus en détails pour tenter d'agir directement au niveau des matrices du système et de réaliser un meilleur conditionnement des matrices.

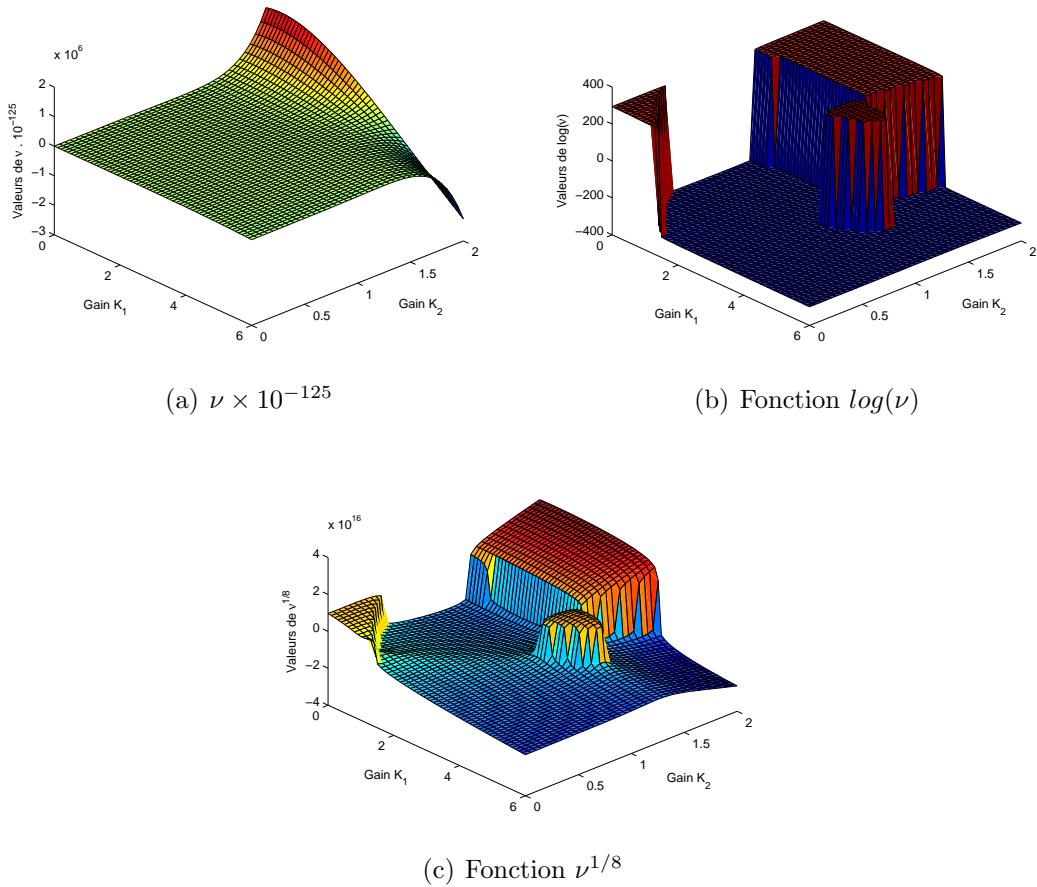


Figure 6.17 Changements d'échelle sur les valeurs de l'application gardienne.

CHAPITRE 7

CONCLUSION

Résultats obtenus

Le premier bilan de cette étude porte sur le mandat initial : stabiliser un lanceur spatial suivant un cahier des charges précis. Ce dernier incluait des contraintes de performance avec le confinement des pôles dans des espaces d'intérêt. Il fallait conserver ces performances à la fois sur le modèle nominal et à la fois sur les types incertains **hf** et **1f** sur l'ensemble du vol. Cette mission a été remplie en réalisant une structure de correction d'ordre 5 seulement comprenant un contrôleur PID avec du roll-off d'ordre 1, et un filtre d'ordre 2. Ces deux améliorations ont permis de modérer considérablement l'effet déstabilisant des modes flexibles. Le réglage du correcteur a ensuite été réalisé par l'utilisation des applications gardiennes. Dans un premier cas, une approche graphique s'appuyait sur la visualisation des zones stables pour le choix du réglage, tandis qu'une seconde approche basée sur l'optimisation de ces fonctions rendait directement un réglage convenable. La validation des structures de correction fut ensuite accomplie au travers du calcul des marges de stabilité et de la simulation du système soumis à une rafale de vent. Dans les deux méthodes de synthèse un correcteur qui respecte l'ensemble du cahier des charges a pu être trouvé et validé. De ce point de vue, le projet a donc atteint son premier objectif.

Le second mandat portait sur le développement des méthodes de synthèse par les applications gardiennes. Il s'est effectué selon deux axes principaux : la synthèse graphique et la synthèse par optimisation. La première repose sur l'analyse automatique des composantes d'une application gardienne. Au travers d'un traitement d'image efficace, il est ainsi possible de déterminer directement les composantes stables de chaque application gardienne. Une superposition de ces composantes pour différentes contraintes permet d'évaluer leur compatibilité pour pouvoir répondre au cahier des charges. L'intérêt de cette méthode est avant tout pédagogique puisqu'elle permet d'avoir une approche visuelle sur les composantes et sur leur rôle dans la synthèse. L'influence des paramètres est ainsi illustrée par leur effet sur l'existence, la position ou la taille de ces composantes stables. Finalement le séquencement des gains est plus aisé par cette approche. Sa limitation principale résidait dans la restriction à deux variables pour permettre une visualisation 2D.

La méthode par optimisation a donc été envisagé pour pallier à cette restriction. Elle reprend les travaux de Saussié *et al.* (2010) en y incorporant une base d'optimisation pour permettre d'obtenir des réglages en quelques itérations seulement. Son intérêt réside dans le nombre de variables de synthèse, en théorie illimité. Elle est basée sur une alternance entre la construction des applications gardiennes liées au réglage courant des gains, et la minimisation de ces dernières dans leurs composantes stables. A terme, le fait d'entrer à chaque fois dans ces composantes amène des réglages plus performants jusqu'à atteindre les exigences fixées. Le seul bémol de cette approche tient aux précisions finies des ordinateurs. L'explosion sur les valeurs des applications gardiennes amène un lot de problèmes qui sont toujours ouverts. Certains sont détaillés dans la suite au travers des perspectives.

Perspectives

Au niveau du projet en lui-même, quelques voies mériteraient une étude plus ample de par l'intérêt qu'elles présentent. L'accent a été mis sur le développement de méthodes de synthèse pour les applications gardiennes. Ainsi, le choix de la structure de correction s'en est tenu au correcteur le plus simple stabilisant le système. Les structures de correction ont été inspirées par la littérature et leur légitimité a été prouvée par les correcteurs qu'elles ont fournies. Néanmoins il serait intéressant de considérer des structures alternatives sur le correcteur en lui-même ou sur la structure du filtre choisi. De même, au niveau des simulations, le profil de vent est fourni par ASTRIUM-ST et une validation plus complète serait possible si des profils différents étaient considérés. On pourrait imaginer des profils de vent typiques apparaissant à la traversée des couches de l'atmosphère, ainsi que des rafales de vent correspondant à des cas habituels et des pires cas.

Au niveau des applications gardiennes, leur atout majeur est leur polyvalence. Elles nous ont surtout servi pour la synthèse des correcteurs alors qu'elles trouvent leur origine dans les méthodes d'analyse robuste. Tous les codes et algorithmes développés pourraient ainsi être utilisés dans l'optique de l'analyse de robustesse. Une dernière amélioration sur la modélisation du lanceur passerait par une paramétrisation des modèles incertains. Une structure de correction synthétisée sur le modèle nominal rendrait ainsi des intervalles extrêmes de variation pour les paramètres considérés comme incertains. Dans le même ordre d'idée, l'utilisation d'un modèle variant dans le temps pour le lanceur permettrait de réaliser un séquencement plus efficace. Le temps serait alors considéré comme un paramètre incertain et les durées de validité des gains seraient obtenus (Saussié *et al.*, 2011).

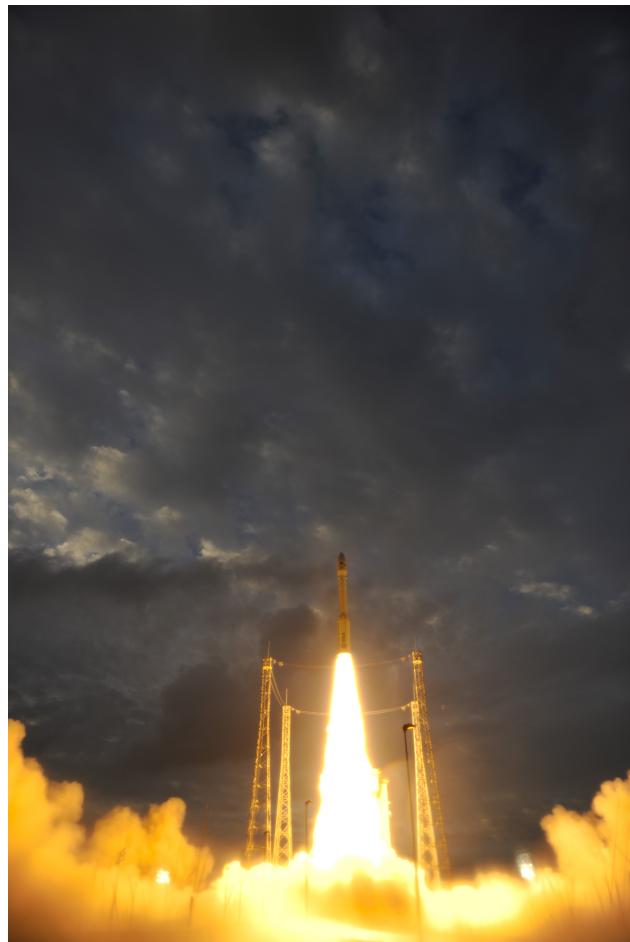
Finalement, les dernières perspectives porteront sur ce remarquable outil de synthèse que sont les applications gardiennes. Le point fondamental pour rendre l'approche graphique plus efficace repose sur l'expression de fonctions. S'il était possible de développer une expression analytique des applications gardiennes usuelles, les calculs seraient considérablement diminués. Une approche par calcul symbolique pourrait être envisagée, même si elle risque d'être limitée avec la taille du système et le nombre de paramètres incertains considérés. Les logiciels de calcul formel sont capables d'effectuer une telle tâche mais l'évaluation des expressions pour tous les points du maillage s'avère très lente sous Matlab[©].

De plus, quelques mots ont été dits sur les variations d'échelle dans les valeurs prises par les applications gardiennes. Il y a là un problème sérieux qui consisterait à conditionner les matrices de départ. Elles sont par définition mal conditionnées quand il y a des dynamiques très différentes comme celle des actionneurs et celle des modes flexibles. La première amène des termes de l'ordre de 10^3 dans les matrices du modèle, tandis que la seconde apporte des corrections en 10^{-6} . De tels écarts sont amplifiés par les calculs des déterminants propulsant les valeurs des applications gardiennes à 10^{120} , voire plus, quand on effectue des produits entre elles. Comme on peut le voir sur les derniers graphes présentés en Fig. 6.17, les variations entre les valeurs positives et négatives sont très brutales, ce qui rend le calcul du gradient plus laborieux. Il serait donc bon d'arriver à « lisser » le comportement de ces fonctions pour faciliter les implantations numériques. De même, un calcul analytique de la hessienne des applications gardiennes apporterait une grande efficacité dans l'algorithme d'optimisation. Il permettrait entre autres d'utiliser des méthodes de Newton avec les gradients conjugués pour obtenir de meilleurs résultats.

Pour conclure, ce projet fortement multidisciplinaire a permis de passer à travers un travail de modélisation du lanceur puis de s'atteler au développement d'une loi de commande efficace. Le pilotage d'un tel véhicule est basé sur le pari risqué de modéliser aussi bien les dynamiques structurelles que les incertitudes de modèle. Ces dernières prennent leur source dans tous les domaines qui touchent au lanceur, depuis la propulsion, la structure, jusqu'au système d'acquisition de données. L'accent est mis ici sur la grande complexité des systèmes spatiaux. Tous les domaines y sont interdépendants et leur synergie est seule capable de donner naissance aux projets les plus ambitieux. Que l'espace soit considéré simplement pour l'observation, ou bien comme un enjeu économique voire militaire, il reste un centre d'inté-

rêt passionnant. Comme le disait *Konstantin Tsiolkovsky* (1857-1935), le père fondateur de l'astronautique :

« The Earth is the cradle of humanity, but mankind cannot stay in the cradle forever »



Crédits - ESA - S. Corvaja, 2012

Figure 7.1 Premier vol du lanceur européen VEGA, le 13 Février 2012

RÉFÉRENCES

- AITKEN, A. (1928). On the latent roots of certain matrices. *Proceedings of the Edinburgh Mathematical Society (Series 2)*. 135–138.
- ALAZARD, D. et APKARIAN, P. (1999). Exact observer-based structures for arbitrary compensators. *International Journal of Robust and Non-Linear Control*, 9, 101–118.
- BALDESI, G. (2008). *Modelling, control design and simulation for a launch vehicle : from linear to nonlinear methods*. Thèse de doctorat, La Sapienza, Université de Rome, Italie and ISAE, Université de Toulouse, France.
- BARMISH, B. (1994). *New tools for robustness of linear systems*. New York.
- BARTLETT, A., HOLLOT, C. et HUANG, L. (1988). Root location of an entire polytope of polynomials : It suffices to check the edges. *Mathematics of Control, Signals and Systems*, 1, 61–71.
- BÉRARD, C., FERRERES, G., JACQUIER, B. et MAGNI, J.-F. (2008). LPV & robustness synthesis concepts for complex space systems. Final Version, FR 6/13198 DCSD.
- BIANNIC, J.-M., DÖLL, C. et MAGNI, J.-F. (2006). Simulink-based tools for creating and simulating interconnected LFR objects. *Proceedings of the IEEE International Symposium on Computer Aided Control System Design*.
- BLAKELOCK, J. (1965). *Automatic control of aircraft and missiles*. New York.
- BÉRARD, C., SAUSSIÉ, D., BIANNIC, J.-M. et DÖLL, C. (2012). LFT modeling and self-scheduled control : application to a launch vehicle. *CASI Conference*. XXXX–XXXX.
- CARNEVALE, C. et RESTA, P. (2007). Vega electromechanical thrust vector control development. *AIAA Joint Propulsion Conference*. vol. 8, 7990–7999.
- CLÉMENT, B., DUC, G. et MAUFFREY, S. (2005). Aerospace launch vehicle control : A gain scheduling approach. *Control Engineering Practice*, 13, 333–347.
- DELMOND, F., ALAZARD, D. et CUMER, C. (2006). Cross standard form : a solution to improve a given controller with H_2 or H_∞ specifications. *International Journal of Control*, 79, 279–287.
- DESESSARD, J.-F. (2002). Sophie mauffrey reçoit le prix robert alkan 2000. <http://www.bulletins-electroniques.com/actualites/36639.htm>. Ministère français des affaires étrangères et européennes.
- GONZALEZ, R., WOODS, R. et EDDINS, S. (2004). *Digital Image Processing Using MATLAB*. New Jersey.

- GREENSITE, A. (1970). *Analysis and design of space vehicle flight control systems*. New York.
- HERNANDEZ, M. (2008). *Launcher study case : Modal self-scheduling control*. Mémoire de maîtrise, French National Aerospace Research Establishment, Toulouse, France.
- KHARITONOV, V. (1981). Distribution of the roots of the characteristic polynomial of an autonomous system. *Automation and Remote Control*, 42, 589–593.
- LEITH, D. J. et LEITHEAD, W. E. (2000). Survey of gain-scheduling analysis and design. *International Journal of Control*, 73, 1001–1025.
- LISCOUËT, J. (2010). *Conception préliminaire des actionneurs électromécaniques - Approche hybride, directe/inverse*. Thèse de doctorat, Institut National des Sciences Appliquées de Toulouse, Université de Toulouse, France.
- MAGNI, J.-F. (2002). Presentation of the linear fractionnal representation toolbox (LFRT). *Proceedings of the IEEE International Symposium on Computer Aided Control System Design*.
- MAILLEBOUIS, M. (1988). Utilisation des actionneurs hydrauliques et électriques dans les chaines de pilotage des missiles et des lanceurs. *Revue de L'Electricite et de L'Electronique*, 2, 55–60.
- MARTY, D. (1986). *Conception des véhicules spatiaux*. Paris.
- MAUFFREY, S., CHABERT, F. et SCHOELLER, M. (1997). H_∞ method applied to launcher control during atmospheric flight. *Proceedings of the 3rd ESA International Conference on Spacecraft Guidance, Navigation and Control Systems*. vol. 1, 349.
- MAUFFREY, S. et SCHOELLER, M. (1999). Non-stationary H_∞ control law for launcher with bending modes. *14th IFAC Symposium on Automatic Control in Aerospace*. vol. 1, 433–438.
- MEHRABIAN, A., LUCAS, C. et ROSHANIAN, J. (2006). Aerospace launch vehicle control : an intelligent adaptive approach. *Aerospace Science and Technology*, 10, 149–155.
- NOCEDAL, J. et WRIGHT, S. (2006). *Numerical Optimization*. New York, seconde édition.
- PING, S. (2010). Dynamic modeling and analysis of a small solid launch vehicle. *2nd International Conference on Information Engineering and Computer Science*. vol. 1.
- RAHMAN, S., KEIM, N. et ZEENDER, P. (2010). Global trends in space access and utilization. *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*. vol. 1.

- RENAULT, C. (2005). Usefulness of a force feedback on electromechanical actuator. *Proceedings of the 6th International ESA Conference on Guidance, Navigation and Control Systems*. vol. 1, 18.1–18.10.
- ROTUNNO, M., FIORENZANI, T., PETRITOLI, P. et BENNANI, S. (2007). Modern analysis techniques for gain scheduled thrust vector controllers of launchers. *Proceedings of the IEEE International Conference on Control Applications*. vol. 1, 2896–2901.
- RUGH, W. J. et SHAMMA, J. S. (2000). Research on gain-scheduling. *Automatica*, 36, 1401–1425.
- SAUSSIÉ, D. (2010). *Contrôle du vol longitudinal d'un avion civil avec satisfaction de qualités de manœuvrabilité*. Thèse de doctorat, Ecole Polytechnique de Montréal, Université de Montréal, Canada.
- SAUSSIÉ, D., AKHRIF, O. et SAYDY, L. (2010). Aircraft pitch rate control design with guardian maps. *18th Mediterranean Conference on Control & Automation*. vol. 1, 1473–1478.
- SAUSSIÉ, D., BALDESI, G., DÖLL, C. et BÉRARD, C. (2008). Self-scheduling controller for a launcher in atmospheric ascent. *Proceedings of the 17th IFAC World Congress*.
- SAUSSIÉ, D., SAYDY, L., AKHRIF, O. et BÉRARD, C. (2011). Gain scheduling with guardian maps for longitudinal flight control. *AIAA Journal of Guidance, Control and Dynamics*, 34, 1045–1059.
- SAYDY, L., TITS, A. et ABED, E. (1988). Robust stability of linear systems relative to guarded domains. *Proceedings of the 27th IEEE Conference on Decision and Control*. vol. 1, 544–551.
- SAYDY, L., TITS, A. et ABED, E. (1990). Guardian maps and the generalized stability of parametrized families of matrices and polynomials. *Mathematics of Control, Signal and Systems*, 3, 345–371.
- SIOURIS, G. (2004). *Missile Guidance and Control Systems*. New York.
- STEPHANOS, C. (1900). Sur une extension du calcul des substitutions linéaires. *Journal de Mathématiques Pures et Appliquées 5^{ème} série*, 6, 73–128.
- TEREN, F., DAVIDSON, K., BORSODY, J. et DANIELE, C. (1968). Thrust-vector control requirements for large launch vehicles with solid-propellant first stages. NASA Technical Note, Lewis Research Center. D-4662.
- TURNER, M. (2009). *Rocket and Spacecraft Propulsion*, Berlin, chapitre Launch Vehicle Dynamics. Seconde édition, 135–164.
- UNITED STATES ARMY, M. C. D. (1990). *Design of aerodynamically stabilized free rockets*. Military Handbook.

VANTHUYNE, T. (2009). An electrical thrust vector control system for the vega launcher. *Proceedings of the 13th European Space Mechanisms and Tribology Symposium*. vol. 670 SP, 7.

VOINOT, O., ALAZARD, D., APKARIAN, P., MAUFFREY, S. et CLÉMENT, B. (2003). Launcher attitude control : discrete-time robust design and gain-scheduling. *Control Engineering Practice*, 11, 1243–1252.

VOINOT, O., APKARIAN, P. et ALAZARD, D. (2002). Gain-scheduling H_∞ control of the launcher in atmospheric flight via linear-parameter varying techniques. *AIAA Guidance, Navigation, and Control Conference*. vol. 1.

ANNEXE A

Composantes des applications gardiennes ν_α

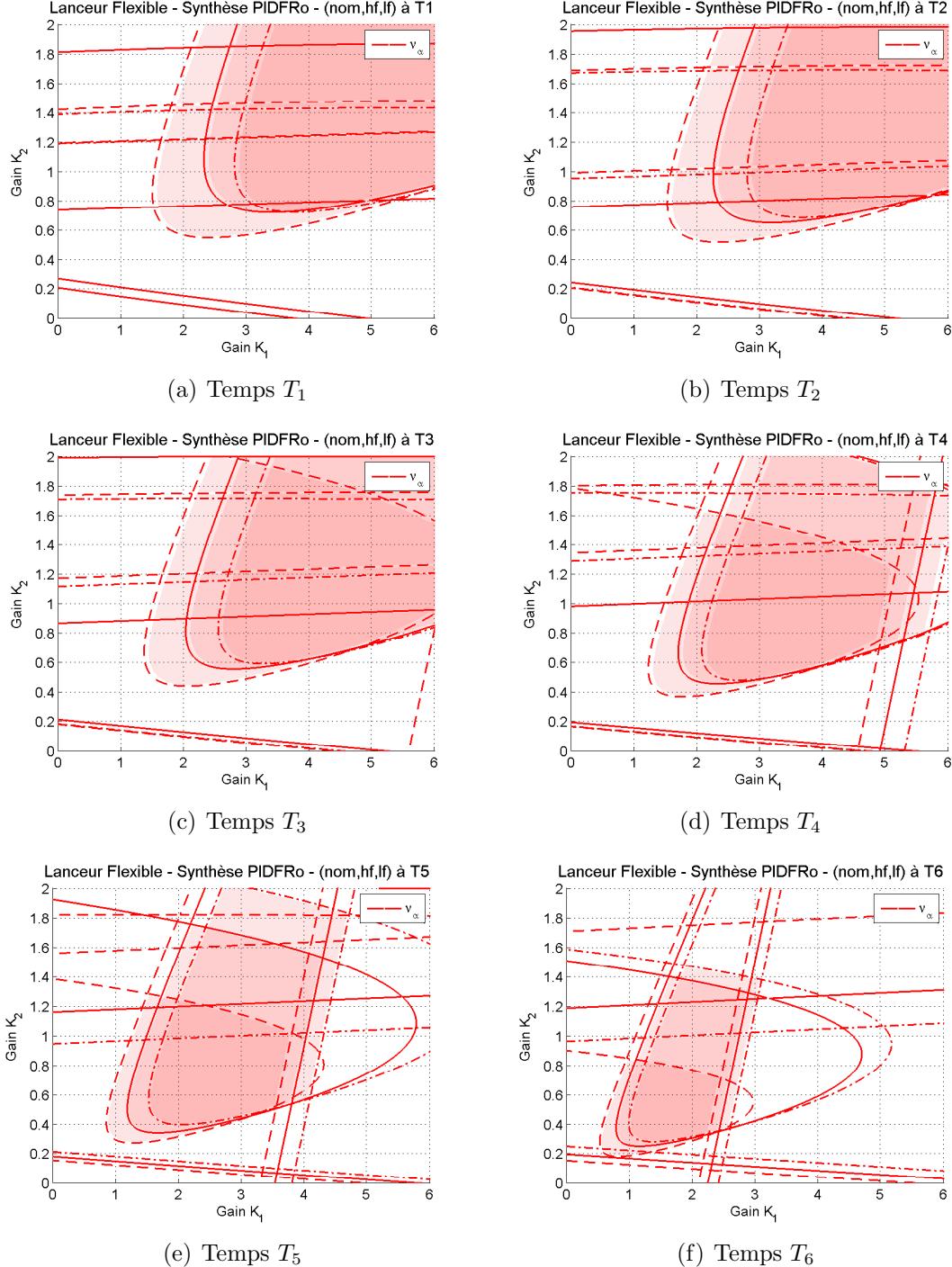


Figure A.1 Intersections des composantes stables sur les trois modèles pour chaque temps
Contrainte en partie réelle sur les pôles rigides

ANNEXE B

Composantes des applications gardiennes ν_β

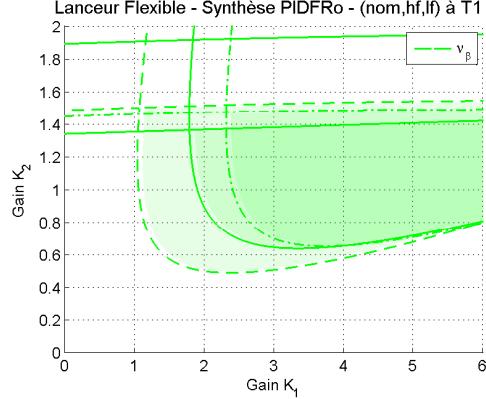
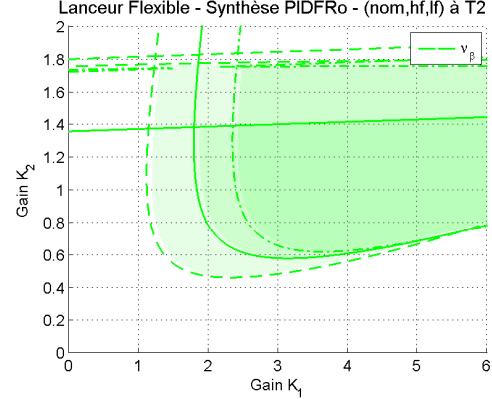
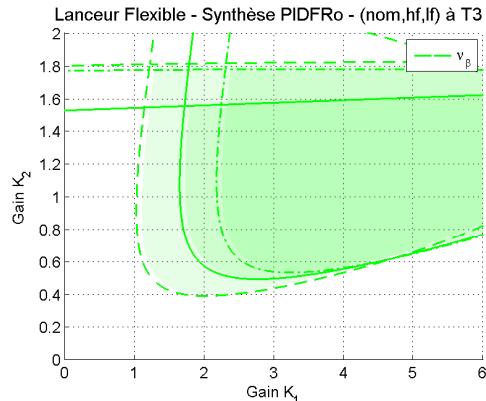
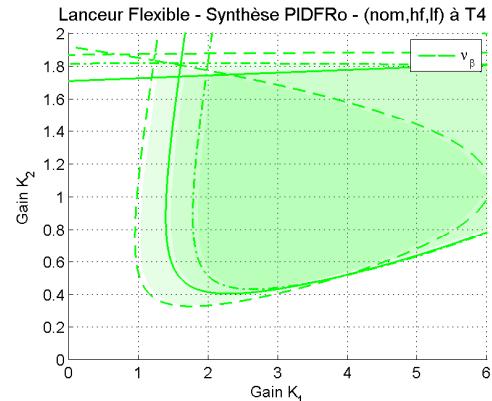
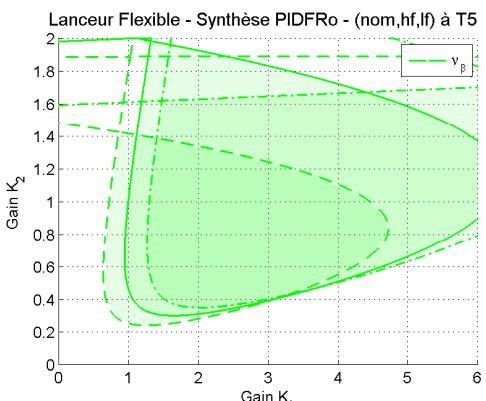
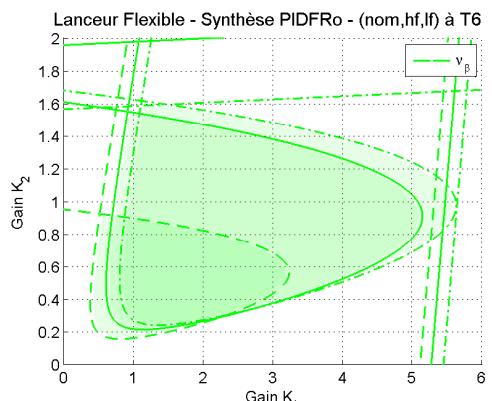
(a) Temps T_1 (b) Temps T_2 (c) Temps T_3 (d) Temps T_4 (e) Temps T_5 (f) Temps T_6

Figure B.1 Intersections des composantes stables sur les trois modèles pour chaque temps
Contrainte en amortissement sur les pôles rigides

ANNEXE C

Composantes des applications gardiennes ν_ξ

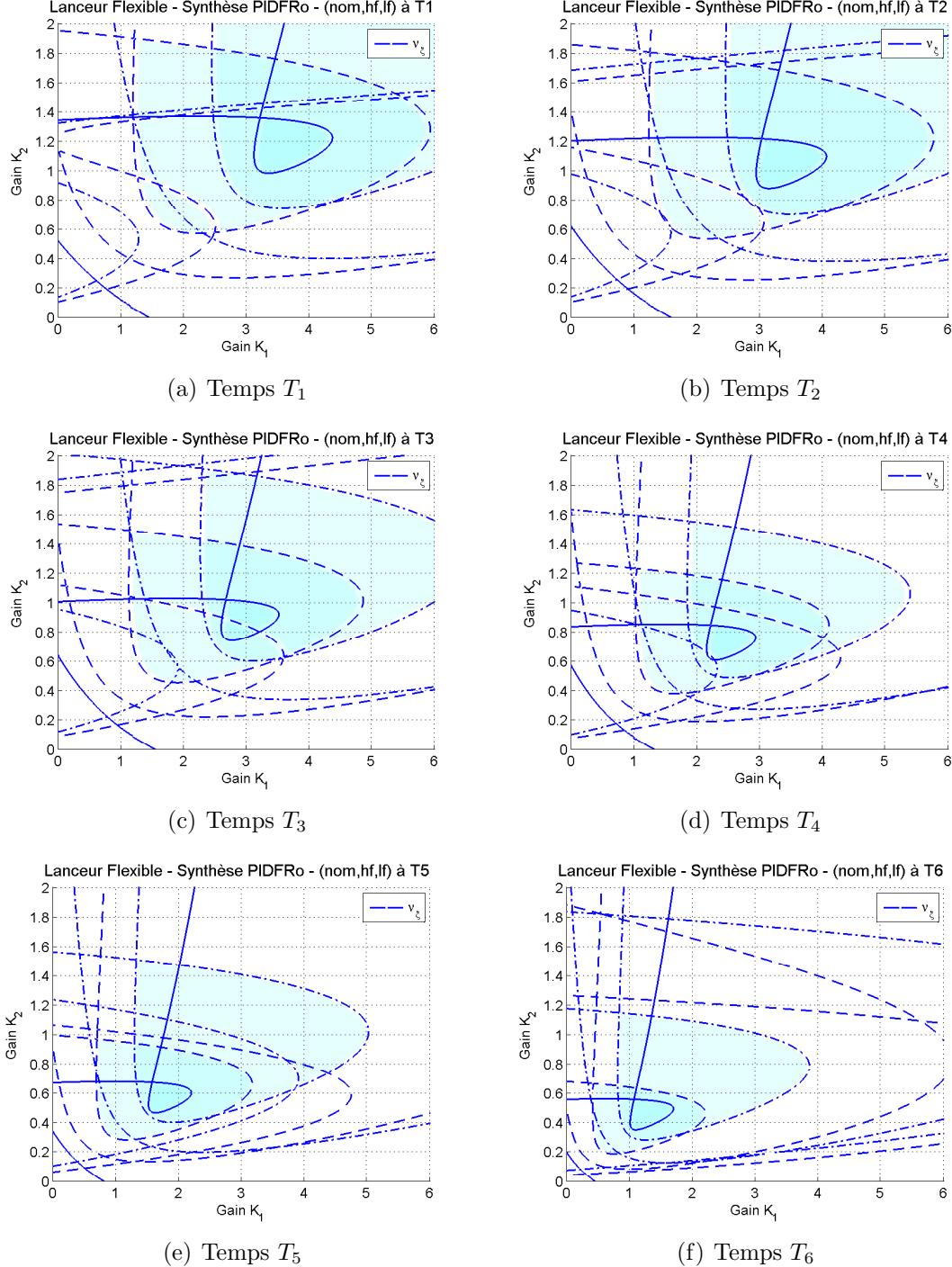


Figure C.1 Intersections des composantes stables sur les trois modèles pour chaque temps
Contrainte en stabilité sur les pôles flexibles

ANNEXE D

Superposition des composantes des applications gardiennes ν_α , ν_β et ν_ξ

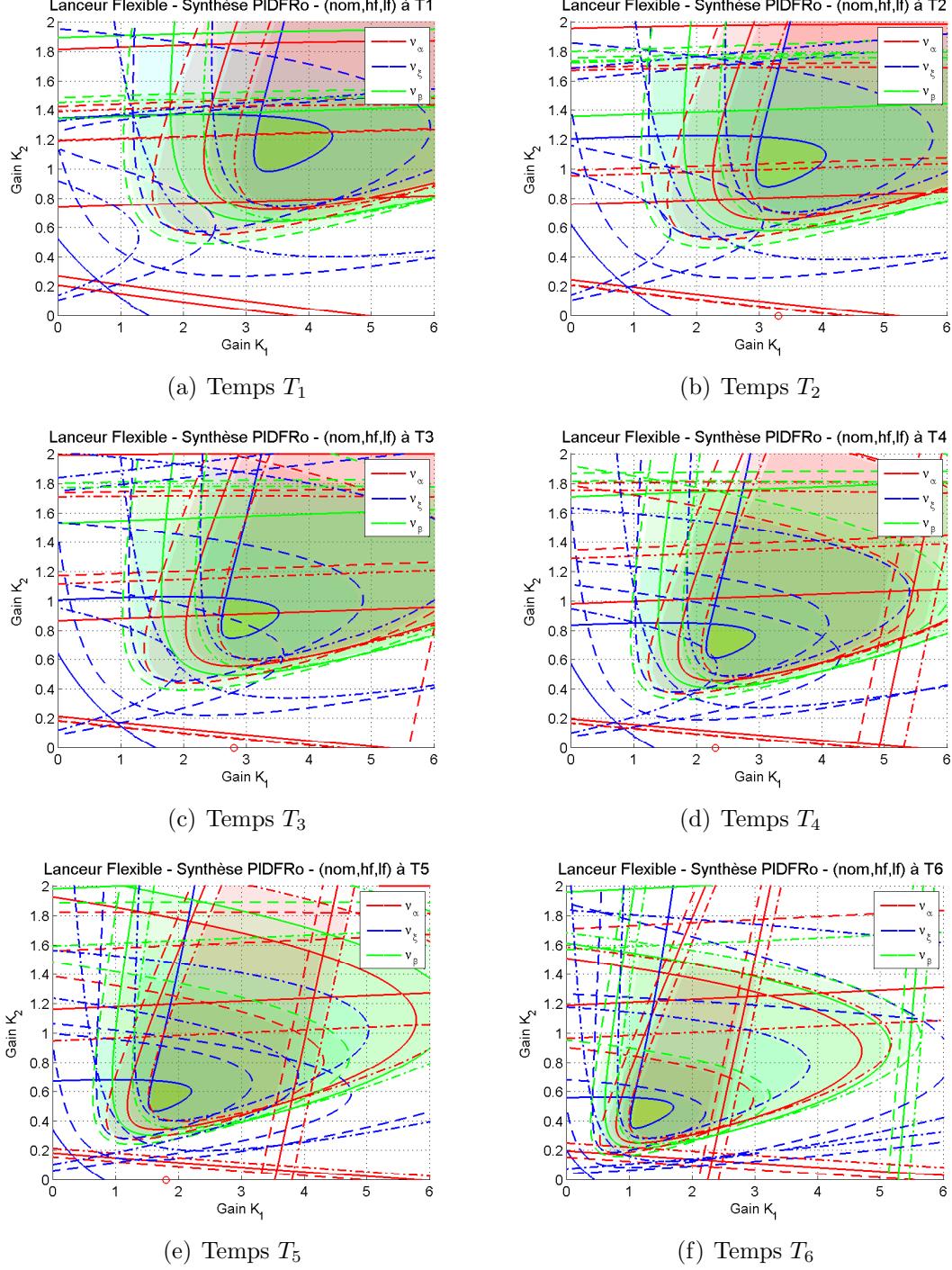


Figure D.1 Intersections des composantes stables sur les trois modèles pour chaque temps