

Titre: Commande structurée H-infinie de la dynamique rigide et aéro-élastique d'un avion civil
Title: [Commande structurée H-infinie de la dynamique rigide et aéro-élastique d'un avion civil](#)

Auteur: Élodie Jacquet
Author: [Élodie Jacquet](#)

Date: 2018

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Jacquet, É. (2018). Commande structurée H-infinie de la dynamique rigide et aéro-élastique d'un avion civil [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. <https://publications.polymtl.ca/3787/>
Citation: [Jacquet, É. \(2018\). Commande structurée H-infinie de la dynamique rigide et aéro-élastique d'un avion civil \[Mémoire de maîtrise, École Polytechnique de Montréal\]. PolyPublie. https://publications.polymtl.ca/3787/](#)

Document en libre accès dans PolyPublie

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/3787/>
PolyPublie URL: [https://publications.polymtl.ca/3787/](#)

Directeurs de recherche: David Saussié, & Lahcen Saydy
Advisors: [David Saussié](#), [Lahcen Saydy](#)

Programme: génie électrique
Program: [génie électrique](#)

UNIVERSITÉ DE MONTRÉAL

COMMANDÉ STRUCTURÉE H_∞ DE LA DYNAMIQUE RIGIDE ET
AÉRO-ÉLASTIQUE D'UN AVION CIVIL

ÉLODIE JACQUET
DÉPARTEMENT DE GÉNIE AÉROSPATIAL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE)
DÉCEMBRE 2018

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

COMMANDÉ STRUCTURÉE H_∞ DE LA DYNAMIQUE RIGIDE ET
AÉRO-ÉLASTIQUE D'UN AVION CIVIL

présenté par : JACQUET Élodie

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

M. GOURDEAU Richard, Ph. D., président

M. SAUSSIÉ David, Ph. D., membre et directeur de recherche

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

M. KRON Aymeric, Ph. D., membre

DÉDICACE

Je dédie ce mémoire de maîtrise à ma famille.

À mes chers parents qui m'ont énormément soutenue tout au long de mes études.

À mes sœurs, vous voir avancer dans vos vies m'a apporté la plus grande des motivations.

À tous ceux qui m'ont manqué, de la rédaction de la première ligne à la dernière.

À mes grands-parents.

REMERCIEMENTS

Mes premiers remerciements vont à Aymeric Kron pour son aide, sa patience et son précieux mentorat pendant mes stages à Bombardier Aerospace. Ma maîtrise a été construite autour de tes enseignements, je t'en remercie grandement.

Je remercie ensuite très sincèrement David Saussié pour les suivis et les conseils avisés. Merci de m'avoir fait confiance pour représenter le laboratoire auprès de Bombardier Aerospace, et de vivre une opportunité incroyable.

Je profite aussi de l'occasion pour remercier Lahcen Saydy, mon co-directeur de maîtrise pour m'avoir offert la chance d'être étudiante en maîtrise dans ce laboratoire. Merci également pour m'avoir recommandé pour ce stage à Bombardier Aerospace. Un immense merci à Mitacs Canada et à leur programme Mitacs Acceleration qui a permis de financer mon projet de recherche avec Bombardier Aerospace.

Merci à Hugo et Frédéric, pour leur aide et leurs nombreux conseils. Hugo, merci pour avoir initié le projet, tes scripts m'ont servi de bon point de départ. Frédéric, merci pour tes modèles flexibles (dans un domaine plus étendu), la problématique aurait été compliquée à adresser sans. Merci à vous deux, on se revoit dans le bus de la ligne 72. Merci à toute l'équipe de Control Laws de Bombardier Aerospace, pour avoir toujours pris du temps pour répondre à mes questions et participer à mes présentations. Bonne chance dans l'apprentissage de cette nouvelle technique, j'espère vraiment qu'elle pourra vous être utile dans de nouveaux projets. Merci à Thomas. L^AT_EX marche à merveille et m'a sauvé un temps précieux. Merci pour ces heures au café conjointement déstressantes.

Merci à Marie, qui a cru en moi et m'a encouragée dès le premier jour de mon stage. Tu fais d'ores et déjà des merveilles.

Merci à Sylvain. Mes figures sont toutes jolies maintenant. Après 1000 "Et ce mémoire, il avance bien?", je peux maintenant répondre "Oui, tout fini!".

Merci à Paul et à mes colocataires pour avoir été à l'écoute et pour avoir apporté le recul parfois nécessaire.

Finalement un grand merci à Polytechnique Montreal et à CentraleSupélec qui, grâce à leur partenariat, m'ont permis de me spécialiser dans le domaine de l'aéronautique. Grâce à ce double-diplôme, je réalise une ambition profonde et rapproche mon parcours académique de mes objectifs professionnels.

RÉSUMÉ

La synthèse des lois de commande d'un avion de transport civil est un défi complexe ; l'avion évolue en effet dans un domaine très large. Les lois de commande sont généralement séquencées en fonction de paramètres comme la vitesse ou l'altitude afin de tenir compte du changement de dynamique d'une condition de vol à une autre. Ces lois doivent en revanche être robustes à des variations de la masse ou de la position du centre de gravité. La synthèse doit finalement trouver le compromis idéal entre robustesse et performance afin d'obtenir la certification de l'avion.

Afin de réduire la consommation de carburant et d'augmenter les distances franchissables, la nouvelle génération d'avions civils est composée de matériaux, plus légers et plus souples qui réduisent la rigidité de la structure de l'avion, et provoquent l'apparition d'oscillations perturbatrices ; l'avion devient alors plus flexible. Avec des bandes passantes larges, les contrôleurs sont alors capables de détecter ces oscillations perturbatrices, qui influencent les commandes. Optimisées pour les dynamiques rigides, les lois de commandes ne sont plus aussi stables et aussi performantes, et peuvent entraîner l'excitation des modes souples de la structure. Il devient alors nécessaire de tenter de modéliser ces dynamiques flexibles et de procéder à un nouveau réglage des lois de commande en incluant le modèle aérolastique de l'avion. Le but de ce projet est donc de produire une loi de commande qui prend en compte les modèles rigides et flexibles, et qui propose un compromis entre performance et robustesse.

Pour effectuer ces nouveaux réglages, on propose d'utiliser une nouvelle méthode de synthèse appelée la synthèse H_∞ structurée. Cette méthode multi-modèle permet de fixer l'architecture du contrôleur à régler, et de chercher la solution optimale qui respecte des contraintes préalablement définies tout en minimisant d'autres contraintes. Cette méthode semble donc bien adaptée à un contexte industriel où l'on cherche à satisfaire un cahier des charges rempli de contraintes de différentes natures, tout en maintenant une architecture de commande la plus simple possible.

À partir des données fournies par Bombardier Aerospace, on prouve d'abord que la technique peut être utilisée pour produire une loi de commande certifiable pour un avion civil. Pour cela, on synthétise un contrôleur robuste auto-séquencé qui satisfait l'ensemble des requis de performance et de stabilité. Dans cette première application, on considère l'ensemble des points de vol de la configuration d'atterrissage de l'avion.

On démontre ensuite la capacité de la méthode de synthèse à prendre en compte les modèles flexibles, ainsi que les contraintes de stabilité supplémentaires qui leurs sont liés. On augmente la boucle interne au réglage d'un gain supplémentaire qui intervient dans la commande en vitesse de roulis, afin de mieux positionner le mode roulis pur. Dans cette seconde application, on considère l'ensemble des points de vol de la configuration de croisière. Finalement, en augmentant à nouveau les degrés de liberté du réglage, on étend la technique pour diminuer l'influence des modes flexibles. Les gains des filtres aéroservoélastiques sont optimisés par la synthèse pour faire coïncider les fréquences de coupures des filtres avec les dynamiques flexibles dominantes. Dans cette dernière application, une seule condition de vol est considérée. On obtient un contrôleur performant et robuste face aux variations de masse et de centrage.

ABSTRACT

The design of a civil transport aircraft control law is a complex challenge; indeed the aircraft operates in a very large flight envelope. Control laws are generally scheduled with respect to parameters such as speed or altitude to account for the change in dynamics from one flight condition to another. However, these laws must be robust to variations in mass or centre of gravity position. The synthesis must finally find the ideal compromise between robustness and performance in order to obtain the aircraft certification.

To reduce fuel consumption and increase flying range, the new generation of civil airplanes is made of lighter and more flexible materials which reduce the rigidity of the aircraft structure, and cause disruptive oscillations to occur; the aircraft then becomes more flexible. With wide bandwidths, controllers are then able to detect these disruptive oscillations, influencing the control law. Optimized for rigid dynamics, control laws are no longer as stable and as efficient as they used to be, and can lead to the excitation of the structure's flexible modes. It then becomes necessary to try to model these flexible dynamics and to make a new adjustment of the control laws by including the aeroelastic model of the aircraft. The purpose of this project is therefore to produce a control law that takes into account both flexible and rigid models, and offers a compromise between robustness and performance.

To make these new settings, it is proposed to use a new synthesis method called the structured H_∞ synthesis. This multi-model method allows to fix the architecture of the controller, and to look for the optimal local solution that respects the constraints previously defined while minimizing other constraints. This method therefore seems to be well-suited to an industrial context where the aim is to satisfy a full set of specifications of different types of constraints, while maintaining a control architecture that is as simple as possible.

Based on the data provided by Bombardier Aerospace, it is first proven that the technique can be used to produce a certifiable control law for a civil aircraft. To do this, we synthesize a robust self-scheduled controller that satisfies all the requirements on performance and stability. In this first application, we consider all the flight cases of the aircraft's landing configuration.

The ability of the synthesis method to take into account the flexible models is then demonstrated, as well as the additional stability constraints associated with them. We are extending the internal loop to an additional tunable gain that occurs in the control of roll speed, in

order to improve the roll mode position. In this second application, all flight cases in cruise configuration are considered.

Finally, by increasing the degrees of freedom of adjustment again, the technique is extended to reduce the influence of flexible modes. The gains of the aeroservoelastic filters are optimized by the synthesis to match the filter cut-off frequencies with the dominant flexible dynamics. In this last application, only one flight condition is considered. We obtain a powerful and robust controller against mass variations and centering.

TABLE DES MATIÈRES

DÉDICACE	iii
REMERCIEMENTS	iv
RÉSUMÉ	v
ABSTRACT	vii
TABLE DES MATIÈRES	ix
LISTE DES TABLEAUX	xiv
LISTE DES FIGURES	xv
LISTE DES SIGLES ET ABRÉVIATIONS	xix
LISTE DES ANNEXES	xx
 CHAPITRE 1 INTRODUCTION	1
1.1 Contexte du projet de recherche	1
1.1.1 Contexte technologique	1
1.1.2 Les techniques de commande robuste	2
1.1.3 Évolution de la synthèse H_∞	3
1.1.4 Création d'un partenariat	8
1.1.5 Mise à disposition de ressources	9
1.2 Définition de la problématique et méthodologie	9
1.2.1 Problématique et objectifs	9
1.2.2 Plan et méthodologie	10
1.3 Confidentialité et divulgation	12
1.3.1 Architecture et schémas	12
1.3.2 Réponses temporelles, fréquentielles et graphiques	12
1.3.3 Valeurs numériques	13
 CHAPITRE 2 MODÉLISATION L'AVION CIVIL	14
2.1 Introduction	14
2.2 Présentation de l'avion	15

2.2.1	Loi normale et commande latérale/directionnelle	15
2.2.2	Angles, vitesses et accélérations	15
2.2.3	Éléments commandables et surfaces de contrôle	18
2.3	Présentation du domaine de vol	19
2.3.1	Configuration de croisière et d'atterrissage	19
2.3.2	Configuration de la charge (<i>Loading Configuration</i>)	20
2.3.3	Condition de vol (<i>Flight condition</i>)	21
2.4	Présentation de l'architecture des lois de commandes	22
2.4.1	Architecture de la loi normale latérale et directionnelle	22
2.4.2	Gains de retour d'état (Feedback)	23
2.4.3	Gains de précommande (Feedforward)	23
2.5	Présentation des modèles d'avion rigides et flexibles	25
2.5.1	Origine des modèles	25
2.5.2	Modèles de l'avion rigide et de l'avion flexible	26
2.5.3	Modèles Boucle-Ouverte (BO) des modèles rigide et flexible	31
2.5.4	Modèles en BO pour les marges de stabilité	37
2.5.5	Modèles en Boucle Fermée (BF) de l'avion rigide et flexible	39
2.6	Présentation des filtres AéroServoÉlastique	43
2.6.1	Objectif des filtres ASE	43
2.6.2	Composition des filtres et éléments réglables	43
2.6.3	Analyse de l'impact sur les modes flexibles	45
2.7	Conclusion	47
CHAPITRE 3	DÉVELOPEMENT DES CONCEPTS ET DU DESIGN	48
3.1	Introduction	48
3.1.1	Contexte et activités précédentes	48
3.1.2	Objectifs	48
3.1.3	Méthodologie suivie	49
3.2	La synthèse H_∞ structurée	49
3.2.1	Synthèse structurée	49
3.2.2	Synthèse multi-objectifs	50
3.2.3	Outils MATLAB® pour la synthèse H_∞ structurée	50
3.2.4	Synthèse robuste et séquencée	52
3.3	Des requis vers les concepts	53
3.3.1	Structure de la loi de commande et séquencement des gains	54
3.3.2	Suivi de référence et découplage de signaux	56

3.3.3	Marges de stabilité	59
3.3.4	Positionnement des pôles	62
3.3.5	Efforts des actionneurs	64
3.4	Optimisations de design	64
3.4.1	Préciser la condition initiale	65
3.4.2	Préciser le réglage des fonctions de pondérations	65
3.5	Conclusion	66
CHAPITRE 4 CONCEPTION D'UN OUTIL LOGICIEL		67
4.1	Introduction	67
4.1.1	Contexte et présentation	67
4.1.2	Quelques rappels de génie logiciel	67
4.2	Requis	68
4.3	Concepts, architecture et codage	69
4.3.1	Principales fonctionnalités	69
4.3.2	Définition des objets	69
4.3.3	Architecture générale	70
4.3.4	Interface utilisateur	72
4.3.5	Application de la synthèse H_∞ structurée	72
4.3.6	Problèmes rencontrés et leurs solutions	74
4.3.7	Codage	75
4.4	Tests et validation	76
4.5	Documentation	76
4.6	Conclusion	77
CHAPITRE 5 SYNTHÈSE D'UN CONTRÔLEUR ROBUSTE ET AUTO-SÉQUENCÉ POUR DES MODÈLES D'AVION RIGIDE		78
5.1	Introduction	78
5.1.1	Contexte et activités précédentes	78
5.1.2	Objectifs de l'application	78
5.1.3	Méthodologie suivie	79
5.2	Requis et design	79
5.2.1	Requis sur les performances de la boucle interne	79
5.2.2	Concepts et canaux de performances	80
5.2.3	Architecture du modèle augmenté	80
5.3	Explicitation des cas utilisés pour l'application	81
5.4	Détermination de la condition initiale	82

5.4.1	Optimisation point-à-point	82
5.4.2	Interpolation polynomiale des gains	84
5.5	Synthèse et optimisations	85
5.5.1	Exclusions d'une partie des conditions de vol	85
5.5.2	Catégorisation des modèles basses/hautes vitesses	86
5.5.3	Améliorations du design	88
5.6	Résultats et performances après réglage de la boucle linéaire	88
5.6.1	Evolution de l'amplitude des gains	88
5.6.2	Suivi de références	91
5.6.3	Découplages de signaux	92
5.6.4	Position des pôles de la boucle fermée	94
5.6.5	Marges de stabilité	94
5.6.6	Efforts des actionneurs	96
5.7	Conclusion	97
CHAPITRE 6 SYNTHÈSE D'UN CONTRÔLEUR ROBUSTE ET SÉQUENCÉ POUR		
DES MODÈLES D'AVION RIGIDES ET FLEXIBLES		98
6.1	Introduction	98
6.1.1	Contexte et activités précédentes	98
6.1.2	Objectifs de l'application	98
6.1.3	Méthodologie suivie	99
6.2	Requis et design	99
6.2.1	Requis sur les performances de la boucle interne	99
6.2.2	Concepts et canaux de performances	100
6.2.3	Architecture du modèle augmenté	100
6.3	Explicitation des cas utilisés pour l'application	101
6.4	Synthèse et optimisations	103
6.4.1	Design par groupe de conditions de vol	103
6.4.2	Ordre des modèles flexibles	103
6.5	Résultats et performances après réglage de la boucle linéaire	103
6.5.1	Evolution de l'amplitude des gains	103
6.5.2	Suivi de références	106
6.5.3	Découplages de signaux	108
6.5.4	Position des pôles de la boucle fermée	110
6.5.5	Marges de stabilité	111
6.5.6	Efforts des actionneurs	115

6.6 Conclusion	117
CHAPITRE 7 RÉGLAGE DES FILTRES AÉROSERVOÉLASTIQUES PAR UNE SYNTHÈSE ROBUSTE	118
7.1 Introduction	118
7.1.1 Contexte et activités précédentes	118
7.1.2 Objectifs de l'application	118
7.1.3 Méthodologie suivie	119
7.2 Requis et design	119
7.2.1 Requis sur les performances de la boucle interne	119
7.2.2 Concepts et canaux de performances	120
7.2.3 Architecture du modèle augmenté	120
7.3 Explication des cas utilisés pour l'application	121
7.4 Synthèse et optimisations	121
7.4.1 Limitations sur les filtres AéroServoÉlastique (ASE)	122
7.4.2 Ajout de configurations de charge différentes	122
7.4.3 Augmentation de l'ordre de réduction des modèles flexibles	124
7.5 Résultats et performances après réglage de la boucle linéaire	124
7.5.1 Evolution de l'amplitude des gains	124
7.5.2 Suivi de références	127
7.5.3 Découplages de signaux	129
7.5.4 Position des pôles de la boucle fermée	131
7.5.5 Marges de stabilité	132
7.5.6 Efforts des actionneurs	135
7.6 Conclusion	136
CHAPITRE 8 CONCLUSION	137
8.1 Synthèse des travaux	137
8.2 Limitations de la solution proposée	137
8.3 Améliorations futures	138
RÉFÉRENCES	139
ANNEXES	144

LISTE DES TABLEAUX

Tableau 2.1	Signaux d'entrées du modèle d'avion	26
Tableau 2.2	Signaux de sortie du modèle d'avion	26
Tableau 2.3	États du modèle d'avion	29
Tableau 2.4	Signaux d'entrées du modèle d'avion BO	31
Tableau 2.5	Signaux de sorties du modèle d'avion BO	31
Tableau 2.6	Signaux d'entrées du modèle d'avion BO de stabilité	37
Tableau 2.7	Signaux de sorties du modèle d'avion BO de stabilité	37
Tableau 2.8	Signaux d'entrées du modèle d'avion BF	39
Tableau 2.9	Signaux de sorties du modèle d'avion BF	40

LISTE DES FIGURES

Figure 1.1	Schéma du problème H_∞	3
Figure 1.2	Schéma du problème augmenté H_∞	5
Figure 1.3	Schéma du problème augmenté H_∞	8
Figure 1.4	Cycle en « V »	11
Figure 2.1	Les trois rotations de l'avion	17
Figure 2.2	Angles de dérapage et d'attaque	17
Figure 2.3	Surfaces de contrôle de l'avion	19
Figure 2.4	Domaine de masse et de charge (FAA)	20
Figure 2.5	Architecture générale de la loi de commande	23
Figure 2.6	Architecture de la dynamique de précommande	24
Figure 2.7	Allures fréquentielles des modèles d'avion rigide et flexible	28
Figure 2.8	Pôles de modèles d'avion rigide et flexible	29
Figure 2.9	Pôles d'un modèle d'avion rigide	30
Figure 2.10	Pôles dominants des modèles d'avion rigide et flexible	30
Figure 2.11	Schéma de l'architecture de la BO	31
Figure 2.12	Pôles d'un modèle BO d'avion rigide	32
Figure 2.13	Pôles dominants d'un modèle BO d'avion rigide	32
Figure 2.14	Allures fréquentielles des modèles BO d'avion rigide et flexible	33
Figure 2.15	Pôles de modèles BO d'avion rigide et flexible	34
Figure 2.16	Pôles dominants de modèles BO d'avion rigide et flexible	34
Figure 2.17	Réduction des modèles BO d'avion rigide et flexible	36
Figure 2.18	Schéma de conception des modèles BO de stabilité	37
Figure 2.19	Allures fréquentielles des BO de stabilité d'un modèle d'avion rigide . .	38
Figure 2.20	Marges de stabilité d'un modèle BO d'avion rigide	39
Figure 2.21	Pôles de modèles BF d'avion	40
Figure 2.22	Allures fréquentielles de modèles BF d'avion rigide et flexible	41
Figure 2.23	Pôles dominants de modèles BF d'avion rigide et flexible	42
Figure 2.24	Pôles dominants de modèles BF d'avion rigide et objectifs	42
Figure 2.25	Allures fréquentielles des filtres ASE sans le filtre anti-bruit	44
Figure 2.26	Allures fréquentielles des filtres ASE avec le filtre Noise	44
Figure 2.27	Impact des filtres ASE sur la dynamique des modèles BO d'avion	46
Figure 3.1	Schéma de la boucle interne de la loi de commande (Co1)	54
Figure 3.2	Concept de la limite sur l'amplitude des gains de retour (Co3)	56

Figure 3.3	Concept du suivi de référence (Co4)	57
Figure 3.4	Concept du découplage de signaux (Co5)	58
Figure 3.5	Concept des marges de gains et de phase (Co6)	60
Figure 3.6	Canal de performance pour les marges de stabilité (Co6)	61
Figure 4.1	Diagramme de cas du logiciel	69
Figure 4.2	Diagramme général des classes du logiciel	70
Figure 4.3	Diagramme général des activités du logiciel	71
Figure 4.4	Diagrammes d'activités du logiciel	73
Figure 5.1	Plan augmenté de la synthèse rigide	81
Figure 5.2	Ensemble des conditions de vol sélectionnées	81
Figure 5.3	Ensemble des conditions de charge sélectionnées	82
Figure 5.4	Gains de feedback obtenus par synthèse point à point	83
Figure 5.5	Gains de feedforward obtenus par synthèse point à point	84
Figure 5.6	Lissage du gain K13 après une synthèse point à point	85
Figure 5.7	Catégorisation de la configuration d'atterrissement	87
Figure 5.8	Extrêums des amplitudes des gains de feedback	89
Figure 5.9	Extremums des amplitudes des gains de feedforward	89
Figure 5.10	Évolution des gains de feedback sur l'enveloppe	90
Figure 5.11	Évolution des gains de feedforward sur l'enveloppe	90
Figure 5.12	Performances du suivi de référence de ϕ_{cmd} vers ϕ	91
Figure 5.13	Performances du suivi de référence de β_{cmd} vers β	92
Figure 5.14	Performances du découplage de signaux de ϕ_{cmd} vers β	93
Figure 5.15	Performances du découplage de signaux de β_{cmd} vers ϕ	93
Figure 5.16	Pôles et zéros de la BF	94
Figure 5.17	Marges de stabilité de la BO de l'aileron	95
Figure 5.18	Marges de stabilité de la BO du rudder	95
Figure 5.19	Efforts des actionneurs pour des commandes de ϕ_{cmd} et β_{cmd}	96
Figure 6.1	Modèle augmenté de la synthèse rigide et flexible	101
Figure 6.2	Ensemble des configurations de charge sélectionnées	102
Figure 6.3	Ensemble des conditions de vol sélectionnées	102
Figure 6.4	Extremums des amplitudes des gains de feedback	104
Figure 6.5	Extremums des amplitudes des gains de feedforward	104
Figure 6.6	Évolution des gains de feedback sur l'enveloppe	105
Figure 6.7	Évolution des gains de feedforward sur l'enveloppe	105
Figure 6.8	Évolution du gains de feedforward $P_{21,Dyn}$ sur l'enveloppe	106
Figure 6.9	Performances du suivi de référence de P_{cmd} vers P	107

Figure 6.10	Performances du suivi de référence de ϕ_{cmd} vers ϕ	107
Figure 6.11	Performances du suivi de référence de β_{cmd} vers β	108
Figure 6.12	Performances du découplage de signaux de P_{cmd} vers β	109
Figure 6.13	Performances du découplage de signaux de ϕ_{cmd} vers β	109
Figure 6.14	Performances du découplage de signaux de β_{cmd} vers ϕ	110
Figure 6.15	Pôles et zéros de la BF	111
Figure 6.16	Marges de stabilité de la BO de l'aileron	112
Figure 6.17	Marges de stabilité de la BO du rudder	112
Figure 6.18	Marges d'amplitude de la BO de l'aileron	114
Figure 6.19	Marges d'amplitude de la BO du rudder	114
Figure 6.20	Efforts des actionneurs pour des commandes de P_{cmd} , ϕ_{cmd} et β_{cmd} . .	116
Figure 7.1	Modèle augmenté de la synthèse réglant les filtres ASE	121
Figure 7.2	Ensemble des configurations de charge sélectionnées	122
Figure 7.3	Optimisation des filtres ASE non restreints	123
Figure 7.4	Marges de stabilité non respectées par la solution	123
Figure 7.5	Amplitudes des gains de feedback	125
Figure 7.6	Amplitudes des gains de feedforward	125
Figure 7.7	Allure fréquentielle des filtres ASE	126
Figure 7.8	Allure fréquentielle du filtre anti-bruit	126
Figure 7.9	Performances du suivi de référence de P_{cmd} vers P	127
Figure 7.10	Performances du suivi de référence de ϕ_{cmd} vers ϕ	128
Figure 7.11	Performances du suivi de référence de β_{cmd} vers β	128
Figure 7.12	Performances du découplage de signaux de P_{cmd} vers β	129
Figure 7.13	Performances du découplage de signaux de ϕ_{cmd} vers β	130
Figure 7.14	Performances du découplage de signaux de β_{cmd} vers ϕ	130
Figure 7.15	Pôles et zéros de la BF	131
Figure 7.16	Pôles avant et après la fermeture de la boucle	132
Figure 7.17	Marges de stabilité de la BO de l'aileron	133
Figure 7.18	Marges de stabilité de la BO de rudder	133
Figure 7.19	Marges d'amplitude de la BO de l'aileron	134
Figure 7.20	Marges d'amplitude de la BO du rudder	134
Figure 7.21	Efforts des actionneurs pour des commandes de P_{cmd} , ϕ_{cmd} et β_{cmd} . .	135
Figure B.1	Représentation en schéma-bloc d'une boucle de commande	145
Figure D.1	Allure fréquentielle d'un modèle d'avion rigide	149
Figure D.2	Allure fréquentielle d'un modèle BO d'avion rigide	150
Figure D.3	Marges de stabilité d'un modèle BO d'avion rigide	151

Figure D.4	Allure fréquentielle d'un modèle BF d'avion rigide	152
Figure D.5	Allures fréquentielles des filtres ASE sans le filtre Noise	153
Figure D.6	Allures fréquentielles des filtres ASE avec le filtre Noise	154
Figure E.1	Gains de feedback obtenus par une synthèse point à point	155
Figure E.2	Gains de feedforward obtenus par une synthèse point à point	156
Figure E.3	Évolution des gains de feedback sur l'enveloppe	157
Figure E.4	Évolution des gains de feedforward sur l'enveloppe	158
Figure F.1	Évolution des gains de feedback sur l'enveloppe	159
Figure F.2	Évolution des gains de feedforward sur l'enveloppe	160

LISTE DES SIGLES ET ABRÉVIATIONS

AFTCG	Aft CG
ALT	Altitude
ALTMAX	Altitude Maximale
ALTMIN	Altitude Minimale
ASE	AéroServoÉlastique
BA	Bombardier Aerospace
BF	Boucle Fermée
BMI	Bilinear Matrix Inequalities (<i>inégalités matricielles bilinéaires</i>)
BO	Boucle Ouverte
CAS	Calibrated Airspeed
CG	Centre de Gravité
CLaws	Control Laws
FBW	Fly-By-Wire
FWDCG	Forward CG
HTML	Hyper Text Markup Language
HIFOO	H-Infinity Fixed Order Optimization
LD	Load
LFT	Linear Fractional Transformation
LMI	Linear Matrix Inequalities (<i>inégalités matricielles linéaires</i>)
LTI	Linear Time Invariant
MFS	Multi-Fonction Spoilers
MFW	Minimum Flight Weight
MLW	Maximum Landing Weight
MPC	Model Predictive Control
MTOW	Maximum Take-Off Weight
POO	Programmation Orientée Objet
RMS	Root Mean Square (racine des moindres carrés)
RP	Régime Permanent
RT	Régime Transitoire
SISO	Single Input Single Output
UML	Unified Modeling Language
VMAX	Vitesse Maximale
VMIN	Vitesse Minimale

LISTE DES ANNEXES

Annexe A	RAPPELS MATHÉMATIQUES	144
Annexe B	RAPPELS DE COMMANDE	145
Annexe C	RAPPELS DE GÉNIE LOGICIEL	147
Annexe D	COMPLÉMENT DE FIGURES DU CHAPITRE 2	149
Annexe E	COMPLÉMENT DE FIGURES DU CHAPITRE 5	155
Annexe F	COMPLÉMENT DE FIGURES DU CHAPITRE 6	159

CHAPITRE 1 INTRODUCTION

À travers une revue de la littérature, ce chapitre présente en premier le contexte et justifie la pertinence du projet de recherche (Section 1.1). Puis, il définit les objectifs et la problématique et propose le plan et la méthodologie à suivre pour y répondre (Section 1.2). Enfin, il précise finalement les mesures prises pour la rédaction de ce mémoire pour respecter les accords de confidentialité (Section 1.3).

1.1 Contexte du projet de recherche

1.1.1 Contexte technologique

Dans le domaine de l'aviation, la conception et la construction des avions de transport civil évolue rapidement pour dépasser plus encore les performances de distance, de consommation et de qualité de vol. Parmi l'ensemble des possibilités d'amélioration, deux problématiques sont abordées dans l'article de Canadian Aeronautics and Space Journal (Kron et al., 2003). Dans cette étude réalisée dans le cadre d'un projet financé par Bombardier Aerospace (BA), il peut être remarqué que l'industrie s'intéresse particulièrement à réduire la masse totale de l'avion ainsi qu'à augmenter l'utilisation de systèmes automatisés pour les commandes de l'appareil.

Dans un premier temps, la nouvelle catégorie d'avion appelée Fly-By-Wire (FBW), se différencie d'un pilotage manuel conventionnel par une interface électronique automatique. La commande des surfaces de contrôle est maintenant due à des signaux électriques régis par des contrôleurs réglés avec précision. Cette nouvelle technologie apporte de grandes améliorations dans la qualité de vol autant dans les domaines temporel que fréquentiel, notamment en élargissant les bandes passantes des contrôleurs (Favre, 1994). Les lois de commandes connaissent cependant des limites dans le nombre et la taille des contrôleurs envisageables simplement à cause du volume occupé et de la vitesse de calcul.

Dans un second temps, alléger la masse permet effectivement de réduire efficacement la consommation en carburant. Cependant, pour ce faire, il est nécessaire d'utiliser des matériaux plus légers, et de limiter leur quantité dans la structure. Les nouvelles structures d'avions, plus souples et moins solides, sont ainsi plus facilement sujettes à des perturbations aéroélastiques provenant d'un couplage entre les dynamiques de la structure et de la mécanique de vol classique (Kozek et Schirrer, 2015).

Finalement, la bande passante élargie génère des signaux de commande qui interagissent avec

les dynamiques flexibles. Ces perturbations structurales peuvent alors être transmises à la commande des surfaces de contrôle et nuire très fortement à la qualité de vol, voire même à la sécurité des passagers. Il est définitivement impératif de considérer ces flexibilités lors de la conception et du réglage des lois de commande afin de les rendre plus robustes.

1.1.2 Les techniques de commande robuste

Par définition, la robustesse vise à garantir performance et stabilité face à un ensemble d'incertitudes, comme les perturbations flexibles. Contrairement à l'altitude et la vitesse de l'avion, la position du Centre de Gravité (CG) et les perturbations flexibles sont difficilement estimables. Si l'on est capable de produire une loi de commande séquencée en vitesse et altitude, un séquencement en fonction de paramètres complexement prévisibles (comme la masse et le centrage) n'est pas envisageable. Cependant ces variations de paramètres autour du point d'équilibre existent bien, et il est nécessaire d'avoir une loi de commande qui leur assure aussi stabilité et performance. Il est nécessaire d'avoir une loi de commande robuste face à ces variations de paramètres de vol. Pour cette raison, les techniques de réglage comme la synthèse H_∞ (Zhou et al., 1998), la μ -synthèse (Zhou et al., 1996) ou encore la commande modale robuste (Magni, 2002; Biannic, 2013) sont développées. En effet, elles recherchent fondamentalement à obtenir la robustesse du système. Ces trois méthodes de synthèses et bien d'autres ont donc été étudiées dans les vingt dernières années pour réussir à résoudre cette problématique.

Des années 90, avec le projet GARTEUR¹ (Magni et al., 1997), aux années 2010, de nombreuses applications se sont succédées pour tester différentes techniques de design. En France (Lavigne, 2003) ou encore au Canada (Kron, 2004; Kron et de Lafontaine, Jun. 2004; Alazard et al., Oct. 1999), des travaux sont publiés sur la commande de la dynamique flexible d'avions civils (Joshi et Kelkar, 1996) mais aussi de satellites d'observation (Kron et al., Sept. 2013; Bennani et Ankersen, 2011; Bates et Postlethwaite, 2002; Falcoz et al., 2013) et de lanceurs spatiaux (Kosut et al., 1983; Saussié et al., 2013). Entre autres, le projet PIROLA (PIlotage RObuste des LAnceurs), a réuni les chercheurs Arzelier et Peaucelle (2004) qui ont réussi à améliorer la robustesse de la loi de commande et à réduire considérablement le temps nécessaire au réglage des contrôleurs des véhicules aérospatiaux.

1. Group for Aeronautical Research and Technology in EURose (GARTEUR) est un groupe d'étude qui réunit sept pays d'Europe autour d'une problématique commune. Les résultats consignés dans l'article ont permis de comparer les meilleures techniques de design alors disponibles, afin d'obtenir une commande robuste.

1.1.3 Évolution de la synthèse H_∞

La théorie de la synthèse H_∞ développée à partir de Zames (1981), propose aujourd’hui une méthode de calcul d’un contrôleur performant. L’évolution de la synthèse classique vers la synthèse H_∞ structurée est présentée ici². Un résumé détaillé est aussi donné dans "The H_∞ Control Problem is Solved" (Apkarian et Noll, 2017). Cet article met en évidence que la synthèse H_∞ structurée résoud le problème posé, mais rappelle que la solution donnée est un optimum local uniquement.

Théorie de la synthèse H_∞

La norme H_∞ d’un système $G(s)$ est définie par :

$$\|G\|_\infty = \sup_{\omega \in \mathbb{R}}(\bar{\sigma}(G(j\omega))) \quad (1.1)$$

telle que pour une matrice à valeurs complexes M , sa valeur singulière maximale soit définie par : $\bar{\sigma}(M) = \sqrt{\lambda_{\max}(M^*M)}$, où M^* est la matrice hermitienne de M (Annexe A).

Plus clairement, la norme H_∞ représente aussi le gain maximum de la fonction de transfert $G(j\omega)$, une amplification maximale de transferts. Dans ce cas, minimiser la norme H_∞ d’un système, revient à minimiser l’amplification des transferts.

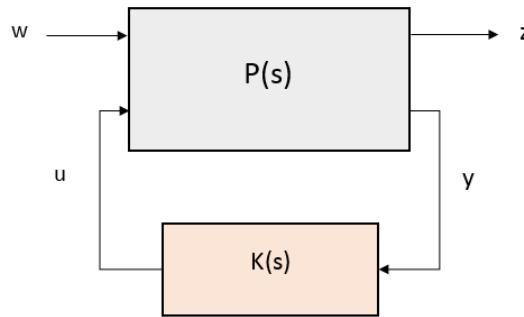


Figure 1.1 Schéma du problème H_∞

La théorie propose de poser le problème selon la figure 1.1 :

- w est le vecteur des entrées exogènes (consignes/perturbations),
- u est le vecteur de commande du système,
- z est le vecteur des sorties régulées,
- y est le vecteur des mesures.

2. On pourra se référer à l’annexe A et B pour des définitions mathématiques et de commande supplémentaires.

Minimiser la norme H_∞ du système revient à un problème de réjection de perturbations. En supposant le système Linear Time Invariant (LTI), il faut alors trouver un contrôleur $K(s)$ qui :

- stabilise internement le système en BF,
- minimise la norme H_∞ du transfert de la perturbation w vers la sortie z .

La synthèse H_∞ , une méthode de synthèse robuste

Comme on l'a vu plus tôt dans le premier paragraphe de la section 1.1.2, il est nécessaire d'avoir une synthèse qui permet la robustesse. Pour représenter l'incertitude liée à la valeur de certains paramètres de $P(s)$ ou à des perturbations non modélisées, on peut ajouter une matrice d'incertitude Δ , connectée en tant que Linear Fractional Transformation (LFT) supérieure (da Silva de Aguiar et al., 2018). Dans le schéma précédent (Fig. 3.1), $P(s)$ est alors remplacé par :

$$\tilde{G} = F_s(P, \Delta) = P_{22} + P_{21}\Delta(I - P_{11}\Delta)^{-1}P_{12} \quad (1.2)$$

$$\text{où } P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}.$$

Dès lors, le théorème du petit gain (*Small Gain Theorem* abordé notamment dans (Zhou et al., 1996, 1998), assure la stabilité de la BF pour toutes les incertitudes $\|\Delta\|_\infty \leq 1$ si la norme H_∞ de la BF est strictement inférieur à 1.

Grâce à ce théorème, on voit apparaître l'intérêt que peut apporter la méthode. En effet, en exprimant l'ensemble des incertitudes dans des sous-matrices Δ_i telles que $\Delta_i \leq 1$, assurer la norme H_∞ de la BF strictement inférieur à 1, revient à assurer la robustesse de la BF face

$$\text{à une perturbation } \Delta = \begin{bmatrix} \Delta_1 & & \\ & \ddots & \\ & & \Delta_n \end{bmatrix}.$$

Plus que la robustesse, des contraintes de performances peuvent aussi être formulées sous des normes H_∞ (concept utilisé en section 3.3). Ces contraintes peuvent donc être traduites en canaux de performances à contraindre. On appelle canal de performance, les fonctions de pondération, fonctions de référence ou transferts supplémentaires nécessaires au design de la synthèse mais qui ne sont pas réellement présents dans le système réel. Le problème peut finalement être représenté par la figure 1.2. On y retrouve la matrice Δ précédente qui se compose de toutes les sous-matrices d'incertitudes et de performances, ainsi que le système P_{aug} qui correspond au système P augmenté des canaux de performances définis.

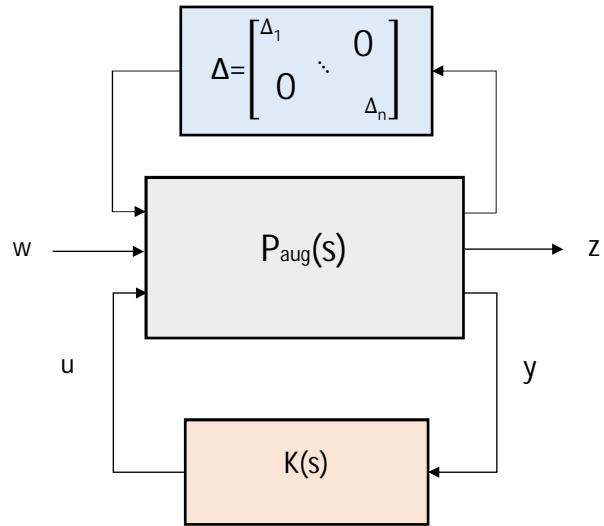


Figure 1.2 Schéma du problème augmenté H_∞

Cependant, on peut se demander si la résolution d'un tel problème est possible. En pratique, assurer que la norme H_∞ est en dessous d'un certain seuil positif γ est parfois suffisant. En effet, on ne cherche pas toujours à obtenir la solution optimale. On peut reformuler en un sous-problème H_∞ : Trouver un contrôleur $K(s)$ qui stabilise internement la LFT inférieure $F_l(P, K)$ et qui assure $\|F_l(P, K)\|_\infty < \gamma$ pour $\gamma > 0$. Ce sous-problème est résolu à l'aide des équations de Riccati (Zhou et al., 1996, 1998).

Cette technique robuste est alors appliquée à des systèmes de l'industrie (Puyou et al., 2004). Bien que la loi de commande soit réglée avec succès, on souligne de nombreuses limitations.

Limitations de la synthèse classique

En effet, la formulation précédente présente trois inconvénients majeurs :

- Cette résolution ne prend en compte que les contraintes formulées sous la forme de contraintes H_∞ . Elle exclut donc les contraintes de nature différente (normes H_2 , placement de pôles, LQR, etc.). Même si le placement de pôles peut être adressé indirectement³ par une norme H_∞ , il pourrait être utile de disposer de techniques multiples pour les contraintes.

3. Le suivi de référence est peut-être exprimé par une norme H_∞ . Par exemple, en forçant la réponse à ressembler à un système de second ordre d'une certaine rapidité et avec un certain amortissement, la paire de pôles complexes dominants de ce transfert sera contrainte à être au voisinage des pôles de la référence afin d'avoir un bon suivi.

- Le contrôleur est obtenu sous la forme d'un modèle d'état, sans structure particulière, d'ordre élevé égal à l'ordre du modèle augmenté. De plus, la structure n'étant pas fixée, le contrôleur obtenu est souvent complexe. Au final, la solution obtenue est une boîte noire qui est difficile à certifier telle quelle dans un contexte industriel. La réduction et la restructuration du contrôleur est un processus lourd (la méthode *Robust Modal Control* est utilisée dans l'article de Puyou et al. (2004)) que l'on préfère éviter en utilisant une synthèse à architecture fixée à priori.
- Les performances sont parfois limitées car elles peuvent être conservatives. La matrice Δ considérée lors de la résolution n'est plus diagonale. Comme la norme s'applique sur l'ensemble du transfert, les transferts croisés sont aussi minimisés alors qu'ils ne sont pas nécessaires. En d'autres termes, les transferts entrées/sorties ne peuvent être considérés indépendamment les uns des autres. La synthèse H_∞ classique minimise la norme H_∞ globale de $F_l(P, K)$, c'est-à-dire que pour un transfert tel que :

$$F_l(P, K) = \begin{bmatrix} T_{w_1 \rightarrow z_1} & T_{w_2 \rightarrow z_1} \\ T_{w_1 \rightarrow z_2} & T_{w_2 \rightarrow z_2} \end{bmatrix} \quad (1.3)$$

afin de minimiser les transferts $T_{w_1 \rightarrow z_1}$ et $T_{w_2 \rightarrow z_2}$, la méthode de synthèse minimise la norme H_∞ de l'ensemble de la LFT, donc les transferts croisés sont inclus aussi. Or, le minimum global obtenu pour K^* n'est pas forcément le minimum particulier des transferts d'intérêt seuls. Par conséquent, les performances peuvent en être réduites.

Il est utile de trouver une technique capable d'imposer des performances particulières, l'emplacement des pôles dans le plan complexe, etc. (Apkarian et al., 2014; Yang et al., 2007). Lorsque la synthèse peut prendre en compte plusieurs objectifs de natures différentes, elle est appelée *multi-objective mixed-synthesis*. L'avantage de ces synthèses est qu'en plus d'imposer plusieurs contraintes au système, parmi un ensemble de contraintes possibles de natures différentes. Dans le cadre de l'étude, il est avantageux d'élargir l'ensemble des concepts possibles. À titre d'exemple, sur un même système P, on pourrait vouloir :

- assurer la robustesse du système (utilisation d'une norme H_∞ à contraindre inférieure à 1),
- assurer des performances en suivi de référence et découplage (utilisation d'une norme H_∞ à minimiser)
- assurer une réduction d'impact de bruit (utilisation d'une norme H_2 à minimiser)
- placer les pôles dans une zone restreinte du plan complexe,
- sans oublier de fixer l'architecture du contrôleur à un ordre limité et une structure spécifique.

La synthèse H_∞ classique n'est pas une technique pertinente, dans le contexte de l'étude, à cause des limitations citées ci-dessus. On choisit une méthode nous permettant de faire une synthèse structurée mixte multi-objectifs. En effet, en 2004, BA n'était alors pas convaincu par la synthèse H_∞ pour le design d'une loi de commande robuste et certifiable d'un avion FBW.

La synthèse H_∞ structurée, la solution

D'autres méthodes de synthèse sont alors recherchées pour résoudre le sous-problème précédent. C'est le cas, entre autres, des approches par Inégalités Matricielles Linéaires (*Linear Matrix Inequalities*) (LMI)s (Gahinet et Apkarian, 1994) qui permettent la synthèse mixte H_2/H_∞ et le placement des pôles (Chilali et Gahinet, 1996; de Abreu et al., 2009). Cette approche est intéressante notamment car le problème est formulé à l'aide de fonctions convexes (Gao et al., 2009), pour lequelles les outils numériques actuels sont capables d'obtenir une solution pertinente efficacement.

Malgré les possibilités offertes par les LMIs, si on utilise la méthode de synthèse pour régler des contrôleurs dont la structure a été définie au préalable, on fait alors intervenir des Inégalités Matricielles Bilinéaires (*Bilinear Matrix Inequalities*) (BMI)s. Contrairement aux LMIs, ces inégalités ne sont plus linéaires, et ne sont plus convexes (Apkarian et Noll, 2006). Sans ces propriétés conservatrices, la résolution est bien plus complexe et par dessus tout, l'existence même d'une solution n'est pas assurée (Safonov et al., 1994).

Parmi toutes les méthodes d'optimisation développées, la technique dénommée H_∞ structurée se démarque car elle résoud les trois limitations présentées à la section précédente :

- Plusieurs natures de contraintes sont acceptées. C'est le cas des suivis de consigne, des réjections de bruit, des gain linéaires quadratiques, des marges de stabilité, du coefficient d'amortissement des pôles, et d'autres.
- L'architecture des contrôleurs peut être fixée a priori et la synthèse vient régler un ensemble de gains déclarés. La plage de variations des gains peut aussi être imposée.
- On peut définir autant de contraintes de performances que désiré. Ces contraintes considèrent seulement des transferts spécifiques.

De plus, plusieurs modèles peuvent être considérés dans une seule synthèse. On peut sélectionner l'ensemble des modèles concernés pour chacunes des contraintes définies. On peut synthétiser un seul contrôleur (robustesse) ou un ensemble de contrôleurs qui suivent une fonction de séquencement (auto-séquencement). Ce dernier aspect est un avantage considérable.

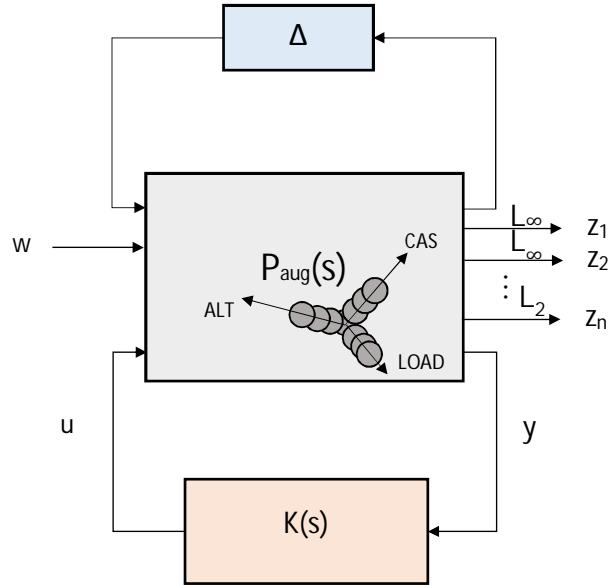


Figure 1.3 Schéma du problème augmenté H_∞

La figure 1.3 présente le schéma augmenté de la méthode de synthèse H_∞ structurée. En comparaison avec la figure 1.2, on insiste sur les avantages multi-modèle, multi-contraintes-mixtes. Les possibilités offertes par cette nouvelle technique font d'elle une méthode de synthèse pertinente pour des applications industrielles, tout comme des applications de commande de vol en aéronautique autant pour un lanceur spatial (Ganet-Schoeller et Desmariaux, 2016) que pour un avion (Lhachemi et al., 2014). En 2017, BA se dit intéressé à renouveler les recherches d'application d'une nouvelle méthode de réglage des lois de commandes.

1.1.4 Création d'un partenariat

David Saussié, professeur agrégé à Polytechnique Montréal, section Automation et Systèmes du département de Génie Électrique, a concentré ses recherches sur les synthèse H_∞ classique et structurée. Il a notamment déjà appliqué la technique à des applications spatiales comme la commande des lanceurs flexibles (Saussié et al., 2013; Knoblauch et al., 2012), et aéronautiques, notamment la commande d'avions (Lhachemi et al., 2015, 2014, 2017).

Aymeric Kron, ingénieur spécialiste chez BA à Montréal, mène des projets de recherche dans la section Control Laws (CLaws) du département Flight Characteristics. Son projet de doctorat portait sur la synthèse H_∞ classique et son application à un avion flexible (Kron

(2004)). On retrouve aussi des études sur la dynamique flexible dans plusieurs de ces articles (Kron et al., 2003; Kron et de Lafontaine, Jun. 2004).

Afin d'améliorer la rapidité de conception des lois de commandes, BA s'intéresse à de nouvelles méthodes de synthèses issues du milieu académique, en partie la synthèse H_∞ structurée. Ce projet a alors pour but de transférer des technologies académiques vers l'industrie.

1.1.5 Mise à disposition de ressources

Dans le cadre du partenariat, des installations ont été mises à disposition autant à l'école Polytechnique, que dans les locaux de BA, auprès de l'équipe CLaws. Il est ainsi donné accès aux ressources matérielles, aux connaissances et aux données de BA (modèles, outils). Les connaissances et ressources matérielles de Polytechnique sont aussi utilisées.

Le projet est mené sous le logiciel Matlab, dans ses versions R2009a et R2016b, selon l'outil manipulé l'une ou l'autre version est nécessaire.

1.2 Définition de la problématique et méthodologie

1.2.1 Problématique et objectifs

Le mémoire tente de répondre à la problématique suivante :

Comment commander les dynamiques rigide et flexible de l'avion par un contrôleur robuste séquencé qui assure performances et stabilité ?

L'objectif principal du projet est donc de concevoir un contrôleur séquencé en vitesse et altitude qui assure stabilité, performance et robustesse sur une large enveloppe de vol à un modèle d'avion comprenant les dynamiques rigide et flexible par la synthèse H_∞ structurée. La réalisation de cet objectif permettra alors de démontrer l'efficacité et la pertinence de la synthèse H_∞ structurée appliquée à des systèmes industriels dans le contexte de BA.

L'objectif principal peut être subdivisé en plusieurs sous-objectifs :

- Associer par un outil logiciel, les données industrielles et la méthode de synthèse académique pour obtenir des résultats concrets et exploitables dans l'environnement de l'entreprise. L'outil a aussi pour but de présenter une interface utilisable par les ingénieurs de CLaws.
- Adresser les requis industriels de performance et de stabilité (sur les dynamiques rigide et flexible) dans un formalisme compatible avec la méthode de synthèse envisagée.

- Appliquer la technique de réglage à un exemple (ensemble de points de vol du modèle rigide tel qu'utilisé aujourd'hui lors du réglage des gains) et analyser la solution obtenue. L'objectif présente ici deux étapes : premièrement, régler la loi de commande, et deuxièmement, la valider par les tests industriels usuels d'évaluation de performance et stabilité. Il convient alors de montrer que la technique peut proposer une solution équivalente ou meilleure au processus déjà en place.
- Appliquer la technique de réglage à un second exemple (ensemble de points de vol des modèles rigide et flexible) et analyser la solution obtenue. L'objectif présente les mêmes étapes que précédemment, régler la loi de commande et enfin la valider par les tests industriels d'évaluation de performance et stabilité usuels. Il convient ici de montrer que la technique peut prendre en compte la dynamique flexible. Pour cela, on considère de nouveaux requis et l'on met en place de nouveaux concepts. En augmentant ainsi ses capacités, on démontre que la synthèse H_∞ est une méthode pratique et fonctionnelle.

1.2.2 Plan et méthodologie

Le **plan** réalisé pour résoudre la problématique est composé d'une série d'étapes. Ces étapes correspondent de manière générale à la division en chapitres de ce mémoire :

1. Comprendre l'environnement des CLaws de BA et les outils à manipuler (Chapitre 2)
2. Comprendre les requis, concepts et l'architecture de la loi de commande (Chapitre 3)
3. Implémenter un outil logiciel intégré dans l'environnement de BA (Chapitre 4)
4. Appliquer la synthèse H_∞ structurée à un exemple usuel (Chapitre 5)
5. Appliquer la synthèse H_∞ structurée à un nouvel exemple incluant la dynamique flexible en utilisant la même architecture des lois de commande (Chapitre 6)
6. Appliquer la synthèse H_∞ structurée à un nouvel exemple incluant la dynamique flexible en réglant aussi les gains des filtres ASE (Chapitre 7)

En amont et durant toutes ces étapes, des activités de recherche sont aussi menées afin d'acquérir les connaissances théoriques nécessaires.

La **méthode** appliquée est la méthode nommée *Cycle en « V »*. Initialement une méthode conceptuelle standard en projet logiciel, le Cycle en « V » est aujourd'hui appliqué à tous les types de projets dans tous les domaines de l'industrie. Il sera utilisée ici autant pour la conception logicielle de l'outil que pour le design des lois de commande qui satisferont les

requis de l'entreprise. La figure 1.4 en expose le principe. Sa structure en « V » permet de gérer facilement les erreurs et les modifications en revenant à l'étape concernée. La succession en cascade est adéquate au projet de cette maîtrise, car l'étude menée est sans précédents (dans le cadre de BA) pour guider les travaux qui rencontrent des problèmes à résoudre. Des allers-retours entre les étapes de design sont à prévoir.

Le côté gauche représente la succession des étapes à suivre pour le design, tandis que le côté droit représente la succession des étapes de vérification, associées à des étapes de design. Le design se construit du général vers le particulier, tandis que la vérification valide le particulier puis le général. Définir les requis revient à déterminer ce que veut BA, les contraintes environnementales, les fonctionnalités majeures, etc. Valider tous les requis revient à s'assurer de la réussite du projet car le produit obtenu satisfera les demandes initiales.

Le design conceptuel définit l'ensemble des concepts théoriques qui parviendront à réaliser les requis demandés. La théorie sous-jacente peut être simple (par exemple, une fonction Matlab commune) ou bien plus complexe et requiert un développement plus long (par exemple, le séquencement des gains).

Il s'agit ensuite d'agencer ces concepts afin de construire la structure d'un logiciel (ou simple script dans le cas de la dernière application - preuves de concept). Quel que soit le produit final, il devrait avoir des entrées, des sorties et un processus composé d'une succession de fonctions reliées par des variables qui implémentent les concepts. La dernière étape consiste à écrire le corps des fonctions (codage).

Méthodologie de Design et de Validation

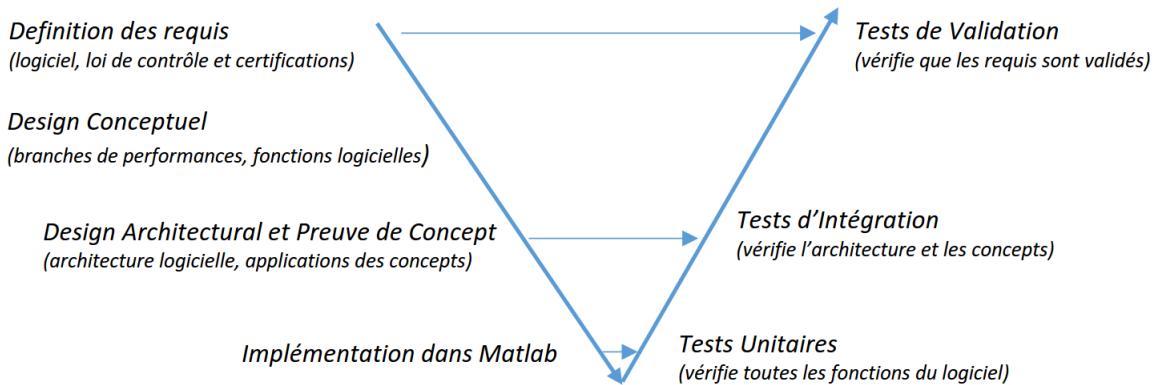


Figure 1.4 Cycle en « V »

1.3 Confidentialité et divulgation

Bombardier Aerospace possède l'entièr e propriété des données utilisées pour ce projet, ainsi que des résultats obtenus. Pour se conformer aux accords de confidentialité, les informations divulguées seront limitées et normalisées.

1.3.1 Architecture et schémas

L'architecture présentée se limite à celle du système étudié, ainsi qu'aux éléments pertinents de celui-ci. La modélisation hors de la boucle de commande (par exemple, en amont) ou celle de la dynamique interne des modèles (par exemple, les actionneurs ou capteurs) ne seront pas dévoilés. Les schémas affichés ne proviennent aucunement de la documentation interne de BA.

1.3.2 Réponses temporelles, fréquentielles et graphiques

Les graphiques représentant des réponses temporelles et fréquentielles offrent l'allure de ces réponses sans pour autant permettre de reconstruire les fonctions et transferts du système. Pour cette raison, les valeurs numériques sont normalisées.

Certains graphiques représenteront plusieurs transferts, afin de pouvoir comparer leurs réponses (référence ou système, rigide ou flexible, nouvelle ou ancienne loi de commande, etc.). On utilisera les notations littérales P pour une phase caractéristique, et F pour une fréquence caractéristique. De cette manière les diagrammes de Bode présenteront des abscisses dépendantes de F (de $10^{-3}F$ à 10^3F rad/sec par exemple). Les diagrammes de Nichols présenteront des abscisses dépendantes de P (de $P - 1440$ à $P + 1440$ degrés par exemple).

Les unités initiales des graphiques (temporels et fréquentiels) sont conservées, pour indiquer des éventuels gains de conversion dûs à des changements d'unités. On rappelle cependant que les valeurs sont normalisées ou rationalisées, et donc que les valeurs numériques réelles ne sont pas données.

En ce qui concerne les graphiques du type des figures 5.4 et 5.8, pour présenter les amplitudes normalisées des gains de feedback ou de feedforward, si A est l'amplitude réelle du gain, le graphique donne le ratio A/X . X est un coefficient choisi, différent pour chacun des graphiques mais identique pour les gains d'un même graphique. Ce processus permet de comparer efficacement l'amplitude relative d'un gain à un autre sans risquer de dévoiler les valeurs numériques des amplitudes réelles des gains.

1.3.3 Valeurs numériques

Pour protéger les données de BA, les valeurs numériques des éléments suivants ne seront pas fournis :

- les modèles d'état de l'avion : matrices A, B, C et D,
- les références de performances : constantes de temps, dépassement, erreur en régime permanent,
- les vitesses, altitude, masse, position du CG,
- la position des modes dans le plan complexe.

Les grandeurs physiques de masse, de vitesse, d'altitude définissant un point de vol pourront être données de la manière suivante :

- une masse : une expression fonction de Maximum Take-Off Weight (MTOW), Minimum Flight Weight (MFW), ou Maximum Landing Weight (MLW),
- une position de CG : une expression fonction de Aft CG (AFTCG) ou Forward CG (FWDCG),
- une altitude : une fonction de Altitude Maximale (ALTMAX) ou Altitude Minimale (ALTMIN),
- une vitesse : une fonction de Vitesse Maximale (VMAX) ou Vitesse Minimale (VMIN).

Afin de comparer numériquement les nouvelles et les anciennes performances, des ratios pourront être donnés et exprimés en pourcentage.

Malgré la censure des données, des gabarits de performances, ou d'autres réponses du système sont fournies pour permettre aux lecteurs de comparer les résultats et de juger correctement le travail effectué. Une annexe confidentielle contenant les figures originales et les données censurées, est délivrée à BA. Seules les personnes autorisées peuvent consulter ce document.

CHAPITRE 2 MODÉLISATION L'AVION CIVIL

Ce chapitre présente l'environnement des lois de commandes. La section 2.1 introduit le chapitre. L'avion est sommairement caractérisé dans la section 2.2. Le domaine de vol considéré et l'ensemble des points d'équilibre sont définis dans la section 2.3. L'architecture des lois de commandes est schématisée dans la section 2.4. Les modèles d'avion utilisés pour cette étude sont définis dans la section 2.5. La section 2.6 présente la protection apportée par les filtres ASE. Finalement la section 2.7 conclut le chapitre.

2.1 Introduction

L'histoire et le concept de la théorie de la synthèse H_∞ structurée ont été vus rapidement à travers la revue de littérature du chapitre 1. L'état de l'art présenté alors, indique que la méthode de synthèse H_∞ structurée montre des résultats prometteurs dans une multitude de disciplines. Notre projet de recherche se concentre sur un cas aéronautique, i.e. la commande d'un avion civil.

Le but de ce chapitre est de présenter l'environnement de la recherche, c'est-à-dire le domaine d'application, une schématisation du système, ainsi que le type de données manipulées.

Les points abordés par ce chapitre seront donc les suivants :

- la présentation de l'avion
- la présentation de l'enveloppe de vol, la division de cet espace et la nomenclature appliquée pour définir précisément un point d'équilibre,
- la présentation des lois de commandes, leur architecture, les caractéristiques des éléments commandables et les limitations,
- la présentation des modèles d'avion utilisés, les signaux d'entrée et de sortie, l'étude des modèles par analyse graphique fréquentielle et l'ajout de la dynamique flexible aux modèles rigides,
- la présentation des filtres ASE, définition et impact sur l'amortissement des modes flexibles.

Une fois tous ces points développés, l'environnement du projet est clairement défini et les étapes de conception sont détaillées.

2.2 Présentation de l'avion

Toute l'étude de ce mémoire repose sur celle d'un modèle d'avion de transport civil FBW, plus particulièrement sur la loi de commande latérale et directionnelle de cet avion. On utilise des modèles représentatifs des dynamiques rigide et flexible à haut niveau de fidélité.

Cette section est un rappel essentiel concernant les éléments structuraux de l'avion commandés et les principales grandeurs physiques mesurées. Il permet une meilleure compréhension des enjeux rencontrés. La plupart du vocabulaire de l'avion sera aussi donné en anglais (noté en *italique*).

2.2.1 Loi normale et commande latérale/directionnelle

Afin de simplifier la commande de l'avion, la dynamique est divisée en deux axes de travail : la commande longitudinale et la commande latérale et directionnelle. La commande longitudinale est associée aux déplacements dans le plan longitudinal de l'avion. La stabilité longitudinale entend une stabilité lors d'une perturbation sur l'axe de tangage¹. La commande latérale/directionnelle (*lateral/directional*) est associée aux déplacements latéraux et directionnels de l'avion. La stabilité latérale se réfère à la stabilité de l'avion autour de son axe de roulis. La stabilité directionnelle (ou verticale) dépend de la stabilité autour de l'axe de lacet.

Cette étude se concentre sur la commande latérale et directionnelle.

2.2.2 Angles, vitesses et accélérations

Principaux repères

Le **repère NED (North-East-Down)** aussi appelé plan tangent local est défini tel que :

- Origine O_0 : point arbitraire à la surface de la Terre,
- Axe x_0 : orienté vers le Nord géographique,
- Axe y_0 : orienté vers l'Est géographique,
- Axe z_0 : orienté vers le centre de la Terre (complète le trièdre direct).

Le **repère avion** est lié à l'avion (Fig. 2.2). Il est défini tel que :

- Origine O_b : point au centre de masse de l'avion,
- Axe x_b : orienté vers le nez de l'avion (plan de symétrie),

1. On peut lire la section 2.2.2 pour une définition des angles d'Euler.

- Axe y_b : orienté vers la droite (complète le trièdre),
- Axe z_b : orienté vers le ventre de l'avion, dans le plan de symétrie et perpendiculaire à x_b .

Le **repère stabilité** est lié à l'avion et est défini par la projection sur le plan de symétrie (plan longitudinal) du vecteur vitesse aérodynamique de l'avion. Il est défini tel que :

- Origine O_s : point au centre de masse de l'avion,
- Axe x_s : orienté selon la projection du vecteur vitesse aérodynamique de l'avion sur le plan de symétrie,
- Axe y_s : similaire à y_b ,
- Axe z_s : orienté vers le bas, complète le trièdre.

Le **repère vent** est lié à l'avion et est défini par le vecteur vitesse aérodynamique de l'avion.

Il est défini tel que :

- Origine O_w : point au centre de masse de l'avion,
- Axe x_w : orienté selon le vecteur vitesse aérodynamique de l'avion,
- Axe y_w : orienté vers la droite (complète le trièdre),
- Axe z_w : orienté vers le bas, dans le plan de symétrie perpendiculaire à y_b , et perpendiculaire à la projection de x_w sur ce même plan.

Grandeur physiques angulaires

Toutes les grandeurs angulaires sont exprimées en radians (rad) ou degrés (deg). L'unité utilisée sera déclarée selon le cas d'application.

Les trois angles suivants positionnent l'avion à partir du sol. La combinaison des trois angles d'Euler qui permet le passage du repère terrestre au repère de l'avion est donnée dans la figure 2.2.

- **Angle de lacet ψ , (Yaw angle)** : associé à la rotation autour de l'axe z , traversant les ailes. La vitesse de lacet (Yaw Rate) sera définie par la lettre R .
- **Angle de tangage θ , (Pitch angle)** : associé à la rotation autour de l'axe y , traversant les ailes. Il représente le nez de l'avion qui se lève ou se baisse. La vitesse de tangage (Pitch Rate) sera définie par la lettre Q .
- **Angle de roulis ϕ , (Roll angle)** : associé à la rotation autour de l'axe x , traversant l'avion du nez à la queue. Il représente une rotation des ailes pour effectuer un virage. La vitesse de tangage (Roll Rate) sera définie par la lettre P .

Les deux angles suivants positionnent la direction aérodynamique de l'avion. La combinaison des angles d'attaque et de dérapage constitue le passage du repère avion au repère vent.

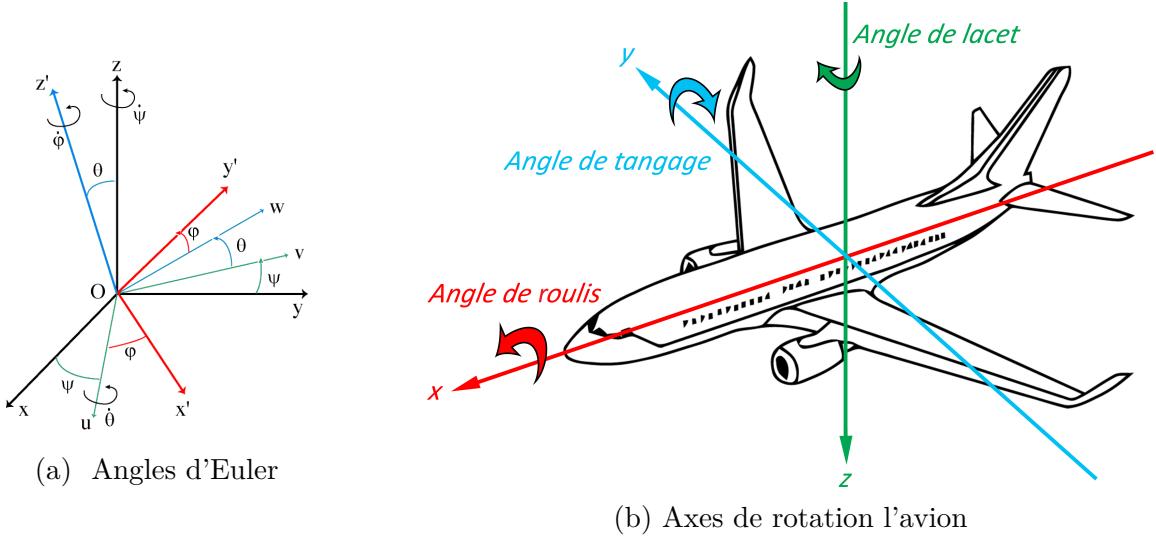


Figure 2.1 Les trois rotations de l'avion

- **Angle d'attaque α , (Angle of attack)** : C'est l'angle formé entre la corde de référence du profil de l'aile et la direction du vent relatif sur l'aile. L'angle de calage entre la corde du profil de l'aile et l'axe longitudinal de l'avion est supposé nul, par conséquent l'angle d'incidence est confondu avec l'angle d'attaque. L'angle d'attaque représente une rotation d'angle $-\alpha$ autour de l'axe y du repère avion.
- **Angle de dérapage β , (Sideslip angle)** : C'est l'angle formé entre l'axe longitudinal de l'avion et la direction du vent relatif. L'angle de dérapage représente une rotation d'angle β autour de l'axe z dans le repère stabilité.

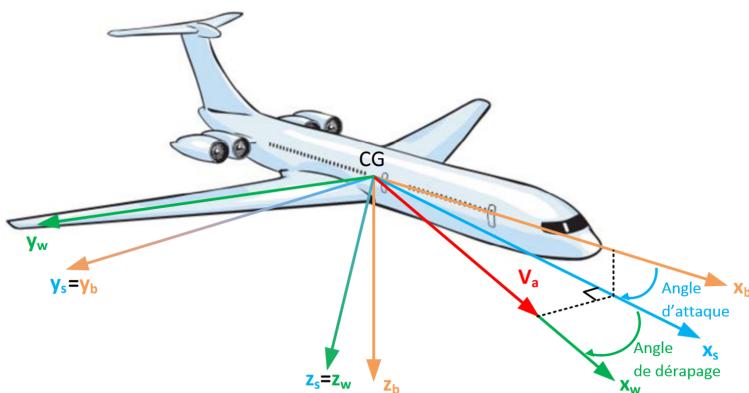


Figure 2.2 Angles de dérapage et d'attaque (source : cours ELE6208A, David Saussié)

Grandeurs physiques linéaires

Pour les grandeurs linéaires, les unités ne sont pas fixées, elles sont déclarées selon le cas d'application.

Les vitesses u, v, w représentent les vitesses des axes x, y et z dans le repère avion. Tandis que les accélérations relatives N_x, N_y, N_z , appelées facteurs de charge (*load factor*) correspondent à la force de portance relative au poids, reportée sur les axes x, y et z . Exprimées en g, elles représentent les forces ressenties par la structure de l'avion selon les directions.

2.2.3 Éléments commandables et surfaces de contrôle

Pour diriger l'avion dans l'espace, il existe des mécanismes movables commandables par le pilote : les surfaces de contrôle. Une surface de contrôle principale permet d'agir directement sur un des trois angles d'Euler présentés à la section 2.2.2. Ces trois actionneurs sont :

- les ailerons (*Aileron*) placés aux extrémités de la partie intérieure de l'aile. Ils permettent de provoquer un mouvement en roulis. Les ailerons (Fig. 2.3) sont actionnés simultanément mais en sens opposé (si l'aileron droit se lève, l'aileron gauche se baisse, et vice versa).
- les élévateurs (ou gouvernes de profondeur) (*Elevator*) placés sur l'empennage horizontal de l'avion. Ils permettent de provoquer un mouvement en tangage, et de modifier l'angle d'attaque (Fig. 2.3).
- la gouverne de direction (*Rudder*) placée sur l'empennage vertical de la queue de l'avion. Elle ne permet pas réellement de créer un angle de lacet. Celui-ci est en réalité combiné avec un angle de roulis pour amorcer un virage. La gouverne de direction permet surtout d'équilibrer l'avion lors de manœuvres assymétriques : les virages, un vol de croisière avec de forts vents, ou une panne d'un des moteurs par exemple.

Afin d'agir sur la vitesse de l'avion, le pilote peut bien évidemment agir aussi sur la poussée (*Thrust*) des moteurs (ou réacteurs).

Les surfaces de contrôle secondaires permettent quant à elles de faciliter certaines manœuvres particulières. Les plus connues sont :

- le train d'atterrissage (*Landing gear*) rétractable. Il est sorti lors des phases d'atterrissage et permet de se déplacer sur les pistes pour les phases de roulage. Il est ensuite rentré une fois le décollage terminé pour réduire la traînée de l'avion.
- les volets d'atterrissage (*Flap*), situés à l'arrière de l'aile. Ils sont déployés pour l'atterrissage et le décollage à des angles fixes. Ils permettent d'augmenter la portance de l'avion lors de ces manœuvres. Ils sont représentés en orange sur l'aile droite de la

figure 2.3.

- les becs de bord d'attaque (*Slat*), situés à l'avant de l'aile. Ils augmentent la surface de l'aile et retardent le décrochage lors de forts angles d'attaque (décollage et atterrissage).
- les destructeurs de portance (*Spoiler*) situés sur le dessus de l'aile. Ils consistent en des panneaux aérofreins qui sont levés une fois l'avion atterri pour ralentir efficacement la vitesse. Ils peuvent être utilisés aussi lors de fortes turbulences.

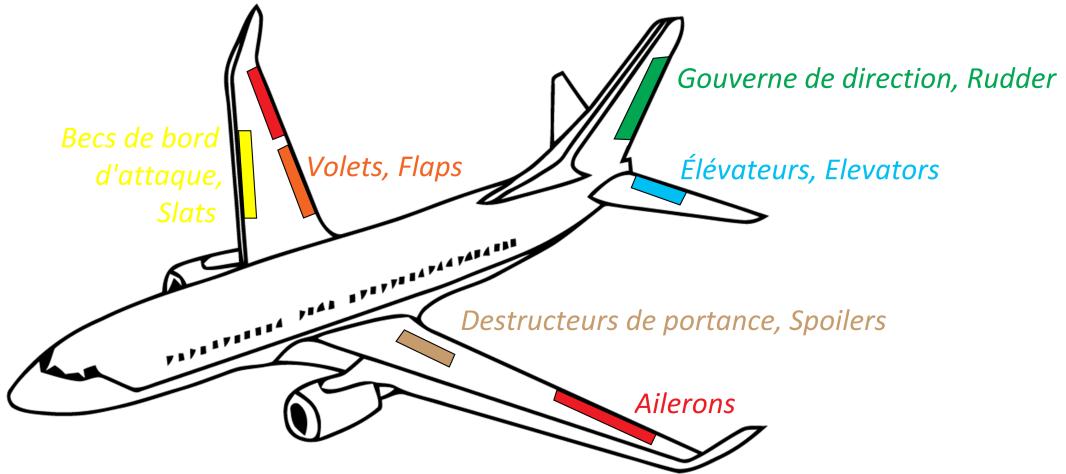


Figure 2.3 Surfaces de contrôle de l'avion

2.3 Présentation du domaine de vol

On spécifie ici l'état de l'avion dans son environnement, plusieurs configurations sont définies.

2.3.1 Configuration de croisière et d'atterrissage

On définit tout d'abord ce que l'on appelle la configuration de l'avion (aussi appelée *Flaps Configuration*). Elle désigne un état particulier des *flaps*, des *slats*, des *spoilers* et du train d'atterrissage. Les états des flaps, slats et spoilers sont désignés par des angles tandis que le train d'atterrissage peut être simplement rentré ou sorti.

Dans la suite, deux configurations de *flaps* sont étudiées. On s'y réfère comme :

- la configuration de croisière : qui représente l'avion en vol de croisière. Le train d'atterrissage, et les *flaps/slats/spoilers* sont tous rentrés.
- la configuration d'atterrissage : qui représente l'avion lors d'une manœuvre d'atterrissage. Les *flaps/slats* sont sortis à leur angle maximum. Le train d'atterrissage est déployé. Les *spoilers* sont toujours rentrés.

2.3.2 Configuration de la charge (*Loading Configuration*)

Le second état à expliciter est la configuration de charge (*Loading Configuration*). Cette configuration désigne la masse totale de l'avion, la position de son centre de gravité ainsi que la matrice d'inertie de rotation.

Les configurations de charge sont numérotées selon la position du centre de gravité sur l'axe x et selon la masse de l'avion. L'enveloppe de masse peut ainsi être représentée sur un graphique où les configurations de charge sont définies. La figure 2.4 donne un exemple de ces graphiques. Il est obtenu du site de la Federation Aviation Agency (FAA), plus particulièrement du chapitre 9 du livre *Weight & Balance Handbook (FAA-H-8083-1B)* (Fed, 2016). La figure 9-46 du livre a été modifiée pour y ajouter des exemples de configurations de charge.

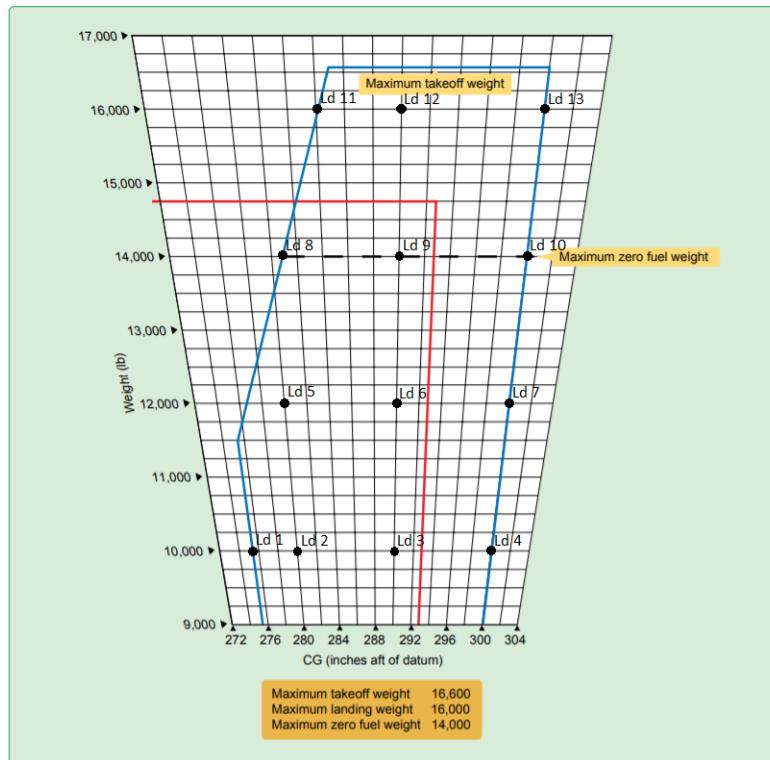


Figure 2.4 Exemple de la FAA d'un domaine de masse et de ses configurations de charge (tiré de Fed (2016))

L'enveloppe de masse élargie comporte toutes les configurations extrêmes de l'avion. La présente étude a pour but de concevoir les lois de commande sur un fonctionnement nominal ; il n'est donc pas nécessaire d'inclure ces cas. L'enveloppe normale sera donc utilisée pour la conception et testée.

2.3.3 Condition de vol (*Flight condition*)

Le dernier état à expliciter est la condition de vol (*Flight Condition*). Cette condition de vol désigne l'altitude (Altitude (ALT)), ainsi que la pression dynamique, et la vitesse de l'avion, avec la vitesse de l'air (Calibrated AirSpeed (CAS)) ou la vitesse en unité Mach. Les conditions de vol sont numérotées selon l'altitude et la vitesse CAS. L'ensemble des conditions de vol est très grand, cependant on n'utilise pas la totalité des modèles disponibles pour ne pas alourdir inutilement les calculs.

Par définition, il existe trois types de conditions de vol. Certaines ne seront pas admissibles pour la synthèse H_∞ :

- les conditions de vol à vitesses fixes : la vitesse CAS de la condition est fixe (par exemple 100 kts). Ces cas peuvent être utilisées pour le design.
- les conditions de vol à vitesses dépendant de la charge : la vitesse CAS est fonction d'une vitesse caractéristique comme V_{sr} (par exemple $1.234 \times V_{sr}$, V_{sr} qui correspond à la vitesse de décrochage de référence). Comme ces cas ont des vitesses variables dans le domaine de vol et comme le séquencement des gains est fonction de la vitesse CAS (Req 2 - section 3.3.1), il n'est pas judicieux de considérer ces points dans l'étude.
- les conditions de vol à vitesses dépendantes des configurations : la vitesse CAS est fonction d'une vitesse caractéristique qui ne varie pas sur une configuration donnée comme V_{fe} , V_{fe} la vitesse maximale permise lorsque les volets sont sortis dans leurs positions maximales. Comme les vitesses sont constantes pour le domaine de charge, ces points sont considérés pour le réglage des gains.

La conception de la loi de commande se fait sur l'enveloppe de vol classique utilisée par BA. Les altitudes de design sont comprises entre un maximum et un minimum. Les vitesses CAS associées sont donc fixes et comprises aussi entre un minimum et un maximum.

Supposons n_a altitudes, n_v vitesses et n_l charges, on obtient $n_a \times n_v \times n_l$ points de design. Cependant une partie d'entre eux sont difficilement réalisables, hors de l'enveloppe à certifier :

- le vol en haute altitude ne peut être réalisé à de trop faibles vitesses ou trop hautes vitesses. La vitesse minimale est limitée par la traînée. Comme la pression est moins grande en altitude, l'avion doit maintenir une vitesse plus élevée pour garantir une portance suffisante. La vitesse maximale, quant à elle, est limitée par la vitesse du son. Le vol supersonique n'étant pas permis (profil d'aile non adéquat pour les ondes de chocs), l'avion ne peut dépasser Mach 1. Comme la vitesse de Mach dépend de la pression de l'air, la vitesse maximale exprimée en noeuds n'est pas la même pour toutes

- les altitudes. En l'occurrence, pour une altitude de 51 000 pieds, Mach 1 correspond à 243 kts, alors qu'il correspond à une valeur de 650 kts pour une altitude de 1 000 pieds. Dans le cas de la configuration de croisière, l'intervalle d'altitudes possibles est très vaste. Par conséquent les vitesses de design ne sont pas les mêmes pour toutes les altitudes. Dans le cas de la configuration d'atterissage, les vitesses extrêmes sont identiques pour toutes les altitudes, car l'intervalle d'altitude est plutôt restreint.
- le vol à basse vitesse ne peut être réalisé pour des avions trop lourds. En effet, plus l'avion est lourd, et donc plus la vitesse de l'avion doit être élevée afin de générer la portance nécessaire pour le maintenir en vol. Pour les plus faibles vitesses du domaine de vol, certaines charges (les plus élevées) sont donc hors du domaine d'utilisation et de certification de l'avion. Ils sont tout de même considérés dans le design mais associés à une pondération plus faible pour réduire leur influence dans le réglage des gains.

Il est judicieux de ne pas forcer tout le système pour obtenir de parfaites performances pour ces modèles non certifiés.

À l'aide des trois renseignements sur la configuration, la charge et les conditions de vol, l'avion est complètement défini et situé dans son environnement. Ce sont donc trois notions essentielles qui sont spécifiées pour chaque application.

2.4 Présentation de l'architecture des lois de commandes

Dans cette section, l'architecture de la loi de commande latérale et directionnelle ainsi que ses principales composantes sont présentées.

2.4.1 Architecture de la loi normale latérale et directionnelle

La loi de commande latérale et directionnelle a pour but de commander l'angle de roulis ϕ , la vitesse angulaire de roulis P et l'angle de dérapage β . En amont, ces consignes sont données par un angle de stick et un angle de pédale, de la part du pilote. Les actionneurs engagés sont alors les ailerons et la gouverne de direction (*rudder*).

L'architecture est donnée à la figure 2.5. Les entrées du système sont donc les trois signaux de référence des commandes en P , ϕ et β . Une quatrième entrée est ajoutée : *PhiHoldIn*. Ce signal d'entrée est utilisée pour le canal de performance ϕ_{cmd} vers β . Elle permet de garantir une référence de ϕ constante lors d'une commande de β .

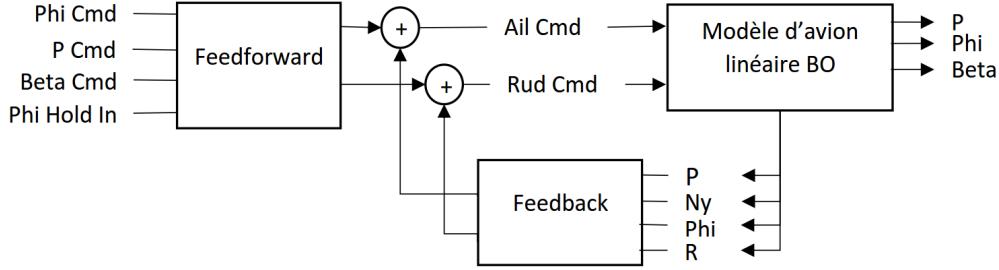


Figure 2.5 Architecture générale de la loi de commande

2.4.2 Gains de retour d'état (Feedback)

Le retour d'état K est une simple matrice de dimension 2×4 décrite par :

$$K = \begin{bmatrix} K_{P \rightarrow Ail} & K_{N_y \rightarrow Ail} & K_{\phi \rightarrow Ail} & K_{r \rightarrow Ail} \\ K_{P \rightarrow Rud} & K_{N_y \rightarrow Rud} & K_{\phi \rightarrow Rud} & K_{r \rightarrow Rud} \end{bmatrix} \quad (2.1)$$

Dans un souci de simplification, on utilisera :

$$K = \begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} \\ K_{21} & K_{22} & K_{23} & K_{24} \end{bmatrix} \quad (2.2)$$

Chaque gain de la matrice est borné par une valeur minimale et une valeur maximale (Req3 - section 3.3.1), qui limite son intervalle de valeurs possibles dans un seul signe (positif ou négatif). En effet, le changement de signe est évité à cause des comportements erratiques qui peuvent être associés à un gain très petit, presque nul.

Les gains de retour associés à N_y (donc K_{12} et K_{22}) sont plus limités que les autres. En plus de la contrainte sur le signe, leur amplitude est de 1 à 4% des contraintes standards sur les autres gains. Il est alors évident que ces restrictions orientent fortement le réglage de la loi de commande.

2.4.3 Gains de précommande (Feedforward)

Contrairement au retour d'état, la dynamique de la précommande est plus complexe qu'une simple matrice. Son architecture est présentée à la figure 2.6. Le bloc $P_{12}(s)$ est composé d'un gain statique P_{12} et d'une fonction de transfert de second ordre. Les gains $P_{21}(s)$ et $P_{21Dyn}(s)$ sont composés des gains statiques (P_{21} et P_{21Dyn}) et d'une fonction de transfert de premier ordre.

Dans un but de simplification, on considère pour la première application (chap. 5), les gains de précommande comme les gains d'une matrice de dimension 2×2 , qui a pour entrées ϕ_{cmd} et β_{cmd} (on ne considère pas P_{cmd}). La matrice de *feedforward* est donc :

$$P = \begin{bmatrix} P_{\phi \rightarrow Ail} & P_{\beta \rightarrow Ail} \\ P_{\phi \rightarrow Rud} & P_{\beta \rightarrow Rud} \end{bmatrix} \quad (2.3)$$

Dans un souci de simplification, on utilisera :

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \quad (2.4)$$

Pour cette simplification, les gains K_p et $P_{21,Dyn}$ sont hors de la précommande, ils sont donc réglés à posteriori. Lorsque la précommande complète est considérée, lors de la seconde et troisième application (chaps. 6 et 7), on inclut le réglage de ces gains dans la synthèse.

Lorsque la dynamique de précommande complète est considérée, le gain K_p est supposé constant. Il pourra être réglé si l'ajout de degrés de liberté est estimé nécessaire. Le gain $P_{21,Dyn}$ est considéré variable mais sa valeur devrait être pratiquement nulle.

Les valeurs des gains de la matrice de *feedforward* ne sont pas contraintes, cependant elles sont analysées une fois le réglage terminé et ne devraient pas être trop élevées.

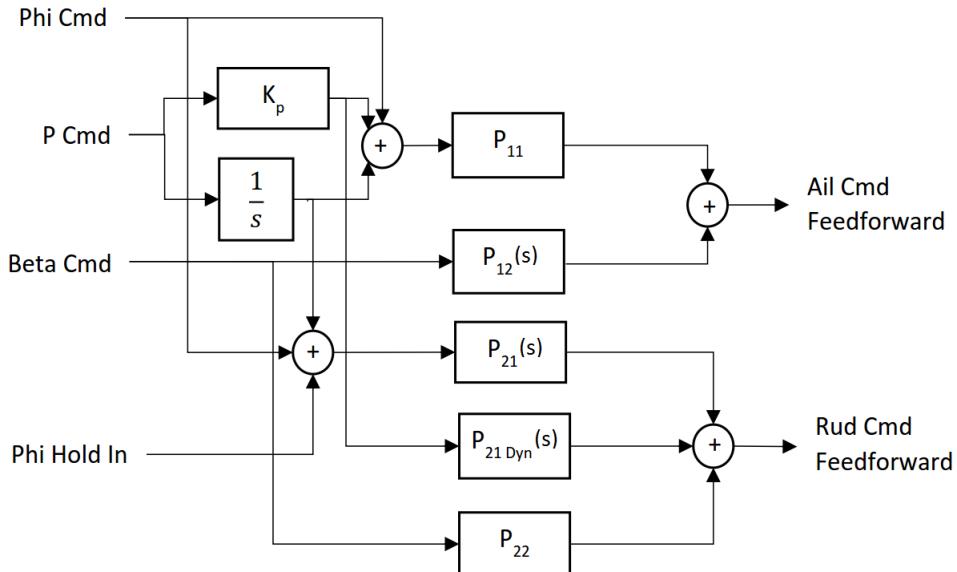


Figure 2.6 Architecture de la dynamique de précommande

2.5 Présentation des modèles d'avion rigides et flexibles

Dans cette section du chapitre, les modèles manipulés sont présentés. Trois types de graphiques seront utilisés :

- les diagrammes de Bode : ils représentent l'allure fréquentielle d'un système. L'amplitude en dB (échelle logarithmique) et la phase en degrés sont données en fonction de la fréquence en radians par seconde (échelle logarithmique.)
- les diagrammes de Nichols : ils représentent l'allure fréquentielle aussi, et permettent une lecture plus facile des marges de stabilités. L'amplitude en dB est donnée en fonction de la phase en degrés.
- les cartes des pôles : ils représentent la position des pôles dans le plan complexe, où un pôle est fonction de sa partie réelle (axe des abscisses) et de sa partie imaginaire (axe des ordonnées).

2.5.1 Origine des modèles

Tous les modèles manipulés dans cette étude proviennent du modèle non linéaire de l'avion conçus par BA. Il modélise tout l'avion et ses composants, des moteurs à l'aérodynamique en passant par l'interaction du pilote. On spécifie les conditions et les configurations désirées dans un outil logiciel, qui calcule la position d'équilibre correspondante et linéarise le système non linéaire autour de cet équilibre. On obtient alors un modèle linéarisé appelé par la suite modèle d'avion rigide qui ne comprend que la dynamique de l'avion provenant des équations cinématiques et dynamiques.

Afin d'obtenir le modèle linéaire de l'avion flexible, un outil logiciel a été spécialement conçu. Il permet de synthétiser les fonctions de transfert représentatives des modes flexibles à partir d'un modèle obtenu par le logiciel NASTRAN (outil de modélisation par éléments finis) qui comporte l'ensemble des modes non entièrement rigides. Il existe des dynamiques strictement rigides et strictement flexibles, mais il existe aussi des dynamiques provoquées par l'interaction des deux ensembles. Le concept du modèle flexible (matrice d'état A) peut être représenté par :

$$A = \begin{bmatrix} A_{rigide \rightarrow rigide} & A_{flexible \rightarrow rigide} \\ A_{rigide \rightarrow flexible} & A_{flexible \rightarrow flexible} \end{bmatrix} \quad (2.5)$$

où $A_{rigide \rightarrow rigide}$ est la dynamique rigide, et $A_{flexible \rightarrow rigide}$, $A_{rigide \rightarrow flexible}$, $A_{flexible \rightarrow flexible}$ est la dynamique flexible. Toute la dynamique flexible contenue dans le modèle de NASTRAN est donc approximée par des fonctions de transfert.

Un deuxième outil parallèle à l'outil initial (cas rigide) est développé. Lors de la linéarisation du modèle rigide, ce deuxième outil cherche si, pour le point d'équilibre considéré, une telle fonction de transfert existe. Si oui, la fonction de transfert est ajoutée au modèle rigide. Le modèle flexible comporte donc la dynamique rigide, flexible et les interactions entre les deux.

2.5.2 Modèles de l'avion rigide et de l'avion flexible

Comme précisé plus haut, les modèles de l'avion rigide et de l'avion flexible, ne comportent que la dynamique de l'avion ; par conséquent les dynamiques des actionneurs et des capteurs n'y apparaissent pas.

On considère dans nos modèles 22 signaux² d'entrée (Tab. 2.1) et 11 signaux de sortie (Tab. 2.2).

Tableau 2.1 Signaux d'entrées du modèle d'avion

Rigide	Positions des surfaces de commande latérales
	Vitesse vent latéral
Flexible	Vitesses des surfaces de commande latérales
	Accélérations des surfaces de commande latérales

Tableau 2.2 Signaux de sortie du modèle d'avion

Rigide et flexible	Vitesse angulaire de roulis P
	Vitesse angulaire de lacet R
	Angle de roulis ϕ
	Angle de lacet ψ
	Angle de dérapage β
	Accélération latérale N_y
	Accélération angulaire de roulis \dot{P}
	Accélération angulaire de lacet \dot{R}
	Angle de rudder Rud
	Position verticale (Altitude) h
	Vitesse de l'air indiquée V_{air}

2. On rappelle que l'ensemble des surfaces de commandes possibles d'un avion sont définies à la section 2.2.3.

La figure 2.7 présente les allures fréquentielles des modèles rigide et flexible d'un même point de vol. Seuls les transferts entrée/sortie d'intérêts sont affichés ci-après. On remarque que le modèle flexible suit la dynamique fréquentielle du modèle rigide (parfois le pic de réponse est même amoindri par la présence de modes souples) mais s'ajoute une série de pics d'excitations à des fréquences plus élevées que pour le modèle rigide. La dynamique flexible est visible sur une plage de fréquence entre 10F et 1000F.

Les allures fréquentielles (Fig. 2.7) et cartes des pôles (Figs. 2.9, 2.8, 2.10b et 2.10a) sont tracés pour le point de fonctionnement suivant :

- Configuration : croisière,
- Masse : 1.125MFW,
- Position du CG : 1.4FWDCG,
- Altitude : 0.686ALTMAX,
- Vitesse : 0.796VMAX.

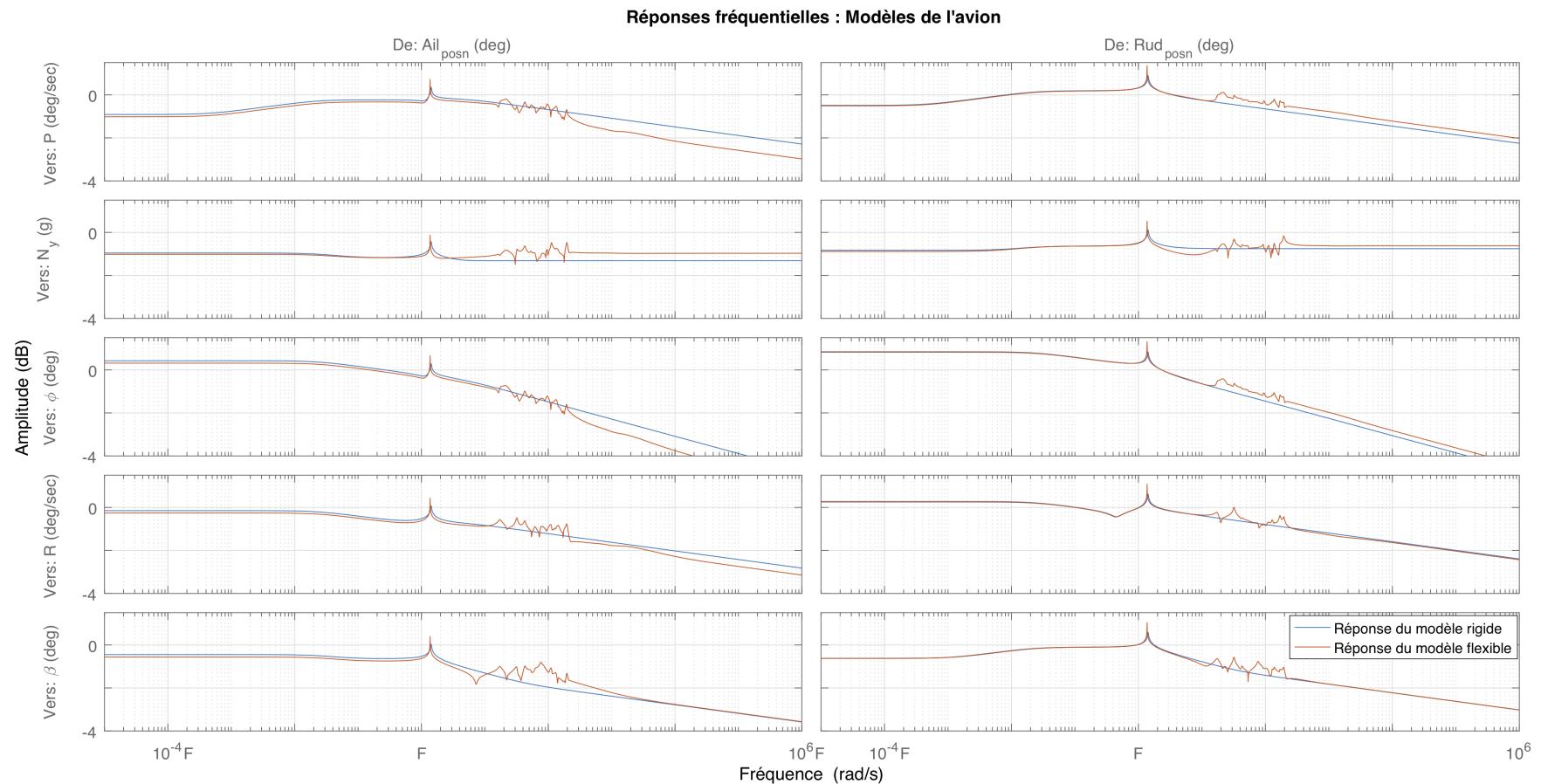


Figure 2.7 Allures fréquentielles des modèles d'avion rigide et flexible

Le modèle de l'avion rigide comporte peu d'états mais tous ont un sens physique car ils représentent une variable physique de l'avion. La même remarque ne pourra pas être faite pour les états du modèle d'avion flexible. Les 8 états du modèle d'avion rigide sont donnés par le tableau 2.3.

Tableau 2.3 États du modèle d'avion

Rigide	Angle de roulis ϕ
	Vitesse angulaire de roulis P
	Vitesse latérale V
	Vitesse angulaire de lacet R
	Angle de lacet ψ
	Position latérale Y
	Position verticale Z
	Altitude de pression h

Lorsque l'on compare la figure 2.9 avec la figure 2.8 qui nous donne la position des pôles de modèles flexible et rigide, on voit que les états flexibles sont bien plus nombreux et bien plus loin de l'origine. Dépendamment du nombre de modes flexibles à prendre en compte, l'ordre du modèle peut être très grand. Les études suivantes sont appliquées à 50 modes flexibles, et le modèle d'avion flexible possède déjà 668 états.

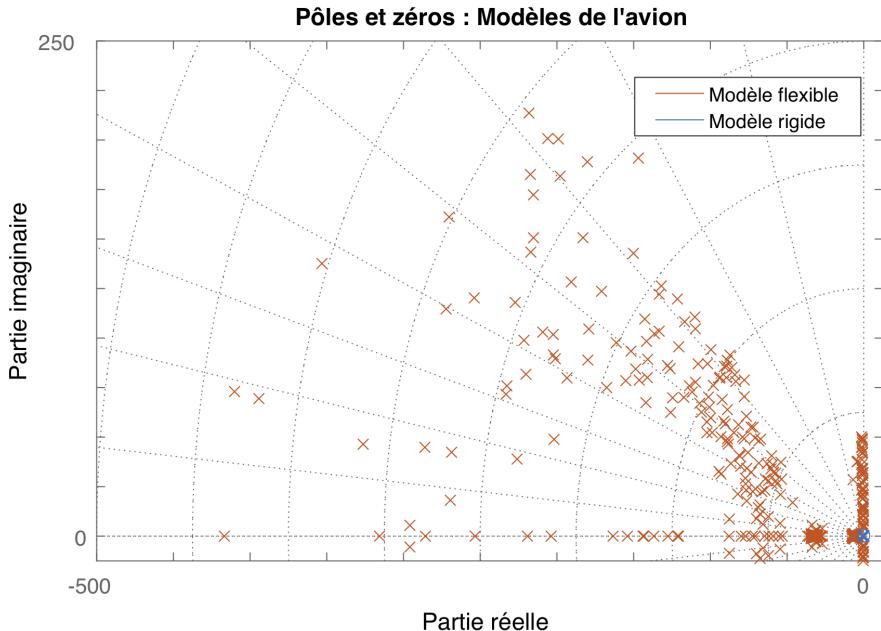


Figure 2.8 Positionnement de l'ensemble des pôles de modèles d'avion rigide et flexible

Dans le cas du modèle rigide, quatre pôles sont nuls (dont le mode spiral), un pôle est un nombre réel (mode roulis pur) et les derniers sont une paire de complexes conjugués (mode roulis hollandais). Si on diminue l'échelle de la figure 2.8, la comparaison rigide/flexible est

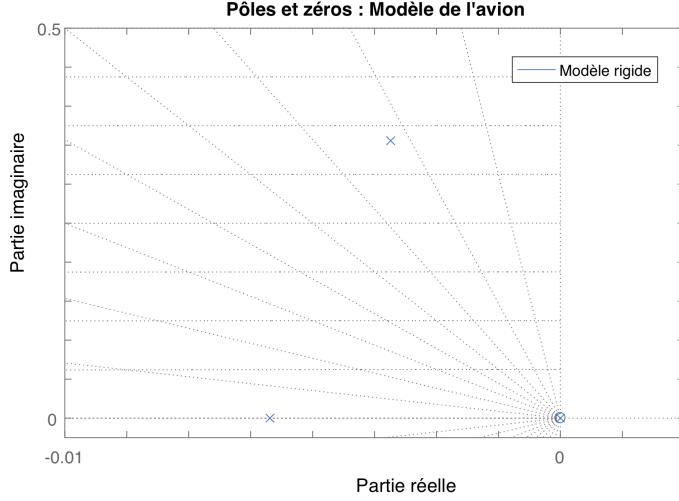


Figure 2.9 Positionnement des pôles d'un modèle d'avion rigide

faite par les figures 2.10a (Zoom $\times 250$) et 2.10b (Zoom $\times 50000$). Dans ces graphiques, on voit premièrement que les pôles flexibles ajoutés (trois dans le coin haut droit du plan complexe) sont mal amortis. On remarque deuxièmement que la paire de pôles complexes est fortement décalée sur la droite, fortement déstabilisée. La dynamique souple vient déstabiliser les modes dominants de l'avion rigide.

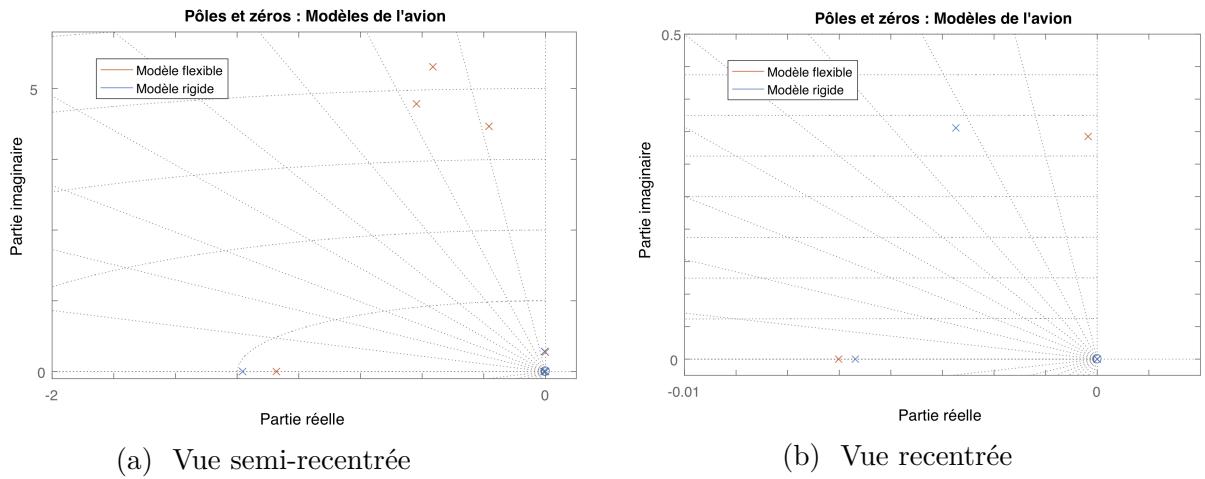


Figure 2.10 Positionnement des pôles dominants des modèles rigide et flexible de l'avion

2.5.3 Modèles BO des modèles rigide et flexible

Les modèles de l'avion BO rigide et flexible, comportent la dynamique de l'avion, celle des actionneurs et des capteurs ainsi que la dynamique des filtres ASE pour réduire l'influence des modes flexibles (Fig. 2.11).

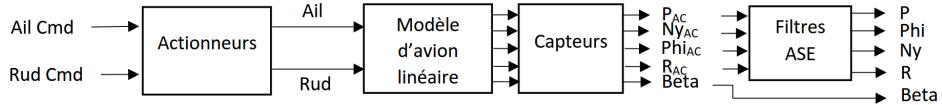


Figure 2.11 Schéma de l'architecture de la BO

Tableau 2.4 Signaux d'entrées du modèle d'avion BO

Rigide et flexible	Braquage d'aileron commandé
	Braquage de rudder commandé

Les deux entrées des modèles sont donnés par le tableau 2.4. La sortie est composée de neuf signaux (dont seulement cinq sont utilisés pour le réglage classique), ces sorties sont présentées dans le tableau 2.5.

Tableau 2.5 Signaux de sorties du modèle d'avion BO

Rigide	Vitesse angulaire de roulis P
	Accélération latérale N_y
	Angle de roulis ϕ
	Vitesse angulaire de lacet R
	Angle de dérapage β
Flexible	Vitesse angulaire de roulis P_{AC} pre-ASE
	Accélération latérale $N_{y,AC}$ pre-ASE
	Angle de roulis ϕ_{AC} pre-ASE
	Vitesse angulaire de lacet R_{AC} pre-ASE

Les allures fréquentielles et cartes des pôles sont tracés pour le point de fonctionnement suivant :

- Configuration : croisière,
- Masse : 0.914MTOW,
- Position du CG : 1.083FWDCG,
- Altitude : ALTMAX,
- Vitesse : 1.827VMIN.

Les figures 2.12 et 2.13 nous montrent la position des pôles du modèle BO rigide dans le plan complexe. On compte plus de pôles que pour le modèle de l'avion. En effet, on compte 121 états. On retrouve les huit états du modèle de l'avion, ainsi que les états des filtres ASE, et des actionneurs. Dans la vue rapprochée, on voit que la paire de pôles complexes est à partie réelle positive. Le système BO possède donc un mode roulis hollandais instable. L'intérêt de fermer la boucle est de stabiliser ces pôles.

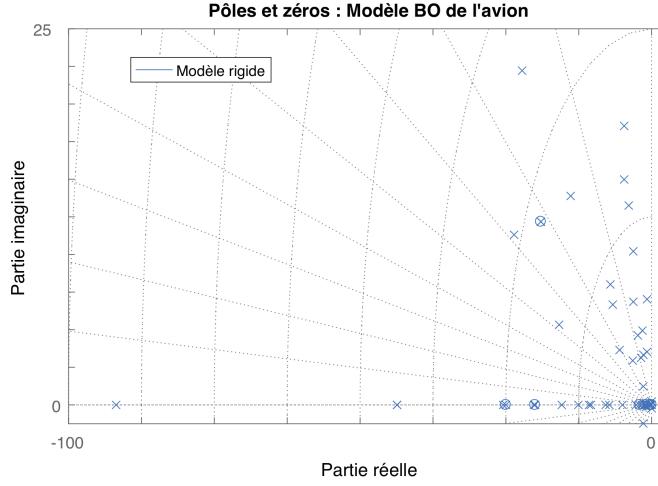


Figure 2.12 Positionnement de l'ensemble des pôles d'un modèle BO d'avion rigide

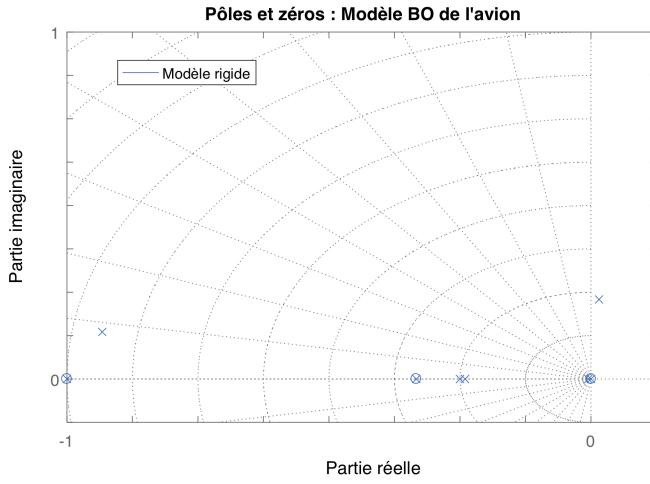


Figure 2.13 Positionnement des pôles dominants d'un modèle BO d'avion rigide

Les allures fréquentielles des modèles rigide et flexible sont comparés dans la figure 2.14. Les mêmes remarques que précédemment peuvent être faites sur l'influence des modes souples.

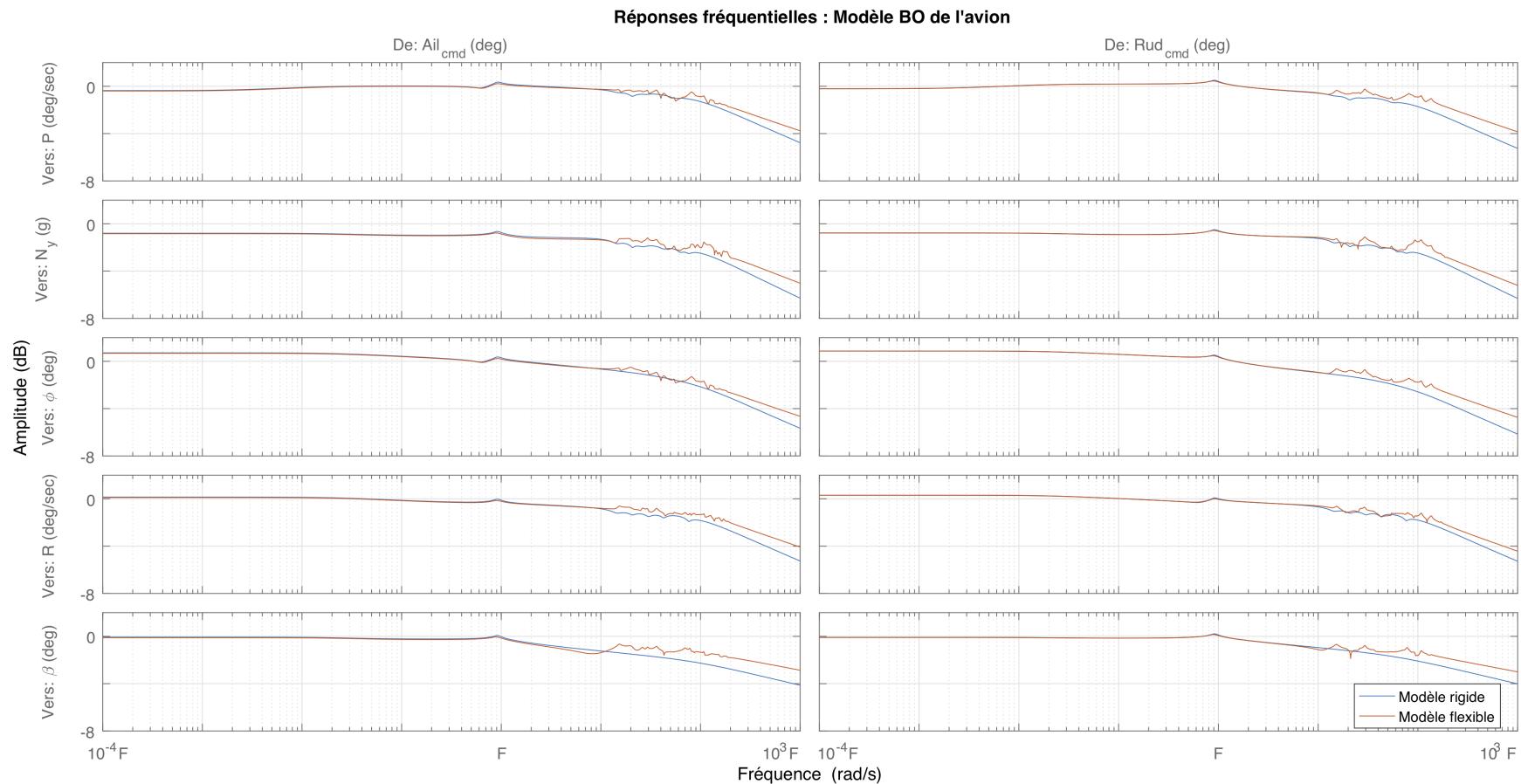


Figure 2.14 Allures fréquentielles des modèles BO d'avion rigide et flexible

Si le nombre d'états a augmenté pour le modèle rigide, il augmente aussi pour le modèle flexible. Le modèle BO flexible manipulé compte 781 états (Fig. 2.15).

Les figures 2.16a et 2.16b présentent aussi la position des pôles dans le plan complexe. Les

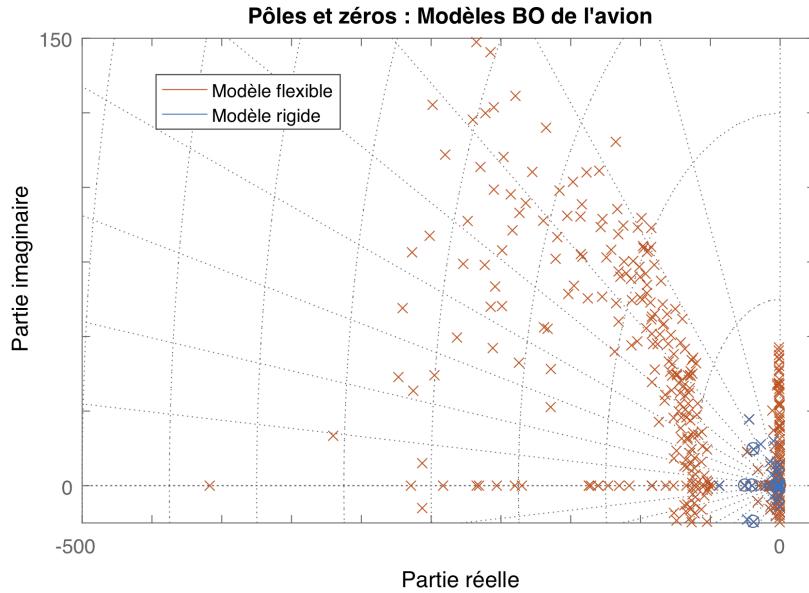


Figure 2.15 Positionnement des pôles de modèles BO rigide et flexible

deux vues proposées permettent de constater que les modes flexibles mal amortis dans le coin haut droit de la première figure, ainsi que la déstabilisation apportée par l'ajout des modes souples (pôles décalés vers la droite dans la seconde figure).

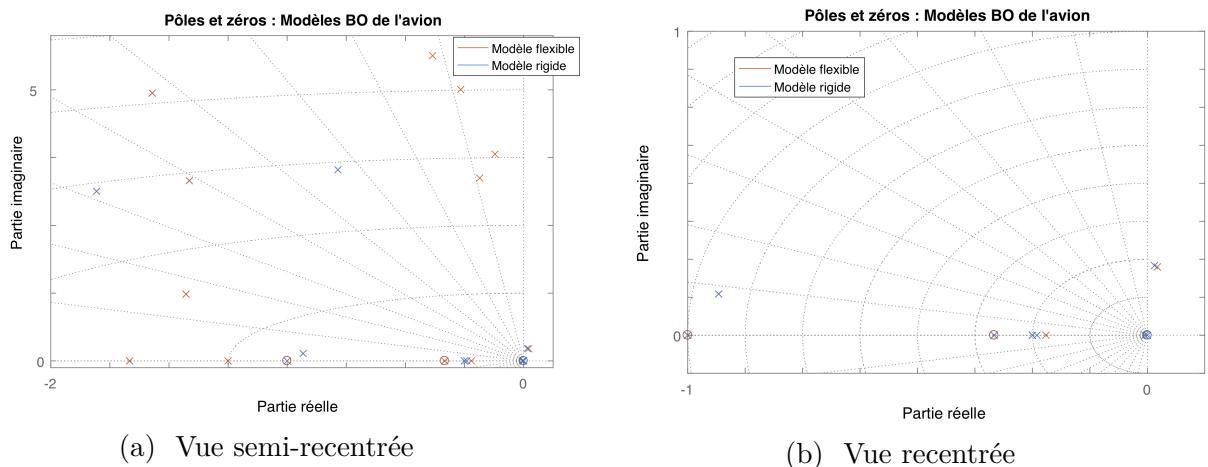


Figure 2.16 Positionnement des pôles principaux de modèles BO d'avion rigide et flexible

Les modèles BO sont ceux manipulés lors du réglage des gains. L'ordre des modèles énoncés plus haut est très élevé et ralentit l'algorithme d'optimisation. Il est nécessaire de réduire l'ordre des modèles.

Pour les modèles rigides, la commande de réduction usuelle `minreal` qui cherche la représentation minimale du système en argument est adoptée. Dans le cas des modèles flexibles, cette fonction ne peut être utilisée car les pôles du roulis hollandais auront tendance à être déplacés à nouveau vers la droite sur le plan complexe. La réduction alors proposée n'est donc pas satisfaisante. La solution proposée est la combinaison de deux fonctions `prescale` et `balred` qui expriment le système dans un repère à l'échelle optimal et appliquent une réduction équilibrée du système à un ordre spécifié. Les deux choix de l'étude sont 55 et 115 états. La figure 2.17 illustre les modèles rigide et flexible ainsi que leurs modèles réduits respectifs. Les courbes bleues claire et foncée sont presque confondues. La commande `minreal` approxime très bien le modèle rigide. La courbe jaune suit bien la courbe rouge pour les plus petites fréquences tandis que la courbe bordeaux la suit mieux sur des fréquences plus élevées et en ce qui concerne le transfert de N_y . Pour les modèles flexibles, la réduction à un ordre 115 est meilleure. On commence par réaliser des synthèses réduisant les modèles à 55 états, puis on utilise les modèles d'ordre 115 pour mieux optimiser la solution.

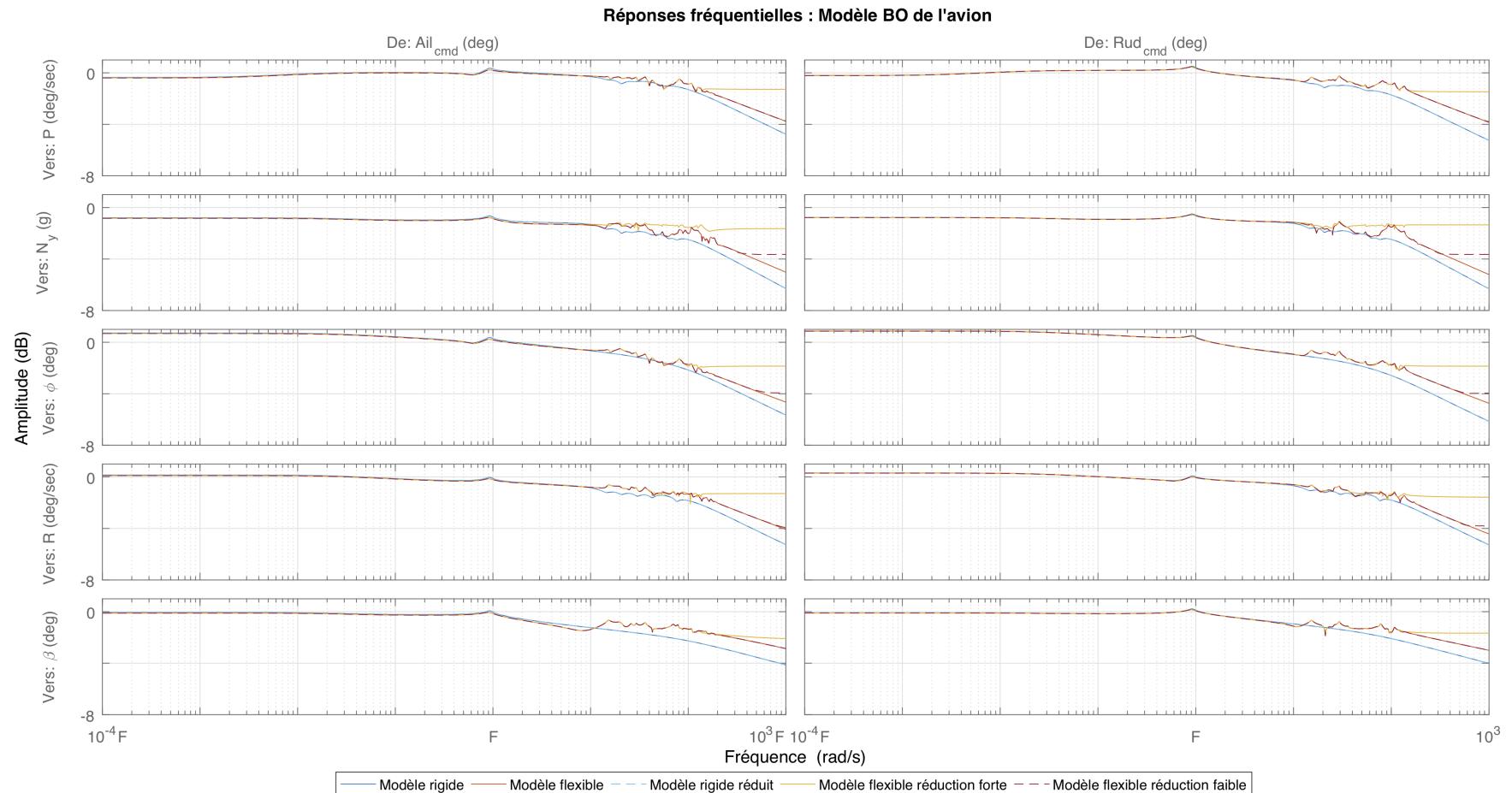


Figure 2.17 Allures fréquentielles de modèles complet et réduit BO d'avion rigide et flexible

2.5.4 Modèles en BO pour les marges de stabilité

Les modèles BO des actionneurs sont nécessaires lors de la vérification des marges de stabilités. Par définition, les marges de stabilité sont celles obtenues au niveau de l'aileron et du rudder et elles sont construites telles que présentées dans la figure 2.18.

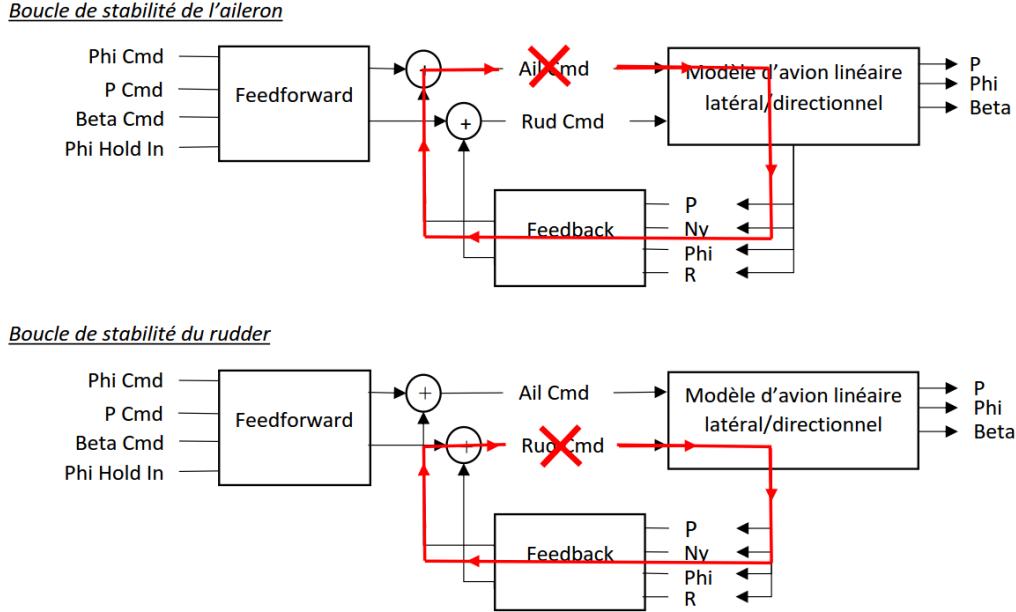


Figure 2.18 Schéma de conception des modèles BO de stabilité d'un modèle d'avion

Les deux entrées et les deux sorties du système sont données par les tableaux 2.6 et 2.7. Les angles “sommés” signifiant les angles obtenus à la sortie des sommateurs de la précommande et du retour d'état.

Tableau 2.6 Signaux d'entrées du modèle d'avion BO de stabilité

Rigide et flexible	Angle d'aileron commandé
	Angle de rudder commandé

Tableau 2.7 Signaux de sorties du modèle d'avion BO de stabilité

Rigide et flexible	Angle d'aileron sommé
	Angle de rudder sommé

On obtient ainsi deux modèles de stabilité : le premier est le transfert de l'entrée Ail_{cmd} (en entrée du modèle BO de l'avion) vers la sortie Ail_{cmd} (sortie du sommateur de l'aileron). La

boucle de commande de l'aile est ouverte. Le deuxième est le transfert de l'entrée Rud_{cmd} (en entrée du modèle BO de l'avion) vers la sortie Rud_{cmd} (sortie du sommateur du *rudder*). La boucle de commande du *rudder* est ouverte.

Les marges de stabilité désirées sont de M_G dB et M_ϕ degrés. Les valeurs numériques ne peuvent être divulguées, la notation ci-dessus est donc employée.

Le diagramme de Bode et de Nichols suivants (Figs. 2.19 et 2.20) présentent l'allure fréquentielle des modèles BO de stabilité rigide. On remarque qu'avec la présence de modes flexibles, l'amplitude maximale dans la bande de fréquence de 10F à 1000F est plus grande. Ainsi elle a plus de risque de dépasser la limite de stabilité imposée. Dans la figure ci-dessous, les marges sont respectées. Le point de fonctionnement choisi pour représenter cette section est tel que :

- Configuration : croisière,
- Masse : MFW,
- Position du CG : 0.798AFTCG,
- Altitude : 0.784ALTMAX,
- Vitesse : 2.205VMIN.

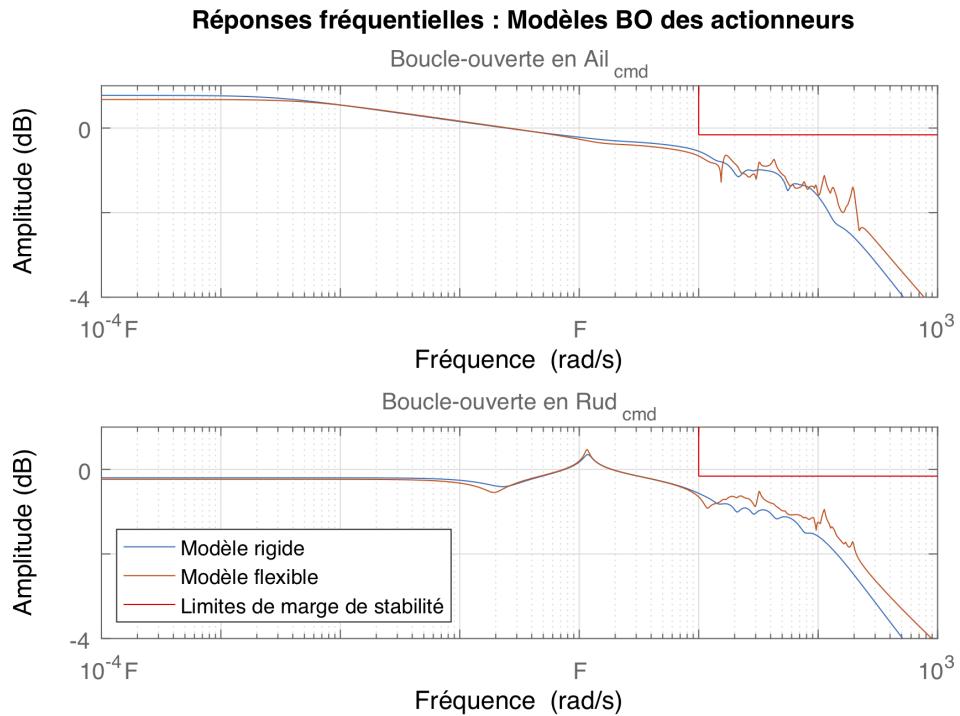


Figure 2.19 Allures fréquentielles des BO de stabilité d'un modèle d'avion rigide

La figure 2.20 montre l'allure fréquentielle de ces BO représentées dans des diagrammes de Nichols. Les marges de stabilités (en phase et en gain) imposées forment des ellipses limites tracées. On peut voir que les courbes respectent aussi ces marges, même si le transfert du rudder semble plus optimisé car plus proche de l'ellipse.

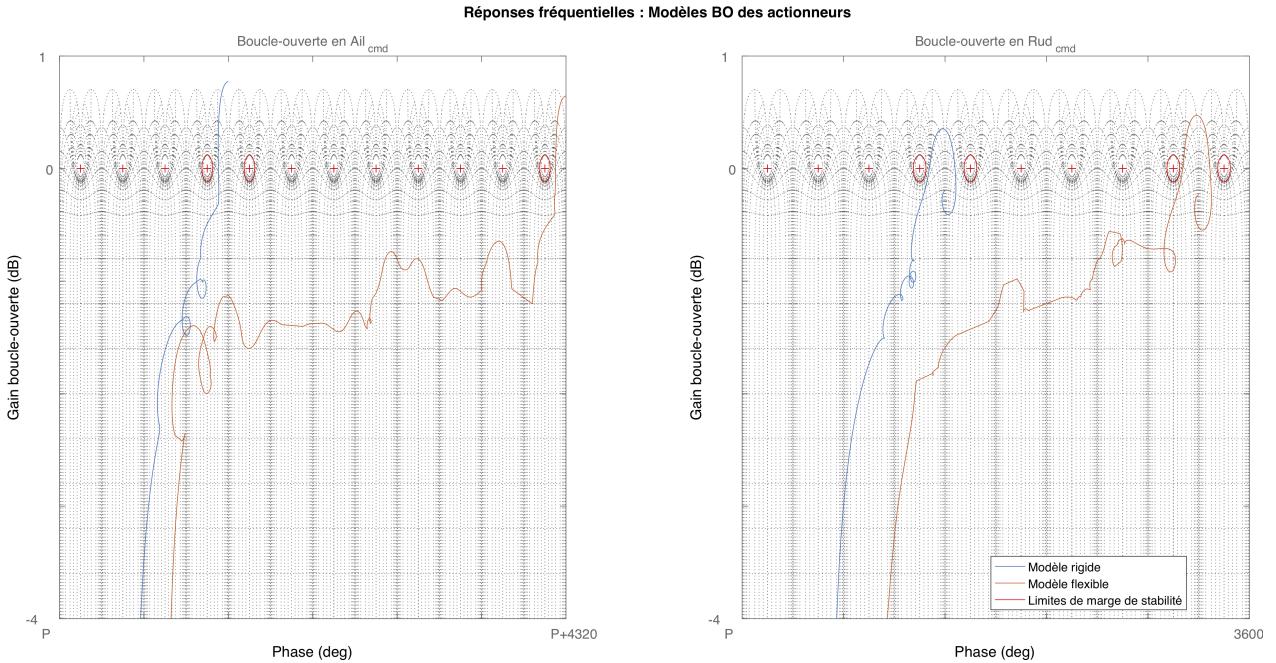


Figure 2.20 Allure fréquentielle et marges de stabilité d'un modèle BO rigide d'avion

2.5.5 Modèles en BF de l'avion rigide et flexible

Les modèles de l'avion BF rigide et flexible, sont obtenus à partir des modèles BO. Le schéma de conception des modèles BF est donnée plus haut dans le chapitre, à la section 2.4.1. Globalement, il consiste simplement à fermer la boucle avec la matrice de feedback et à ajouter la dynamique de précommande en amont.

Tableau 2.8 Signaux d'entrées du modèle d'avion BF

Rigide et flexible	Angle de roulis commandé ϕ_{cmd}
	Vitesse angulaire de roulis commandé P_{cmd}
	Angle de roulis de maintien ϕ_{HoldIn}
	Angle de dérapage commandé β_{cmd}

Les 4 signaux d'entrée des modèles sont donnés dans le tableau 2.8. Comme mentionné plus haut, l'entrée ϕ_{HoldIn} est utilisé pour le réglage, il n'est pas jugé utile de présenter les allures

fréquentielles des transferts qui lui sont liées. Les 5 signaux de sortie sont donnés par le tableau 2.9. Les allures fréquentielles et les cartes des pôles sont tracés pour le point de

Tableau 2.9 Signaux de sorties du modèle d'avion BF

Rigide et flexible	Vitesse angulaire de roulis P
	Accélération latérale N_y
	Angle de roulis ϕ
	Vitesse angulaire de lacet R
	Angle de dérapage β

fonctionnement suivant :

- Configuration : croisière,
- Masse : MTOW,
- Position du CG : 1.224FWDCG,
- Altitude : 0.686ALTMAX,
- Vitesse : 0.817VMAX.

Le fait de fermer la boucle par un retour d'état statique n'augmente pas l'ordre du système ; il permet de positionner correctement les pôles dominants de la BF, comme le roulis hollandais. La dynamique de précommande en revanche apporte cinq nouveaux états. Par conséquent, les modèles BF ont 126 états pour les modèles rigides (comme montré sur la figure 2.21a) et respectivement 786 états pour les modèles flexibles (Fig. 2.21b).

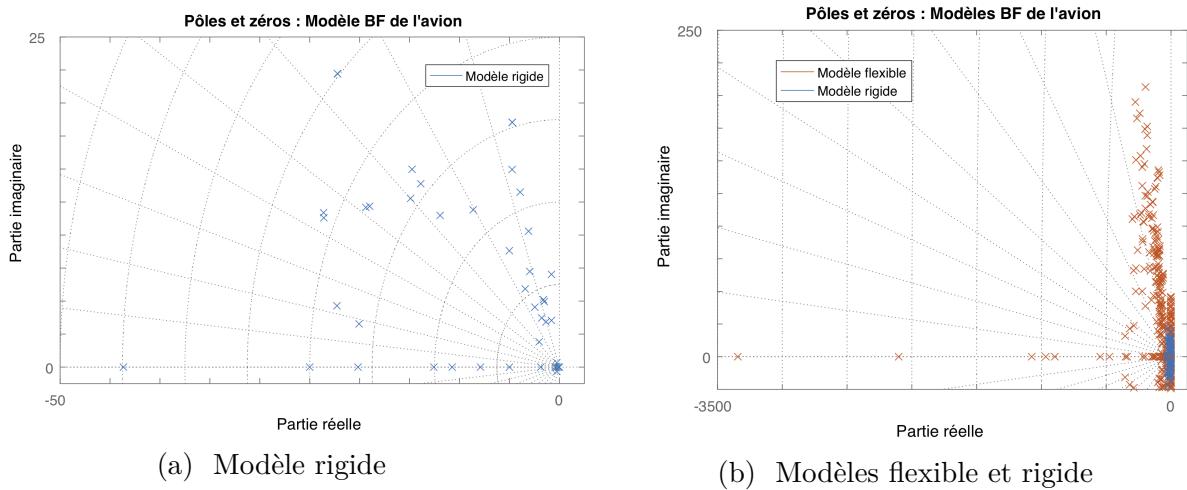


Figure 2.21 Positionnement de l'ensemble des pôles de modèles BF d'avion

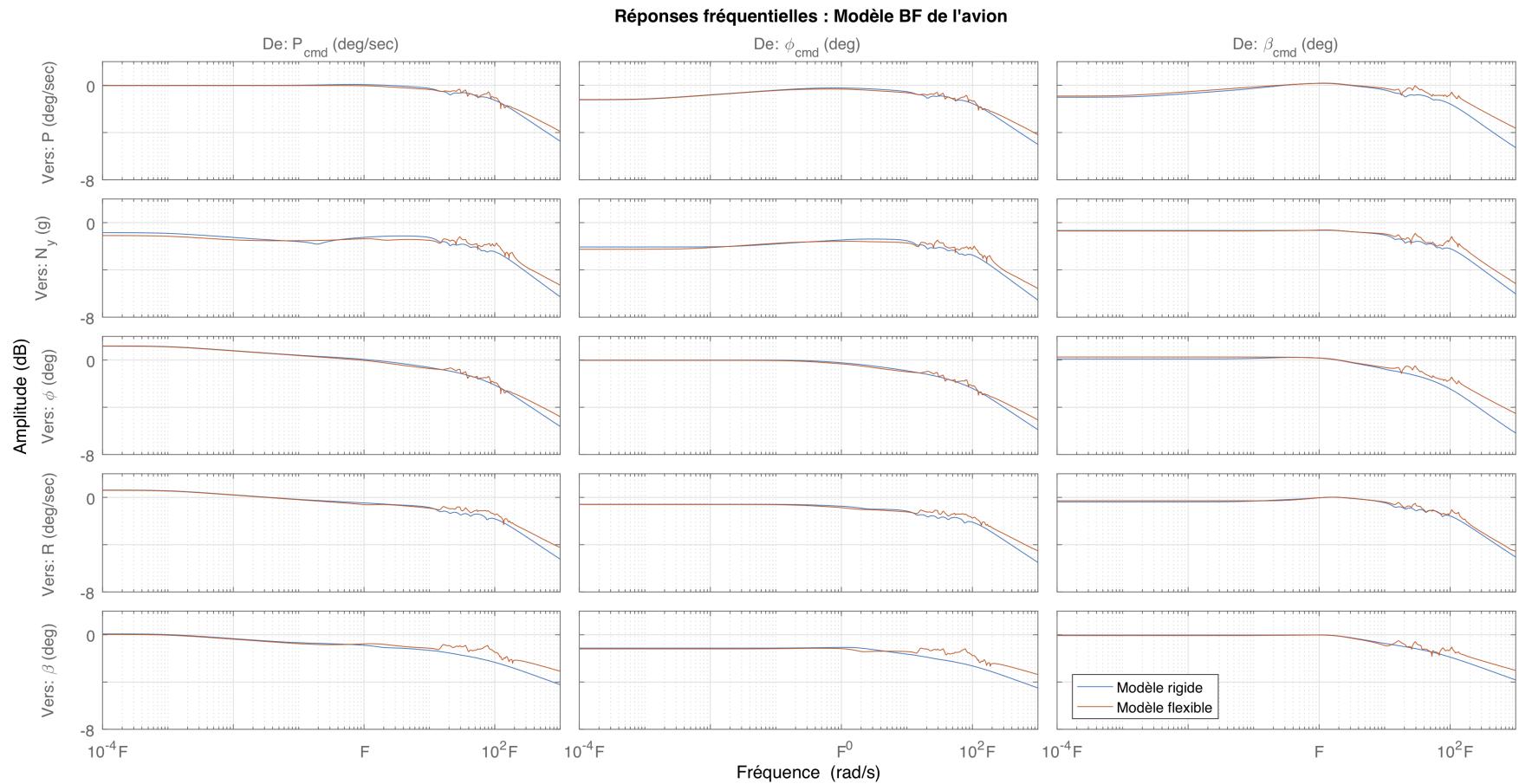


Figure 2.22 Allures fréquentielles de modèles BF d'avion rigide et flexible

Les modèles flexibles et rigides en BF (allures fréquentielles et positionnement des pôles) sont maintenant comparés à travers les figures 2.22 à 2.23b.

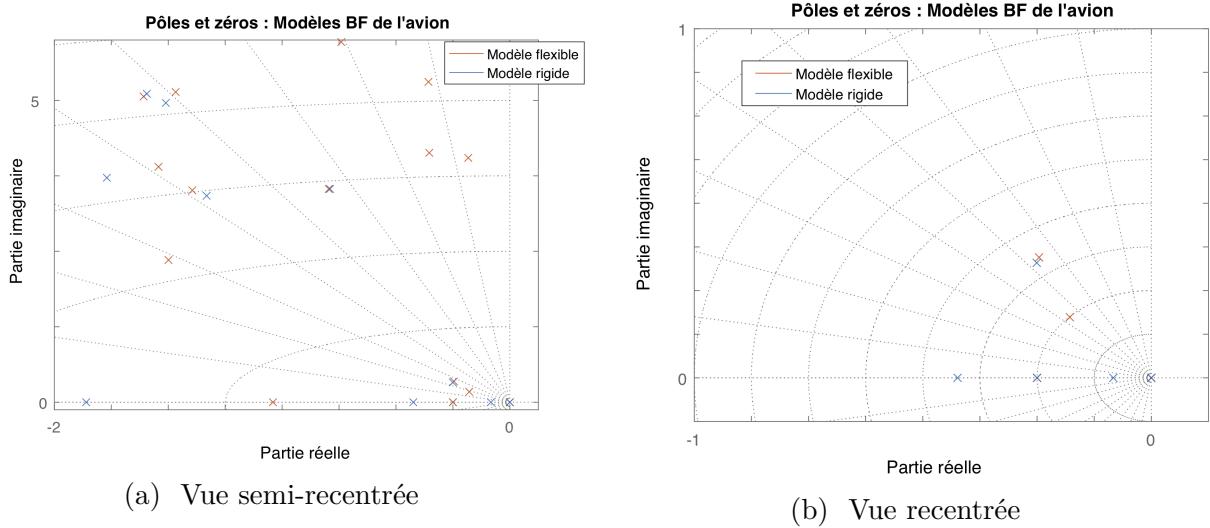


Figure 2.23 Positionnement des pôles principaux de modèles d'avion BF rigide et flexible

On remarque avec la figure 2.24 que les objectifs concernant le placement de pôles définissent des zones. Ces zones forment un patron d'objectif de placement à plusieurs couleurs : vert pour l'objectif initial, violet et jaune pour objectif suffisant, bleu pour objectif à améliorer et rouge pour toutes les zones à éviter. On peut alors noter qu'en BF les pôles sont plutôt bien placés, sauf pour le roulis pur qui est à côté de son objectif.

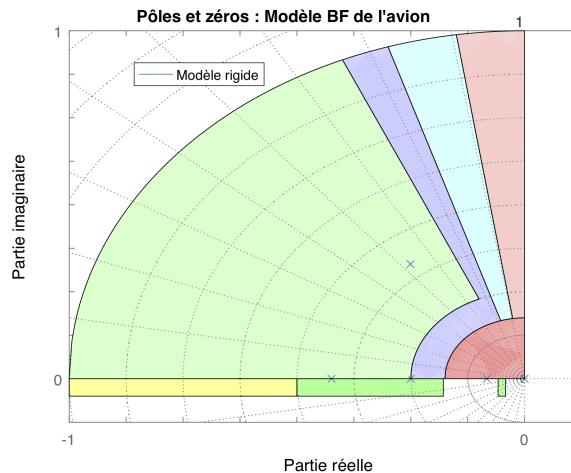


Figure 2.24 Positionnement des pôles de modèles d'avion BF rigide dans le patron objectif

2.6 Présentation des filtres AéroServoÉlastique

Pour finir, les filtres ASE sont détaillés dans cette section du chapitre.

2.6.1 Objectif des filtres ASE

Une fois le réglage des gains effectués pour obtenir une BF performante, les résultats expérimentaux ont démontré une dégradation de ces performances notamment à cause de modes souples. Les filtres ASE alors ont été mis en place pour venir contrer les modes flexibles les plus oscillants en rejetant ces bandes de fréquences.

2.6.2 Composition des filtres et éléments réglables

Les filtres ASE sont appliqués sur les quatres principaux canaux de la loi de commande, c'est-à-dire sur P , N_y , ϕ et R . Ces filtres sont principalement composés de filtres coupe-bande (pour limiter les fréquences des modes souples problématiques), mais aussi de filtres passe-bas de second ordre et de filtres anti-bruit (filtre passe-bas et passe-bande).

On rappelle que la définition des filtres énoncés ci-dessus sont :

- le filtre coupe-bande (*Notch Filter*) :

$$F_{notch} = \frac{s^2 + \omega_0^2}{s^2 + 2\omega_c s + \omega_0^2} \quad (2.6)$$

où ω_0 est la fréquence centrale de réjection et ω_c est la largeur de bande de réjection.

- le filtre passe-bas (*Low-Pass Filter*) :

$$F_{low} = \frac{K}{s^2 + 2\omega_c s + \omega_0^2} \quad (2.7)$$

- le filtre anti-bruit (*Noise Filter*) :

$$F_{noise} = \frac{\omega_1 s + 1}{\omega_2 s^2 + \omega_0 s + 1} \quad (2.8)$$

Les signaux P , N_y , ϕ et R ont chacun leur canal de filtres (filtres passe-bas et coupe-bande) et le filtre anti-bruit est appliqué sur tous. La figure 2.25 présente les filtres ASE appliqués avant le filtre anti-bruit. On voit que les bandes de coupures ciblent des séries de fréquences spécifiques. L'allure ressemble finalement à un inverse des allures des modèles flexibles sur la même bande de 10F à 1000F. La combinaison des filtres aux perturbations flexibles devraient supprimer les amplitudes des pics des diagrammes de Bode. On note donc l'importance de la précision du placement de ces fréquences de coupure.

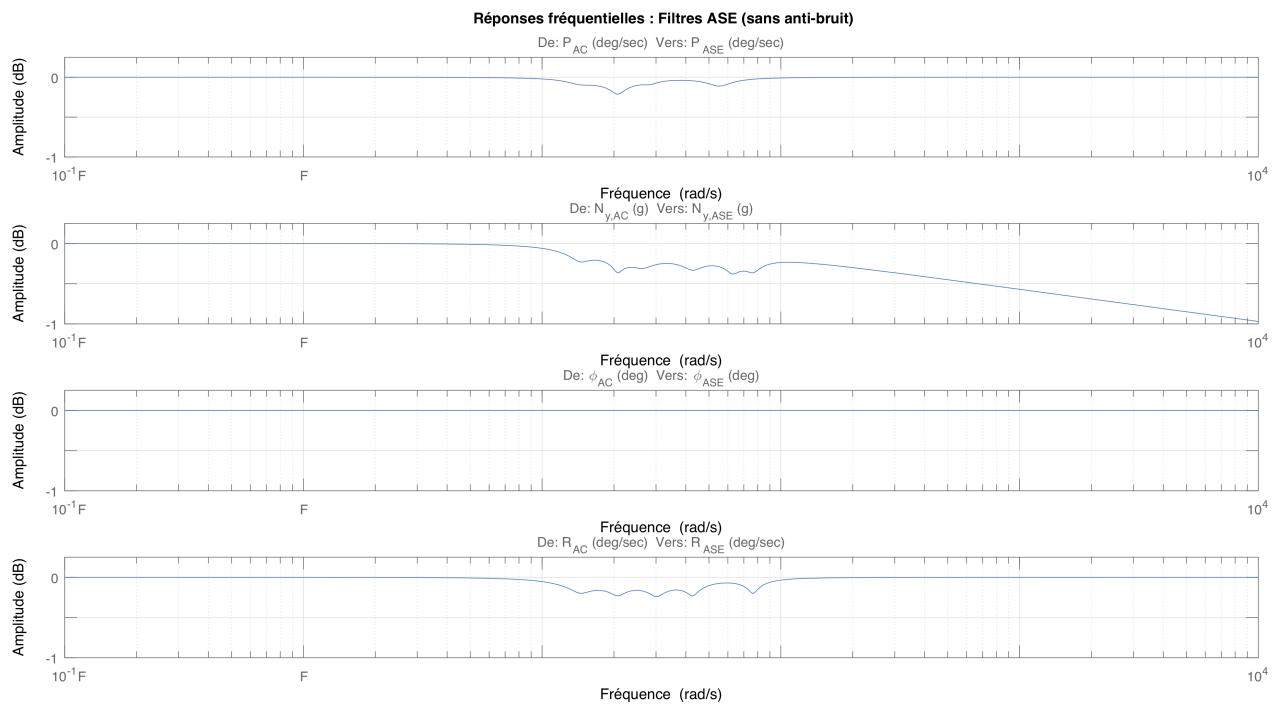


Figure 2.25 Allures fréquentielles des filtres ASE sans le filtre anti-bruit

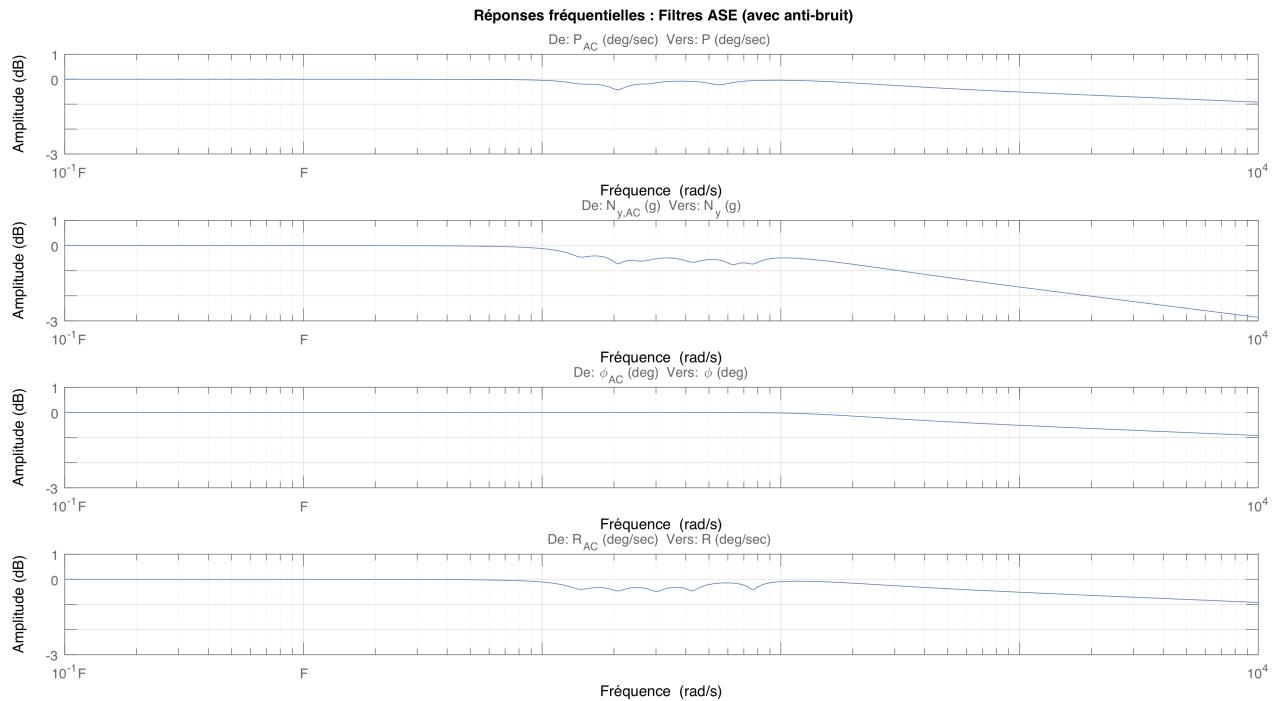


Figure 2.26 Allures fréquentielles des filtres ASE avec le filtre Noise

La figure 2.26 ajoute le filtre anti-bruit et montre l'allure fréquentielle des filtres ASE appliqués à chacun des canaux de retours. On remarquera que seul le filtre anti-bruit est appliqué au signal ϕ . Le filtre anti-bruit est finalement un simple passe-bas qui permet de couper toutes les perturbations à hautes fréquences.

2.6.3 Analyse de l'impact sur les modes flexibles

Le diagramme de Bode ci-après (Fig. 2.27) illustre l'intérêt des filtres ASE. Même si les pics d'amplitude flexibles ne sont pas totalement supprimés, les amplitudes sont largement diminuées sur l'ensemble des fréquences des modes souples. Obtenir une fonction inverse pour contrer ces perturbations, qui soit de plus de taille raisonnable n'est pas réalisable. Il convient donc de trouver des filtres qui ciblent suffisamment bien les modes flexibles, dans le but de réduire leur influence au mieux.

Le point de fonctionnement utilisé pour cette section est le suivant :

- Configuration : croisière,
- Masse : 0.853MTOW,
- Position du CG : 1.167FWDCG,
- Altitude : ALTMAX,
- Vitesse : 1.951VMIN.

Les filtres ASE ont effectivement une action positive sur la réponse du système pour gérer les flexibilités. Il conviendrait de se demander si le réglage de ces filtres peut être fait lors du réglage de la loi de commande et non à posteriori. Cette éventualité sera abordée lors du chapitre 7.

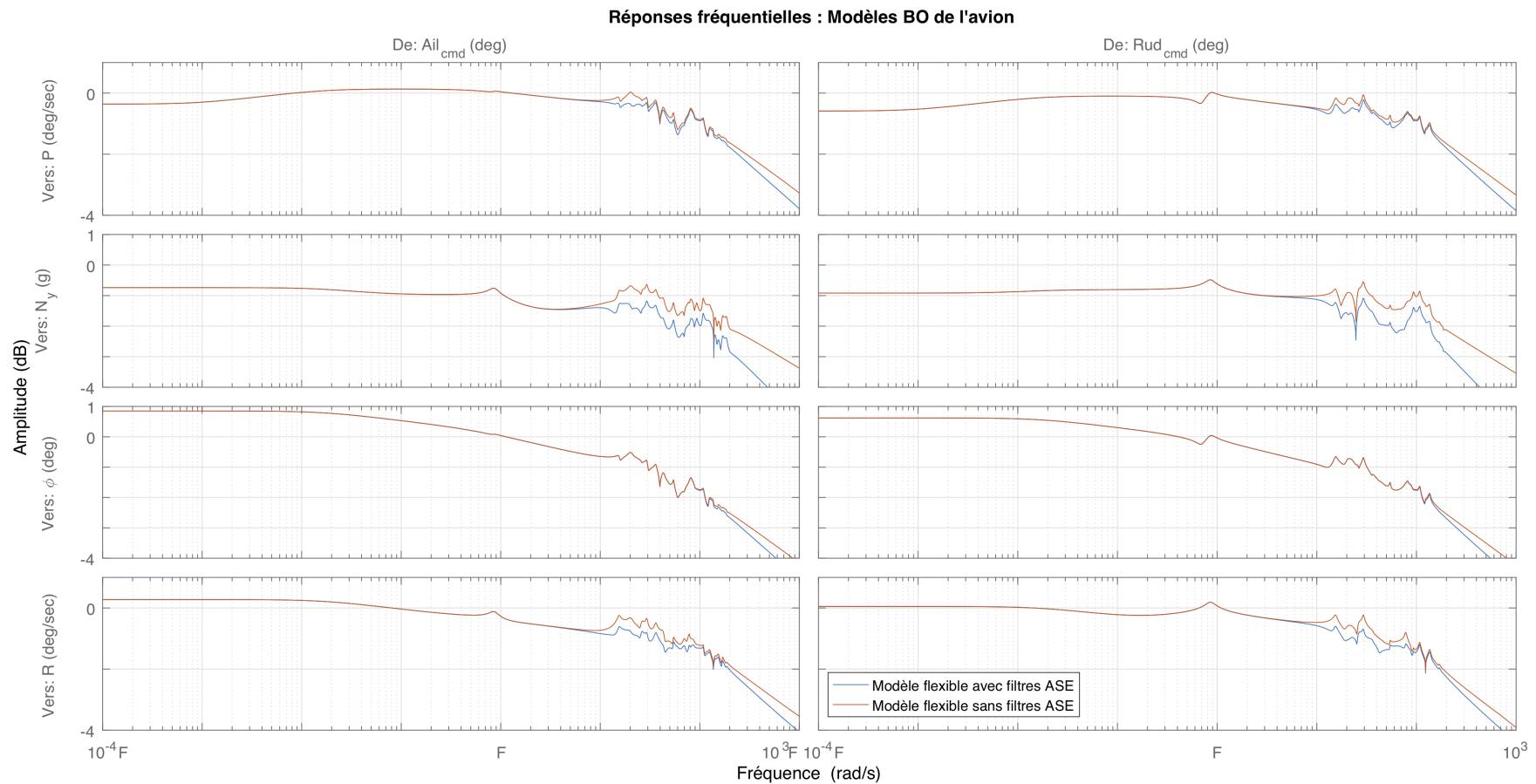


Figure 2.27 Allures fréquentielles d'un modèle d'avion flexible BO avec ou sans les filtres ASE

2.7 Conclusion

Pour conclure rapidement, l'ensemble des notions nécessaires à la compréhension des chapitres suivants ont été développés dans ce chapitre. L'environnement de travail, les variables et modèles manipulés ont aussi été détaillés. De plus, toute la loi de commande, son architecture et ses éléments réglables ont été présentés.

Tous les aspects de ce chapitre permettent par la suite de pouvoir définir les requis et les concepts qu'applique la synthèse H_∞ structurée aux modèles tout en s'inscrivant parfaitement dans l'environnement de BA.

CHAPITRE 3 DÉVELOPEMENT DES CONCEPTS ET DU DESIGN

Ce chapitre présente l'ensemble des concepts utilisés pour résoudre la problématique de l'étude. La section 3.1 introduit le chapitre et divise son contenu. La section 3.2 sert à rappeler l'essentiel de la théorie liée à la méthode de synthèse H_∞ . Les requis imposés par l'industrie ainsi que les concepts utilisés par la synthèse pour adresser les requis sont listés dans la section 3.3. En section 3.4, on ajoute quelques techniques pour optimiser le design. Finalement la section 3.5 conclut le chapitre.

3.1 Introduction

3.1.1 Contexte et activités précédentes

Au cours du premier chapitre, l'histoire de la synthèse H_∞ a été résumée. On y apprend que cette méthode de réglage de loi de commande robuste se montre tout à fait efficace et performante. Cependant certaines entreprises excluent la synthèse classique à cause des limitations qu'elle présente, particulièrement la lourdeur de la réduction et la restructuration du contrôleur. L'arrivée de la synthèse H_∞ structurée résoud les problèmes soulevés, car elle permet notamment de fixer au préalable l'architecture de la loi de commande. De précédents travaux (Lhachemi et al., 2014; Saussié et al., 2013) démontrent que la méthode est applicable à l'industrie. On montre qu'elle l'est aussi dans le contexte de BA.

Finalement dans le chapitre 2, l'environnement a été présenté au complet : des conditions de vol à l'architecture des boucles de commande de l'avion. Maintenant que le système est bien connu, on peut définir les concepts qui serviront à l'asservir, afin d'adresser la totalité des requis imposés.

3.1.2 Objectifs

Les objectifs de cette partie sont donc de lister l'ensemble des requis, et de présenter les concepts théoriques. Leur application au système est donnée sous la forme de schémas augmentés.

Le corps du chapitre est divisé de la manière suivante :

- premièrement, dans la section 3.2, un récapitulatif de la théorie de la synthèse H_∞ structurée, de ses outils et avantages,

- deuxièmement, dans la section 3.3, le développement de la traduction des requis en concepts,
- finalement, dans la section 3.4, quelques guides pour optimiser le réglage des canaux de performances.

3.1.3 Méthodologie suivie

La méthodologie suivie a été d'abord de rassembler la documentation de BA et de lister l'ensemble des requis qui sont appliqués au système. Ensuite, les requis ont été regroupés selon la nature des contraintes. Les groupes sont traités les uns après les autres.

Pour la traduction des requis en concepts, la plupart des solutions sont trouvées dans la documentation Matlab *TuningGoal*¹.

3.2 La synthèse H_∞ structurée

Comme on l'a vu au chapitre 1 et dans la figure 1.3, la synthèse H_∞ structurée possède les avantages suivants :

- la structure du contrôleur est imposée,
- plusieurs contraintes de différentes natures sont acceptées,
- plusieurs contraintes peuvent être imposées,
- seules les canaux de performances spécifiés sont optimisés,
- plusieurs modèles peuvent être considérés dans une seule synthèse.

On développe ici l'ensemble des concepts utiles à la compréhension² de la théorie de la méthode de synthèse et qui n'ont pas été présentés dans la section 1.1. L'outil choisi pour résoudre le problème d'optimisation est aussi détaillé.

3.2.1 Synthèse structurée

La synthèse H_∞ structurée permet de fixer l'architecture du contrôleur en définissant des gains réglables et en optimisant leur valeur. Si on reprend l'exemple de la figure 3.1, le transfert de w vers z et sa norme $\| \cdot \|_\infty$ dépend du choix de $K(s)$. Or le contrôleur $K(s)$ lui-même est exprimé en fonction de gains réglables \mathcal{K} , ainsi $K(s, \mathcal{K})$, et donc :

$$\| T_{w \rightarrow z}(P, K) \|_\infty = \| T_{w \rightarrow z}(P, K(s, \mathcal{K})) \|_\infty \quad (3.1)$$

1. Documentation consultée le 23/11/2018 à l'adresse url suivante :<https://www.mathworks.com/help/slcontrol/tuning-goals.html>.

2. On pourra se référer à l'annexe A et B pour des définitions mathématiques et de commande supplémentaires.

Pour mieux illustrer la théorie, prenons l'exemple classique du PID :

$$K(s, \mathcal{K}) = r_k + \frac{r_i}{s} + \frac{r_d}{\tau s + 1} \quad (3.2)$$

et donc :

$$\|T_{w \rightarrow z}(P, K)\|_\infty = \|T_{w \rightarrow z}(P, r_k, r_i, r_d, \tau)\|_\infty \quad (3.3)$$

En utilisant cette définition, il est alors possible de contraindre la structure du contrôleur à un PID ou à tout autre structure désirée sans se limiter aux contrôleurs classiques.

3.2.2 Synthèse multi-objectifs

Les performances à obtenir sont déclarées sous la forme de deux fonctions, une fonction de contrainte et une fonction de coût. La fonction de contrainte doit impérativement être inférieure à 1, c'est l'ensemble des performances que l'on souhaite absolument. La fonction coût rassemble le reste des performances souhaitables ; l'optimisation consiste alors à minimiser sa valeur. La résolution d'un tel problème utilise une approche de type descente de gradient afin de trouver le minimum local d'une fonction.

L'augmentation de cette méthode à la synthèse mixte est aussi possible. Néanmoins les fonctions coûts et contraintes associées ne sont alors plus forcément dérivables et continues à tout ordre n (\mathcal{L}^n). Dès lors, le gradient peut ne pas exister en certains points, et les algorithmes classiques ne peuvent déterminer de solutions. L'outil numérique actuel de la méthode de synthèse utilise la dérivée différentielle de Clarke (1987). Cette dérivée existe toujours car elle ne presuppose pas l'existence d'une limite ; cependant elle peut avoir une valeur infinie. Des algorithmes avancés d'optimisation sont finalement nécessaires. De plus, puisque le problème posé est NP-complexe³, non-convexe (Gahinet et Apkarian, 2011a) , la recherche du minimum est limitée à la recherche d'un minimum local (sans certitude que ce minimum local soit le minimum global).

3.2.3 Outils MATLAB® pour la synthèse H_∞ structurée

Un des premiers outils implémentés qui permet de régler des lois de commandes tout en respectant les critères sur la structure du contrôleur, la multiplicité des canaux de performances et leur priorité dans le réglage face aux autres transferts non contraints, est la librairie H-

3. NP signifie "Non-deterministic Polynomial". C'est une classe qui exprime la complexité du problème. Un problème de résolution est dit NP-complexe si l'on peut vérifier « rapidement » (complexité d'ordre polynomiale) que la solution proposée est bien solution.

Infinity Fixed Order Optimization (HIFOO)⁴. Plusieurs applications de cet outil ont montré son efficacité dans le contexte de synthèse à multi-objectifs (Gumussoy et Overton, 2008; Gumussoy et al., 2009). Plus récemment de nouveaux outils ont été développés dans les dernières versions de la toolbox (boîte à outils) de Matlab (Gahinet et Apkarian, 2011b), Control System Toolbox et aussi Robust Control Toolbox (Balas et al., 2014). Ces derniers outils rassemblent un ensemble de fonctions plus complètes, numériquement plus efficaces et rapides, et soumises à des mises à jour bi-annuelles pour faire améliorer l'outil. Pour manipuler des modèles flexibles (ordre élevé), on a choisi de travailler avec cette toolbox, sous Matlab R2016b. Cette version de Matlab apporte des modifications importantes par rapport aux versions précédentes dans la rapidité des calculs. Des problèmes simples qui pouvaient prendre plusieurs heures de compilation avec Matlab 2014 ne nécessitent que quelques minutes avec la version de 2016.

Afin de résoudre le problème d'optimisation, on utilisera la fonction **systune** de la Control System Toolbox. La fonction cherche à trouver le vecteur x^* qui satisfait localement la propriété suivante :

$$x^* \in \arg \min_{g(x) \leq 1} f(x) \quad (3.4)$$

où x est composé de l'ensemble des paramètres réglables de la loi de commande, la fonction f est la fonction coût (définie dans la section 3.2.2) qui rassemble les critères de performances à optimiser, la fonction g est la fonction de contrainte (définie dans la section 3.2.2) qui rassemble les critères de performances qui doivent strictement être assurés, et $\arg \min$ représente l'ensemble des valeurs x_{min} qui respectent la condition $g(x_{min}) \leq 1$ et $f(x_{min})$ est minimum.

Dans l'environnement de Matlab, f et g sont appelées *SoftGoal* et *HardGoal*. Ces dénominations traduisent plus facilement la souplesse affectée à la contrainte. On rappelle que cette technique accepte les critères mixtes. Dans ce cas, on peut considérer que N_s *soft goals* et N_h *hard goals* sont appliqués à la loi de commande. Alors :

$$\gamma_f = \max_{i \in [1, N_s]} f_i$$

$$\text{et de même, } \gamma_g = \max_{j \in [1, N_h]} g_j$$

γ_f et γ_g sont finalement les indicateurs de succès de la synthèse. Si $\gamma < 1$ alors l'ensemble des contraintes (soft ou hard) sont assurées. La synthèse cherche d'abord à assurer que les

4. Documentation consultée le 23/11/2018 à l'adresse url suivante : <https://cs.nyu.edu/overton/software/hifoo/>.

hard goals soient satisfaits, *i.e.* $\gamma_g < 1$; on cherche ensuite à minimiser γ_f et donc il est plus habituel (voire nécessaire) d'avoir $\gamma_g < 1$. On cherchera ensuite à réduire au maximum γ_f .

“Control System Tuning in Simulink Made Easy” (Gahinet et Turevskiy, consulté le 23.11.18) donne un exemple d'utilisation de la Robust Control Toolbox, de ses fonctions et objets. La fonction **systune** s'utilise ainsi :

```
[SYS, fSoft, gHard, info] = systune(SYS_0, softGoals, hardGoals, opt)
```

La variable *SYS_0* porte le modèle augmenté de la boucle interne (connexion de modèles, paramètres réglages, fonctions de transfert, blocs logiques, augmenté de branches de performances, etc.) initialisé à une condition particulière. L'initialisation est importante car comme les fonctions à résoudre sont non-convexes, la résolution est locale uniquement. Une mauvaise condition initiale fait converger la synthèse vers une mauvaise solution. Si aucune condition initiale n'est connue, on peut demander une initialisation aléatoire dans les options de la fonction.

Les fonctions coûts *softGoals* et *hardGoals* sont instanciées par la librairie de *TuningGoal* qui comporte plusieurs natures de contraintes possibles. On y trouve des contraintes sur les gains H_∞ , les gains H_2 , le placement des pôles, les marges de stabilités, la réjection de signal, le suivi de référence et d'autres encore.

3.2.4 Synthèse robuste et séquencée

Un des grands intérêts de la méthode est sa possibilité d'effectuer des synthèses multi-modèles. Au cours d'une seule synthèse, l'outil est capable de traiter plusieurs instances du modèle augmenté. Cela devient pertinent dans le cas de système incertains, si l'on ne connaît pas exactement la valeur de certains paramètres, ou si la valeur de ces paramètres est amenée à évoluer selon le point d'équilibre choisi. Dans les applications du vol d'un avion, les conditions et paramètres qui évoluent sont nombreux sur l'enveloppe de vol (Chapitre 2). En une synthèse, on est alors capable d'optimiser le contrôleur pour des modèles répartis sur la totalité du domaine.

Pour les synthèses multi-modèles, la méthode permet aussi d'appliquer des contraintes à une sous-sélection des modèles, étant donné que les critères de performances peuvent être différents selon les cas centraux et les cas les plus extrêmes de l'enveloppe de vol. Dans ce cas, le problème que Matlab résout pour un modèle augmenté de dimension N_m -modèles est :

$$x^* \in \arg \min_{\max_{i \in [1, N_m]} g_i(x) \leq 1} \max_{j \in [1, N_m]} f_j(x) \quad (3.5)$$

Il existe deux possibilités (qui peuvent aussi être combinées) pour l'évolution des gains sur le domaine des points d'équilibre. La première possibilité est de définir les gains réglables comme une valeur unique à régler, pour tous les points d'équilibre. Dans ce cas, on cherchera un contrôleur commun à tous ces points, et pour lesquels les critères de performances sont respectés. On cherchera alors à développer la robustesse de la loi de commande (Puyou et Eezerzere, 2012).

La seconde possibilité est de définir les gains réglables comme une fonction de paramètres variant dans le domaine (Lhachemi et al., 2015). Cette définition est appelée "séquencée" (*scheduled*). Alors qu'usuellement le séquencement des gains était obtenu par un outil complémentaire qui utilisait les gains de N_m synthèses sur un modèle unique, les coefficients du séquencement sont maintenant déterminés directement par la synthèse. On parle de synthèse "auto-séquencée" (Vito et al., 2010; Saussié et al., 2013). La fonction de séquencement (souvent polynomiale) est définie au préalable lors de la définition des gains réglables ainsi que le domaine des modèles utilisés. En supposant que le gain K réglable dépend du point d'équilibre $\theta \in \mathbb{R}^n$:

$$K = K_0 + \sum_{k=1}^{n_\sigma} K_k \sigma_k(\theta) \quad (3.6)$$

où K_0, K_1, \dots sont les coefficients de la fonction de séquencement déterminés lors de la synthèse, n_σ est l'ordre de séquencement, et $\sigma_k(\theta)$ est une fonction définie au préalable qui utilise les valeurs des paramètres caractéristiques du point d'équilibre (notamment de sa position dans le domaine). En supposant que le point d'équilibre θ est caractérisé par deux paramètres θ_1 et θ_2 , des exemples de $\sigma_k(\theta)$ pourraient être : $\sigma_i(\theta) = \theta_1$, $\sigma_i(\theta) = \cos(\theta_2)$, $\sigma_i(\theta) = \theta_1 \theta_2^2$, ou encore $\sigma_i(\theta) = e^{\theta_1}$. L'ensemble des coefficients K_0, K_1, \dots sont inclus dans l'ensemble des paramètres réglables. Le problème revient donc au même que celui de l'équation 3.5. L'ajout de paramètres réglables augmente la taille du problème à résoudre, la résolution est seulement plus lente.

En combinant les deux approches, on peut alors définir une commande robuste selon le paramètre θ_1 et séquencée avec θ_2 et θ_3 . Les possibilités de construction du modèle augmenté sont alors infinies.

3.3 Des requis vers les concepts

Dans cette section, on exprime l'ensemble des requis imposés par l'industrie ainsi que leur réalisation. Les requis listés ci-dessous sont regroupés par sujet et numérotés, afin d'y faire

référence.

3.3.1 Structure de la loi de commande et séquencement des gains

Les premiers requis imposés concernent la loi de commande, sa structure et les gains réglables.

Requis n°1 (Req1) : Le modèle en BO réglé par la synthèse doit représenter l'architecture de la loi de commande de l'avion présentée dans la documentation de BA⁵.

Concept n°1 (Co1) : Le concept associé à ce requis est le schéma de la boucle interne de la loi de commande qui sera utilisée.

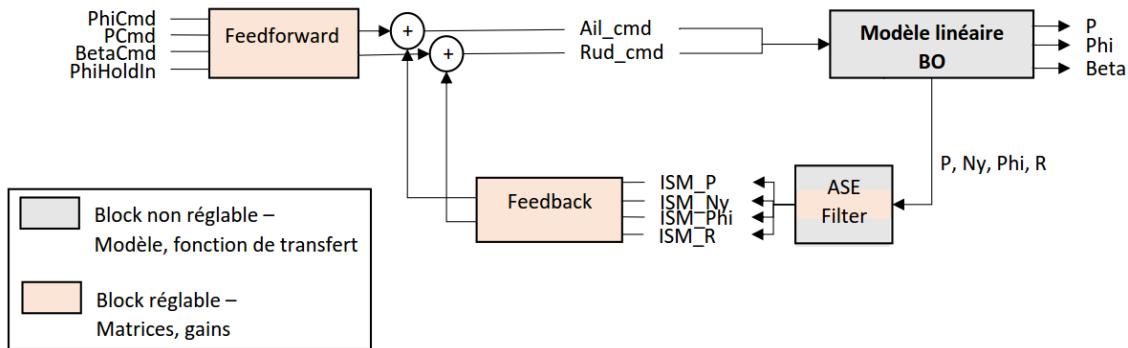


Figure 3.1 Schéma de la boucle interne de la loi de commande (Co1)

Le retour est implémenté comme une matrice réglable, et les gains de feedforward sont implémentés comme des gains. Même lorsque l'on procède à la schématisation simplifiée sans certains gains dynamiques, on gardera la matrice feedforward 2×2 déclarée à l'aide de quatre gains indépendants (notamment à cause des pré-filtres qui peuvent être associés à certains gains).

Selon l'application, on utilisera les signaux à la sortie des filtres ASE du modèle linéaire BO, ou on modélisera les filtres par des fonctions de transferts équivalentes et à gains réglables.

Requis n°2 (Req2) : Les gains de feedback et de feedforward doivent être séquencés en vitesse et altitude (*i.e.*, selon la condition de vol) pour une configuration donnée, mais robustes en chargement. L'évolution des gains doit être douce et non erratique.

Les gains des filtres ASE et le gain K_p pourront être réglés afin d'augmenter le nombre de degrés de liberté de la synthèse, mais ils restent robustes pour toutes les conditions de vol et de charge, pour une configuration de flap donnée.

5. Cette architecture est présentée dans les figures 2.5 et 2.6.

Concept n°2 (Co2) : Un séquencement polynomial apporte des courbes suffisamment douces. Après une étude des gains déjà en place, on remarque qu'ils sont plus sensibles à un changement de vitesse (CAS) qu'à l'altitude (ALT). On choisira les polynômes suivants :

- ordre 2 : $K_{i,j} = \alpha_{0,i,j} + \alpha_{1,i,j} * CAS + \alpha_{2,i,j} * ALT + \alpha_{3,i,j} * CAS^2 + \alpha_{4,i,j} * CAS * ALT$
- ordre 3 : $K_{i,j} = \alpha_{0,i,j} + \alpha_{1,i,j} * CAS + \alpha_{2,i,j} * ALT + \alpha_{3,i,j} * CAS^2 + \alpha_{4,i,j} * CAS * ALT + \alpha_{5,i,j} * CAS^3 + \alpha_{6,i,j} * CAS^2 * ALT$
- ordre 4 : $K_{i,j} = \alpha_{0,i,j} + \alpha_{1,i,j} * CAS + \alpha_{2,i,j} * ALT + \alpha_{3,i,j} * CAS^2 + \alpha_{4,i,j} * CAS * ALT + \alpha_{5,i,j} * CAS^3 + \alpha_{6,i,j} * CAS^2 * ALT + \alpha_{7,i,j} * CAS^4 + \alpha_{8,i,j} * CAS^3 * ALT$

On remarque que les valeurs des gains ne dépendent pas de la masse de l'avion ou de la position du CG ; la solution de l'optimisation doit y être robuste. Les fonctions polynomiales ci-dessus seront utilisées autant pour les gains de feedforward que pour les gains de feedback. Les gains des filtres ASE seront réglés de manière unique sur toute l'enveloppe de vol considérée. Le gain K_p ne sera libéré dans aucune des applications (comme il est fixe dans les tableaux de gains de BA), on gardera la valeur proposée par l'entreprise.

Requis n°3 (Req3) : Les gains de feedback doivent rester des valeurs acceptables afin de limiter la commande de l'aileron et du rudder par le pilote, et de ne pas saturer les actionneurs. Des matrices K_{min} et K_{max} dont les coefficients limitent les amplitudes de chacun des gains sont données par BA.

Concept n°3 (Co3) : Lorsqu'une synthèse est séquencée, ce ne sont plus les gains de la matrice qui sont des paramètres réglables mais les coefficients des fonctions de séquencement. Limiter l'amplitude de ces gains ne peut pas agir de manière efficace (non-conservative) sur l'amplitude des gains finalement obtenus. On ne peut définir de limite ainsi lors de la déclaration de la matrice réglable.

La limitation des amplitudes des gains de la matrice de feedback est alors assurée par 8 canaux de performances chacun appliqué sur un coefficient de la matrice. La contrainte $K_{ij,min} \leq K_{ij} \leq K_{ij,max}$ est alors traduite par une norme H_∞ en utilisant l'inégalité suivante :

$$limK_{ij} = \left| \frac{K_{ij} - \frac{K_{ij,max} - K_{ij,min}}{2}}{\frac{K_{ij,max} + K_{ij,min}}{2}} \right| \leq 1 \quad (3.7)$$

D'après l'équation précédente, on peut représenter dans la figure 3.2, la structure qui sera utilisée pour contraindre les amplitudes. Les 8 fonctions de pondération sont de simples fonctions identité et les 8 canaux de performances proviennent bien des 8 transferts entrée/sortie

possibles par la matrice de dimension 2×4 dont les coefficients sont définis dans l'équation 3.7.

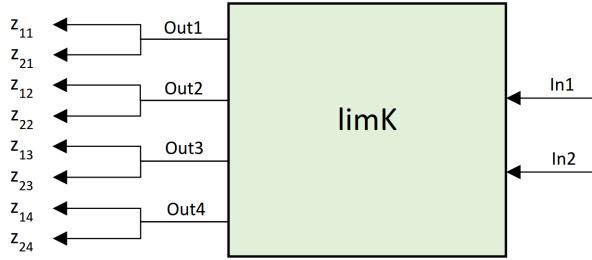


Figure 3.2 Concept de la limite sur l'amplitude des gains de retour (Co3)

Afin de réduire l'encombrement des schémas des modèles augmentés utilisés, on ne représentera pas ces canaux de performances, pour les applications à suivre.

3.3.2 Suivi de référence et découplage de signaux

Requis n°4 (Req4) : Les réponses en BF de certains transferts doivent satisfaire les critères de performance suivants :

- Req4a : La réponse de P à une commande en échelon de P_{cmd} doit être similaire à un transfert du premier ordre. Le dépassement doit être inférieur à $D_{\%,P}$. L'erreur en Régime Permanent (RP) acceptée est $\epsilon_{\%,P}$.
- Req4b : La réponse de ϕ à une commande en échelon de ϕ_{cmd} doit être similaire à un transfert du premier ordre. Le dépassement doit être inférieur à $D_{\%,\phi}$. L'erreur en RP acceptée est $\epsilon_{\%,\phi}$.
- Req4c : La réponse de β à une commande en échelon de β_{cmd} doit être similaire à un transfert du second ordre. Le dépassement doit être inférieur à $D_{\%,\beta}$. L'erreur en RP acceptée est $\epsilon_{\%,\beta}$.

Concept n°4 (Co4) :

Rappel : Un dépassement (en %) correspond à la valeur maximum y_{max} d'une sortie y d'un système stable en réponse à un échelon, relativement à sa valeur en RP y_∞ . Il se calcule : $D_{\%,y} = 100 \times \frac{y_{max} - y_\infty}{y_\infty}$.

Rappel : Une erreur en RP (en %) correspond à l'erreur relative en RP d'une sortie y d'un système stable en réponse à un échelon d'amplitude A . Il se calcule : $\epsilon_{\%,y} = 100 \times \frac{y_\infty - A}{A}$.

Afin de forcer les performances temporelles d'une réponse en BF, on utilisera un signal de référence et l'on forcera le système à suivre au mieux cette référence temporelle. L'approche

usuelle consiste à minimiser la norme H_∞ entre la dite-réponse en BF et la réponse de la référence à la même entrée. La référence est construite à partir des objectifs de performances : dépassement, temps de réponse, placement de pôles.

On ajoute alors un canal qui produit la réponse de référence $y_{ref} = T_{ref}r$ et la différence $y_{err} = y_{ref} - y$ est prise en compte par la synthèse, pour minimiser la norme H_∞ du transfert de r vers y_{err} . On peut ajouter une fonction de pondération W dans le but d'optimiser le suivi de référence sur les plages de fréquences souhaitées (plus souvent des filtres passe-bas). Ce nouveau transfert lié uniquement à un objectif de performance, est implémenté dans le modèle augmenté utilisé par la synthèse et non dans le système réel.

Prenons en guise d'exemple le système donné à la figure 3.3. Dans cette configuration T_{ref} est la fonction de transfert de référence (en vert) qui indique l'objectif à suivre, W est une fonction de pondérations (en bleu). La fonction de coût à minimiser est alors :

$$f(K) = \|W(T_{ref} - (1 + PK)^{-1}PK)\|_\infty \quad (3.8)$$

La configuration présentée ici est simple et pourrait être augmentée à une deuxième fonction de pondération comme préfiltre au transfert d'intérêt.

Un canal de performance tel que dans la figure 3.3 est ajouté pour chacun des suivis, il existera donc jusqu'à trois fonctions de référence et signaux d'erreur dans le modèle augmenté.

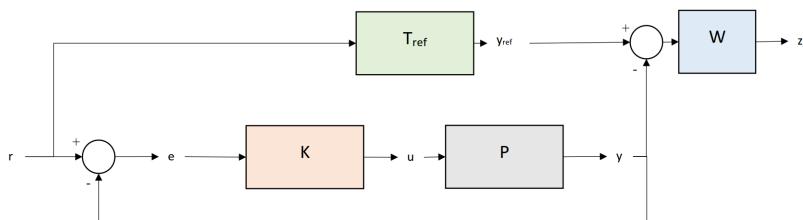


Figure 3.3 Concept du suivi de référence (Co4)

Requis n°5 (Req5) : Les réponses en BF suivantes doivent être minimalement affectées lors d'excitations d'autres entrées. Les transferts croisés suivants doivent être découplés :

- Req5a : Pour la réponse de ϕ à une commande en échelon β_{cmd} , le dépassement doit être inférieur à $D_{\%, \beta \rightarrow \phi}$. L'erreur en RP acceptée est $\epsilon_{\%, \beta \rightarrow \phi}$.
- Req5b : Pour la réponse de β à une commande en échelon de P_{cmd} , le dépassement doit être inférieur à $D_{\%, P \rightarrow \beta}$. L'erreur en RP acceptée est $\epsilon_{\%, P \rightarrow \beta}$.

- Req5c : Pour la réponse de β à une commande en échelon de ϕ_{cmd} , le dépassement doit être inférieur à $D_{\%, \phi \rightarrow \beta}$. L'erreur en RP acceptée est $\epsilon_{\%, \phi \rightarrow \beta}$.

Concept n°5 (Co5) :

Rappel : Des sorties sont découplées lorsque l'on peut commander chacune des sorties indépendamment l'une de l'autre. Un découplage parfait d'une entrée u vers une sortie y est atteint lorsque cette entrée n'a aucun impact sur la sortie. Pour un système LTI pour lequel la fonction de transfert de u vers y est notée $P(s)$, un découplage parfait est atteint si $P(s) = 0$. Comme un tel découplage est quasiment impossible à atteindre pour des systèmes réels, on évalue la performance en découplage selon des critères de dépassement et d'erreur en RP.

Dans le cas d'un découplage de signaux, l'approche utilisée est de venir minimiser la norme H_∞ de la réponse du signal à découpler pour une excitation du signal d'entrée. À part une fonction de pondération pour affiner l'optimisation, il n'y a pas de fonctions supplémentaires ou de connexions à faire (Fig. 3.4). Ce canal de performance est simplement constitué de la sortie du signal et d'une ou deux éventuelles fonctions de pondération, qui comme précédemment ne sont pas réellement implémentées dans le système réel mais dans le modèle augmenté donné à la fonction de synthèse. Le schéma augmenté pourra contenir jusqu'à trois canaux de découplage.

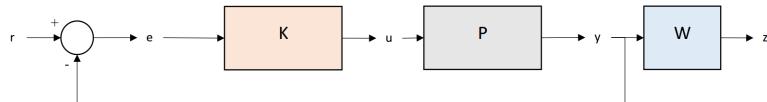


Figure 3.4 Concept du découplage de signaux (Co5)

On peut remarquer que ce concept revient finalement au concept précédent Co4 (suivi de référence) dans l'éventualité où la fonction de référence est la fonction nulle. En effet, le canal contenant le sommateur est alors simplifié par un gain -1, qui peut finalement être enlevé par le passage à la norme H_∞ (par définition d'une norme : $|-x| = |x|$).

3.3.3 Marges de stabilité

Requis n°6 (Req6) : Les BO suivantes doivent assurer les marges de stabilités M_G dB en gain et M_ϕ degrés en phase :

- En ouvrant la boucle au niveau de Ail_{cmd} , le transfert BO doit respecter les marges de stabilité (M_G dB, M_ϕ deg).
- En ouvrant la boucle au niveau de Rud_{cmd} , le transfert BO doit respecter les marges de stabilité (M_G dB, M_ϕ deg).

Concept n°6 (Co6) :

Rappel : La marge de gain d'un système Single Input Single Output (SISO) LTI dont la fonction de transfert est $P(s)$, indique la stabilité relative de la BF $P(s)/(1 + P(s))$ (fermeture par retour négatif). Elle représente l'augmentation ou la diminution de gain nécessaire pour déstabiliser le système (amener le gain de la boucle égal à 1) à la fréquence particulière ω_{-180} pour laquelle l'angle de phase est de -180° (modulo 360°).

Elle se calcule : $\arg(P(j\omega_{-180})) = -180^\circ \rightarrow M_G = 1/|P(j\omega_{-180})|$.

Rappel : La marge de phase d'un système SISO LTI dont la fonction de transfert est $P(s)$, indique la stabilité relative de la BF $P(s)/(1 + P(s))$ (fermeture par retour négatif). Elle représente la différence entre la phase -180° et la phase de la réponse lorsque le gain est unitaire. Cette fréquence caractéristique, lorsque $|P(j\omega_{-1})| = 1$ est appelée la fréquence de gain de coupure (*gain crossover frequency*).

La marge de phase est calculée ainsi : $|P(j\omega_{-1})| = 1 \rightarrow M_\phi = \arg(P(j\omega_{-1})) + 180^\circ$.

Pour ajouter les critères de marge de gain M_G et marge de phase M_ϕ à la synthèse, le concept est de traduire les deux contraintes en région d'exclusion dans le diagramme de Nichols afin de les intégrer dans un critère de marges de disque (*disk margin*⁶) (Bates et Postlethwaite, 2002). Notons que l'utilisation de marges de disque est ingénieuse car elle permet de considérer les variations de gains et de phase simultanément et leur impact sur la stabilité de la BF. Pour cette raison, notons aussi que ce critère est par conséquent plus conservatif que la mesure même des marges de gain et de phase.

Dans le diagramme de Nichols, la région d'exclusion (M_G dB, M_ϕ deg) est une ellipse de longueur M_ϕ et de largeur M_G . L'équation de cette ellipse est :

6. Une schématisation du concept de marge de disque est donnée dans la documentation en ligne de Matlab de la fonction `dmplot`. La documentation est accessible à l'adresse url suivante : <https://www.mathworks.com/help/robust/ref/dmplot.html> (documentation consultée le 23/11/2018).

$$\frac{Y^2}{M_G^2} + \frac{(X + 180)^2}{M_\phi^2} \leq 1 \quad (3.9)$$

où Y le gain et X la phase de la BO. Si on traduit cette équation dans le diagramme de Nyquist, la région d'exclusion équivalente a la forme d'un cercle de centre a et de rayon r , tels que :

$$a = \frac{(10^{M_G/20})^2 - 1}{2(10^{M_G/20} - \cos(M_\phi \frac{\pi}{100}))}, \quad r = 10^{M_G/20} - a \quad (3.10)$$

Finalement, la perturbation peut être représentée comme un ensemble de perturbations où Δ est dans la boule complexe unitaire, c'est-à-dire $\|\Delta\|_\infty < 1$ afin de pouvoir appliquer le théorème du petit gain (Bates et Postlethwaite, 2002).

$$P_{aug} = P(a + r\Delta) \quad (3.11)$$

Par identification :

$$P_{aug} = PW_G(1 + W_\phi\Delta), \quad W_G = a \text{ et } W_\phi = \frac{r}{a} \quad (3.12)$$

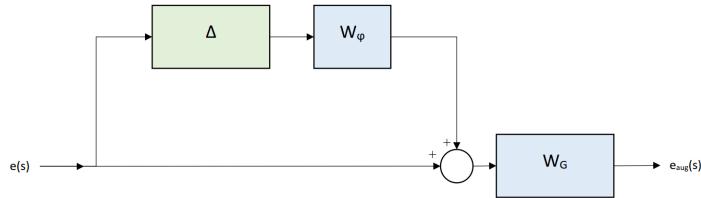


Figure 3.5 Concept des marges de gains et de phase (Co6)

De cette manière, et en utilisant le canal de performance de la figure 3.5, on peut prendre en compte les marges de stabilités en tant que contraintes de la synthèse. Dans l'exemple de la figure 3.6, la contrainte sera respectée si :

$$\left\| \frac{W(s)}{Z(s)} \right\|_\infty \leq 1 \quad (3.13)$$

Cette contrainte est notamment prise en charge par la fonction *TuningGoal.Margins*⁷ de la toolbox Control System de Matlab, qui permet de spécifier des marges de phase et de gain à imposer sur une BO. On indique l'ouverture de la boucle à l'aide d'un point d'analyse inséré

7. Documentation consultée le : 23/11/2018 à l'adresse url suivante : <https://www.mathworks.com/help/control/ref/tuninggoal.margins-class.html>.

lors de la création du modèle augmenté.

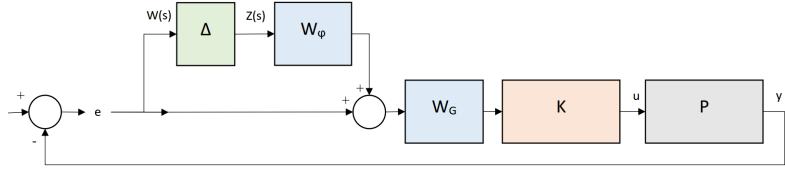


Figure 3.6 Exemple du canal de performance utilisé pour les marges de gain et de phase (Co6)

Requis n°7 (Req7) : Les BO suivantes doivent assurer des marges de stabilité suffisantes pour toutes fréquences supérieures ou égales à celle du premier mode flexible. Les marges appliquées sont M_{flex} dB ($M_{flex} \in \mathbb{R}$) pour un modèle flexible et $M_{rigide}(s)$ dB (un gabarit fréquentiel) pour un modèle rigide :

- En ouvrant la boucle au niveau de Ail_{cmd} , le transfert BO doit respecter les marges de stabilité (M_{flex} dB, $M_{rigide}(s)$ dB).
- En ouvrant la boucle au niveau de Rud_{cmd} , le transfert BO doit respecter les marges de stabilité (M_{flex} dB, $M_{rigide}(s)$ dB).

Concept n°7 (Co7) : Le concept de cette marge est plus simple à exprimer car la limite imposée concerne ici une limite d'amplitude sur une réponse fréquentielle. On choisira :

$$W_{flex,M_A} = \frac{1}{10^{\frac{-M_{flex}}{20}}} = 10^{\frac{M_{flex}}{20}} \quad (3.14)$$

ce qui revient à décaler de M_{flex} dB vers le haut la réponse fréquentielle de la BO. Ainsi assurer que $\|W_{flex,M_A} T_{Ail \rightarrow Ail}\|_\infty < 1$, revient à assurer que la valeur du pic de $W_{flex,M_A} T_{Ail \rightarrow Ail}(s)$ est plus petite en norme que 1, donc que 0dB ($\log(1) = 0$). Si la valeur du pic du transfert précédent est plus petit que 0, alors l'amplitude du transfert en dB sera plus petit que 0, pour toutes les fréquences, et ainsi l'amplitude du transfert $T_{Ail \rightarrow Ail}(s)$ sera plus petit que M_{flex} dB pour toutes les fréquences. De cette manière, on parvient alors à contraindre la marge de stabilité désirée à l'aide d'une expression utilisant la norme H_∞ .

La marge précédente est donnée afin que la stabilité ne soit pas impactée lorsque les modes flexibles sont considérés. On ne peut donc pas appliquer la même marge M_{flex} dB sur les modèles rigides, car une fois les flexibilités ajoutées par dessus la dynamique rigide, la réponse fréquentielle est poussée vers le haut et les marges risquent fortement de ne plus être respectées. Dans le cas des modèles rigides, on ne dispose pas de requis fixé par BA. On décide alors

de tracer toutes les réponses fréquentielles des BO de Ail_{cmd} puis de Rud_{cmd} et de dessiner un gabarit fréquentiel (une fonction de transfert) $M_{rigide}(s)$ qui approxime le comportement fréquentiel maximal de tous les autres transferts. Si on impose les modèles rigides à rester sous cette fonction, l'ajout des flexibilités ne devrait donc pas risquer de compromettre la stabilité.

Reprendons le raisonnement précédent avec le transfert $T_{Ail \rightarrow Ail}$, que l'on appellera simplement T pour cette démonstration. Finalement, on cherche à assurer que :

$$\begin{aligned} \forall \omega, |T(s)| &\leq |M_{rigide}(s)| \\ |M_{rigide}(s)|^{-1}|T(s)| &\leq 1 \\ |M_{rigide}(s)^{-1}| |T(s)| &\leq 1 \\ |M_{rigide}(s)^{-1}T(s)| &\leq 1 \\ \|M_{rigide}(s)^{-1}T(s)\|_\infty &\leq 1, \text{ par définition de } \|\cdot\|_\infty \end{aligned}$$

On peut alors considérer $W(s) = M_{rigide}(s)^{-1}$. Avec cette définition de la fonction de pondération, on pourra assurer les marges d'amplitude désirées.

3.3.4 Positionnement des pôles

Requis n°8 (Req8) : Les pôles en BF des modes caractéristiques de la commande latérale-directionnelle doivent être placés dans leur zone de confinement :

- Req8a : le mode roulis pur doit être un pôle réel négatif dont la constante de temps est comprise entre $T_{Roll,min}$ et $T_{Roll,max}$ secondes.
- Req8b : le mode spirale doit être un pôle réel négatif dont la constante de temps est inférieure à $T_{SP,max}$ secondes.
- Req8c : le mode roulis hollandais doit être une paire de pôles complexes conjugués dont la pulsation naturelle doit être supérieure à $\omega_{DR,min}$ et le coefficient d'amortissement supérieur à $\xi_{\beta,min}$.

Concept n°8 (Co8) :

Le placement des pôles pourra être assuré par deux techniques. En effet, un bon suivi de référence permettra d'avoir des dynamiques proches de la référence, donc des pôles centrés dans les zones désirées. On rappelle qu'un transfert du premier ordre donne un pôle réel dominant, et un transfert du second ordre, deux pôles complexes conjugués.

Les requis Req4 en section 3.3.2 sont liés au placement des pôles du Req8 de la manière suivante :

- Les requis Req4a et Req8a sont liés. Le pôle du roulis pur domine la dynamique du transfert de P_{cmd} vers P .
- Les requis Req4b et Req8b sont liés. Le pôle du spiral domine la dynamique du transfert de ϕ_{cmd} vers ϕ .
- Les requis Req4c et Req8c sont liés. Les pôles complexes du roulis hollandais dominent la dynamique du transfert de β_{cmd} vers β .

La seconde technique consiste à utiliser la fonction *TuningGoal.Poles*⁸ de Matlab. Elle permet d'ajouter des requis sur la position des pôles dans le plan complexe, notamment un amortissement minimal ou un écart minimal entre l'axe des imaginaires et le premier pôle (partie réelle maximale).

Les problèmes rencontrés par cette technique proviennent du fait que les modèles manipulés possèdent parfois des pôles difficilement observables et contrôlables. Ces derniers, très mal amortis, proches de l'axe des imaginaires ne peuvent parfois pas être déplacés par la synthèse. Cette contrainte trop forte ne peut être réalisée et pénalise les autres requis.

Une solution est de procéder à une réduction des modèles pour s'en séparer et pouvoir agir directement sur les pôles dominants du système.

Requis n°9 (Req9) : Afin d'assurer la stabilité, les pôles en BF doivent être :

- à partie réelle strictement négative,
- à partie réelle positive inférieure à X_{max} et une partie imaginaire (en valeur absolue) inférieure à Y_{max} .

Concept n°9 (Co9) : Un des requis initiaux de la synthèse H_∞ est de trouver un contrôleur qui stabilise internement le système en BF. La fonction *systune* cherche par défaut à stabiliser la BF. Si certains pôles ne peuvent être stabilisés tout en respectant les *hard goals*, la synthèse indique **Failed to stabilize** avec la valeur du pôle réel le plus instable.

Cependant, certains systèmes réels ne peuvent être stabilisés pour tous les cas. En l'occurrence, un avion est un système qui évolue dans un domaine très large d'altitude, de vitesse, etc. et il est impossible d'avoir des pôles à partie réelle négative pour chacune des conditions. Si la synthèse veut prendre en compte l'ensemble des conditions de vol, elle devra accepter la présence de pôles instables. Bien entendu, ces pôles devront être faiblement instables pour

8. Documentation consultée le : 23/11/2018 à l'adresse url suivante : <https://www.mathworks.com/help/control/ref/tuninggoal.poles-class.html>.

ne pas perturber les performances obtenues, et ne pas être perceptibles par le pilote.

Il n'existe aucune option possible directement dans les options de *systune* pour accepter ces pôles. L'option *minDecayRate* de *TuningGoal.Poles* ne prend que des valeurs positives. L'écart accepté entre l'axe des imaginaires et le premier pôle ne peut être que pour imposer une marge de stabilité et non autoriser une déstabilisation.

La solution trouvée est d'utiliser l'option *Stabilize* proposée lors de la déclaration des canaux de performances utilisant la fonction *TuningGoal.WeightedGain*⁹. Cette option permet d'enlever le requis de stabilité sur le transfert de la BF de l'entrée spécifiée à la sortie spécifiée dans la déclaration du canal de performance. Si la stabilité n'est pas un requis essentiel ou ne peut être atteinte, il faut alors définir l'option *Stabilize* à faux, *i.e.* à 0.

3.3.5 Efforts des actionneurs

Requis n°10 (Req10) : L'effort demandé aux actionneurs doit rester dans des amplitudes raisonnables et réalisables par les surfaces de commandes.

On définit :

- Ail_{max} , l'angle maximal possible pour l'aileron ;
- Rud_{max} , l'angle maximal possible pour le rudder.

Concept n°10 (Co10) : Ces limitations comme pour le concept Co7 de la section 3.3.3 (pour la marge d'amplitude des modèles flexibles) sont prises en compte par des fonctions de pondérations.

En pratique, on se rend compte que le seul transfert qui provoque des efforts trop importants, est le transfert de ϕ_{cmd} vers Ail_{cmd} . On agira seulement sur ce transfert pour ne pas surcharger le modèle augmenté. La fonction $W = \frac{1}{10^{-\frac{\gamma}{20}} Ail}$ est utilisée. La constante γ est choisie pour diminuer suffisamment l'effort de commande sans dégrader les performances atteintes.

3.4 Optimisations de design

Dans cette section, on présentera quelques guides pour améliorer le design, affiner les réglages et synthétiser une meilleure solution.

9. Documentation consultée le 23/11/2018 à l'adresse url suivante : <https://www.mathworks.com/help/control/ref/tuninggoal.weightedgain-class.html>.

3.4.1 Préciser la condition initiale

Comme mentionné précédemment, le problème à résoudre est non convexe et donc la solution optimale vers laquelle converge la synthèse est en général un minimum local. La condition initiale choisie a donc une grande influence sur la solution synthétisée.

Afin d'obtenir une meilleure condition initiale, on pourra partir d'une solution que l'on sait performante (si on la connaît). Dans notre situation, on connaît la solution utilisée par BA, et on sait qu'elle propose un compromis intéressant entre les critères de performances et de stabilité. C'est donc un bon point de départ pour l'optimisation.

Dans le cas du contrôle séquencé, on pourra utiliser deux points de départ intéressants : l'interpolation polynomiale des gains de BA ou celle formée à partir des gains optimisés point à point par la synthèse H_∞ . Cette deuxième méthode est notamment utilisée dans le chapitre 5, tandis qu'on préfère la première pour le chapitre 6.

3.4.2 Préciser le réglage des fonctions de pondérations

Afin d'obtenir une meilleure solution synthétisée, il faudra bien choisir tous les éléments de réglage, comme les fonctions de référence, choix entre *hard* et *soft goals*, etc. Parmi eux, régler les fonctions de pondérations correctement peut s'avérer plus complexe que choisir une valeur de marge de gain ou définir un amortissement minimal, venant du fait que le domaine fréquentiel est moins intuitif que le domaine temporel.

On rappelle que les fonctions de pondérations sont des gabarits fréquentiels. On choisira de manière générale des filtres passe-bas. On conseille de diminuer l'allure fréquentielle pour les plus basses fréquences (vers 0), pour renforcer les contraintes sur l'erreur en RP. Pour un meilleur suivi de consigne pendant le Régime Transitoire (RT), on conseillera au contraire de renforcer les contraintes sur les fréquences moyennes (de 0.1 à 100 rad/s environ).

Pour contraindre une erreur en dessous de $e\% > 0$ d'un système $T(s)$, il faut que :

$$|T(0) - 1| \leq \frac{e\%}{100}$$

i.e., $\frac{e\%}{100} |T(0) - 1| \leq 1$

On reconnaît alors une forme qui peut être contrainte par H_∞ . On choisira une fonction de référence T_{ref} telle que $T_{ref}(0) = 1$ et une fonction de pondération W telle que $W(0) = \frac{100}{e\%}$.

Dans ce cas, contraindre $\|W(T - T_{ref})\|_\infty \leq 1$ implique $\frac{100}{e\%}|T(0) - 1| = |W(0)(T(0) - T_{ref}(0))| \leq \|W(T - T_{ref})\|_\infty \leq 1$. Ceci assure une erreur en RP en dessous de $e\%$ si la contrainte H_∞ est satisfaite.

De la même manière, dans le cas d'un découplage, si l'on désire limiter l'amplification du signal d'entrée en dessous de $A > 0$, on prendra $W(0) = \frac{1}{A}$.

3.5 Conclusion

La synthèse H_∞ structurée apporte les améliorations nécessaires à l'application des problèmes industriels. Les outils sont maintenant suffisamment performants pour résoudre les problèmes d'optimisation dans un temps relativement court. La technique est aujourd'hui suffisamment évoluée pour pouvoir présenter un intérêt pour l'entreprise. De plus, on a démontré que l'ensemble des requis imposés peuvent être traduits sous forme de contraintes prises en charge par la synthèse et son outil. Moyennant quelques itérations de réglages, les solutions proposées aux problèmes réels de BA devraient être performantes et valider les avantages de la méthode de synthèse.

Avant de s'attaquer à une synthèse complète de la loi de commande sur l'ensemble des conditions de vol, on implémente d'abord un outil supplémentaire pour appliquer la synthèse H_∞ structurée sur les modèles de BA.

CHAPITRE 4 CONCEPTION D'UN OUTIL LOGICIEL

Ce chapitre présente l'outil logiciel implémenté afin de synthétiser des contrôleurs performants par la synthèse H_∞ structurée. Après une introduction (Section 4.1) qui rappelle notamment des points théoriques de la conception logicielle, la section 4.2 liste les requis et contraintes imposées. Par la suite, les concepts sont développés dans la section 4.3, accompagnés par quelques schémas de l'architecture, suivis dans la partie suivante (Section 4.4) de la procédure de tests et vérifications appliquées à l'outil. Finalement, une conclusion en Section 4.5 prend position sur la validité de l'outil implémenté et sur les possibilités d'amélioration de celui-ci.

4.1 Introduction

4.1.1 Contexte et présentation

Dans le contexte du projet de commande de l'avion par la synthèse H_∞ structurée, un outil intégré à l'environnement de BA a du être implémenté afin de synthétiser une loi de commande performante.

Si on considère les outils existants, ils s'inscrivent déjà complètement dans l'environnement de l'entreprise et remplissent la fonction principale qui est de synthétiser une loi de commande pour un ensemble donné de points de fonctionnement. Seule la méthode appliquée pour obtenir le réglage des contrôleurs semble différer. Cependant, il apparaît rapidement que toute la structure interne de l'outil doit en réalité être modifiée notamment grâce aux nouvelles capacités de la méthode (réglage simultané sur toute l'enveloppe de vol, et non point par point). Un tout nouvel outil est donc nécessaire.

Il réutilise autant que possible les outils déjà existants tout en s'adaptant à la nouvelle méthode. Des fonctionnalités supplémentaires sont ajoutées afin d'analyser grossièrement les résultats obtenus. Le logiciel sera manipulé par des ingénieurs de l'équipe CLaws de BA afin de régler rapidement et de manière performante la loi de commande.

4.1.2 Quelques rappels de génie logiciel

Quelques rappels de génie logiciel concernant la programmation orientée objet et la lecture de diagramme Unified Modeling Language (UML) sont donnés en annexe C. Pour une meilleure compréhension des informations partagées dans ce chapitre, on pourra s'y référer.

4.2 Requis

La méthodologie impose de commencer par lister l'ensemble des requis qui délimiteront les fonctionnalités du logiciel et apporteront des contraintes cruciales. Voici quelques exemples de requis :

- Le logiciel doit pouvoir fonctionner sous l'environnement Windows.
- Le logiciel doit être implémenté sous Matlab, pour les versions Matlab R2016b et postérieures.
- Le logiciel doit utiliser les toolboxes commerciales ou celles accessibles à l'équipe de CLaws.
- Le logiciel doit être composé d'un ensemble de fonctions et de scripts. Il ne doit pas nécessiter d'interface graphique.
- Le logiciel doit s'intégrer parfaitement à son environnement. Ses entrées et sorties devront être les formats de fichiers similaires aux documents usuellement utilisés par l'équipe et les outils de CLaws.
- Le logiciel doit permettre à un utilisateur/développeur de facilement ajouter, enlever ou modifier des variables, des fonctions d'analyses, des canaux de performances, etc.
- Lors de la compilation, le logiciel doit indiquer les causes de l'erreur si l'exécution s'interrompt.
- Si la compilation est réussie, le logiciel doit afficher le temps écoulé de la tâche de chargement des modèles jusqu'à la sauvegarde des résultats finaux.
- Le logiciel doit être capable de charger des modèles d'avion aux points de fonctionnement listés par l'utilisateur.
- Le logiciel doit être capable de linéariser des modèles d'avion en des modèles BO aux points de fonctionnement listés par l'utilisateur.
- Le logiciel doit être capable d'appliquer une série d'analyses prédéfinies par l'utilisateur aux points de fonctionnement listés par celui-ci.
- Le logiciel doit être capable de permettre à l'utilisateur de choisir comment initialiser la synthèse H_∞ structurée.
- Le logiciel doit être capable de synthétiser une loi de commande (gains de feedback et feedforward, et possiblement d'autres gains réglables) optimisée selon des branches de performances définies par l'utilisateur.
- Le logiciel doit être capable d'évaluer si la solution proposée remplit les premiers critères de succès.

4.3 Concepts, architecture et codage

4.3.1 Principales fonctionnalités

Afin de déterminer les principales fonctionnalités et les secondaires, le diagramme des cas (Fig. 4.1) est réalisé. Les principales fonctionnalités sont alors listées ci-après :

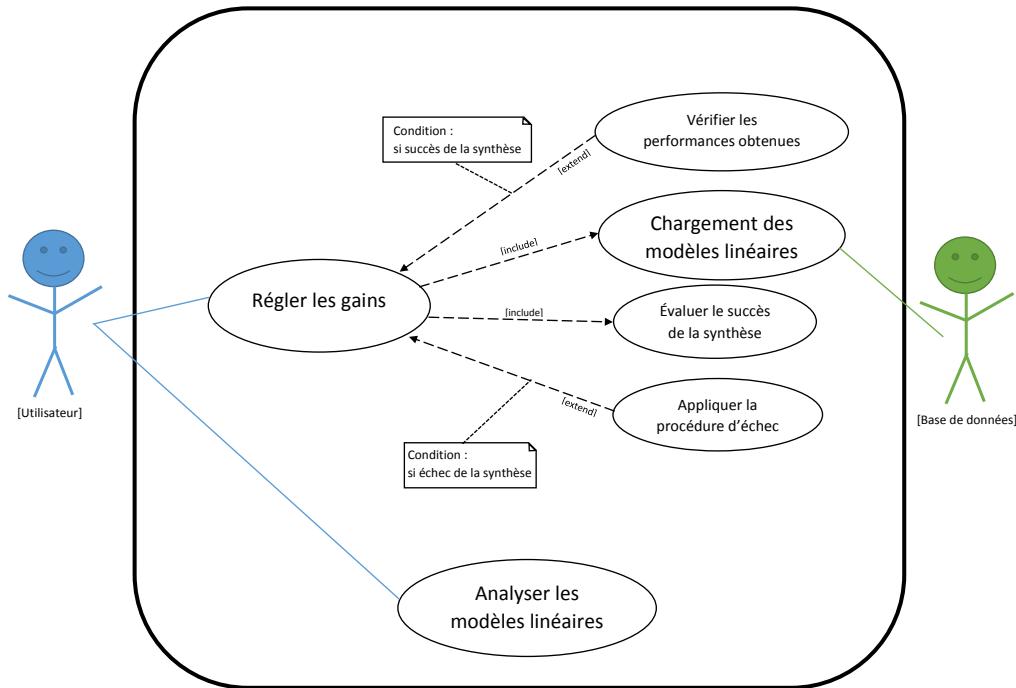


Figure 4.1 Diagramme de cas du logiciel

- Charger et linéariser les modèles d'avions (rigides et flexibles).
- Régler les gains des contrôleurs en utilisant la synthèse H_∞ structurée.
- Analyser la solution proposée.
- Sauver les résultats dans des fichiers au format approprié.

4.3.2 Définition des objets

Pour sélectionner les objets nécessaires à définir, on se réfère aux chapitres 2 et 3. En l'occurrence, les éléments manipulés par le logiciel sont en partie grandement détaillés dans le chapitre sur l'environnement de BA et sur les concepts.

De toute évidence l'élément central est le **modèle**. Un modèle est un objet mère de trois autres modèles, le modèle d'avion, le modèle BO et le modèle BF, qui héritent des propriétés et méthodes de la classe mère. Parmi les propriétés on peut retrouver les signaux d'entrées et de sorties, le modèle d'état (A, B, C, D), ou encore le point de vol utilisé pour la linéarisation.

Le **point de vol** est une autre classe pour laquelle une instance définit un état de l'avion et du vol, un point d'équilibre dans l'enveloppe de vol. Il a comme propriétés, les configurations de *flaps*, les configurations de charge, l'altitude et la vitesse (condition de vol).

Le **gain réglable** possède entre autres un tableau des valeurs prises par le gain sur l'enveloppe de vol considéré ainsi que le polynôme schédule qui le définit. La **synthèse** et l'**analyse** traduisent des actions majeures transparentes qui relient plusieurs classes entre elles. Les méthodes associées, dont les entrées/sorties sont déclarées en propriétés des classes, permettent d'appliquer différentes fonctions d'analyse et de synthèse. Finalement, le **back-up** est une sauvegarde indexée des **conditions initiales**, qui viennent initialiser la synthèse.

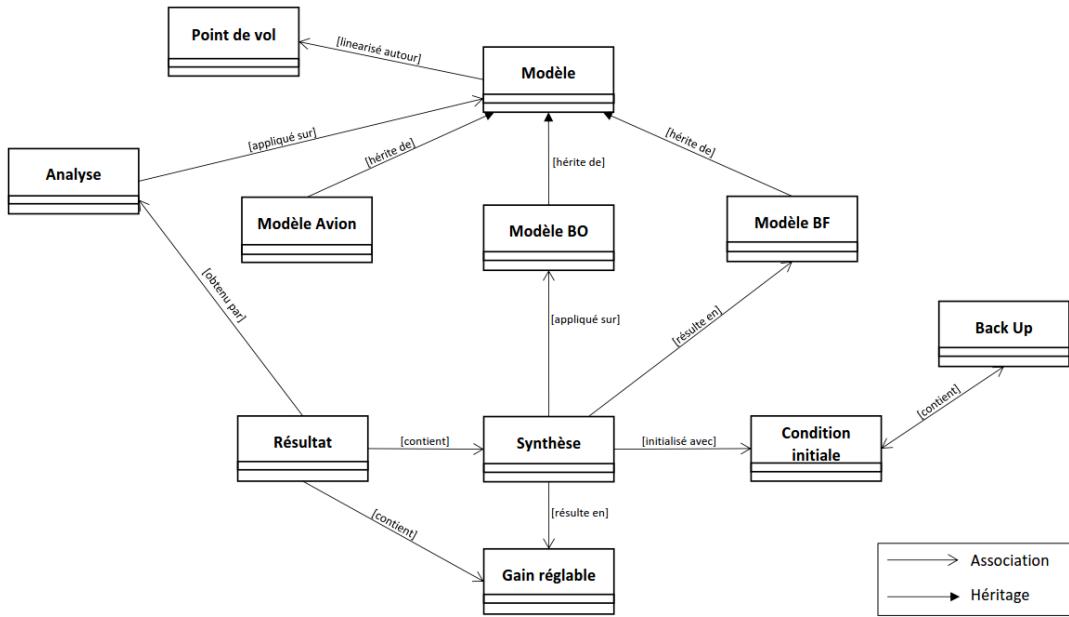


Figure 4.2 Diagramme général des classes du logiciel

4.3.3 Architecture générale

Une fois que le concept global du logiciel est trouvé, il est possible de dessiner un diagramme d'activités qui présente l'architecture générale choisie (Fig. 4.3). L'architecture se construit du général au particulier, une descente dans les niveaux de codes de l'outil.

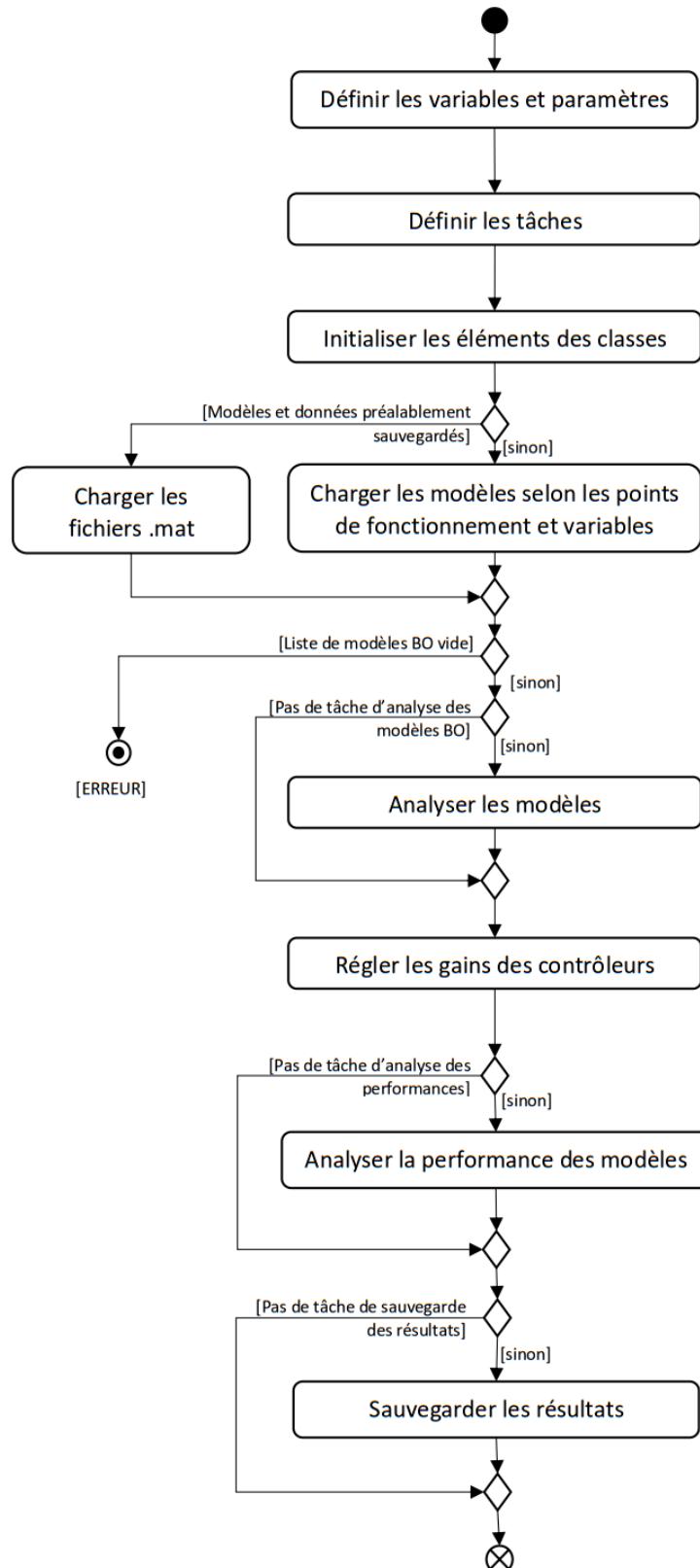


Figure 4.3 Diagramme général des activités du logiciel

4.3.4 Interface utilisateur

Les entrées du logiciel sont :

- des schémas-blocks Simulink de la modélisation linéaire de l'avion, fonctionnant avec Matlab R2009a¹.
- des fichiers de données Matlab .mat contenant les modèles linéaires de l'avion à de nombreux points de fonctionnement de toute l'enveloppe de vol.
- un fichier script Matlab qui spécifie les conditions de vol à considérer ainsi que d'autres paramètres du vol et de l'avion.
- un fichier script Matlab qui spécifie l'ensemble des tâches à exécuter (analyses et sauvegardes particulièrement).
- un fichier script Matlab qui spécifie les paramètres de design à utiliser pour la synthèse.

Les sorties du logiciel sont :

- les données résultant de la synthèse dans des fichiers Matlab .mat, qui contiennent les valeurs des gains des contrôleurs, les modèles en BF obtenus, les analyses et les caractéristiques de la synthèse.
- un fichier Excel .xls qui expose tous les cas traités, les gains obtenus et les positions des pôles dominants du système en BF.
- un fichier texte .txt qui fournit la valeur des gains de la loi de commande calculés à des points équidistants de l'enveloppe de vol pour le réglage des gains non linéaires.

Tout au long de l'exécution du logiciel, de nombreux messages sont affichés pour que l'utilisateur puisse être averti du déroulement de la synthèse, des activités actuellement terminées et en cours.

4.3.5 Application de la synthèse H_∞ structurée

La structure et le flux logique des activités conçues pour réaliser la synthèse H_∞ sont fournis par la figure 4.4. Dans cette figure, le premier diagramme (a) présente l'architecture de l'activité de synthèse. Une fois la synthèse préparée (voir le second diagramme (b)), la synthèse est effectuée. La fonction de réglage des gains appelle principalement la fonction `systune`. À chaque itération de `systune`, un back-up est sauvegardé et le précédent, s'il existe, est écrasé. Cette procédure permet alors de repartir de la dernière optimisation si le logiciel arrête de

1. Astuce : Les modèles Simulink des versions R2009a ne peuvent être ouverts sous Matlab R2016b. Par conséquent, par le biais d'une fenêtre Windows, le logiciel Matlab sous sa version R2009a est appelé à charger les modèles BO à partir de ces fichiers Simulink.

s'exécuter (coupure de courant, beugue de Matlab, etc). Le nombre d'appel à `systune` est compté afin de ne pas entrer dans une boucle infinie, si la synthèse échoue. La dernière activité consiste en une vérification rapide de la validité des critères de réussite, notamment les *soft* et *hard goals* obtenus et si l'amplitude des gains n'est pas en dehors des limitations imposées.

Le second diagramme (b) développe la structure de la sous-activité "Préparer les entrées pour la synthèse H_∞ " du diagramme a). L'ensemble des étapes de b) est suivie afin d'obtenir tous les éléments (modèles et variables) nécessaires à la synthèse. Ces entrées obtenues sont finalement sauvegardées pour permettre à l'utilisateur de connaître le point de départ de la synthèse et de pouvoir la relancer si les résultats obtenus sont intéressants.

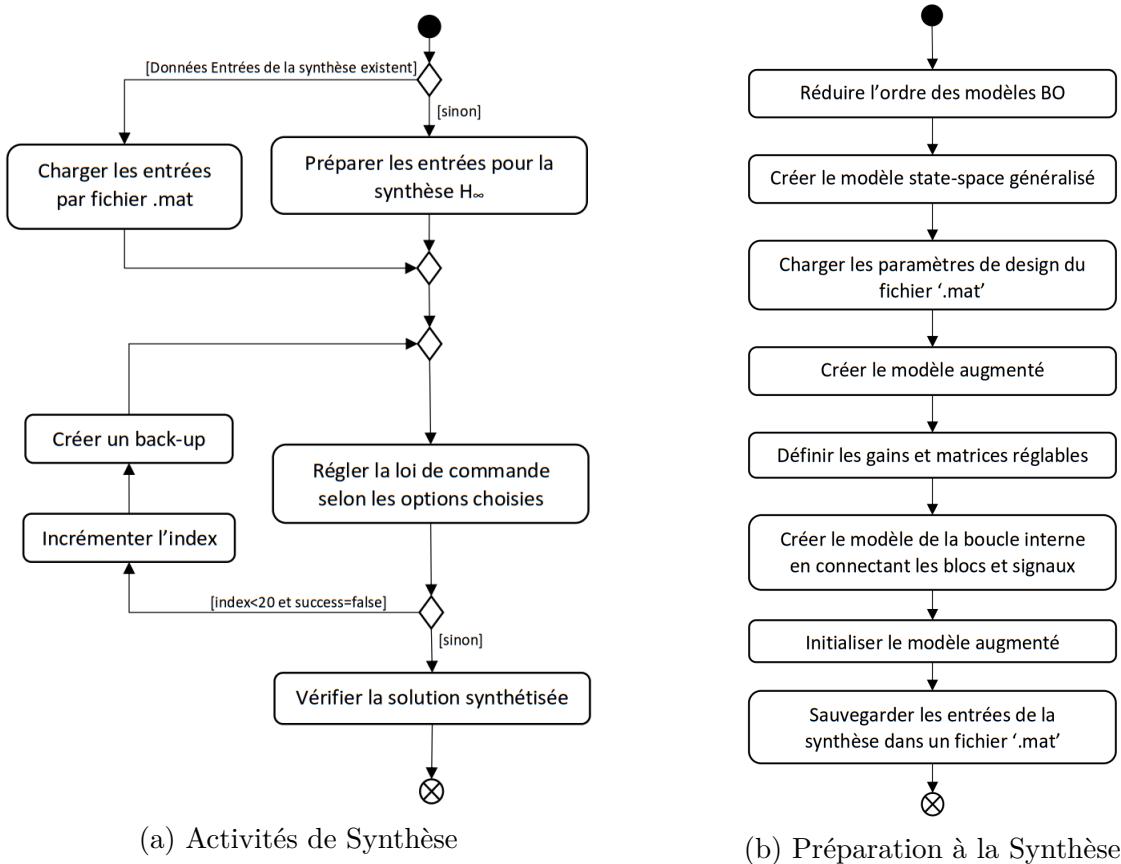


Figure 4.4 Diagrammes d'activités du logiciel

4.3.6 Problèmes rencontrés et leurs solutions

Durant la conception du logiciel, de nombreux problèmes ont été soulevés. Les principaux challenges liés à la méthode sont listés ci après :

Modèles manquants dans l'enveloppe de vol

Contexte : Les modèles sont stockés dans une variable de type `genss` (*generalised state-space*), de dimension 3. Les modèles sont alors rangés dans la cellule correspondant à la bonne configuration de charge LD, altitude ALT et vitesse CAS. Lors de la déclaration des conditions de vol à considérer, l'utilisateur définit un ensemble de valeurs pour les trois conditions précédentes. Il définit donc n_{LD} cas de configuration de charge (*Load*), n_{ALT} altitudes et n_{CAS} vitesses d'équilibres. Dans le meilleur des cas, un modèle linéaire existe pour chacune des possibilités, il y a donc $n_{LD} \times n_{ALT} \times n_{CAS}$ modèles. Toutes les cellules du modèle généralisé sont donc remplies. Or, il est aussi possible que certains des modèles linéaires ne soient pas valides, par conséquent pour des combinaisons particulières des trois conditions (LD, ALT et CAS), il est possible que la cellule du modèle généralisé soit vide.

Problème : La fonction `systune` n'accepte pas que des cellules du modèle d'état généralisé soient vides.

Solution : Les cellules vides sont comblées par des modèles existants. Pour une cellule c , telle que $c = (i_{LD}, j_{ALT}, k_{CAS})$, on définira le modèle de défaut $(i_{LD,0}, j_{ALT,0}, k_{CAS,0})$ tel que :

- $j_{ALT,0} = j_{ALT}$,
- $k_{CAS,0} = k_{CAS}$,
- $i_{LD,0} \in I_{LD}$ pour lesquels un modèle linéaire existe.

Bien évidemment dans la mesure du possible, un modèle de dynamique rigide uniquement sera remplacé par un autre modèle de même nature. Dans le cas où l'une ou l'autre des catégories de modèle est vide, le modèle de remplacement est de nature différente mais le changement est tout de même suivi pour le design de la synthèse, afin de ne pas chercher à contraindre des dynamiques flexibles comme celles rigides.

Données volumineuses pour la sauvegarde

Contexte : Il a été exposé dans le chapitre précédent que dans le cas des modèles incluant la dynamique flexible, l'ordre des modèles linéaires est alors très grand. La nécessité d'en réduire l'ordre pour procéder à la synthèse des gains est alors inévitable. Cependant, pour vérifier

les performances réellement atteintes par le système, il est jugé intéressant de conserver les modèles complets et de les sauvegarder avant de les réduire. Ces modèles sont alors rassemblés dans des variables et stockés sur le disque de l'espace de travail. Pour les applications à une enveloppe de vol trop large, le nombre de modèles regroupés dans ces variables peut être suffisamment important pour causer des difficultés lors de la sauvegarde des variables.

Problème : Les variables sont trop volumineuses pour être sauvegardées.

Solutions : La première solution est l'utilisation d'une option lors de la sauvegarde, avec la commande : `load('fichier.mat', '-v7.3')`. L'option '`-v7.3`', permet de sauvegarder des fichiers de taille supérieure à 2GB avec une compression des données. Cette solution, qui débloque les limitations de Matlab ne suffit cependant pas à régler totalement le problème car pour les plus grosses synthèses, la sauvegarde est alors empêchée par les capacités physiques de l'ordinateur.

La solution suivante face à l'insuffisance d'espace de calcul dans l'ordinateur pourrait être résolue en modifiant les réglages internes de celui-ci. Cependant, il est peu judicieux de pousser les capacités de la machine pour permettre de tels calculs et stockage de données. On ne veut pas que la puissance nécessaire à l'exécution soit une limitation de l'outil. En effet, il doit pouvoir être exécuté sur les ordinateurs des membres de l'équipe CLaws. Une telle demande de puissance n'est pas réaliste. Il est donc préférable de trouver une solution tangeante à celle-ci.

L'ultime solution est alors de diviser les variables, en d'autres de plus petite taille à stocker sur le disque dur, libérant ainsi la mémoire vive de Matlab pour sauvegarder les données de grande taille sur le réseau BA. Actuellement, la division adoptée par l'équipe des CLaws est le stockage des modèles selon la configuration de charge. Il est décidé de ne pas changer cette répartition.

4.3.7 Codage

La tâche d'écriture des fonctions pures n'apporte pas d'éléments jugés constructifs dans cette étude. Elle ne sera donc pas détaillée d'avantage.

Il est seulement précisé que l'implémentation a été réalisé en suivant les règles de codage usuelles et en combinant la nomenclature recommandée par Matlab et celle officielle de BA. Toutes les fonctions sont commentées, structurées et claires pour faciliter la navigation dans le logiciel de toute personne extérieure.

4.4 Tests et validation

Une série de tests sont réalisés afin de vérifier que l'ensemble des requis sont correctement adressés. Des cas fructueux comme des cas d'échec de la synthèse sont simulés pour vérifier un maximum de scénarios d'exécution. Il est nécessaire que tous les requis soient couverts par au moins un test. Une matrice de traçabilité est utilisée pour s'en assurer. La matrice est un tableau à double entrées, les tests d'une part et les requis de l'autre, qui permet d'associer chaque requis aux tests qui le valident. Tous les tests une fois effectués sont valides ce qui confirme la réalisation des objectifs et la prise en compte des contraintes imposées.

Si l'on considère l'ensemble de l'enveloppe de vol, l'exécution du logiciel prend environ une heure pour des modèles à dynamique rigide uniquement. Dans les cas de modèles rigides et flexibles, une synthèse sur un sous-ensemble restreint de l'enveloppe de vol nécessite environ deux heures (inclut aussi l'obtention des modèles). Les performances de l'outil en temps de calcul sont donc plutôt satisfaisantes, car le logiciel effectue plus de travail (réglage de tous les points de vol, et assurance de certaines performances) que l'outil actuel de l'équipe en moins de temps que l'outil simple de réglage.

Un script supplémentaire est implémenté hors du logiciel pour effectuer une vérification des performances réelles des gains synthétisés.

La synthèse du chapitre 5 a été faite par l'outil. Cette application est vue comme un test ultime validant l'outil.

Les synthèses flexibles des chapitres 6 et 7 ne sont pas produites par l'outil. Ces applications toujours en phase expérimentale ne seront pas implémentées prochainement par l'équipe CLaws. Cependant, grâce à la structure du logiciel, quelques heures de travail devraient suffir pour augmenter les fonctions concernées.

Pour ces deux expériences des scripts de synthèse et de vérification des performances sont créés.

4.5 Documentation

Dans le but d'aider à la possible extension et amélioration des capacités de l'outil, un guide d'utilisateur et une documentation pour développeurs sont rédigés. Le guide d'utilisateur spécifie les entrées et sorties du logiciel, donne des conseils, méthodes et exemples de synthèse H_∞ structurée. La documentation orientée développeurs détaille l'architecture du logiciel. On

y retrouve toutes les fonctions utilisées avec des précisions sur les entrées/sorties/procédure associées ainsi que les fonctions appelées et celles qui l'appellent. On utilisera l'outil M2HTML (Flandin, consulté le 23.11.18) qui génère automatiquement une documentation Hyper Text Markup Language (HTML) à partir des fichiers .m du logiciel. Avec ces deux manuels, la manipulation du logiciel est facilitée ; l'outil pourra alors être utilisé et entretenu par l'équipe.

4.6 Conclusion

L'outil implémenté est valide et satisfait les critères imposés. Le temps de calcul nécessaire à l'obtention de modèles et à la synthèse de contrôleurs performants est relativement rapide comparé à ce qui existe. Il est rappelé que l'intérêt majeur de cette méthode et outil est la possibilité d'obtenir les lois de commandes pour toute l'enveloppe de vol en une seule itération.

Il est néanmoins concevable d'améliorer davantage la rapidité, notamment en optimisant les calculs effectués (du ressort d'un spécialiste en génie informatique et/ou logiciel). Les améliorations possibles concernant le logiciel en soit, portent sur l'augmentation du design à la loi longitudinale de l'avion ou encore à la loi directe.

CHAPITRE 5 SYNTHÈSE D'UN CONTRÔLEUR ROBUSTE ET AUTO-SÉQUENCÉ POUR DES MODÈLES D'AVION RIGIDE

Ce cinquième chapitre présente la première application de la synthèse H_∞ structurée dans le contexte de BA. La section 5.1 présente les objectifs de l'application et la méthodologie suivie pour le design de la loi de commande. La section 5.2 suivante liste l'ensemble des requis imposés sur les performances de la boucle interne, mais aussi les différents concepts utilisés pour construire le modèle augmenté de l'application. L'ensemble des cas de design utilisés sont définis dans la section 5.3. Dans la section 5.4, il est ensuite spécifié comment concevoir une condition initiale au problème. Les itérations de design ainsi que les résultats et les graphiques de performances obtenus pour la synthèse linéaire sont fournis dans les sections 5.5 et 5.6. Finalement la conclusion sur l'expérience est apportée en section 5.7.

5.1 Introduction

5.1.1 Contexte et activités précédentes

La synthèse présentée dans ce chapitre manipule les modèles de l'avion exposés dans le chapitre 2, applique les concepts définis dans le chapitre 3 et règle la loi de commande à l'aide de l'outil développé dans le chapitre 4. À travers cette première application, on démontre les capacités de l'outil. La traduction des concepts dans l'environnement de BA est testée sur un exemple concret de réglage. La synthèse est effectuée sur la totalité de l'enveloppe de vol pour un avion en configuration d'atterrissement.

5.1.2 Objectifs de l'application

Afin de démontrer le potentiel de la technique et de l'outil, on voudra obtenir une solution performante rapidement. En effet, l'outil devra proposer une solution dans un temps relativement court comparativement au temps actuellement nécessaire à l'outil existant pour effectuer les mêmes réglages. Ce temps est estimé à au moins 2 heures pour linéariser et régler les gains point à point sur l'ensemble de l'enveloppe de vol.

Le deuxième intérêt est la performance de la solution obtenue. Il sera difficile de synthétiser une solution meilleure dans tous les critères évalués. Il existera des canaux plus ou moins optimisés. On estimera la qualité de la réponse proposée à partir de l'ensemble des critères. Si la plupart des résultats sont meilleurs (respectivement moins bons), la solution générale proposée par la synthèse H_∞ sera jugée plus performante (respectivement moins performante).

Par conséquent, une solution peut aussi être considérée comme équivalente au réglage actuel si aucune des deux ne se démarque particulièrement.

5.1.3 Méthodologie suivie

Les grandes lignes de la méthode définies dans la section 1.2.2 du chapitre d'introduction, imposent les étapes suivantes :

- Définition des requis (section 5.2.1)
- Définition des concepts (section 5.2.2)
- Définition de l'architecture (section 5.2.3)
- Implémentation et optimisations (sections 5.3 et 5.4)
- Tests et vérifications (section 5.5)

Optimisation du design

La méthode utilisée pour l'optimisation du design est une méthode itérative. À partir des résultats obtenus, un critère de performance particulier est optimisé, puis le second, puis le troisième et ainsi de suite jusqu'à ce que les performances soient satisfaisantes. Bien entendu, un critère peut être optimisé plusieurs fois si le changement des paramètres de design d'un autre canal dégrade les performances atteintes précédemment. La solution obtenue par l'algorithme dépend des paramètres de design mais aussi de la condition initiale comme mentionné dans les chapitres 1 et 3). Par conséquent les deux éléments sont amenés à subir plusieurs itérations de design.

5.2 Requis et design

5.2.1 Requis sur les performances de la boucle interne

D'après la méthode “Cycle en V” suivie, il est nécessaire de poser les requis à valider. Ces contraintes permettent de définir sous quelles conditions une solution est performante ou non. Les critères sur les performances désirées sont listées ici :

- Requis 1 : voir 3.3.1.
- Requis 2 : voir 3.3.1.
- Requis 3 : voir 3.3.1.
- Requis 4 : voir 3.3.2. Le requis 4 ne concerne que les sous-requis Req4b et Req4c.
- Requis 5 : voir 3.3.2. Le requis 5 ne concerne que les sous-requis Req5a et Req5c.
- Requis 6 : voir 3.3.3.
- Requis 8 : voir 3.3.4.

— Requis 10 : voir 3.3.5.

5.2.2 Concepts et canaux de performances

Afin d'assurer les requis listés plus haut, plusieurs canaux de performances sont à construire, notamment des fonctions de pondérations mais aussi des points d'analyse. La série de concepts choisis sont définis ci-après :

1. Concept 1 (associé au requis 1) : voir 3.3.1. On considérera la structure simplifiée au niveau de l'entrée et de la commande feedforward. Les dynamiques des filtres ASE sont incluses dans les modèles d'avion en BO (on prendra les sorties correspondantes).
2. Concept 2 (associé au requis 2) : voir 3.3.1. On ne modifiera ni les filtres ASE, ni le gain K_p .
3. Concept 3 (associé au requis 3) : voir 3.3.1.
4. Concept 4 (associé au requis 4) : voir 3.3.2. On construit seulement deux canaux de performances avec modèles de référence.
5. Concept 5 (associé au requis 5) : voir 3.3.2. On construit seulement deux canaux de performances.
6. Concept 6 (associé au requis 6) : voir 3.3.3.
7. Concept 8 (associé au requis 8) : voir 3.3.4. On contrôlera la position des pôles avec le suivi de références et avec l'outil *TuningGoal.Poles*.
8. Concept 10 (associé au requis 10) : voir 3.3.5.

On rappelle que les canaux de performances et leurs architectures, sont abordés dans la section 3.3 du chapitre 3. Pour réduire la taille et augmenter la lisibilité du schéma du modèle augmenté de la synthèse (Fig. 5.1), on ne présente pas tous les canaux de performances.

5.2.3 Architecture du modèle augmenté

La figure 5.1 présente le schéma du modèle augmenté pour la synthèse sur la configuration d'atterrissement complète dans le cas rigide. On rappelle que les transferts pour l'étude des marges de stabilité au niveau de la commande en aileron et en rudder, ainsi que les transferts de limitations d'amplitude des gains de feedback ne sont pas représentées. Le placement des pôles et la délimitation de zones performantes dans le plan complexe sont les deux dernières contraintes ajoutées.

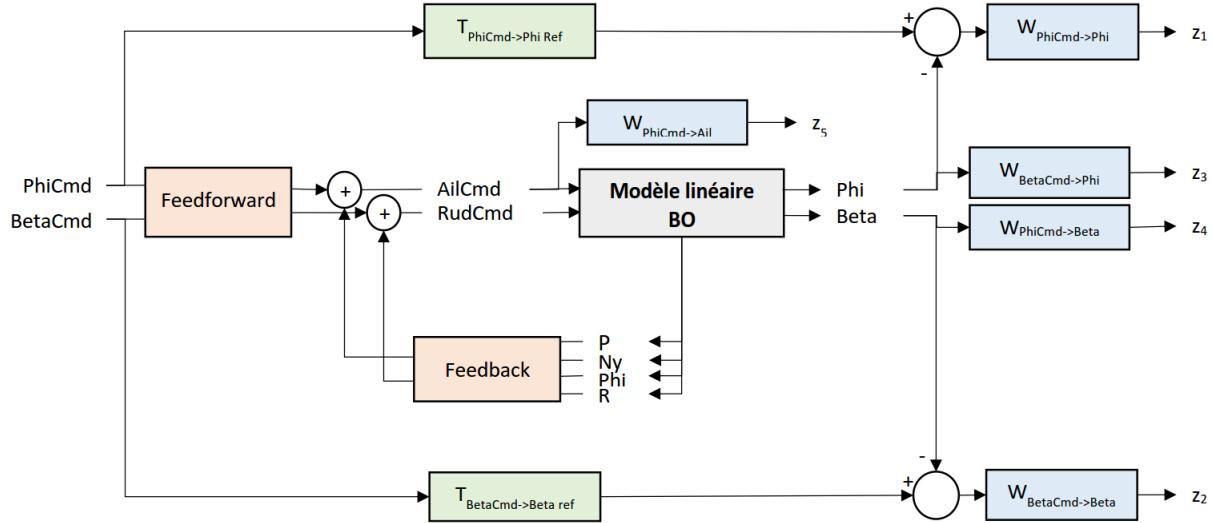


Figure 5.1 Plan augmenté de la synthèse rigide

5.3 Explication des cas utilisés pour l'application

Comme dit plus haut, la synthèse est appliquée sur les cas de vol en configuration d'atterrissement. L'objectif est ici de couvrir l'ensemble des conditions de vol possibles pour cette configuration. Tous les modèles disponibles ne seront pas utilisés car pour une cinquantaine de configuration de charge et de conditions de vol, nous aurions 2500 modèles à prendre en compte dans la synthèse ce qui ralentirait très fortement l'optimisation. On choisit alors des points suffisamment représentatifs de l'ensemble des cas possibles mais en nombre raisonnable. Les cas choisis sont alors présentés dans les figures 5.3 et 5.2, où on peut compter 13 configurations différentes en charge et 30 conditions de vol.

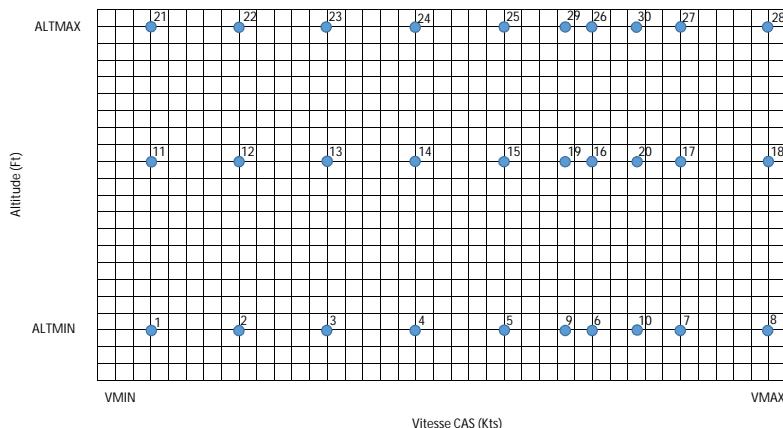


Figure 5.2 Position des conditions de vol sélectionnées dans leur ensemble

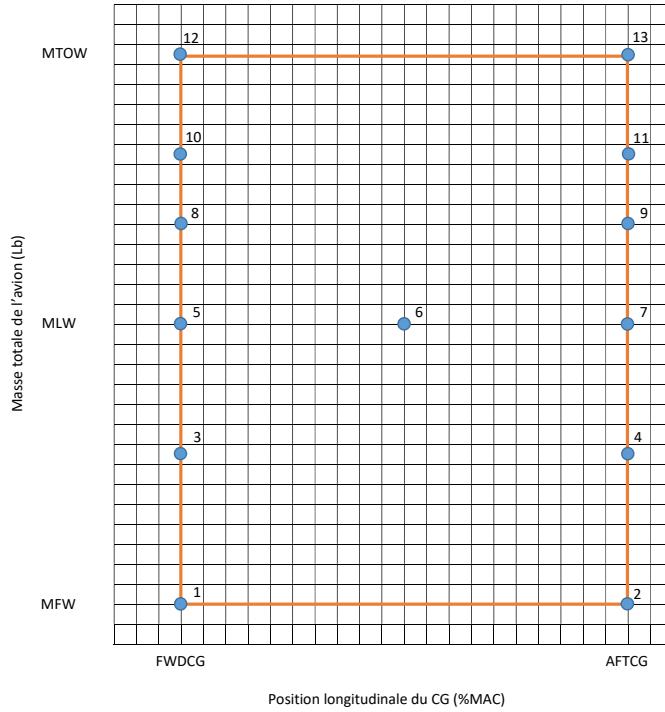


Figure 5.3 Position des configurations de charge sélectionnées dans leur ensemble

5.4 Détermination de la condition initiale

Le succès de la synthèse dépend des canaux de performances construits et de leurs gains de pondération mais aussi de la condition initiale choisie. En effet, l'importance de partir d'une bonne condition initiale provient de la nature non convexe du problème (Section 3.2). Le processus utilisé pour la réalisation de cette condition initiale comprend deux étapes : l'optimisation point à point et l'interpolation en un polynôme solution.

5.4.1 Optimisation point-à-point

Comme mentionné aussi à la section 3.4, la condition initiale choisie influence le résultat de la synthèse. En partant d'une condition initiale optimisée, les chances d'arriver à la meilleure des solutions possibles sont augmentées, mais ne sont pas assurées. Pour obtenir une bonne condition initiale, il s'agit premièrement de trouver les solutions les plus optimales. En l'occurrence, un par un, tous les points de vol sont étudiés et pour chacun, une combinaison de gains performants est synthétisée.

Cependant, il est aussi mentionné que les objectifs de performances (donc les canaux de performances), influencent le résultat de la synthèse. Il est alors nécessaire de fixer tous ces paramètres lors du processus de détermination de la condition initiale.

La condition initiale est par conséquent optimisée pour un design particulier. Si les objectifs ou le design changent, il est conseillé de construire une nouvelle condition initiale, plus proche du nouveau point optimum recherché.

Les figures 5.4 et 5.5 représentent les valeurs des gains obtenues pour les trois altitudes et les dix vitesses de design. On remarque que pour certains gains comme le gain K_{13} , une évolution générale est suivie quand pour d'autres gains, cette tendance générale est moins lisse (voir le gain K_{22}). Le point suivant (sous-section 5.4.2) a justement pour but de lisser ces courbes en cherchant un polynôme de séquencement, quitte à s'éloigner de la solution optimale.

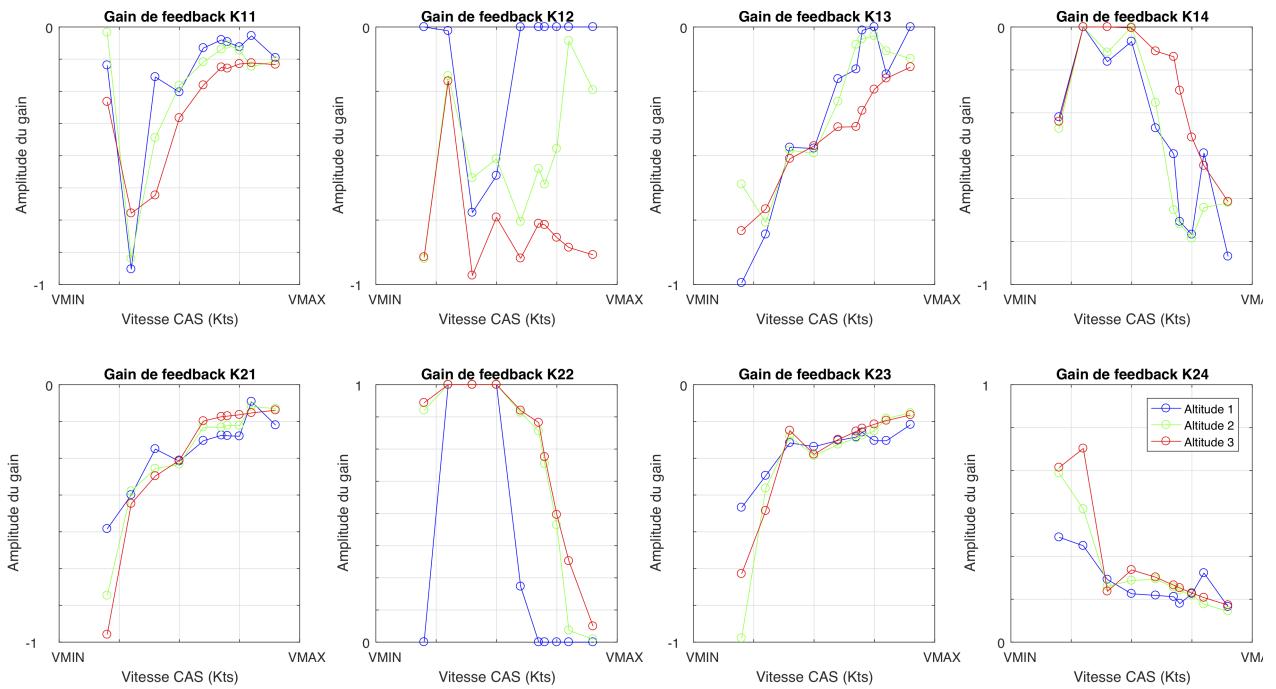


Figure 5.4 Évolution des gains de feedback selon l'enveloppe de vol après une synthèse point à point

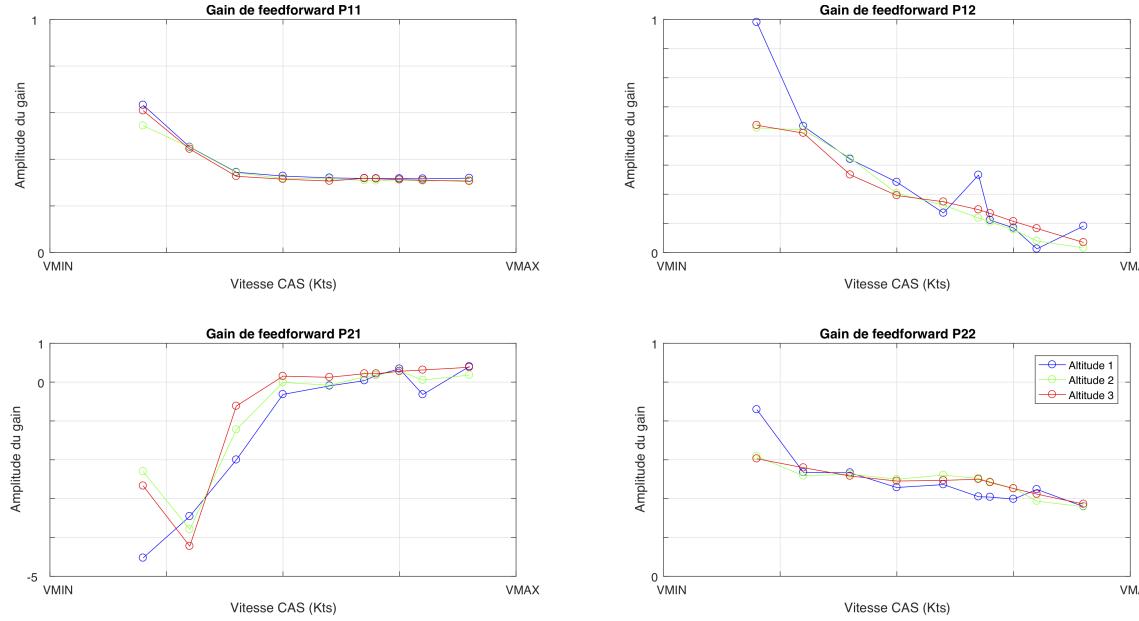


Figure 5.5 Évolution des gains de feedforward selon l'enveloppe de vol après une synthèse point à point

5.4.2 Interpolation polynomiale des gains

Avec ce point de départ alors obtenu, les contraintes de performances sont satisfaites mais pas le critère de séquencement des gains, et d'évolution douce sur l'enveloppe de vol. Il est alors nécessaire de procéder à l'étape suivante, c'est-à-dire interpoler un polynôme de séquencement à partir des gains précédents pour obtenir la condition initiale optimisée.

Le séquencement est effectué en fonction de l'altitude et de la vitesse, comme le montre la figure 5.6. Comme il apparaissait déjà dans les graphiques 5.4 et 5.5 (disponible en pleine page en Annexe E), l'altitude ALT de vol influence moins la valeur des gains que la vitesse CAS. Il est possible de se contenter d'un ordre 1 pour le séquencement de l'altitude mais il est nécessaire de choisir un ordre 3 pour le séquencement de la vitesse. L'ordre 3 est choisi car la différence de performances obtenue est suffisamment élevée pour justifier l'augmentation de l'ordre du polynôme de 2 à 3, tandis que l'amélioration obtenue lors du passage à l'ordre 4 n'est pas significative. Le polynôme de séquencement est alors :

$$G = \alpha_1 + \alpha_2 \cdot CAS + \alpha_3 \cdot ALT + \alpha_4 \cdot CAS^2 + \alpha_5 \cdot CAS \cdot ALT + \alpha_6 \cdot CAS^3 + \alpha_7 \cdot CAS^2 \cdot ALT \quad (5.1)$$

où G représente un gain des matrices de feedback ou de feedforward, et les α_i sont les coefficients de séquencement.

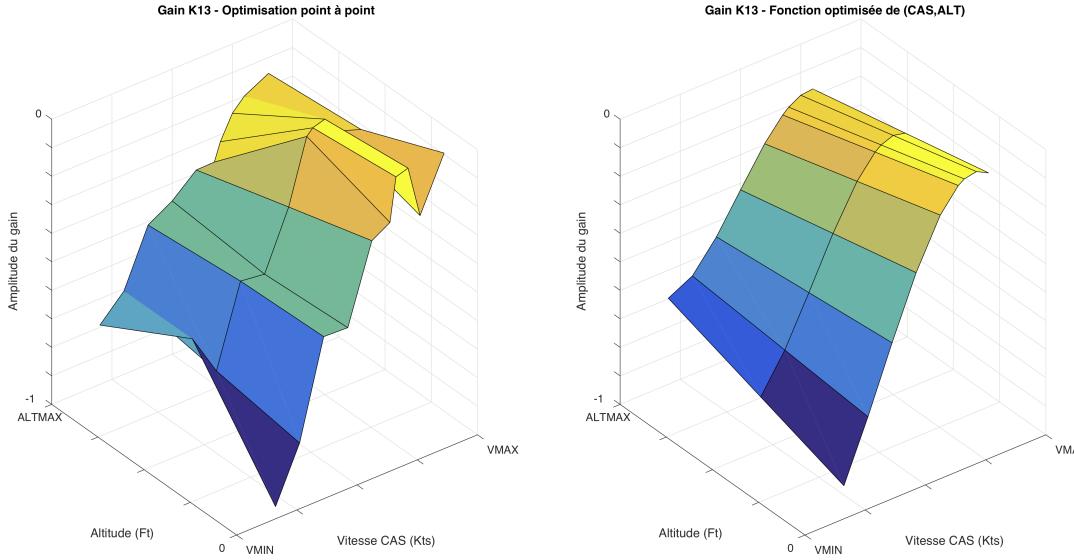


Figure 5.6 Évolution du gain de feedback K13 selon l'enveloppe de vol après une synthèse point à point (gauche) et après interpolation polynomiale (droite)

5.5 Synthèse et optimisations

Afin d'obtenir la solution présentée plus loin, de nombreuses itérations de réglages ont été appliquées. Les modifications les plus importantes dans le design sont détaillées dans cette section.

5.5.1 Exclusions d'une partie des conditions de vol

Comme défini dans la section 2.3.3, il existe des conditions de vol dont la vitesse est dépendante du chargement. Ces cas peuvent avoir une vitesse CAS variable de plusieurs dizaines de noeuds ce qui donne des modèles très différents. Alors que la synthèse cherche une combinaison de gains uniques pour assurer la robustesse sur une condition de vol à vitesse fixe (en pratique un large intervalle de vitesse) et une charge variable, ces modèles trop différents ajoutent des contraintes de robustesse trop importantes et viennent fortement dégrader la solution produite.

Toutes les conditions de vol dont les vitesses sont fonction de V_{sr} ne sont pas utilisées pour le design. On garde cependant les vitesses fonction de V_{fe} qui sont fixes sur une configuration donnée mais varient, par exemple, entre la configuration d'atterrissage et celle de croisière.

À la suite de la modification, l'énoncé “condition de vol donnée à altitude et vitesse fixe” est

vérifié pour toutes les conditions de vol de l'application. Dès lors, les résultats sont définitivement meilleurs car les compromis à faire sont moindres, pour obtenir une solution robuste en charge.

5.5.2 Catégorisation des modèles basses/hautes vitesses

Une fois, les modèles précédents supprimés, il est remarquable que le réglage des canaux de performances est plus ou moins adapté selon la partie de l'enveloppe de vol concernée. Étant donné que pour cette application l'ensemble de l'enveloppe de vol est considérée, l'intervalle de vitesse, d'altitude, et de charge est très grand. Afin d'avoir un réglage plus performant il est donc nécessaire de diviser ce grand ensemble en sous-ensembles qui répondront à des réglages de canaux de performances différents.

Lors de la recherche de catégorisation, il apparaît que certains cas de vol sont situés hors de l'enveloppe de certification. Par exemple, pour la configuration d'atterissage, dans le cas d'une charge élevée (c'est-à-dire un avion lourd), il n'est pas physiquement possible d'atterrir aux vitesses les plus basses. Ces cas sont jusqu'à présent considérés par la synthèse et intégrés au même titre que les autres cas de vols.

On récupère les requis exacts de certification pour délimiter précisément quels sont les cas concernés. Ces cas sont placés dans un groupe de design particulier. Comme la synthèse préfère obtenir un taux d'erreur maximal le plus petit possible, elle aura tendance à beaucoup dégrader certaines conditions de vol pour améliorer très peu d'autres si l'erreur maximale en est diminuée. Étant donné les cas extrêmes qui sont considérés, il n'est pas judicieux de laisser le design ainsi. Il est nécessaire de relaxer les requis imposés sur ces modèles extrêmes. On ne désire pas que les contraintes de performances soient absolument respectées mais il est tout de même attendu que le comportement ne soit pas trop éloigné des cas certifiés. Le fait de relaxer les contraintes permet de prioriser la performance des cas à certifier devant les autres.

Finalement, l'enveloppe de vol est divisée en 6 parties : certifiés (oui ou non) et masse (basse, moyenne, élevée). Le critère de la certification sépare les conditions de vol en fonction de leur vitesse. Le schéma qui représente l'emplacement de ces modèles dans l'enveloppe de vol est donné par la figure 5.7. Pour les 390 cas considérés par l'application, on y observe la répartition suivante :

- Catégorie masse haute et non certifiés : 36 cas.
- Catégorie masse haute et certifiés : 84 cas.
- Catégorie masse moyenne et non certifiés : 30 cas.

- Catégorie masse moyenne et certifiés : 120 cas.
- Catégorie masse basse et non certifiés : 12 cas.
- Catégorie masse basse et certifiés : 108 cas.

ALT 3		Loading Configuration												
		1	2	3	4	5	6	7	8	9	10	11	12	13
Flight Condition	1	ALT 2		Loading Configuration										
	2			1	2	3	4	5	6	7	8	9	10	11
	3	ALT 1		Loading Configuration										
	4			1	2	3	4	5	6	7	8	9	10	11
	5			1										
	6			2										
	7			3										
	8			4										
	9			5										
	10			6										
		Flight Condition		7										
		Flight Condition		8										
		Flight Condition		9										
		Flight Condition		10										

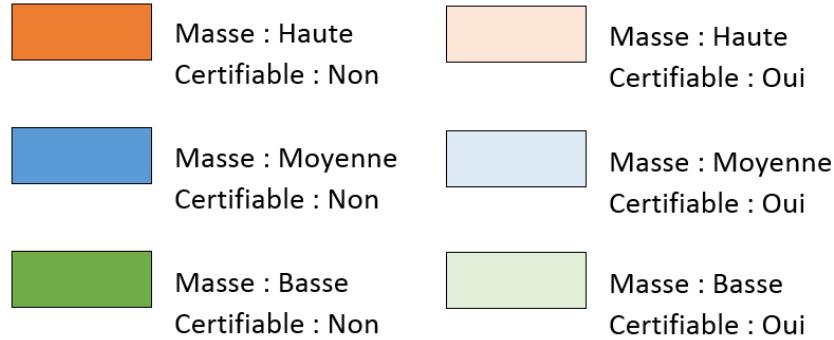


Figure 5.7 Schéma de la catégorisation de l'enveloppe de vol appliquée à la configuration d'atterrissage

Pour chaque canal de performance des concepts 1, 2, 3 et 6, il est créé 6 fonctions de pondérations pour le design particulier de chacune des catégories de l'enveloppe de vol. Le résultat obtenu est convaincant : les critères de performances sont bien plus respectés et ce, sur toute l'enveloppe. Les cas non certifiés sont dégradés mais leurs réponses restent tout à fait convenables.

Une catégorisation supplémentaire peut être appliquée (notamment en vitesses - basses, moyennes, hautes). Il appartient à l'ingénieur de décider de la performance voulue en fonction du temps associé au design et de la complexité du code à implémenter pour associer les bonnes fonctions de pondérations aux bons modèles.

5.5.3 Améliorations du design

Des dernières modifications sont finalement appliquées afin d'améliorer les performances de la solution.

1. Les fonctions de références pour le suivi de consigne sont renforcées. La constante de temps du transfert de ϕ précédemment à la limite de validité, comprend maintenant une marge de 15%. Pour le transfert de β , l'amortissement cible est augmenté de 40%, et la fréquence naturelle de 30%.
2. L'amortissement minimum exigé est aussi augmenté de 70% afin de stabiliser encore plus le système final.
3. L'ordre du polynôme d'interpolation passe de 2 à 3 (comme mentionné à la section 5.3).

5.6 Résultats et performances après réglage de la boucle linéaire

Dans les graphiques présentés ci-après : les courbes les plus claires représentent les modèles des points de vol qui appartiennent à l'enveloppe de certification (Section 5.5.2), tandis que les courbes bleues foncées sont les modèles hors des requis de certification. Pour apporter une analyse des graphiques, on utilise la notation :

- + : concernant le critère jugé, les nouvelles solutions sont meilleures que les anciennes.
- = : concernant le critère jugé, les nouvelles solutions sont équivalentes aux anciennes.
- - : concernant le critère jugé, les anciennes solutions sont meilleures que les nouvelles.

5.6.1 Evolution de l'amplitude des gains

Les quatre figures 5.8 à 5.11 permettent de comparer l'amplitude des nouveaux gains avec les anciens. Les deux premières figures présentent les extrema. On constate que les amplitudes minimales et maximales sont équivalentes pour les deux solutions. Les deux figures suivantes présentent l'évolution des gains dans l'enveloppe de vol. On remarque ici que l'allure des nouveaux gains de feedforward suivent l'allure des anciens. Les nouveaux gains de feedback

suivent plus ou moins ceux de la solution initiale. Il est normal que la tendance générale soit suivie car le design effectué est optimisé par les outils de CLaws pour satisfaire les performances que l'on cherche à atteindre. De plus, la solution de BA est utilisée comme le point de départ pour obtenir la condition initiale. Il est donc cohérent d'avoir un tracé proche de l'ancienne solution, avec des différences d'optimisation.

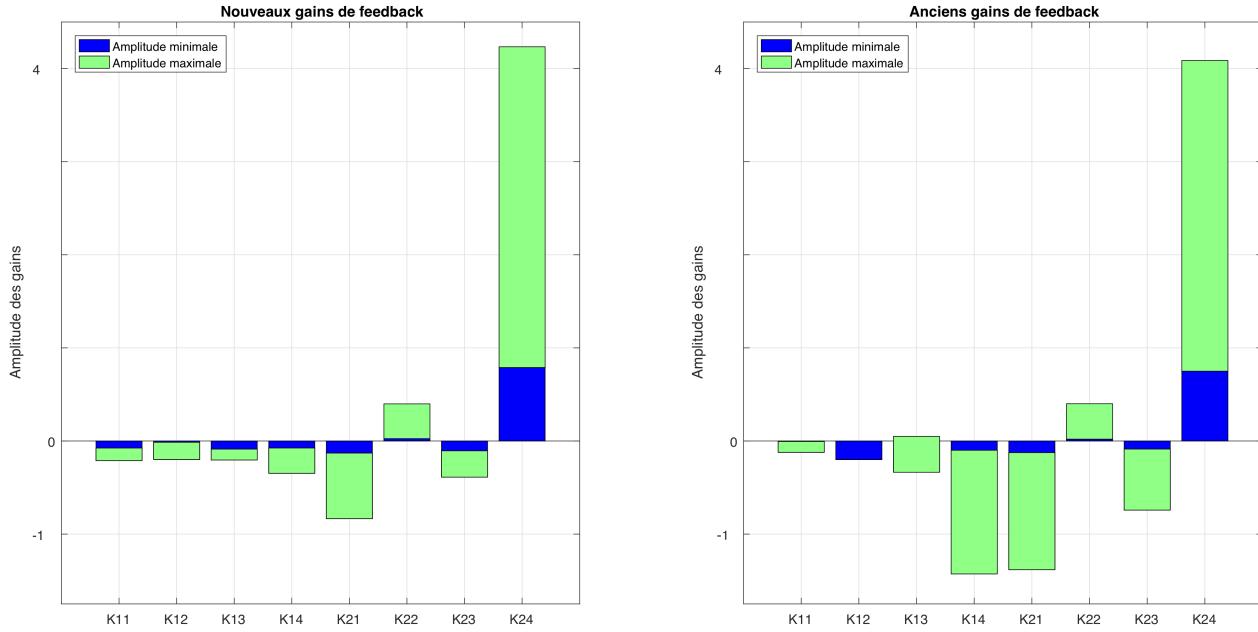


Figure 5.8 Extrémums des amplitudes des gains de feedback

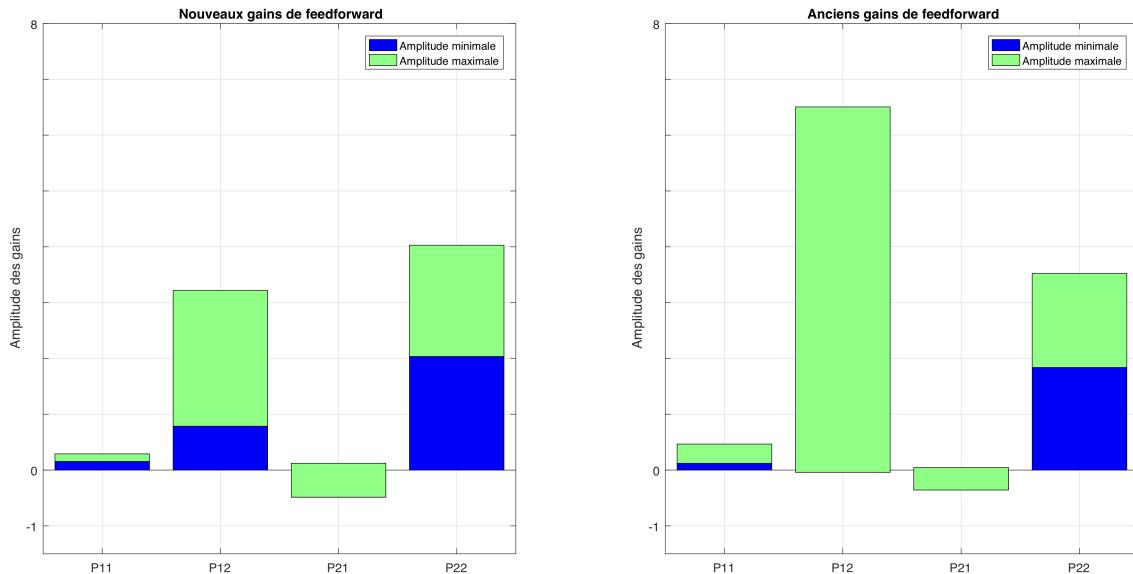


Figure 5.9 Extremums des amplitudes des gains de feedforward

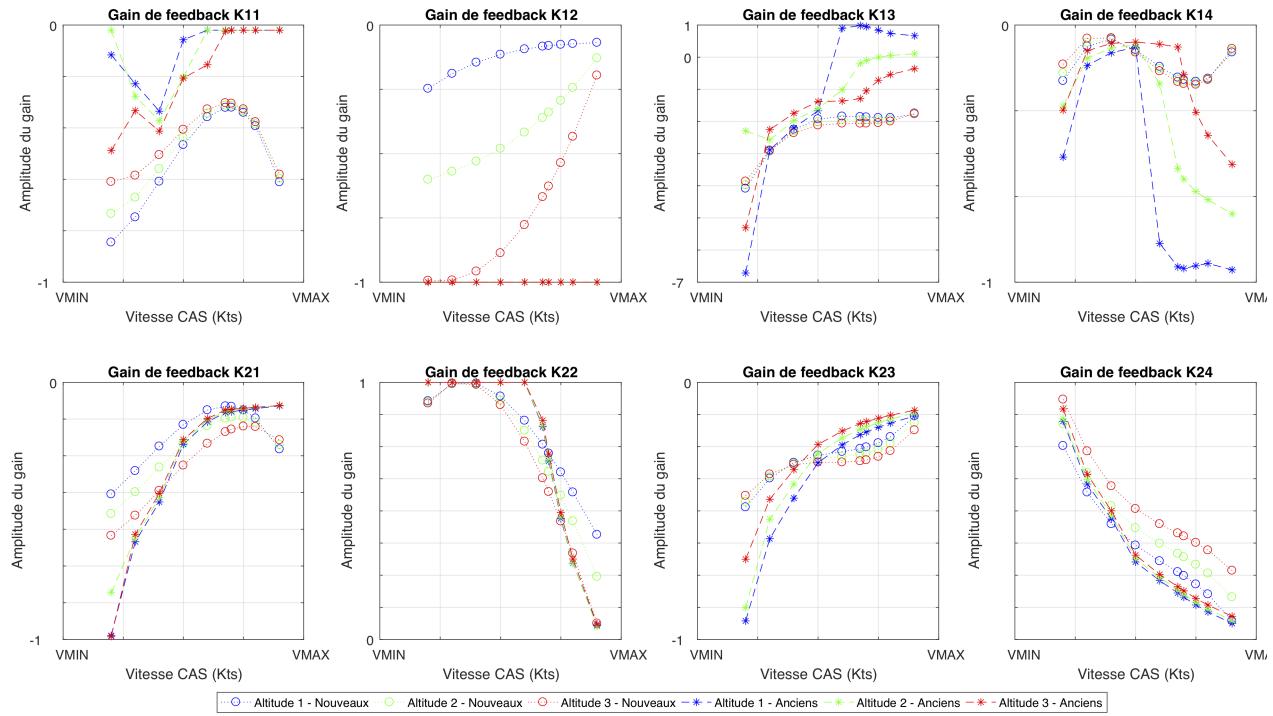


Figure 5.10 Évolution des gains de feedback sur l'enveloppe

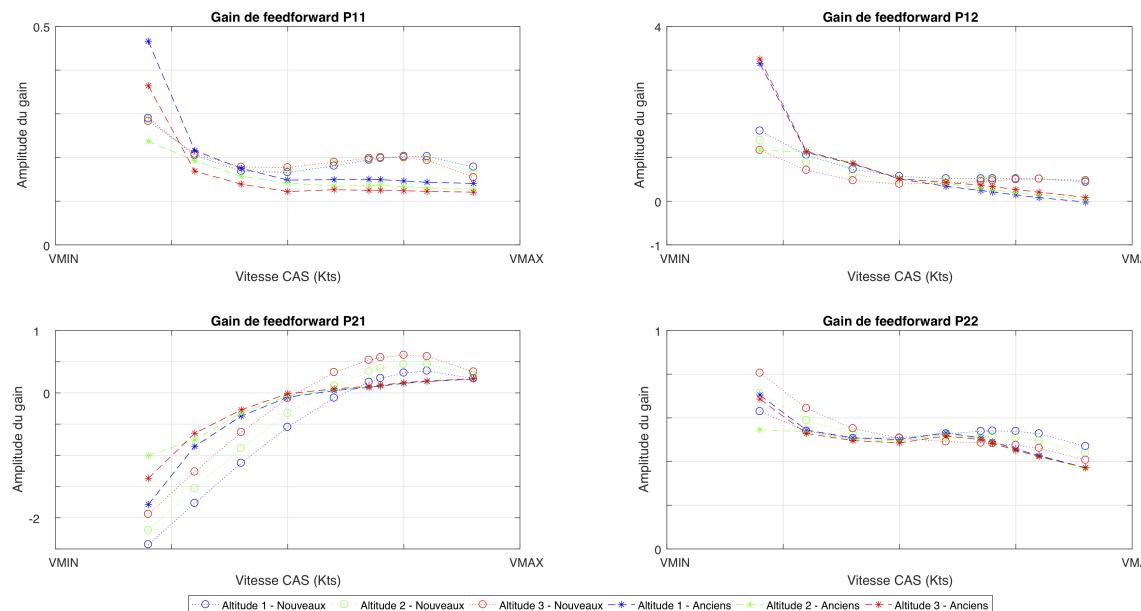


Figure 5.11 Évolution des gains de feedforward sur l'enveloppe

5.6.2 Suivi de références

Les figures 5.12 et 5.13 affichent les réponses temporelles des transferts de ϕ_{cmd} et β_{cmd} , respectivement vers ϕ et β . Les graphiques rendent compte de la performance obtenue en suivi de référence.

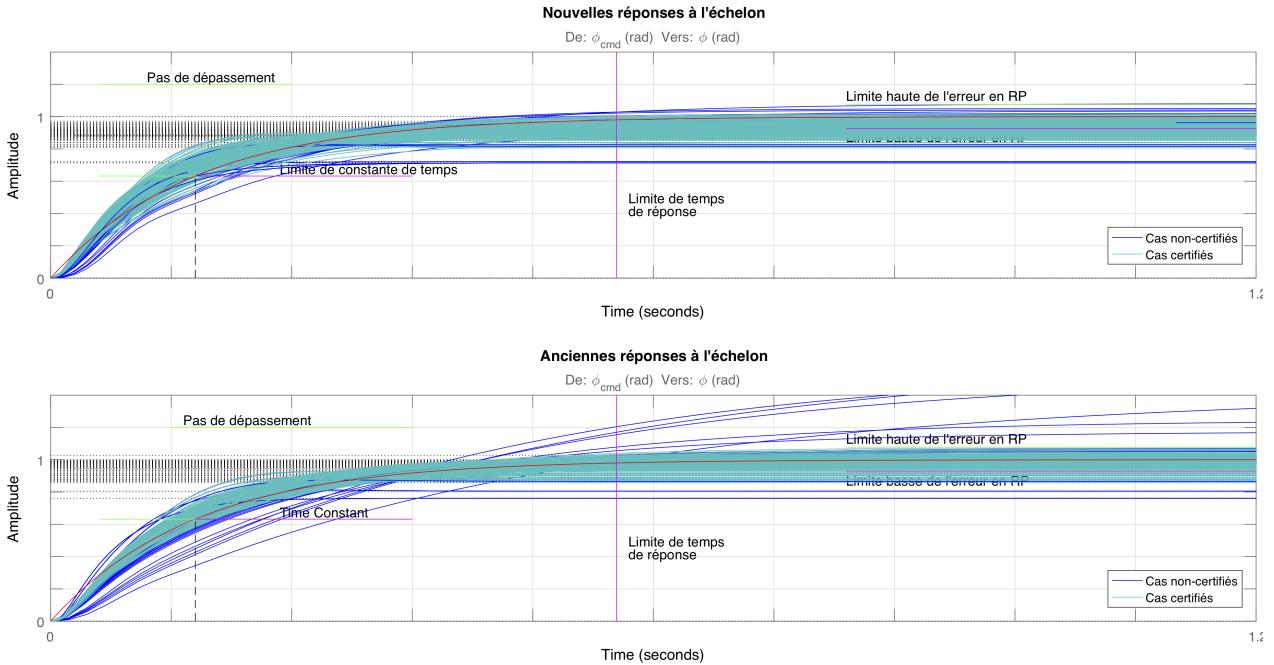


Figure 5.12 Performances du suivi de référence pour une commande de ϕ_{cmd} vers ϕ

Pour la figure 5.12, les commentaires sont les suivants :

- + : Les nouvelles constantes de temps (rapidité du système) sont légèrement meilleures. Les cas non certifiés (en bleu foncé) proposent des réponses temporelles bien meilleures après la nouvelle synthèse ; les réponses sont plus rapides avec une erreur bien plus faible.
- = : Les transferts suivent une réponse d'un système de premier ordre. Elles ont des erreurs en RP du même ordre de grandeur.
- - : Le RP des nouvelles réponses a une valeur finale moyenne non centrée autour de la référence.

Pour la figure 5.13, les commentaires sont les suivants :

- + : Le RP des nouvelles réponses a une valeur finale moyenne bien centrée autour de la référence. Les cas non certifiés proposent des réponses temporelles meilleures après la nouvelle synthèse, car l'intervalle d'erreur en RP est plus petit.
- = : Les réponses suivent une réponse d'un système de second ordre. Elles ont les

mêmes performances en terme de rapidité.

- $-$: Le RP des nouvelles réponses a un intervalle d'erreur plus grand que celui des réponses initiales.

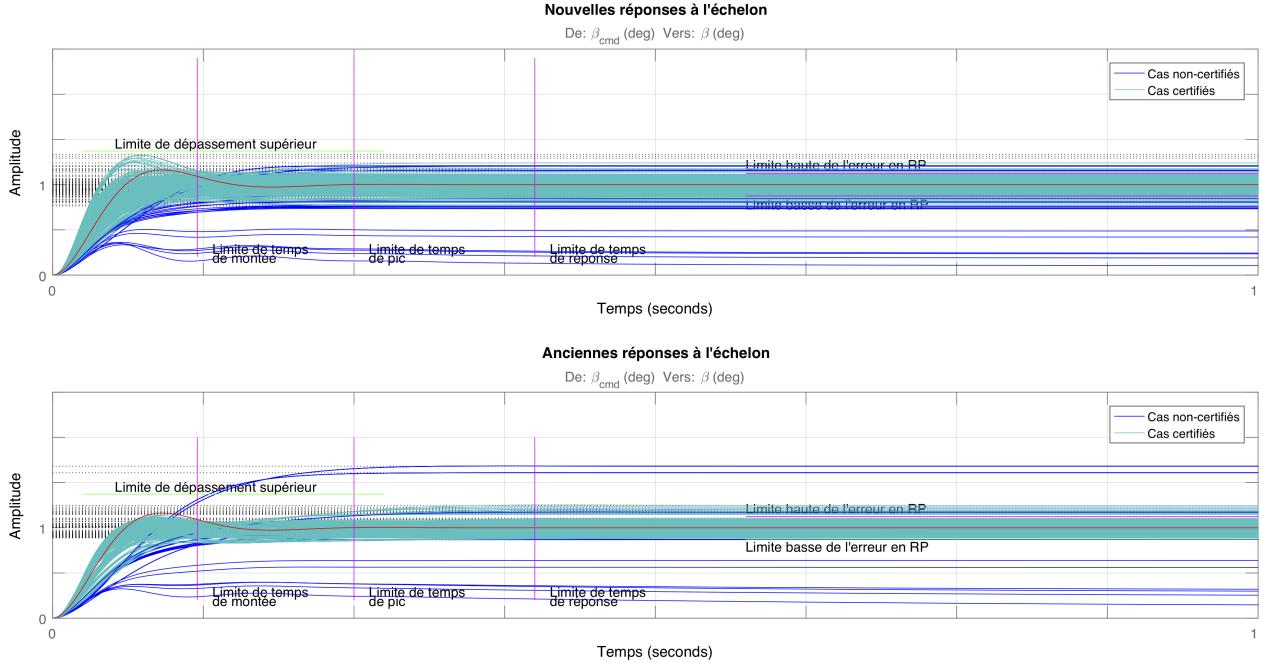


Figure 5.13 Performances du suivi de référence pour une commande de β_{cmd} vers β

5.6.3 Découplages de signaux

Les figures 5.14 et 5.15 affichent les réponses temporelles des transferts croisés de ϕ_{cmd} et β_{cmd} . Les graphiques rendent compte de la performance obtenue en découplage de ces signaux. Pour la figure 5.14, les commentaires sont les suivants :

- $+$: Les cas non certifiés proposent des réponses temporelles meilleures après la nouvelle synthèse, l'erreur en RP est plus faible.
- $-$: Les dépassements supérieurs sont légèrement plus grands pour les nouvelles réponses, et les dépassements inférieurs sont considérablement plus élevés. L'intervalle accueillant les erreurs en RP des nouvelles réponses est aussi plus grand pour les réponses de la nouvelle synthèse.

Pour la figure 5.15, les commentaires sont les suivants :

- $+$: L'intervalle accueillant les erreurs en RP des nouvelles réponses est un peu meilleur pour les réponses de la nouvelle synthèse. Les cas non certifiés proposent des réponses temporelles meilleures après la nouvelle synthèse, encore une fois selon l'erreur en RP.
- $=$: Les enveloppes des réponses sont équivalentes.

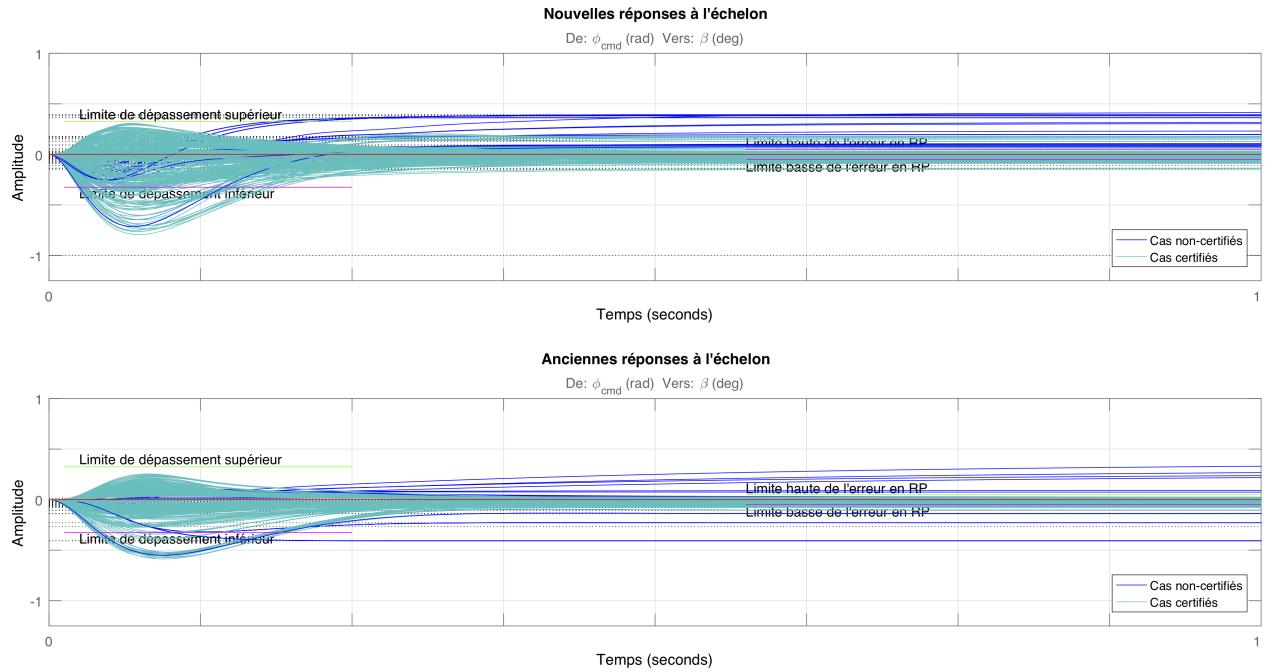


Figure 5.14 Performances du découplage de signaux pour une commande de ϕ_{cmd} vers β

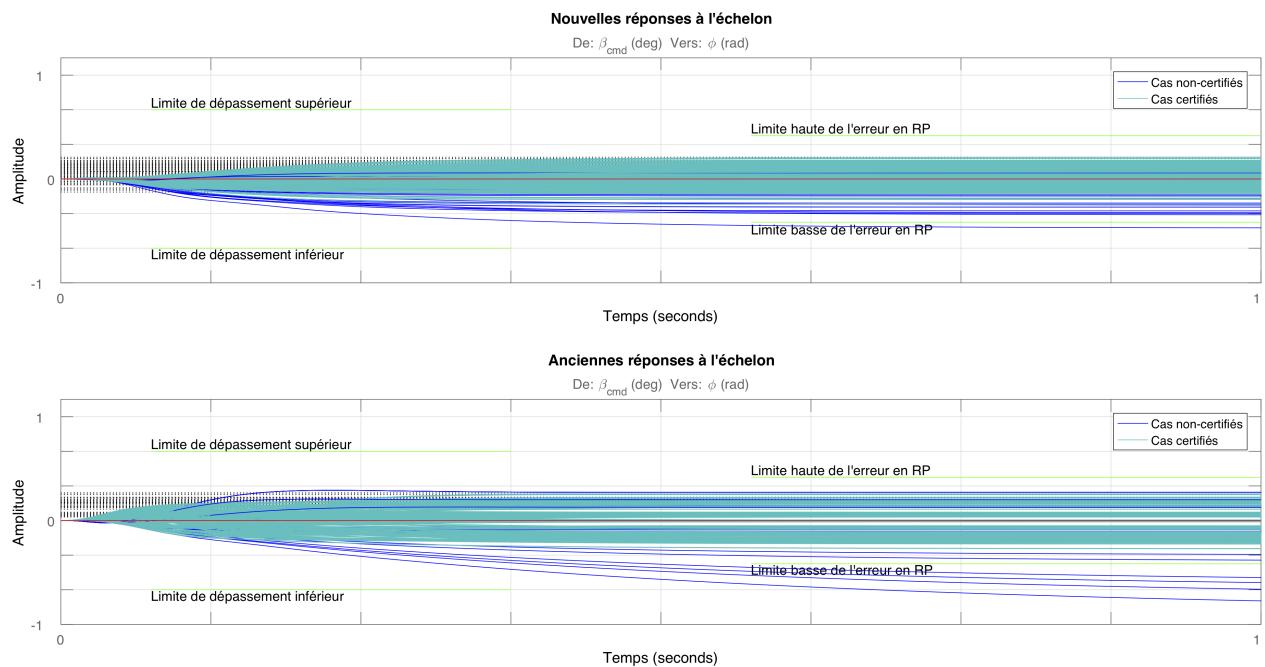


Figure 5.15 Performances du découplage de signaux pour une commande de β_{cmd} vers ϕ

5.6.4 Position des pôles de la boucle fermée

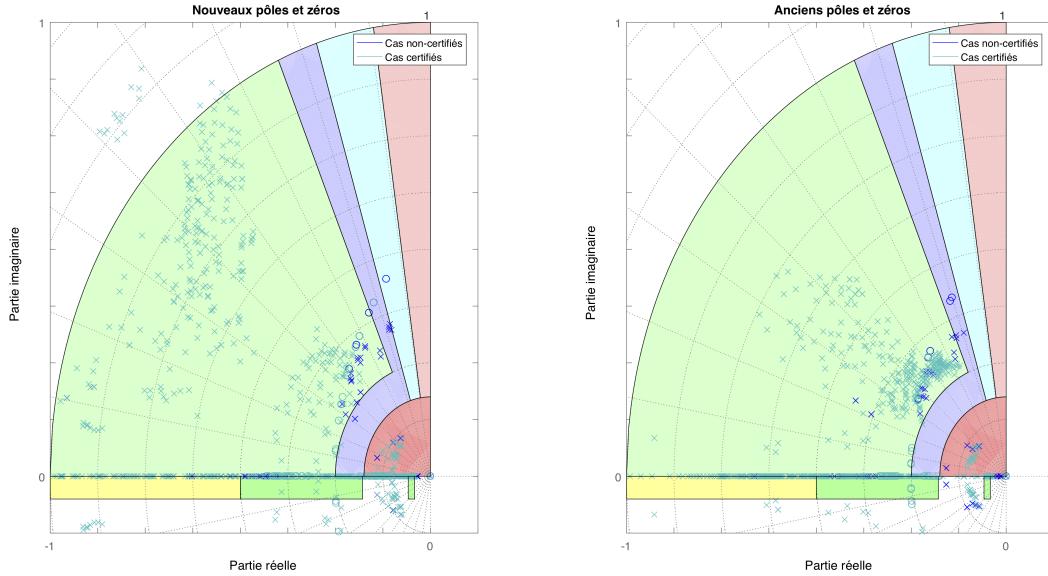


Figure 5.16 Positionnement des pôles et des zéros de la BF

La figure 5.16 présente le placement des pôles dominants de la nouvelle et de l'ancienne solution. Les pôles sont placés dans leur gabarit afin de mieux pouvoir juger. Pour cette figure, les commentaires sont les suivants :

- + : Le pôle réel le plus grand (en anglais : *minimum decay rate*) est plus petit pour les nouvelles réponses. Les cas non certifiés proposent de meilleurs placements de pôles, notamment pour le roulis hollandais.
- = : La plupart des pôles sont dans les zones de placement requises. Les deux solutions possèdent néanmoins quelques pôles mal placés (faible pulsation naturelle). Ce placement est accepté car les pôles ont un amortissement suffisamment important pour ne pas avoir d'influence.

5.6.5 Marges de stabilité

On donne les tracés des réponses fréquentielles des BO en Ail_{cmd} et Rud_{cmd} dans les figures 5.17 et 5.18. Elles renseignent sur le respect des marges de stabilité. Pour la figure 5.17, les commentaires sont les suivants :

- = : Toutes les marges de stabilité sont respectées.

Pour la figure 5.18, les commentaires sont les suivants :

- = : Toutes les marges de stabilité sont respectées. Les réponses sont même optimisées, car tangentes aux ellipses limites.

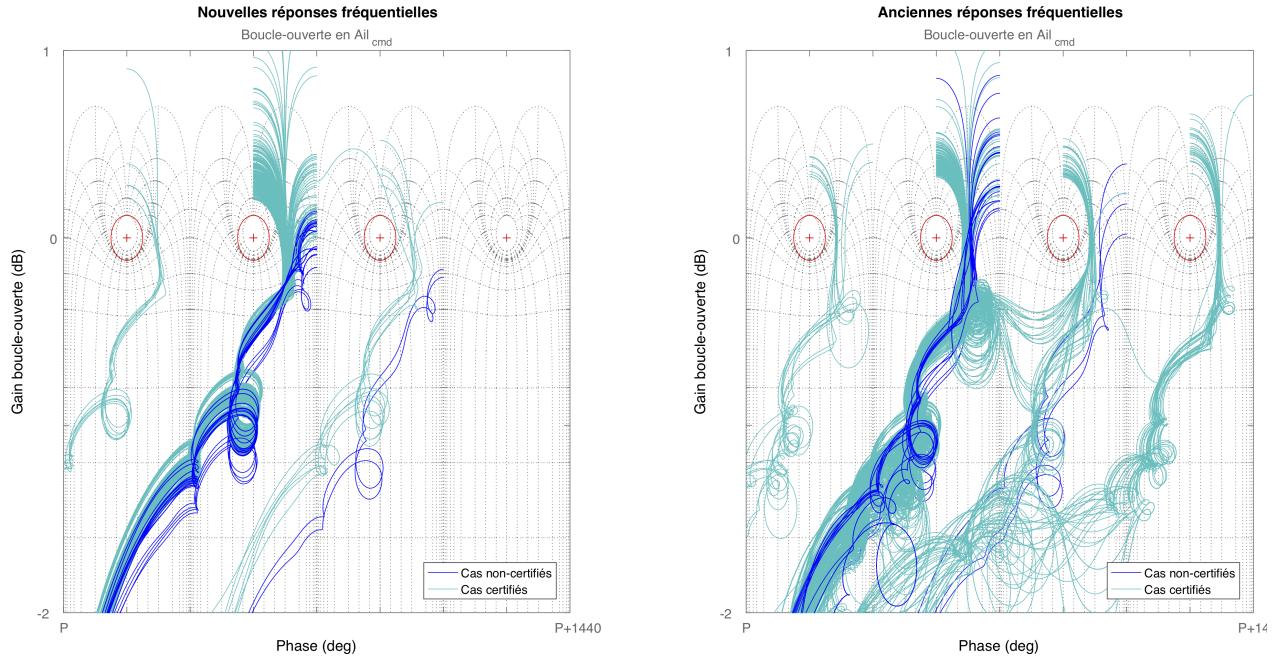


Figure 5.17 Réponses fréquentielles et diagramme de Nichols de la BO de l'aileron

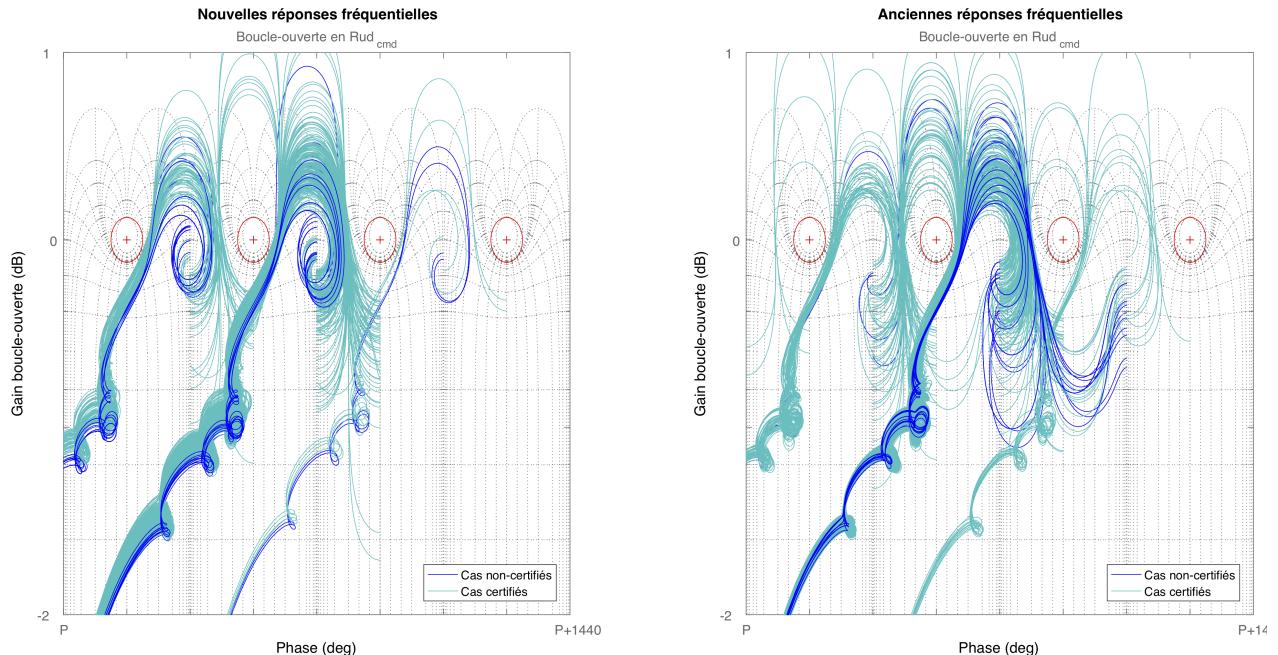


Figure 5.18 Réponses fréquentielles et diagramme de Nichols de la BO du rudder

5.6.6 Efforts des actionneurs

Finalement, la figure 5.19 rassemble les efforts demandés aux actionneurs lors de commandes en ϕ_{cmd} et β_{cmd} . Ces efforts sont du même ordre de grandeur, ce qui indique que l'on ne surcommande pas le système. Les efforts demandés sont réalisables.

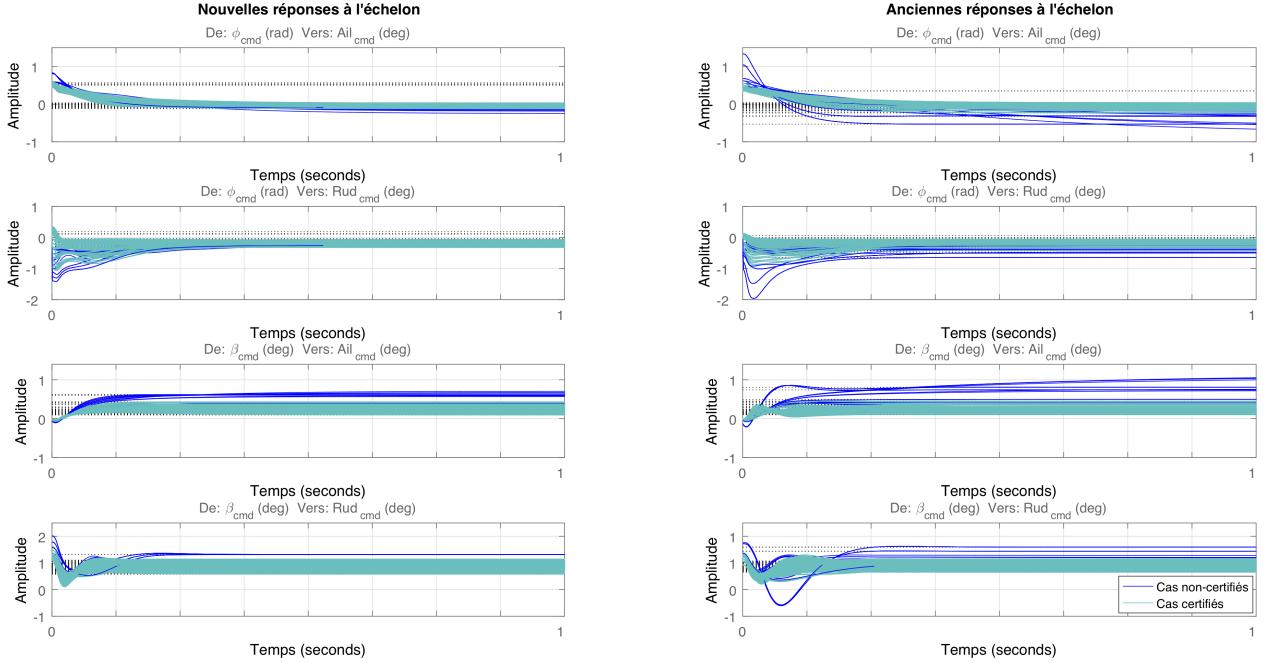


Figure 5.19 Efforts des actionneurs pour des commandes de ϕ_{cmd} et β_{cmd}

Pour la figure 5.19, les commentaires sont les suivants :

Pour ϕ_{cmd} vers la commande en aileron :

- + : Les nouvelles réponses sont meilleures car elles sont légèrement plus rapides, et de plus avec des valeurs en RP plus centrées. Les cas non certifiés provoquent des efforts de l'aileron bien plus raisonnables.

Pour ϕ_{cmd} vers la commande en rudder :

- + : Les nouvelles réponses ont des valeurs en RP plus centrées. Les cas non certifiés provoquent des efforts du rudder bien plus raisonnables.
- - : Les efforts du rudder sont un peu plus grands en amplitude pour les nouvelles réponses. Les efforts négatifs de ces réponses tendent aussi à osciller un peu plus.

Pour β_{cmd} vers la commande en aileron :

- + : Les réponses de la nouvelle synthèse sont plus rapides. Elles ne présentent pas d'oscillations au départ de la simulation. Les cas non certifiés sont globalement meilleurs autant en RP qu'en RT.

— = : Les valeurs en RP sont du même ordre.

Pour β_{cmd} vers la commande en rudder :

— + : Les cas non certifiés sont meilleurs pour les nouvelles réponses temporelles.

— = : Les valeurs en RP sont du même ordre.

— - : Les nouvelles amplitudes de commande du rudder sont plus importantes.

5.7 Conclusion

En tant que première application, on a présenté ici une synthèse rigide robuste et auto-séquencée par la méthode H_∞ structurée sur l'ensemble des conditions de vol de la configuration d'atterrissement. Les gains obtenus et les graphiques des performances (stabilité, suivi de consigne, etc.) ont été donnés pour juger des résultats synthétisés.

Les résultats obtenus pour le réglage linéaire sont très satisfaisants. Les contraintes imposées par l'entreprise ont pu être traduites en requis et concepts, et une synthèse appliquée à l'ensemble de l'enveloppe produit rapidement des résultats atteignant (voire dépassant) les performances attendues.

Cependant l'application démontre aussi que son efficacité dépend d'une excellente connaissance de l'architecture du système (découverte des cas non certifiés). Bien que la modification du système augmenté ou des *softs/hard* goals à prendre en compte ne soit pas très coûteuse en temps, elle nécessite une augmentation du nombre de synthèse à effectuer, ce qui diminue l'intérêt de la synthèse.

Le réglage du modèle non-linéaire est effectué avec les outils de BA. Elle permet de déceler des modifications de contraintes et de convaincre l'équipe de CLaws de la validité de la méthode universitaire appliquée à leur problématique.

On remarque aussi que la synthèse permet d'agir efficacement sur le positionnement du roulis hollandais et du mode spiral mais difficilement sur le roulis pur. Il est conseillé d'ajouter un réglage supplémentaire (architecture ou canaux de performances) afin d'agir directement sur ce mode. C'est d'ailleurs pour cette raison que l'architecture de feedforward au complet sera utilisée dans les applications suivantes.

Étant donné la pertinence des résultats produits pour cette application, l'étude se poursuit au chapitre suivant avec le contrôle de modèles à dynamiques rigides et flexibles.

CHAPITRE 6 SYNTHÈSE D'UN CONTRÔLEUR ROBUSTE ET SÉQUENCÉ POUR DES MODÈLES D'AVION RIGIDES ET FLEXIBLES

Ce sixième chapitre présente la seconde application de la synthèse H_∞ structurée dans le contexte de BA. Maintenant que l'on a vu que la méthode et l'outil sont capables d'obtenir rapidement des résultats performants dans le cadre d'activités de design traditionnel, on démontrera ses capacités à intégrer des concepts supplémentaires pour prendre en compte les requis liés aux modes flexibles. En l'occurrence, cette application considère les modèles flexibles lors de la synthèse pour assurer stabilité et performances malgré l'ajout de ces dynamiques perturbatrices. Comme précédemment, en section 6.1 l'introduction présente les objectifs de l'application. La section 6.2 liste l'ensemble des requis imposés sur les performances de la boucle interne, mais aussi les différents concepts utilisés pour construire le modèle augmenté de l'application. La section 6.3 suivante donne les cas de design utilisés. Les itérations de design ainsi que les résultats et les graphiques de performances obtenus pour la synthèse linéaire sont fournis dans les sections 6.4 et 6.5. Finalement la conclusion sur l'expérience est apportée en section 6.6.

6.1 Introduction

6.1.1 Contexte et activités précédentes

La synthèse présentée dans ce chapitre porte sur les modèles à dynamiques rigides et flexibles, de l'avion exposés dans le chapitre 2. Des scripts sont implémentés pour récupérer les entrées nécessaires, construire le modèle augmenté à partir des concepts du chapitre 3, puis procéder à la synthèse et à l'analyse.

Cette démonstration des capacités de la technique dans l'environnement de BA augmenté des modèles flexibles disponibles, est testée sur l'ensemble de l'enveloppe de vol pour un avion en configuration de croisière. On considère aussi plusieurs cas de chargements différents. Comme dans l'application précédente, la loi de commande est séquencée en altitude et vitesse mais robuste en charge.

6.1.2 Objectifs de l'application

Cette nouvelle application a pour but de démontrer le potentiel de la synthèse en présence de dynamiques flexibles. L'objectif est de prendre en compte ces perturbations lors du réglage, pour limiter leurs influences déstabilisatrices. On veut obtenir une solution plutôt rapidement,

même si on sait que la taille des modèles pris en compte et l'étendue de cette configuration donnent un modèle augmenté d'ordre très élevé qui nécessite plus de temps à optimiser.

On ne dispose pas d'une synthèse classique qui inclut les modèles flexibles dans le design. On prend comme référence de performance les résultats actuels pour un réglage classique, tout en sachant que l'influence des modèles flexibles est alors prise en compte a posteriori.

6.1.3 Méthodologie suivie

La même méthodologie que celle définie dans la section 1.2.2 du chapitre d'introduction est appliquée ici. Les étapes suivantes sont imposées :

- Définition des requis (section 6.2.1)
- Définition des concepts (section 6.2.2)
- Définition de l'architecture (section 6.2.3)
- Implémentation et optimisations (sections 6.3 et 6.4)
- Tests et vérifications (section 6.5)

Optimisation du design

La méthode utilisée pour l'optimisation du design est toujours une méthode itérative. Les deux points de départ de cette application sont :

- pour le design : celui obtenu dans l'application précédente (Chapitre 5),
- pour la condition initiale : les gains de feedforward (non linéaires) et feedback d'un design de référence obtenu par un réglage classique. Comme les gains proviennent du modèle non linéaire un coefficient K_β a été ajouté aux gains $P_{1,2}$ et P_{21} pour remplacer la commande β_{cmd} en une commande Rud_{cmd} .

6.2 Requis et design

6.2.1 Requis sur les performances de la boucle interne

D'après la méthode “Cycle en V” suivie, il est nécessaire de poser les requis à valider. Ces contraintes permettent de définir sous quelles conditions une solution est performante ou non. Elles s'appliquent ici sur un transfert entrée/sortie de la boucle interne linéaire : un transfert BO ou fermée. Parmi les requis de BA (voir 3.3), ceux considérés dans cette application sont listées ci-après :

- Requis 1 : voir 3.3.1.
- Requis 2 : voir 3.3.1.
- Requis 3 : voir 3.3.1.

- Requis 4 : voir 3.3.2.
- Requis 5 : voir 3.3.2.
- Requis 6 : voir 3.3.3.
- Requis 7 : voir 3.3.3.
- Requis 8 : voir 3.3.4.
- Requis 9 : voir 3.3.4.
- Requis 10 : voir 3.3.5.

6.2.2 Concepts et canaux de performances

Afin d'assurer les requis listés plus haut, plusieurs canaux de performances sont à construire, notamment des fonctions de pondérations mais aussi des points d'analyse. La série de concepts choisis sont définis ci-après :

- Concept 1 (associé au requis 1) : voir 3.3.1. On considère la structure globale au niveau de l'entrée et de la commande feedforward. On n'ajoute pas cependant les filtres ASE, on utilise les sorties des modèles d'avion qui considèrent déjà leurs dynamiques.
- Concept 2 (associé au requis 2) : voir 3.3.1. On ne modifie ni les filtres ASE, ni le gain K_p .
- Concept 3 (associé au requis 3) : voir 3.3.1.
- Concept 4 (associé au requis 4) : voir 3.3.2.
- Concept 5 (associé au requis 5) : voir 3.3.2.
- Concept 6 (associé au requis 6) : voir 3.3.3.
- Concept 7 (associé au requis 7) : voir 3.3.3.
- Concept 8 (associé au requis 8) : voir 3.3.4. On contrôle la position des pôles avec le suivi de références et non avec l'outil *TuningGoal.Poles*, dû aux modes flexibles mal-amortis et difficilement contrôlables.
- Concept 9 (associé au requis 9) : voir 3.3.4.
- Concept 10 (associé au requis 10) : voir 3.3.5.

On rappelle que les canaux de performances et leurs architectures, sont abordés dans la section 3.3 du chapitre 3. Pour réduire la taille et augmenter la lisibilité du schéma du modèle augmenté de la synthèse (Fig. 6.1), on ne présente pas toutes les canaux de performances.

6.2.3 Architecture du modèle augmenté

La figure 6.1 présente le schéma du modèle augmenté de la synthèse sur la configuration de croisière complète dans le cas flexible. On rappelle que les transferts pour l'étude des marges de stabilité au niveau de la commande en aileron et en rudder, ainsi que les transferts de

limitations d'amplitude des gains de feedback ne sont pas représentées.

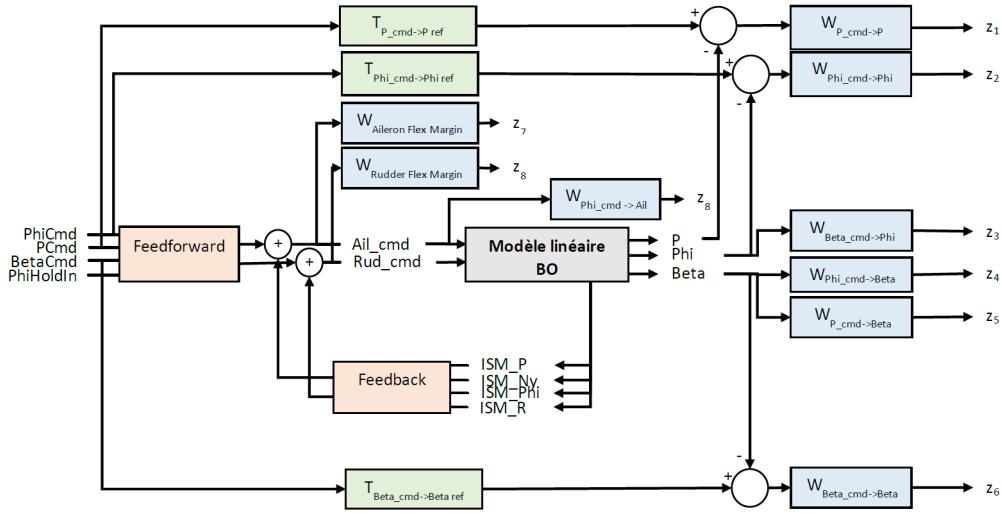


Figure 6.1 Modèle augmenté de la synthèse rigide et flexible

6.3 Explication des cas utilisés pour l'application

La synthèse est appliquée sur l'ensemble des conditions de vol à la configuration de croisière. Les points de vol choisis sont donnés à la figure 6.3. On manipule donc des modèles provenant de 80 conditions de vol différentes (8 altitudes et 10 vitesses CAS). Chaque point de vol est représenté par 5 modèles rigides (modèles bleus) à des configurations de charge différentes (Fig. 6.2). Pour certains cas, des modèles flexibles (modèles oranges) existent, on prendra alors trois configurations de charge. L'objectif est de couvrir l'ensemble des conditions de vol possibles pour cette configuration, et de démontrer que la synthèse de la loi de commande peut mixer les modèles rigides et flexibles dans une seule et même synthèse afin de prendre en compte toutes les contraintes industrielles. La commande est séquencée en vitesse et altitude, mais robuste en conditions de charge.

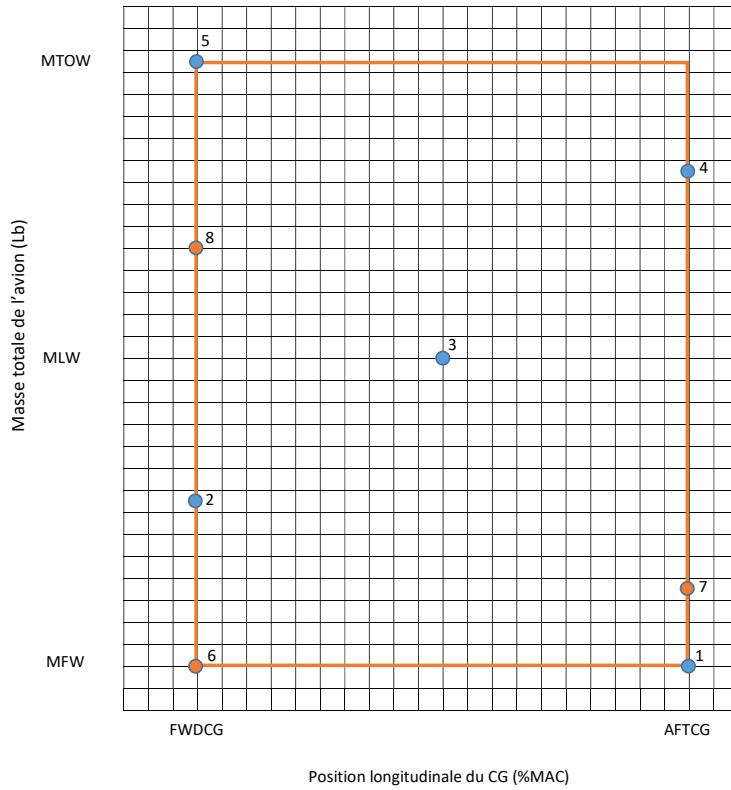


Figure 6.2 Position des configurations de charge sélectionnées dans leur ensemble

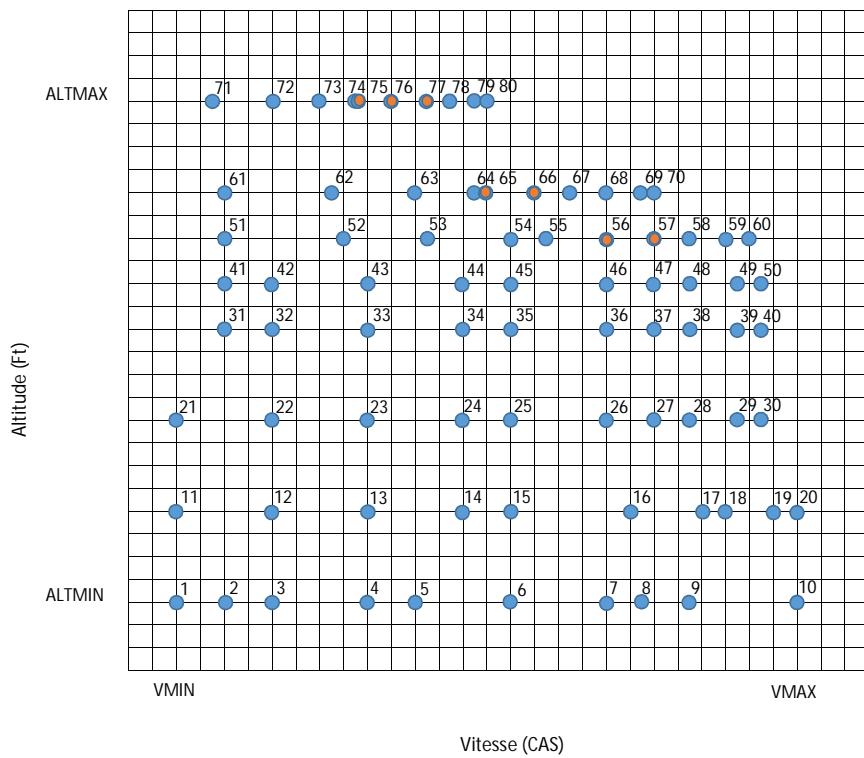


Figure 6.3 Position des conditions de vol sélectionnées dans leur ensemble

6.4 Synthèse et optimisations

6.4.1 Design par groupe de conditions de vol

Étant donné que le domaine (ALT,CAS) considéré est vaste, on juge nécessaire de le diviser en groupe de design (3 altitudes et 3 vitesses). On se concentre sur le réglage des trois groupes à basses vitesses car ils sont les plus problématiques. À faible vitesse, l'effort demandé aux actionneurs est plus grand.

6.4.2 Ordre des modèles flexibles

Les modèles flexibles étaient initialement réduits à 55 états. Lors de la vérification indépendante sur les modèles complets, on se rend compte que les *hard goals* ne sont pas bien vérifiés. Le modèle était finalement trop réduit pour capturer certaines dynamiques déstabilisatrices. On choisit d'augmenter l'ordre de réduction à 115 états. Comme vu à la section 2.5, cette réduction approxime mieux la dynamique des modèles flexibles. Cependant, l'augmentation allonge le temps nécessaire pour optimiser le système augmenté.

6.5 Résultats et performances après réglage de la boucle linéaire

On rappelle que les résultats présentés ci-dessous sont obtenus grâce à un outil de vérification indépendant. Les modèles en BF sont reconstruits et les modèles initiaux sont des modèles complets, à des cas de chargement dans le domaine disponible. Ce ne sont donc pas nécessairement ceux pris pour la synthèse. Pour apporter une analyse des graphiques, on utilise la notation :

- + : concernant le critère jugé, les nouvelles solutions sont meilleures que les anciennes.
- = : concernant le critère jugé, les nouvelles solutions sont équivalentes aux anciennes.
- - : concernant le critère jugé, les anciennes solutions sont meilleures que les nouvelles.

6.5.1 Evolution de l'amplitude des gains

Les cinq premières figures (de Fig. 6.4 à Fig. 6.8) présentent les nouveaux gains de feedback et de feedforward, ainsi que la solution de BA comme point de comparaison.

Les deux premières figures montrent que l'amplitude des nouveaux gains reste équivalente à la solution existante. On remarque que les nouveaux gains P_{12}, P_{22} sont plus importants. La différence peut-être expliquée par le gain K_β appliqué aux anciens gains. Il faudra s'attendre à avoir des anciennes réponses temporelles plus petites pour les transferts venant de β_{cmd} .

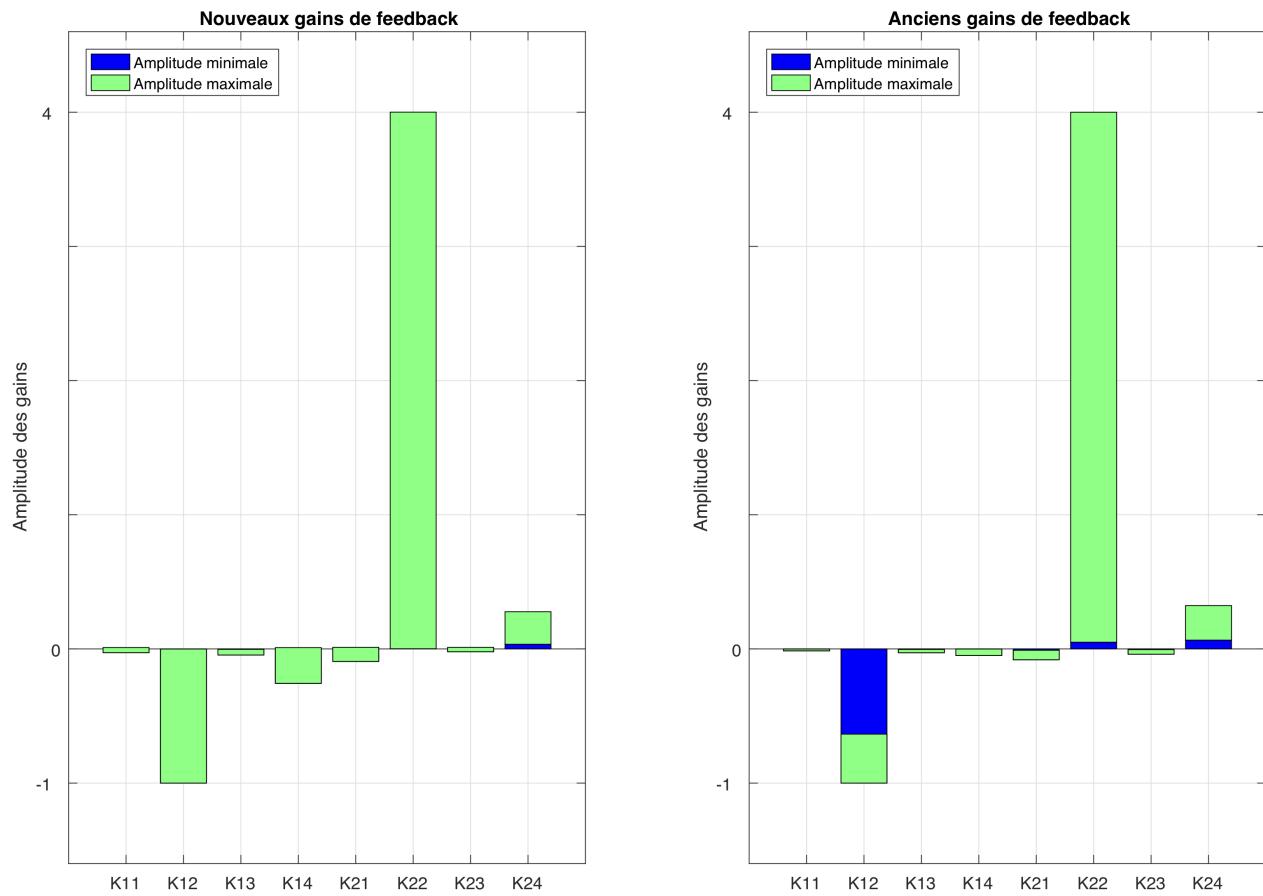


Figure 6.4 Extremums des amplitudes des gains de feedback

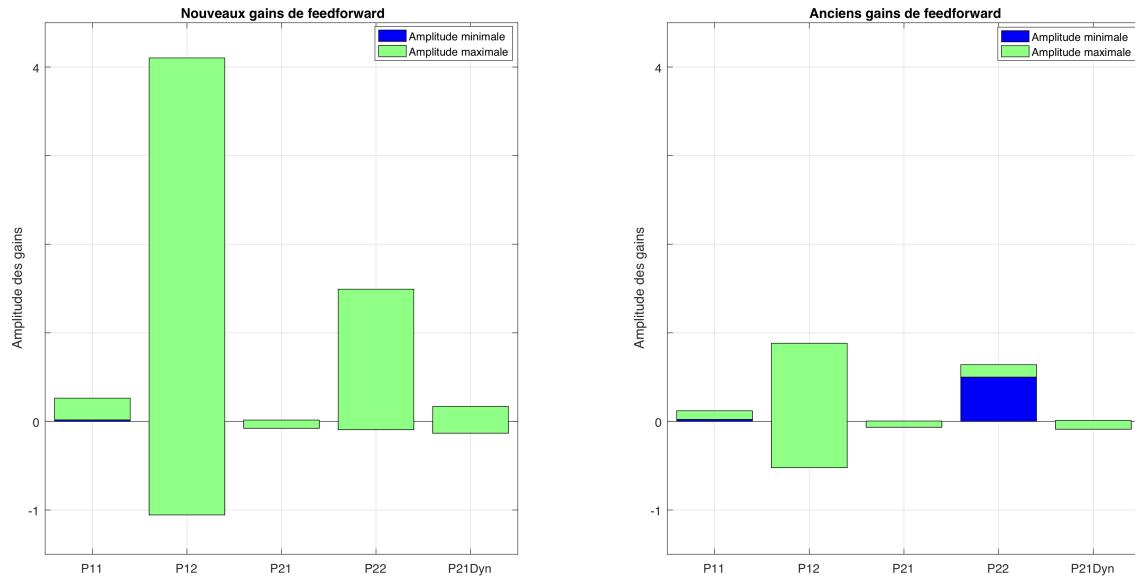


Figure 6.5 Extremums des amplitudes des gains de feedforward

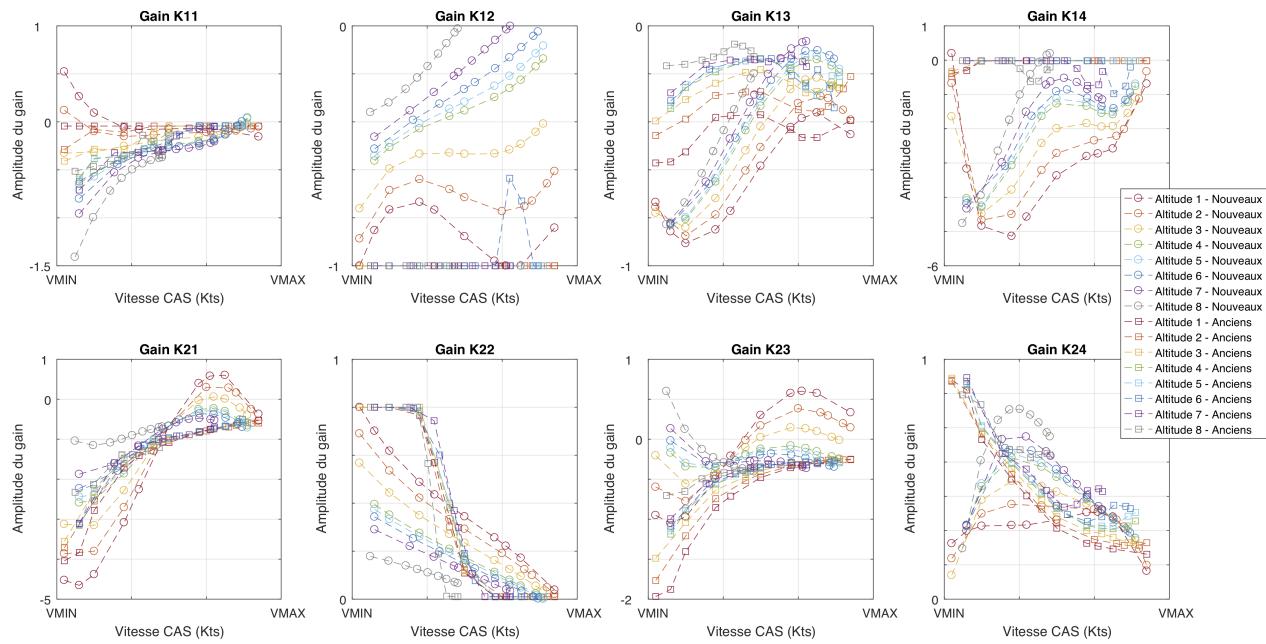


Figure 6.6 Évolution des gains de feedback sur l'enveloppe

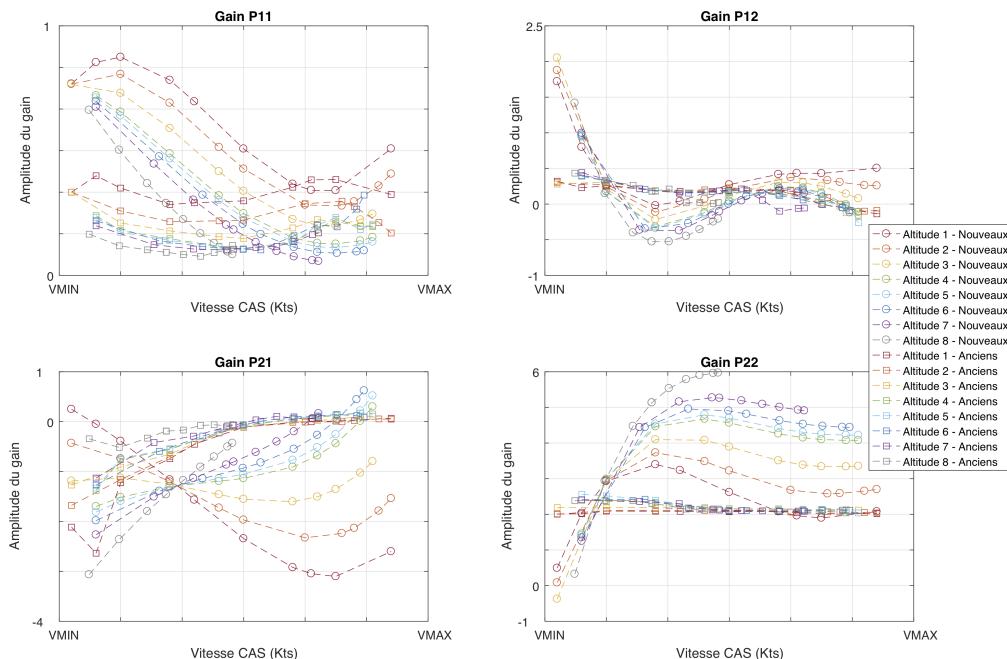


Figure 6.7 Évolution des gains de feedforward sur l'enveloppe

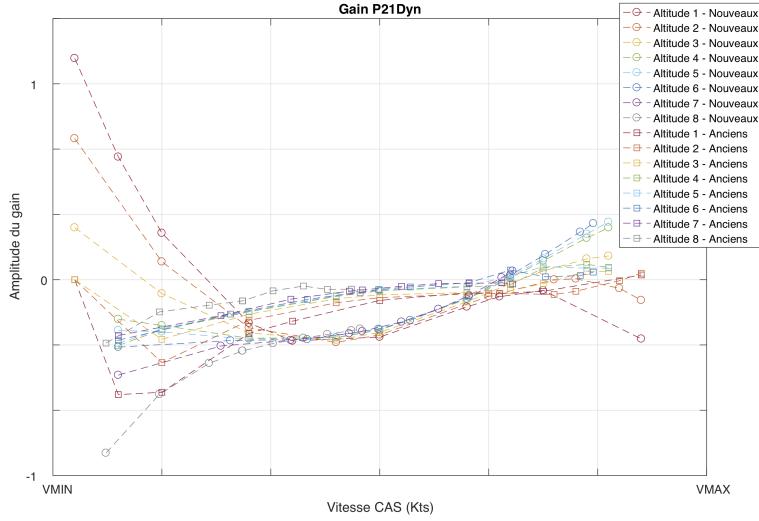


Figure 6.8 Évolution du gains de feedforward $P_{21,Dyn}$ sur l'enveloppe

La nouvelle évolution des gains suit généralement la tendance proposée initialement, ce qui est cohérent car la solution ne devrait pas être trop éloignée de la solution de BA. Cependant, les allures ne sont pas aussi semblables que pour l'application précédente. La solution proposée s'éloigne donc plus du point de départ.

6.5.2 Suivi de références

Les trois figures suivantes de la figure 6.9 à 6.11, affichent les réponses temporelles des signaux de commandes P_{cmd} , ϕ_{cmd} et β_{cmd} , vers leurs sorties respectives P , ϕ et β . On jugera de la capacité de la solution à suivre une référence.

Pour la figure 6.9, les commentaires sont les suivants :

- + : L'erreur en RP des nouvelles réponses est bien moins grande. La nouvelle solution est meilleure en RP.
- - : La nouvelle solution est cependant moins bonne en RT, car elle a des forts dépas- sements (inférieurs et supérieurs).

Pour la figure 6.10, les commentaires sont les suivants :

- + : Les nouvelles solutions n'ont pas de réponses qui s'écartent grandement de la référence en RP. Il semblerait aussi que plus de modèles respectent les critères de rapidité (constante de temps).
- - : En contre-partie, les constantes de temps vont se situer dans un intervalle large pour la nouvelle solution. De plus, si on exclut les modèles extrêmes de l'ancienne solution, ils sont finalement plus recentrés autour de la référence.

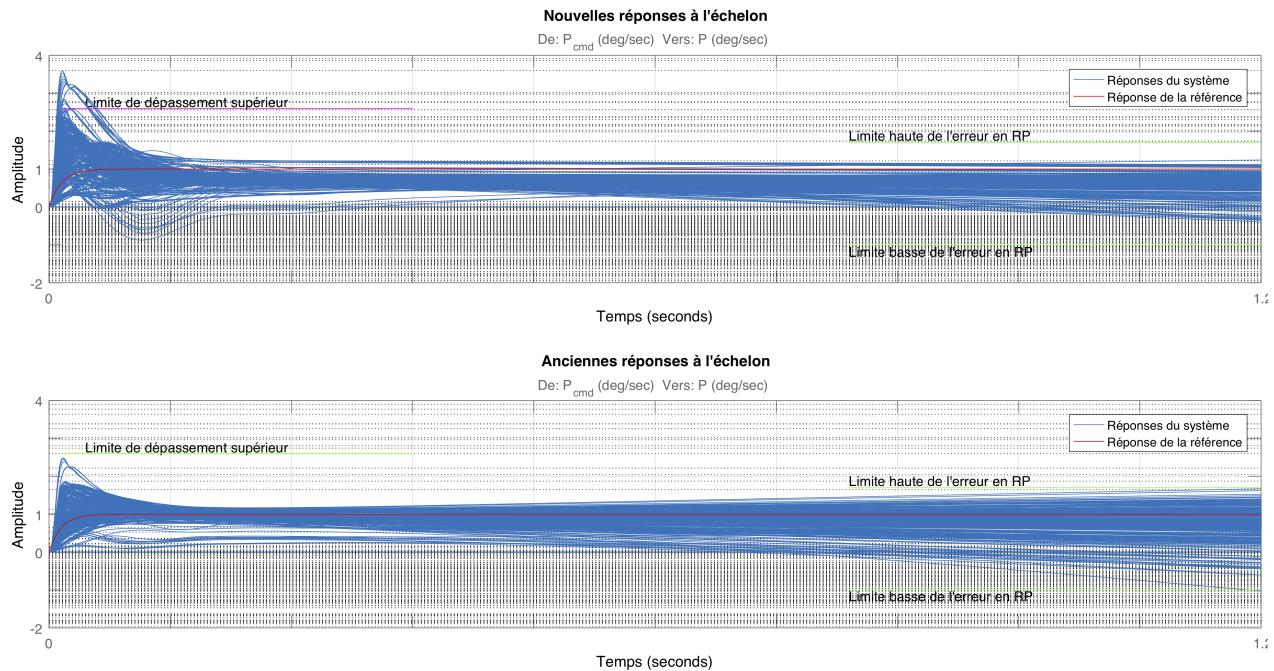


Figure 6.9 Performances du suivi de référence pour une commande de P_{cmd} vers P

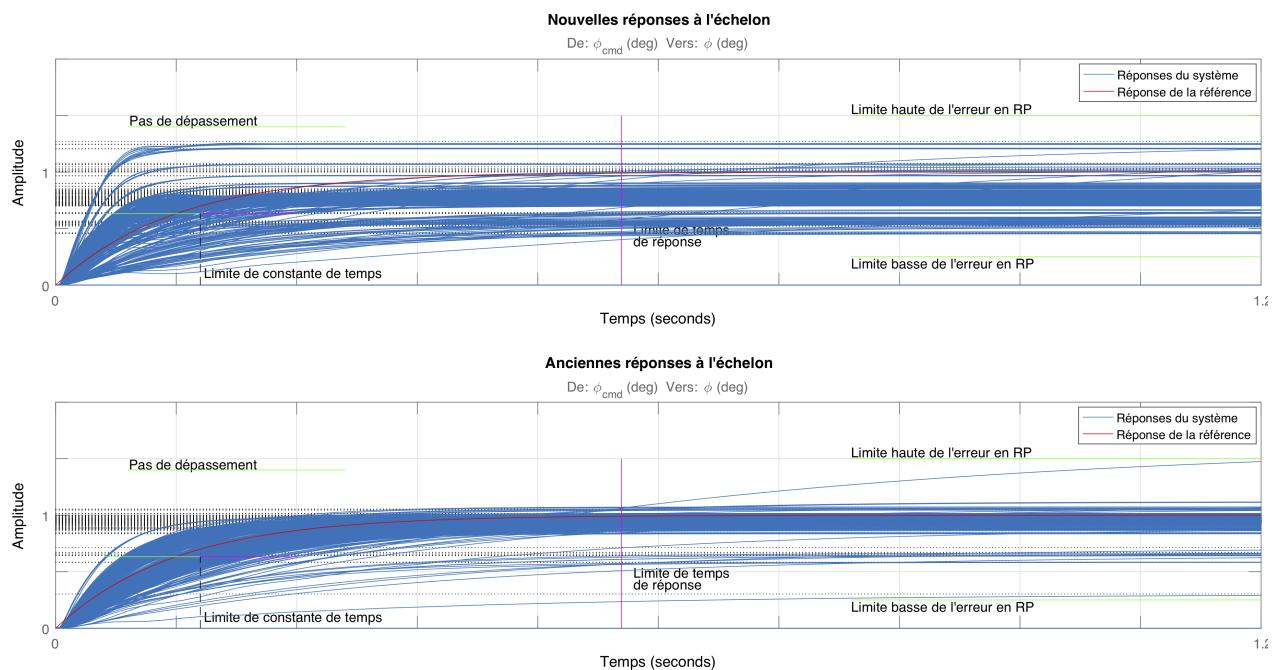


Figure 6.10 Performances du suivi de référence pour une commande de ϕ_{cmd} vers ϕ

Pour la figure 6.11, les commentaires sont les suivants :

- + : Même commentaire que précédemment, les nouvelles réponses n'ont pas de cas qui s'écartent autant de la référence que les cas à valeur en RP très basses. Les anciennes solutions ne sont pas du tout recentrées autour de la consigne. Ceci est dû au coefficient K_β qui intervient en RP. Comme les gains en feedforward étaient réduits, la réponse est β est aussi diminuée, tout comme les efforts liés à la commande β_{cmd} , comme on le verra par la suite.
- = : En général, l'écart de l'erreur en RP est semblable.
- - : Certains nouveaux cas ont un dépassement trop important, donc un amortissement encore trop faible.

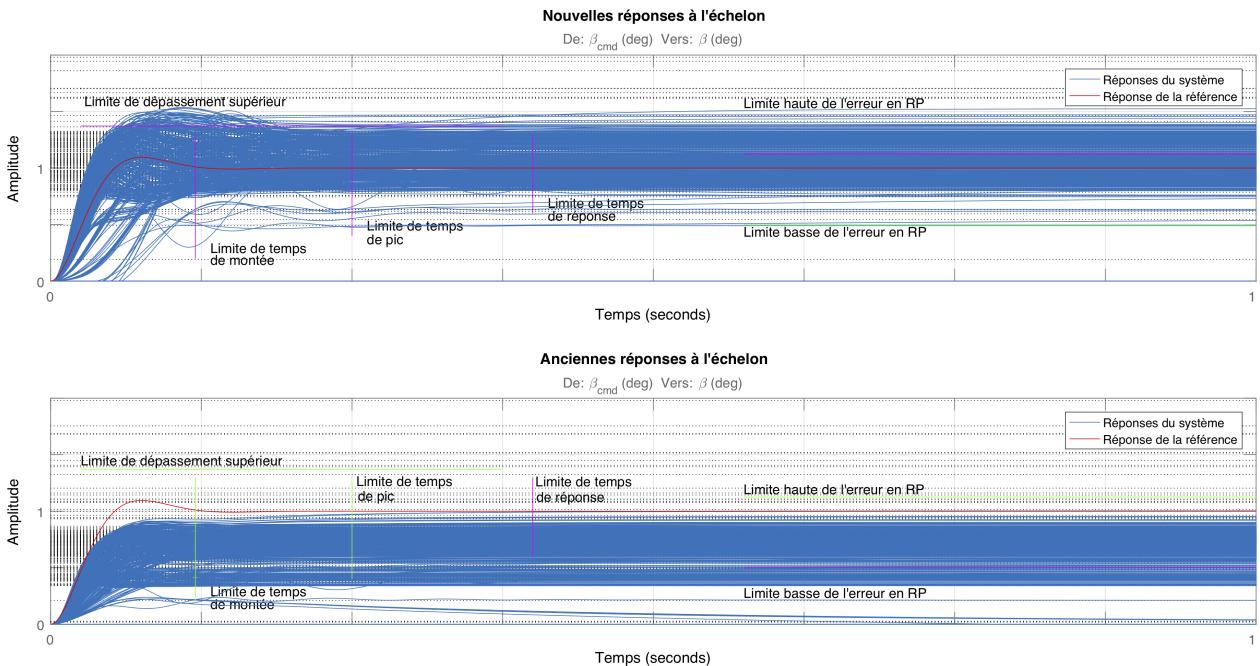


Figure 6.11 Performances du suivi de référence pour une commande de β_{cmd} vers β

6.5.3 Découplages de signaux

Les trois figures suivantes (de Fig. 6.12 à Fig. 6.14) affichent les réponses temporelles des signaux de commandes P_{cmd} , ϕ_{cmd} et β_{cmd} , vers leurs sorties croisées respectives β et ϕ . On jugera de la capacité de la solution à découpler les signaux lors de ces transferts. Pour la figure 6.12, les commentaires sont les suivants :

- = : Les deux solutions ont la même allure, un bon découplage en général et quelques modèles qui divergent avec des valeurs en RP élevées.
- - : Les anciennes réponses sont mieux centrées autour de la référence, et les cas

extrêmes divergent moins rapidement. Tandis que les nouvelles réponses présentent aussi un léger dépassement.

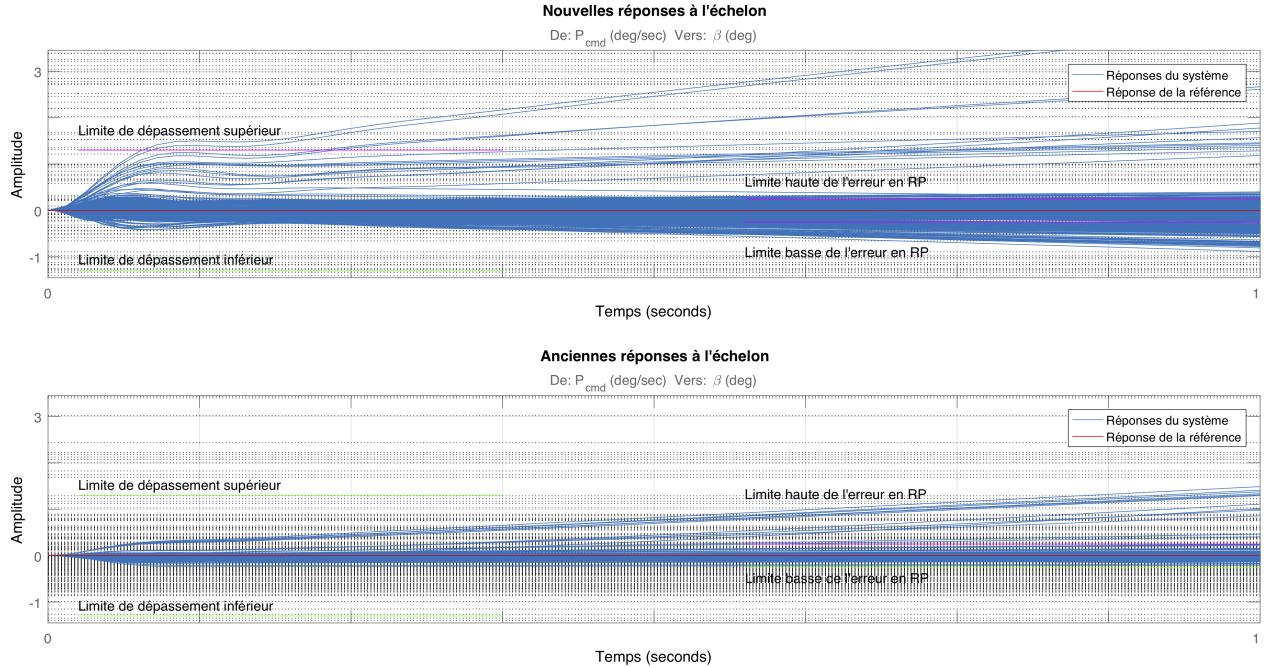


Figure 6.12 Performances du découplage de signaux pour une commande de P_{cmd} vers β

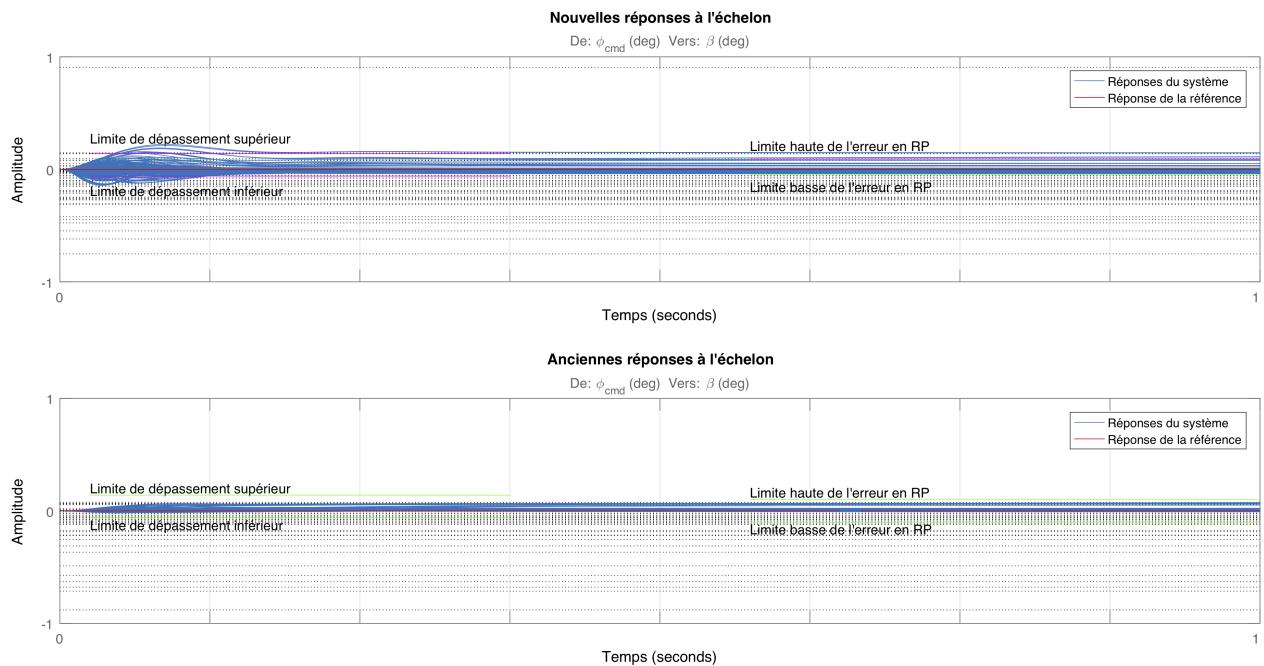


Figure 6.13 Performances du découplage de signaux pour une commande de ϕ_{cmd} vers β

Pour la figure 6.13, les commentaires sont finalement semblables à la figure précédente :

- + : Sans considérer les cas les plus extrêmes, la nouvelle solution a des valeurs en RP mieux recentrées autour de la référence.
- = : Les deux solutions ont la même allure, un bon découplage en général et quelques modèles qui divergent avec des valeurs en RP élevées.
- - : Les dépassements inférieurs et supérieurs sont plus grand pour les nouvelles réponses. L'intervalle accueillant les erreurs en RP est aussi plus grand pour les réponses de la nouvelle synthèse.

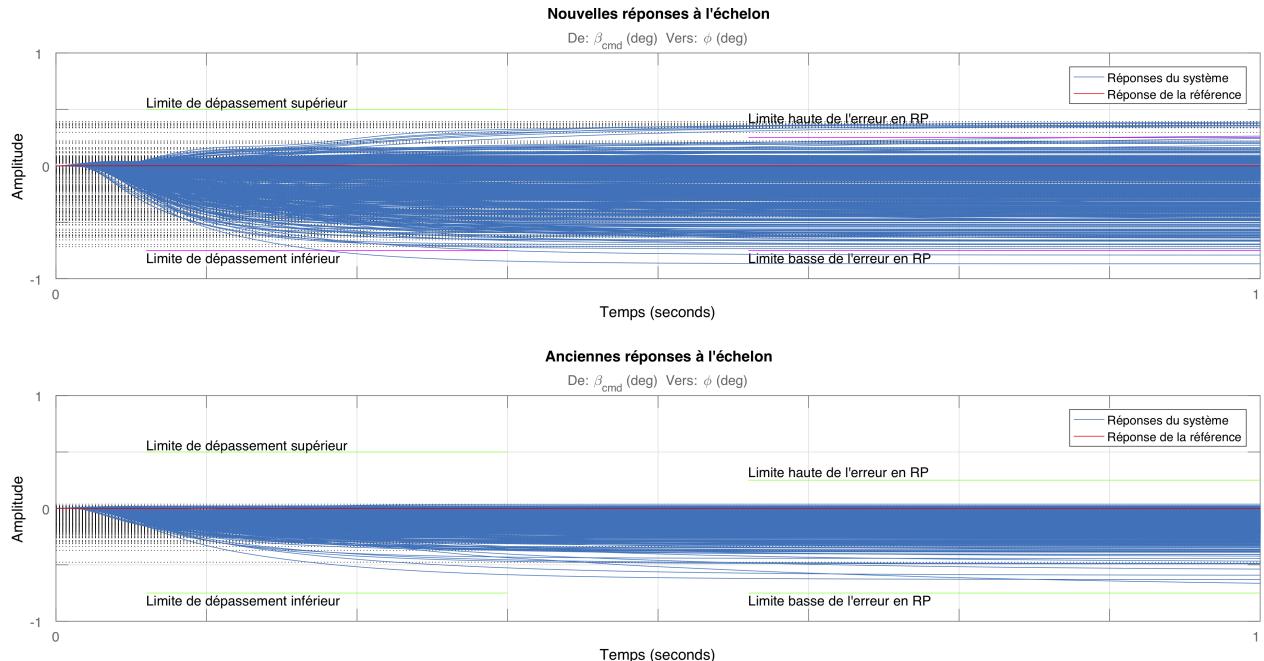


Figure 6.14 Performances du découplage de signaux pour une commande de β_{cmd} vers ϕ

Pour la figure 6.14, le coefficient K_β intervient et réduit l'amplitude des réponses de l'ancienne solution. Les commentaires sont alors les suivants :

- + : Les solutions du nouveau système sont plus recentrées autour de la référence.
- = : Les enveloppes des réponses sont équivalentes.
- - : L'intervalle des erreurs en RP est plus large pour les nouvelles réponses. L'ajout du coefficient K_β dans la nouvelle loi de commande réduirai cependant cet intervalle.

6.5.4 Position des pôles de la boucle fermée

Pour la figure 6.15, les commentaires sont les suivants :

- + : Les pôles réels du roulis pur sont en général plus rapide pour la nouvelle solution.
- = : Les deux solutions ont quelques modèles extrêmes qui se placent dans la zone rouge et bleu claire. Les contraintes ont été relâchées sur ces modèles à basse-vitesse

pour ne pas dégrader les placements des autres modèles. En général, les modes roulis hollandais sont bien placés.

- $-$: Certains pôles de la nouvelle solution sont placés dans la région rouge à très basses fréquences naturelles. Certains n'ont pas un amortissement suffisant pour être ignorés. Un réglage supplémentaire est nécessaire ici pour augmenter la pulsation naturelle de ces pôles. On pourrait faire de même pour les pôles dans la zone violette. Certains roulis purs sont un peu trop rapide, ils mériteraient d'être ralenti un peu pour avoir une meilleure séparation entre le placement du mode roulis et du mode spiral.

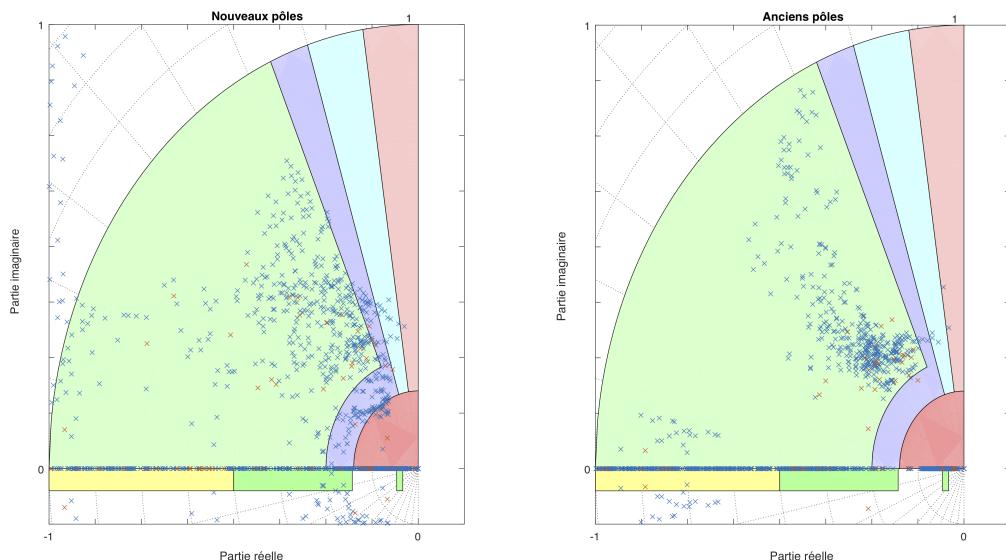


Figure 6.15 Positionnement des pôles et des zéros de la BF

6.5.5 Marges de stabilité

On donne les tracés des réponses fréquentielles des BO en Ail_{cmd} et Rud_{cmd} . Les diagrammes de Nichols des figures 5.17 et 5.18 renseignent sur le respect des marges de gain et de phase. Les diagrammes de Bode des figures 6.18 et 6.19 renseignent sur le respect des marges de stabilité en amplitude.

Pour la figure 6.16, les commentaires sont les suivants :

- $+$: Les courbes des nouvelles réponses tangentes plus les ellipses limites. Ce système est donc plus optimisé.
- $=$: Toutes les marges de stabilité sont respectées, sauf pour un seul modèle dans les deux solutions.

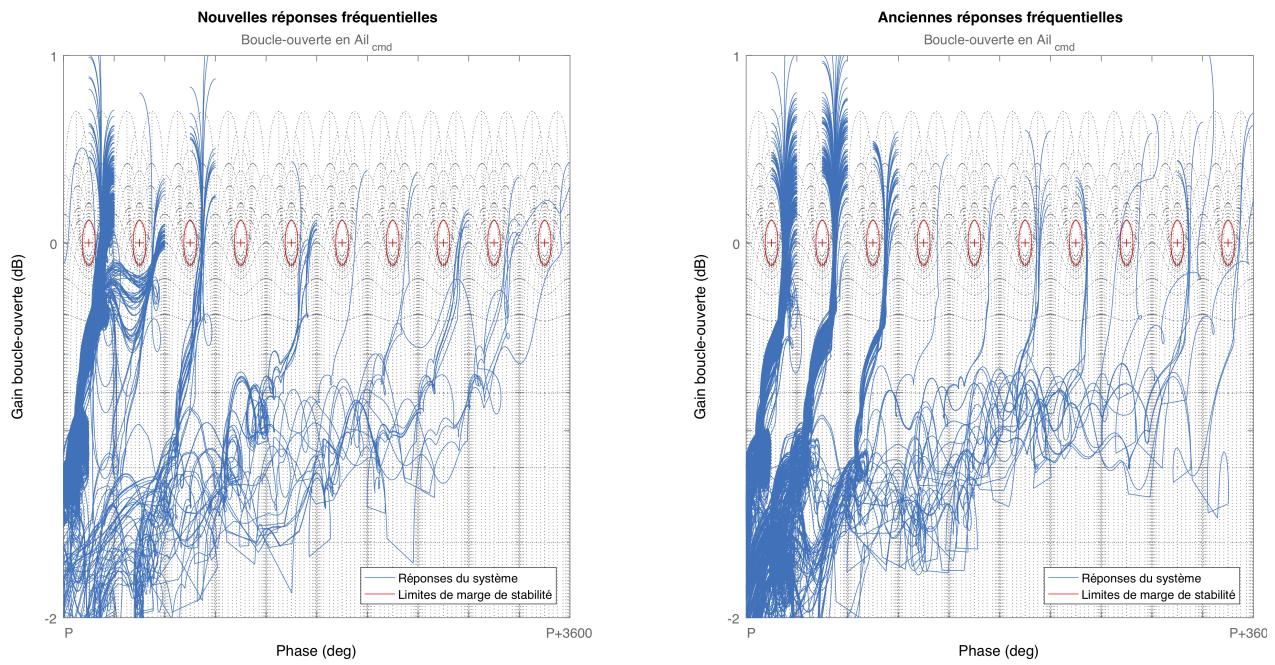


Figure 6.16 Réponses fréquentielles et diagramme de Nichols de la BO de l'aileron

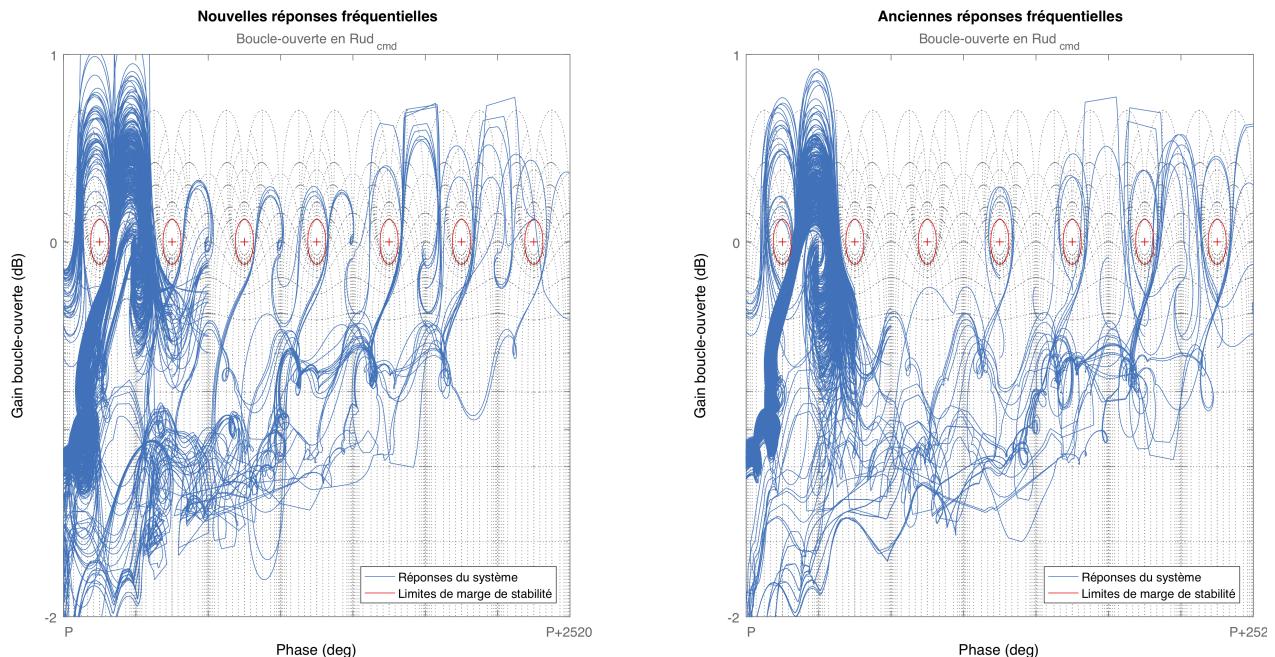


Figure 6.17 Réponses fréquentielles et diagramme de Nichols de la BO du rudder

Pour la figure 6.17, les commentaires sont les suivants :

- + : Toutes les marges de stabilité sont respectées sauf pour 2 modèles dans la nouvelle solution et 3 dans l'ancienne. La même remarque que précédemment peut être faite, les modèles viennent tangenter de plus près les ellipses limites, ce qui indique une meilleure optimisation.

Pour la figure 6.18, les commentaires sont les suivants :

- + : Toutes les nouvelles réponses fréquentielles respectent les limites d'amplitude (autant flexible que rigide). On voit que les contraintes liées aux modèles flexibles sont bien prises en compte.

Pour la figure 6.19, les commentaires sont les suivants :

- = : Les allures fréquentielles des deux solutions se ressemblent. La nouvelle solution vérifie les marges d'amplitude imposées. Tandis que l'ancienne solution ne respecte pas le critère sur les modèles rigides. Le fait que la limite d'amplitude des modèles rigides ne soit pas validée par l'ancienne réponse signifie que le système est alors sur-contraint ici, et que le requis mériterait d'être allégé afin d'améliorer les performances.

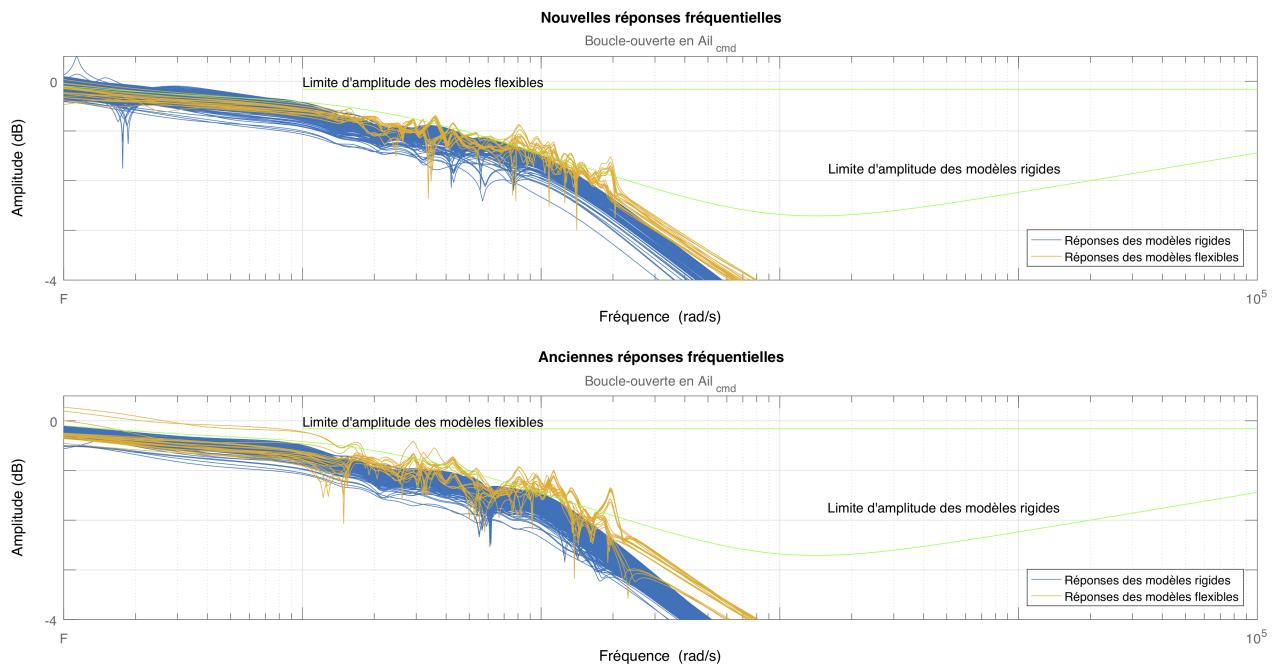


Figure 6.18 Réponses fréquentielles et marges d'amplitude pour la stabilité des modèles flexibles de la BO de l'aile

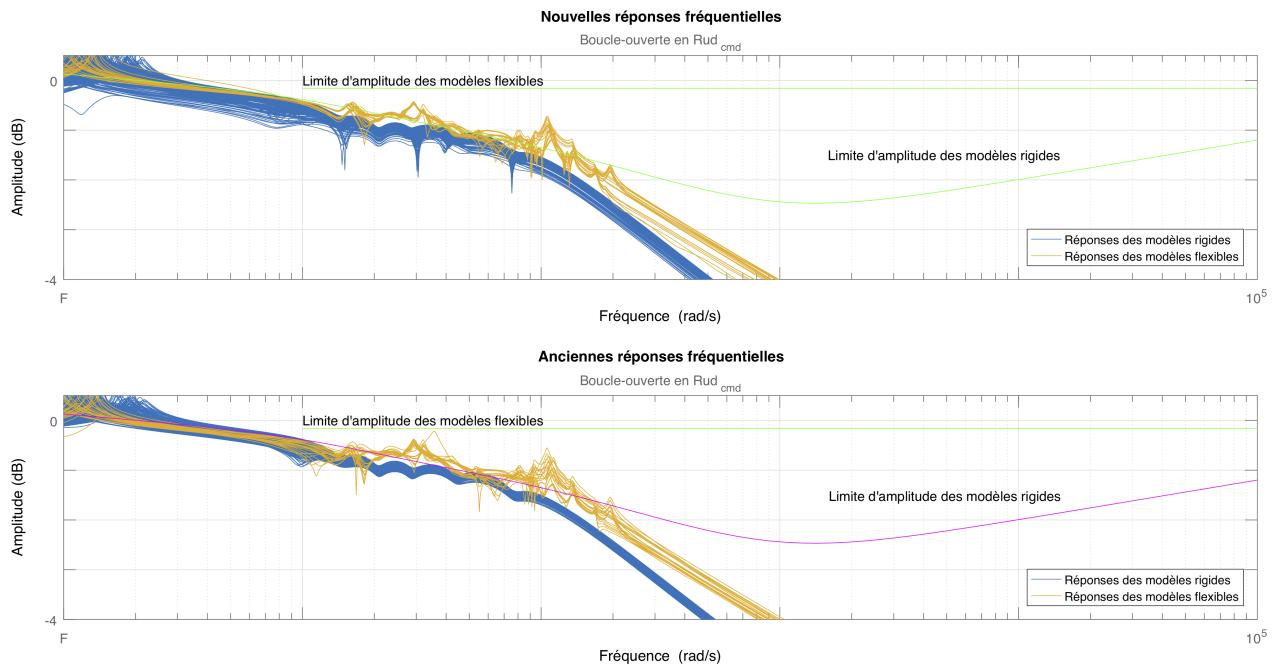


Figure 6.19 Réponses fréquentielles et marges d'amplitude pour la stabilité des modèles flexibles de la BO du rudder

6.5.6 Efforts des actionneurs

Finalement, la figure 6.20 rassemble les efforts demandés aux actionneurs lors de commandes en P_{cmd} , ϕ_{cmd} et β_{cmd} .

Pour la figure 6.20, les commentaires sont les suivants :

Pour P_{cmd} vers la commande en aileron :

- = : Les deux solutions ont les mêmes allures de commandes.
- - : Les nouvelles réponses ont des amplitudes en moyenne plus fortes.

Pour P_{cmd} vers la commande en rudder :

- = : Les deux solutions ont les mêmes allures de commandes.
- - : Les efforts de commande demandés par les nouvelles réponses sont plus importantes, pour quelques modèles extrêmes. En RT, les nouvelles réponses ont des dynamiques mal-amorties donnant lieu à des plus grands dépassements.

Pour ϕ_{cmd} vers la commande en aileron :

- = : Les deux solutions ont les mêmes allures de commandes.
- - : Les efforts de commande demandés par les nouvelles réponses sont plus importants.

Pour ϕ_{cmd} vers la commande en rudder :

- = : Les deux solutions ont les mêmes allures de commande, la même rapidité et le même ordre d'erreur en RP.
- - : Les dépassements inférieurs atteints par les nouvelles réponses sont néanmoins plus grands.

Pour β_{cmd} vers la commande en aileron :

- + : Les réponses de l'ancienne synthèse sont réduites par le coefficient K_β , ce qui explique les efforts plus importants demandés par la nouvelle synthèse.
- = : En prenant les réponses conjointes de la nouvelle synthèse (sans les cas plus extrêmes), l'erreur en RP est équivalente à celle des anciennes réponses.
- - : Les réponses de la synthèse ont un dépassement inférieur et supérieur plus important que la solution initiale. L'intervalle des erreurs en RP des nouvelles réponses est aussi plus large.

Pour β_{cmd} vers la commande en rudder :

- + : Les réponses de l'ancienne synthèse sont réduites par le coefficient K_β , ce qui explique les efforts plus importants demandés par la nouvelle synthèse.
- - : Les réponses de la synthèse ont un dépassement inférieur plus important que la solution initiale. L'intervalle des erreurs en RP des nouvelles réponses est cette fois bien plus large.

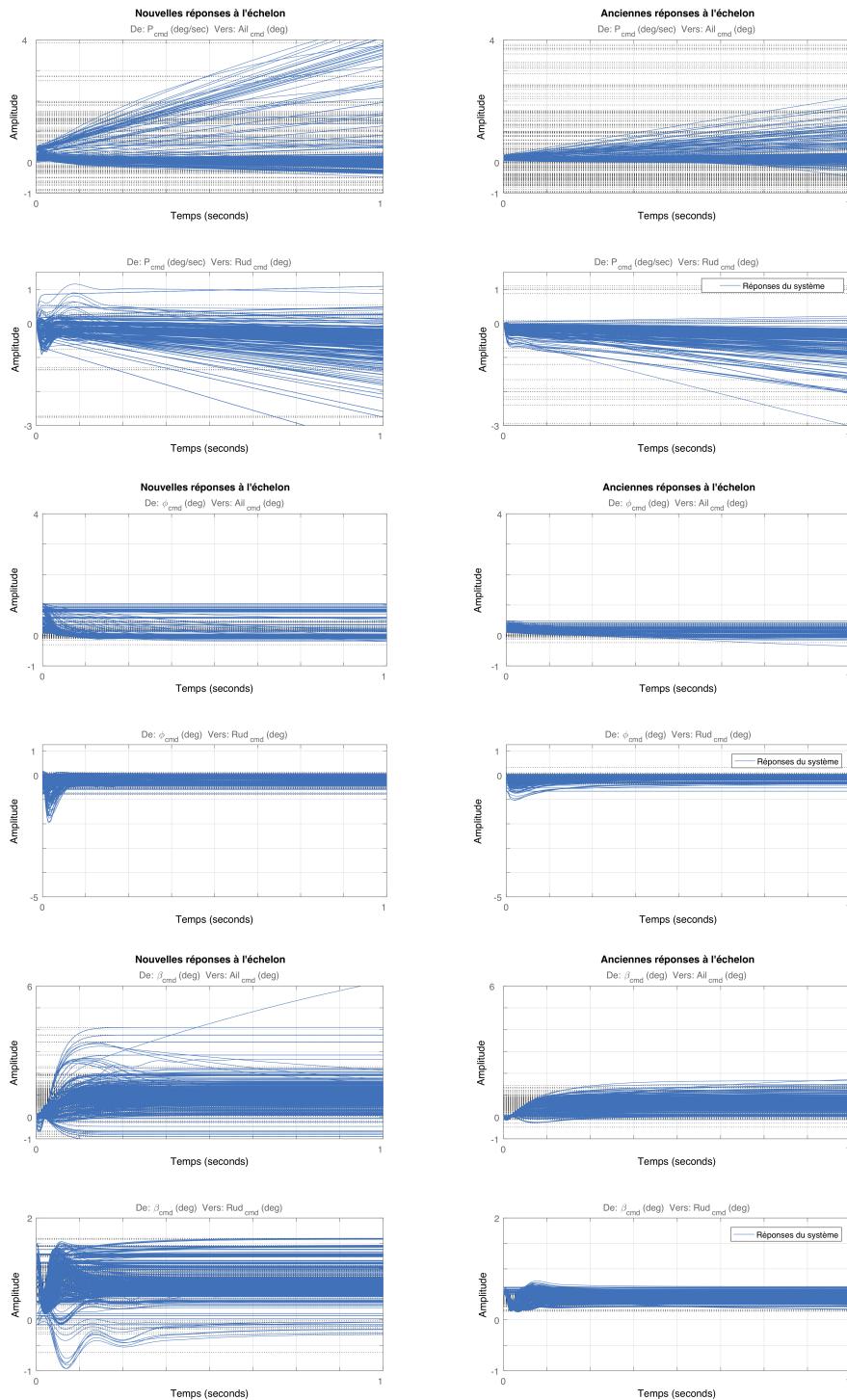


Figure 6.20 Efforts des actionneurs pour des commandes de P_{cmd} , ϕ_{cmd} et β_{cmd}

6.6 Conclusion

Après la première application, on a présenté ici une augmentation de la méthode de synthèse au réglage des modèles rigides et flexibles. Les gains obtenus et les graphiques des performances (stabilité, suivi de consigne, etc.) ont été donnés pour juger des résultats synthétisés.

Les performances obtenues ne sont pas meilleures que la solution de référence. On peut même dire qu'elles sont généralement moins bonnes. Les performances en découplage et en effort de commande mériteraient d'être améliorées.

Afin d'améliorer à nouveau le réglage, on pourrait diminuer la contrainte (*hard goal*) sur la marge d'amplitude de la BO du rudder, et probablement retravailler le placement du roulis hollandais pour les pôles à faibles pulsations naturelles.

Malgré tout, cette application prouve que les contraintes appliquées aux modèles flexibles, notamment la stabilité, peuvent être prises en compte. On peut concevoir des synthèses mixtes qui contiennent des modèles rigides et flexibles. Le temps dépensé à préciser le design de cette application est approximativement évalué à 30% du temps alloué au réglage de la première application. Avec très peu d'itérations de réglages, on peut même conclure que la solution proposée donne finalement une bonne piste pour une loi de commande performante.

On démontre, à travers cette application, qu'on a la capacité de faire un design avec des modèles rigides et flexibles. Ainsi dans des applications futures où il ne sera plus possible d'appliquer la synthèse de référence parce que la dynamique flexible ne pourra pas être traitée séparément de la dynamique rigide, on a ici la preuve de concept d'une solution applicable sans changement majeur de la structure du contrôleur.

Le concept démontré ici est vraiment intéressant. Il ouvre sur une multitude de possibilités concernant les applications possibles, et apporte une plus-value certaine à la méthode. Essayons d'étendre encore cet expérience avec le chapitre suivant, en procédant aux réglages des filtres ASE dans la synthèse.

CHAPITRE 7 RÉGLAGE DES FILTRES AÉROSERVOÉLASTIQUES PAR UNE SYNTHÈSE ROBUSTE

Ce septième chapitre présente l'ultime application de la synthèse H_∞ structurée dans le contexte de BA. Il démontre la possibilité d'ajouter les filtres ASE lors de la synthèse afin de régler précisément les fréquences de coupures sur les modes flexibles. Ce chapitre, à l'image des deux précédents, commence par une introduction en section 7.1 qui présente les objectifs de l'application. La section 7.2 liste l'ensemble des requis imposés sur les performances de la boucle interne, mais aussi les différents concepts utilisés pour construire le modèle augmenté de l'application. La section 7.3 suivante donne les cas de design utilisés. Les itérations de design ainsi que les résultats et les graphiques de performances obtenus pour la synthèse linéaire sont fournis dans les sections 7.4 et 7.5. Finalement la conclusion sur l'expérience est apportée en section 7.6.

7.1 Introduction

7.1.1 Contexte et activités précédentes

La synthèse présentée dans ce chapitre porte sur les modèles à dynamiques rigides et flexibles, de l'avion exposés dans le chapitre 2. Les gains des contrôleurs ne sont pas obtenus à l'aide de l'outil développé dans le chapitre 4. Des scripts sont implémentés pour récupérer les entrées nécessaires, construire le modèle augmenté à partir des concepts définis dans le chapitre 3, puis procéder à la synthèse et à l'analyse.

Cette dernière démonstration des capacités de la technique dans l'environnement de BA est donc testé sur un point de vol de l'enveloppe pour un avion en configuration de croisière. On considère par contre plusieurs cas de chargements différents et construisons une synthèse robuste en ce point de vol.

7.1.2 Objectifs de l'application

Cette dernière application a pour unique but de démontrer le potentiel de la synthèse. On veut obtenir une solution rapidement tout en réglant les filtres ASE. On ne dispose pas d'une synthèse classique qui inclut les modèles flexibles dans le design. Comme précédemment, on prend comme référence de performance les résultats actuels pour un réglage classique, tout en sachant que l'influence des modèles flexibles et le réglage des filtres ASE est alors pris en compte a posteriori.

7.1.3 Méthodologie suivie

La même méthodologie que celle définie dans la section 1.2.2 du chapitre d'introduction est appliquée ici. Les étapes suivantes sont imposées :

- Définition des requis (section 7.2.1)
- Définition des concepts (section 7.2.2)
- Définition de l'architecture (section 7.2.3)
- Implémentation et optimisations (sections 7.3 et 7.4)
- Tests et vérifications (section 7.5)

Optimisation du design

La méthode utilisée pour l'optimisation du design est toujours une méthode itérative. Les deux points de départ de cette application sont :

- pour le design : celui obtenu dans l'application précédente (Chapitre 6) pour la zone de l'enveloppe de vol concernée,
- pour la condition initiale : les gains feedforward (non linéaires) et feedback de la solution de référence au point de vol concerné.

7.2 Requis et design

7.2.1 Requis sur les performances de la boucle interne

Comme dans les applications précédentes, on commence par lister les requis considérés dans cette étude. Les critères sur les performances désirées appliqués ici sont listés ci-après :

- Requis 1 : voir 3.3.1.
- Requis 2 : voir 3.3.1.
- Requis 3 : voir 3.3.1.
- Requis 4 : voir 3.3.2.
- Requis 5 : voir 3.3.2.
- Requis 6 : voir 3.3.3.
- Requis 7 : voir 3.3.3.
- Requis 8 : voir 3.3.4.
- Requis 9 : voir 3.3.4.
- Requis 10 : voir 3.3.5.

7.2.2 Concepts et canaux de performances

Afin d'assurer les requis listés plus haut, les canaux de performances sont construits en s'appuyant sur la solution choisie pour l'application du chapitre précédent (Section 6.2.2), notamment dans le choix des fonctions de pondérations. La série de concepts choisis pour cette application sont définis ci-après :

- Concept 1 (associé au requis 1) : voir 3.3.1. On considère la structure globale au niveau de l'entrée et de la commande feedforward. On utilise les sorties des modèles d'avion qui n'incluent pas les filtres ASE. Ils sont construits à partir de gains réglables.
- Concept 2 (associé au requis 2) : voir 3.3.1. On ne modifie pas le gain K_p , mais le réglage des filtres ASE est fait par la synthèse.
- Concept 3 (associé au requis 3) : voir 3.3.1.
- Concept 4 (associé au requis 4) : voir 3.3.2.
- Concept 5 (associé au requis 5) : voir 3.3.2.
- Concept 6 (associé au requis 6) : voir 3.3.3.
- Concept 7 (associé au requis 7) : voir 3.3.3.
- Concept 8 (associé au requis 8) : voir 3.3.4. On contrôle la position des pôles avec le suivi de références et non avec l'outil *TuningGoal.Poles*, dû aux modes flexibles mal-amortis et difficilement contrôlables.
- Concept 9 (associé au requis 9) : voir 3.3.4.
- Concept 10 (associé au requis 10) : voir 3.3.5.

On rappelle que les canaux de performances et leurs architectures, sont abordés dans le chapitre 3. L'architecture du concept 6 n'est pas représentée pour réduire la taille et augmenter la lisibilité du schéma du modèle augmenté de la synthèse (Fig. 7.1).

7.2.3 Architecture du modèle augmenté

La figure 7.1 présente le schéma du modèle augmenté pour la synthèse flexible incluant le réglage des filtres ASE. Dans sa généralité, ce schéma ressemble beaucoup au modèle augmenté utilisé pour l'application précédente (Fig. 6.1). La singularité de cette preuve de concept provient de l'ajout des filtres ASE dans les gains de la loi de commande que la synthèse vient optimiser. On rappelle que les canaux de stabilité (marge de phase et de gain) au niveau de la commande en aileron et en rudder ne sont pas représentés sur le schéma augmenté.

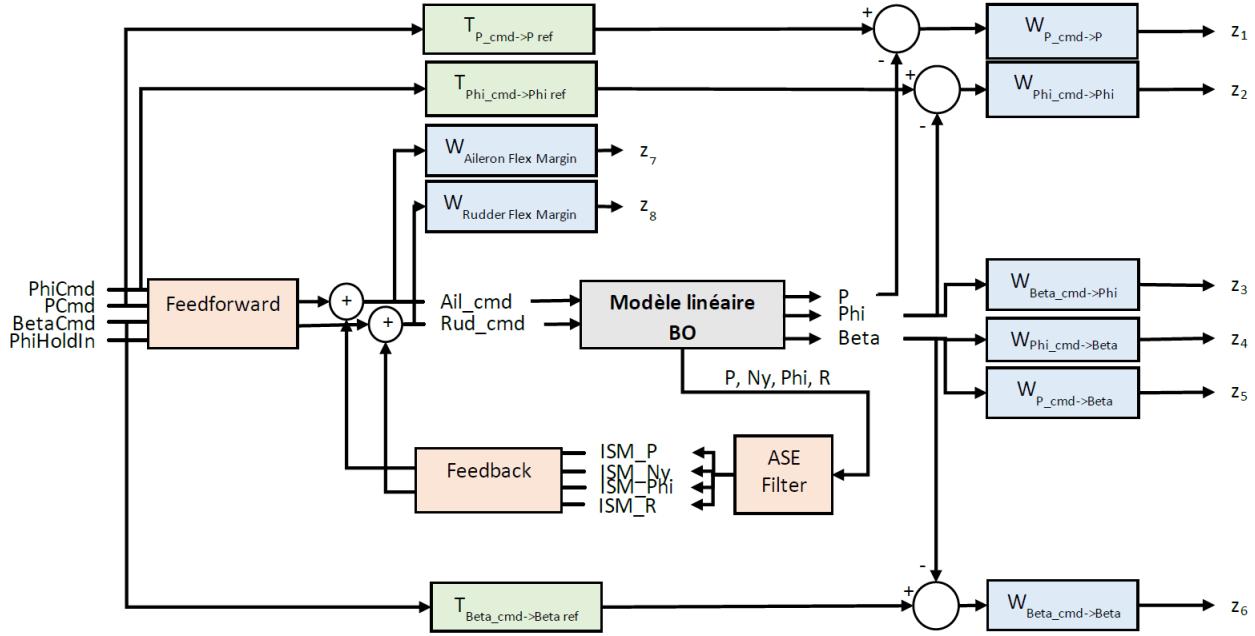


Figure 7.1 Modèle augmenté de la synthèse réglant les filtres ASE

7.3 Explication des cas utilisés pour l'application

La synthèse est appliquée sur un unique point de vol à la configuration de croisière. Le point de vol choisi est un point de l'enveloppe de vol pour lequel des modèles rigides et flexibles sont disponibles. Si l'on reprend la figure 6.3 au chapitre précédent, le point choisi est le numéro 57. L'objectif ici n'est pas de couvrir l'ensemble des conditions de vol possibles pour cette configuration, mais de démontrer que la synthèse de la loi de commande peut s'étendre au réglage des gains des filtres ASE. La commande n'est alors pas séquencée en vitesse et altitude, mais robuste en charge. On prendra 10 cas de chargement pour cette application (Fig. 7.2). Cinq modèles rigides et cinq flexibles sont manipulés.

7.4 Synthèse et optimisations

Afin d'obtenir la solution présentée, le design appliqué est celui assigné au point de vol 57 lors du chapitre précédent. La condition de vol faisait partie d'un groupe de l'enveloppe dont le design avait été optimisé (grâce à l'inclusion des modes flexibles). Il n'est donc pas nécessaire de modifier le réglage précédent pour démontrer la capacité de la synthèse à régler les filtres ASE.

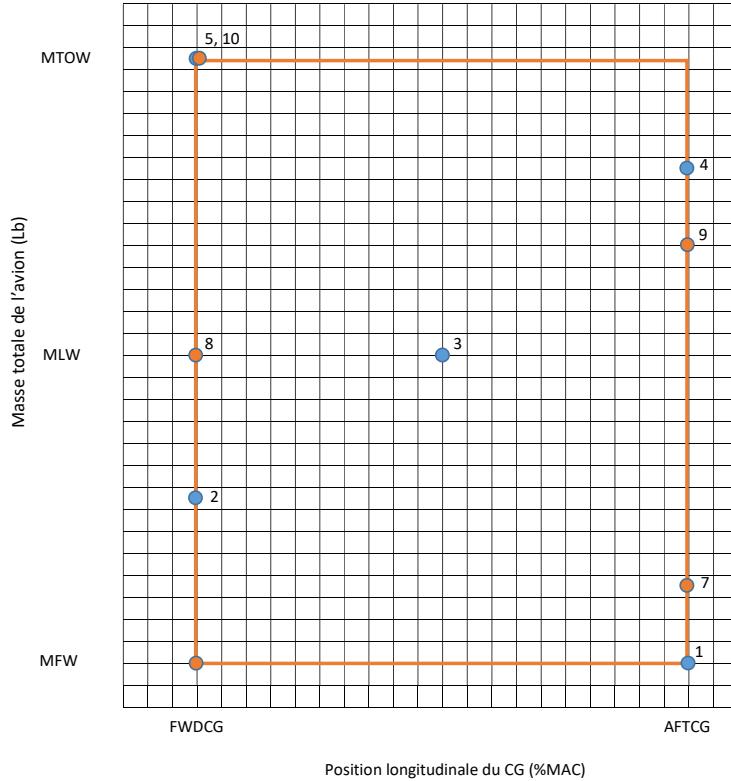


Figure 7.2 Position des configurations de charge sélectionnées dans leur ensemble

7.4.1 Limitations sur les filtres ASE

Si aucune limitation n'est donnée sur les valeurs des gains des filtres, même en conservant la structure, on remarque que l'allure des filtres ASE n'est pas conservée et n'est pas satisfaisante physiquement. En effet, les nouveaux filtres présentent des amplifications de signal (gain en dB positif) sur certaines plages de fréquences, comme le montre la figure 7.3. On choisit de limiter l'amplitudes des gains possibles à $\pm 10\%$ ¹. La figure montre l'allure fréquentielle de nouveaux filtres ASE proposés par une synthèse lors des étapes de design.

7.4.2 Ajout de configurations de charge différentes

Les résultats obtenus sont meilleurs une fois les filtres ASE restreints au voisinage des anciens filtres. On décide d'ajouter plus de modèles (configurations de charge supplémentaires) dans la synthèse. On garde la sélection de modèles rigides de l'application précédente, et

1. La limitation s'impose pour plusieurs raisons : 1) on estime que la solution initiale propose un bon réglage pour l'ensemble des points de vol. Le design appliqué à un point unique pourrait proposer un optimum plutôt éloigné. 2) dans le cas d'amplitude positives, le signal entrant est amplifié. Ceci soulève de grandes inquiétudes pour une éventuelle certification.

choisit cinq modèles flexibles mieux répartis sur l'enveloppe (Fig. 7.2). Cette amélioration permet de prendre en compte des dynamiques initialement non-considérés. Ces modèles de charge éloignées de la sélection initiale ne remplissaient pas les contraintes de marges de stabilité. Dans la figure 7.4 qui présente par un diagramme de Nichols, les allures fréquentielles d'une solution synthétisée, on voit clairement que de nombreux modèles ne respectent pas les conditions de stabilité.

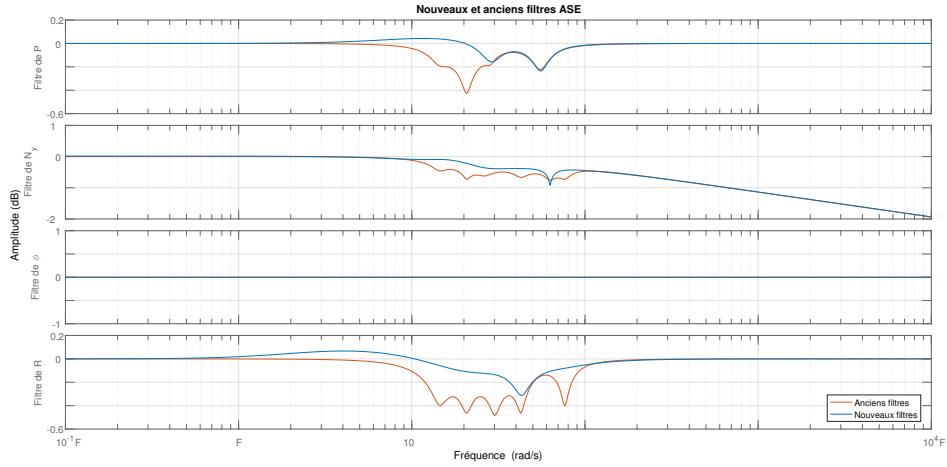


Figure 7.3 Allures fréquentielles des filtres ASE sans restrictions sur les amplitudes des gains

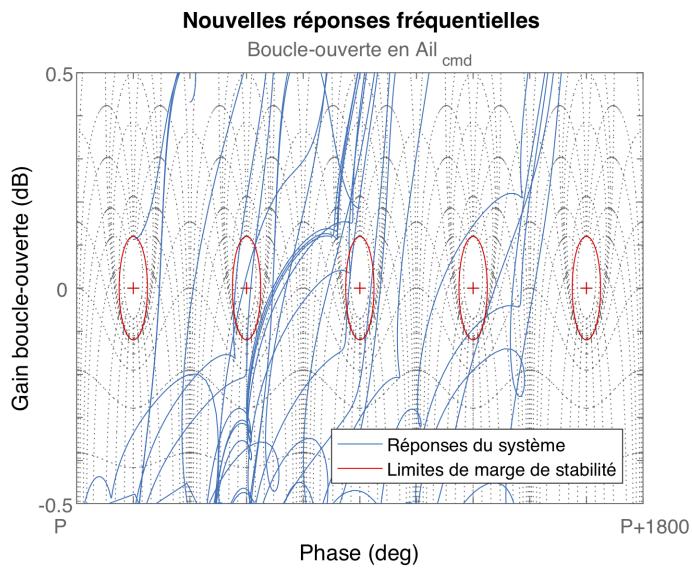


Figure 7.4 Allures fréquentielles et non-respect des marges de stabilité de la solution synthétisée

7.4.3 Augmentation de l'ordre de réduction des modèles flexibles

L'ajout de configurations de charge supplémentaires améliore de beaucoup les marges de stabilité obtenues. Comme la vérification est appliquée aux modèles complet, plus le modèle utilisé pour la synthèse est proche du modèle complet, plus les performances réelles sont assurées. L'augmentation de l'ordre des modèles flexibles permet donc aussi d'améliorer le respect des marges de stabilités, mais allonge le temps nécessaire à la synthèse. Il y a un compromis à faire. Dans ce cas-ci, le nombre de modèles considérés est petit, et la synthèse est rapide (quarantaine de minutes), on peut se permettre de moins réduire l'ordre des modèles flexibles. On passe de 55 états à 115, avec une synthèse qui prend maintenant une soixantaine de minutes .

7.5 Résultats et performances après réglage de la boucle linéaire

Dans ce chapitre, la vérification de la stabilité et des performances peut se permettre de tester les modèles complets pour toutes les configurations de charge disponibles.

Pour apporter une analyse des graphiques, on utilise la notation :

- + : concernant le critère jugé, les nouvelles solutions sont meilleures que les anciennes.
- = : concernant le critère jugé, les nouvelles solutions sont équivalentes aux anciennes.
- - : concernant le critère jugé, les anciennes solutions sont meilleures que les nouvelles.

7.5.1 Evolution de l'amplitude des gains

Les deux figures 7.5 et 7.6 présentent les valeurs des gains de feedback et de feedforward obtenus pour cette application. Les nouvelles amplitudes des gains suivent généralement la tendance proposée initialement. Les gains liés à la commande en *Rudder* sont plus grands pour la nouvelle solution, probablement à cause du gain K_β qui intervient pour les gains (utilisés en non linéaire) de l'ancienne solution. On peut néanmoins statuer que les amplitudes atteintes sont tout-à fait raisonnables.

Les nouveaux filtres ASE et anti-bruits et leurs allures fréquentielles sont affichées dans les deux figures 7.7 et 7.8. L'allure des nouveaux filtres suit aussi de près l'ancienne proposition. Les pics de fréquences à couper sont parfois décalés mais toujours proches du gabarit initial. Il est certain que la limite à $\pm 10\%$ intervient dans ce résultat.

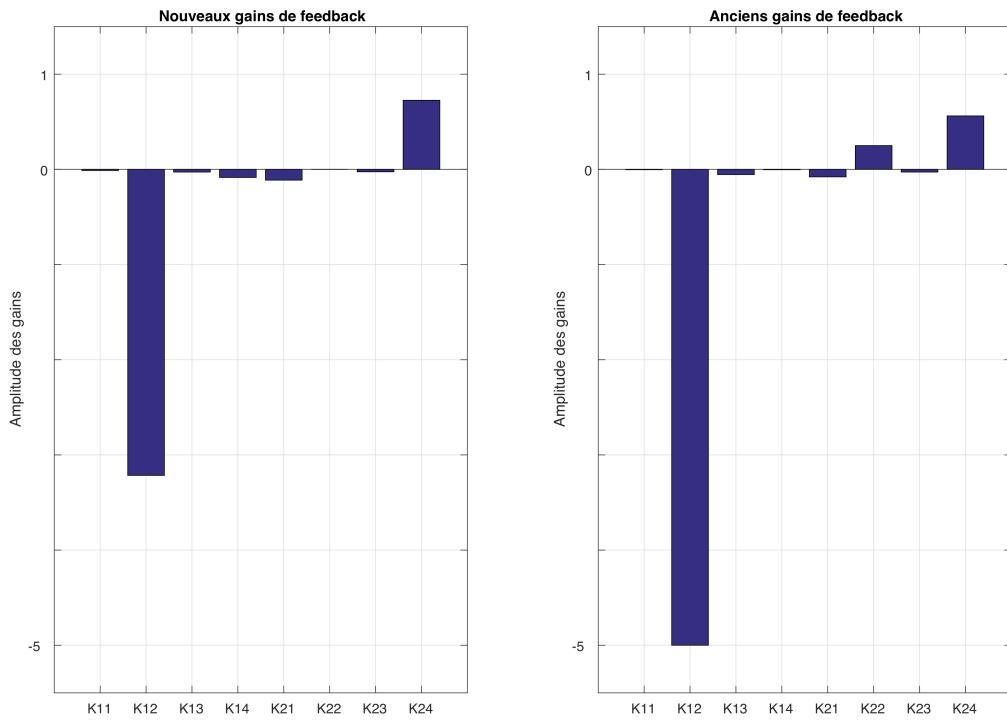


Figure 7.5 Amplitudes des gains de feedback

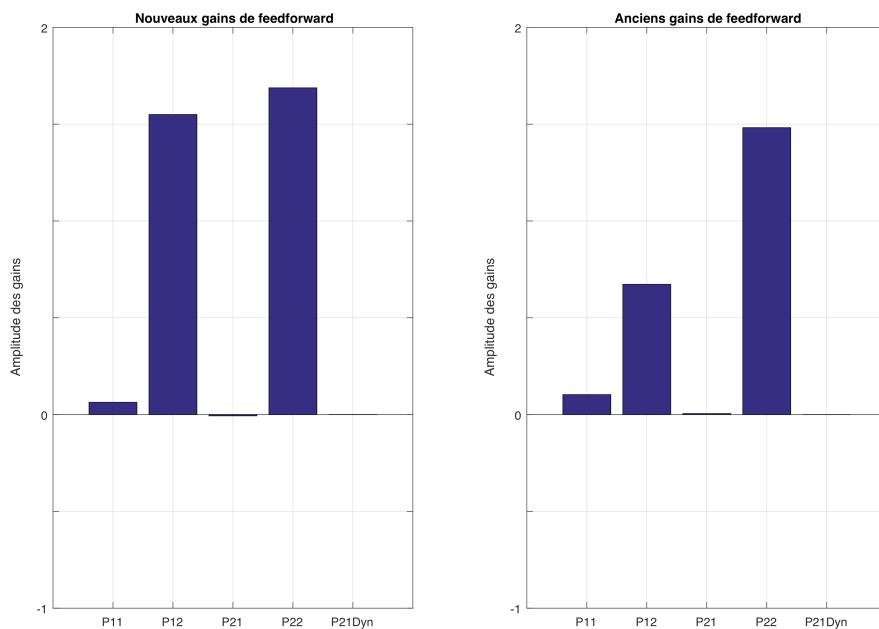


Figure 7.6 Amplitudes des gains de feedforward

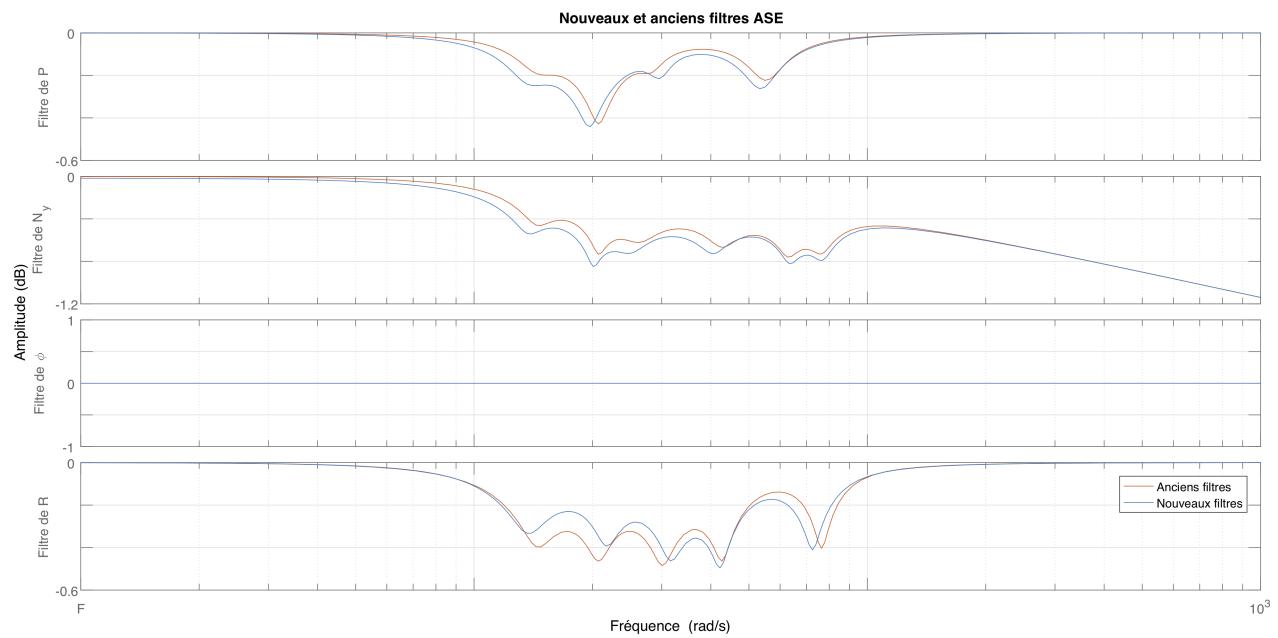


Figure 7.7 Allure fréquentielle des filtres ASE

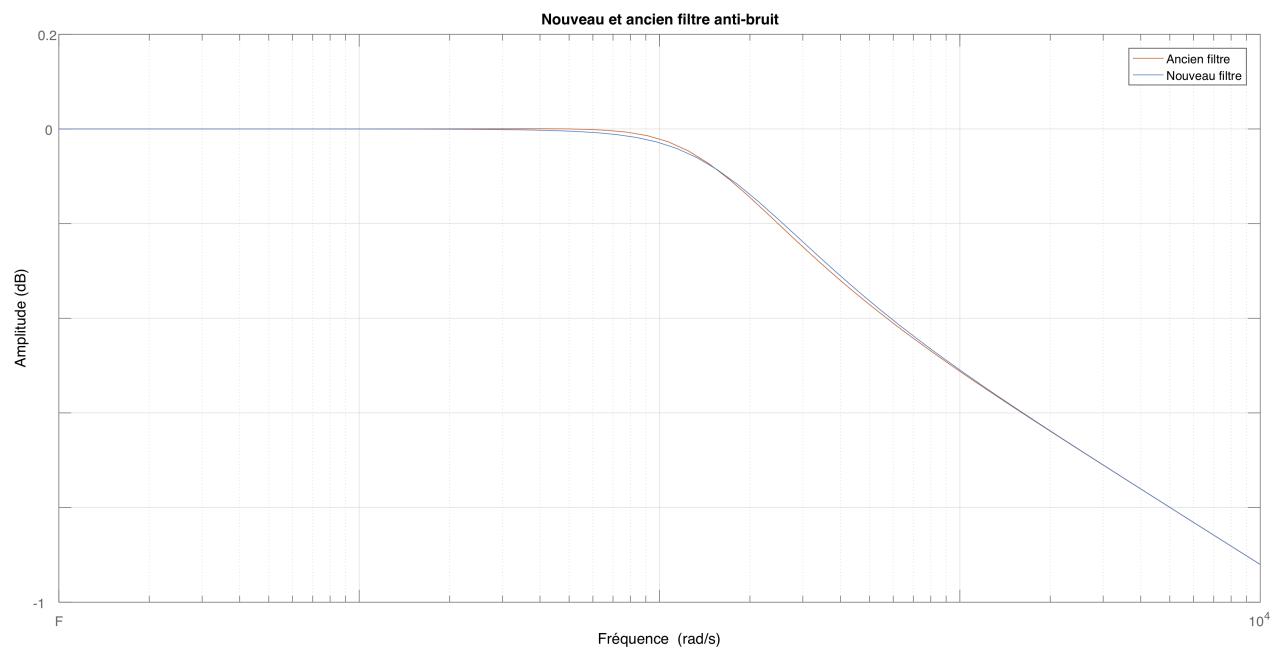


Figure 7.8 Allure fréquentielle du filtre anti-bruit

7.5.2 Suivi de références

Comme pour les chapitres précédents, on commence à présenter les réponses temporelles des transferts de P_{cmd} , ϕ_{cmd} et β_{cmd} vers les sorties respectives P , ϕ et β . Ces tracés permettent de juger des capacités de suivi de références.

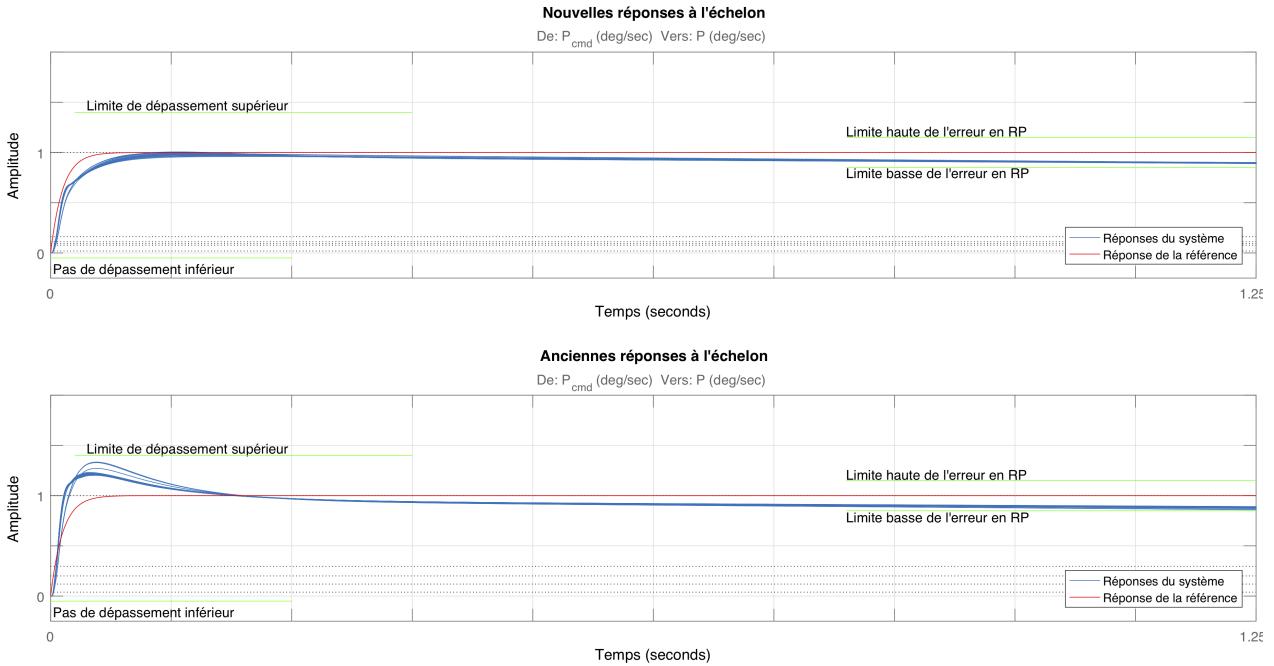


Figure 7.9 Performances du suivi de référence pour une commande de P_{cmd} vers P

Pour la figure 7.9, les commentaires sont les suivants :

- + : La nouvelle solution suit mieux l'enveloppe de la réponse désirée, c'est-à-dire un transfert de premier ordre sans dépassement. Les réponses en RP sont bien plus proches les unes des autres.
- = : Les deux solutions sont équivalentes en matière d'erreur en RP.

Pour la figure 7.10, les commentaires sont les suivants :

- + : Les nouvelles réponses en RP ont une erreur plus petite.
- = : Les deux réponses ont l'allure d'un transfert de premier ordre, et respectent la limite de constante de temps.
- - : Les anciennes réponses en RP sont plus rapides.

Pour la figure 7.11, les commentaires sont les suivants :

- + : En RP, la nouvelle solution respecte l'erreur limite acceptée et est mieux centrée autour de la référence. Le coefficient K_β intervient ici et réduit les amplitudes des anciennes réponses temporelles.

— = : Les deux solutions ont l'allure d'un transfert de second ordre qui respecte les contraintes de dépassement et de rapidité imposées.

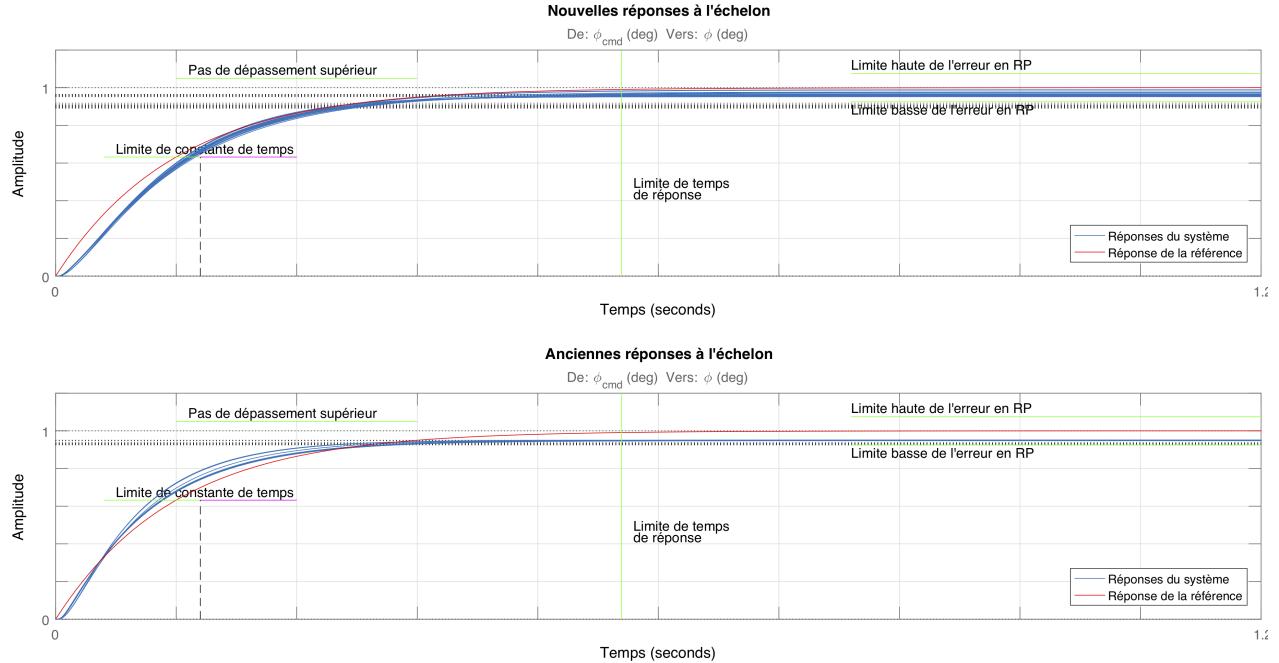


Figure 7.10 Performances du suivi de référence pour une commande de ϕ_{cmd} vers ϕ

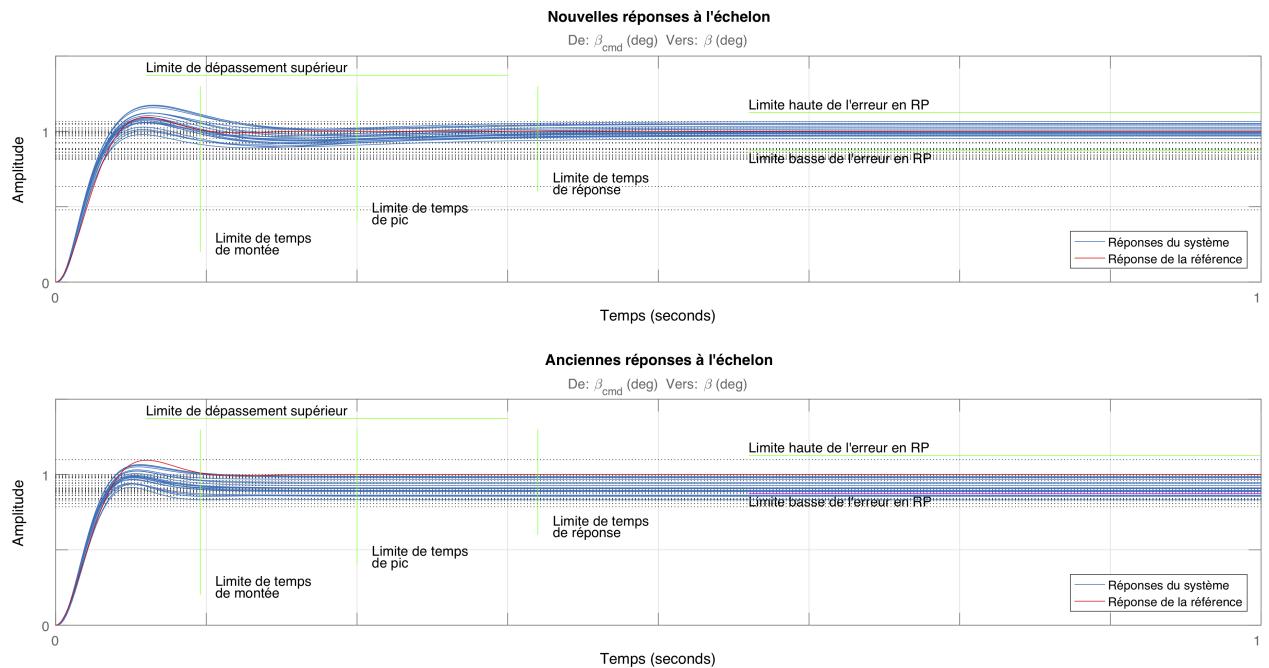


Figure 7.11 Performances du suivi de référence pour une commande de β_{cmd} vers β

7.5.3 Découplages de signaux

On continue l'analyse des résultats avec les réponses temporelles des transferts croisés de P_{cmd} , ϕ_{cmd} et β_{cmd} vers les sorties respectives β et ϕ . Ces tracés, des figures 7.12 à 7.14, permettent de juger des capacités de découplage des sorties.

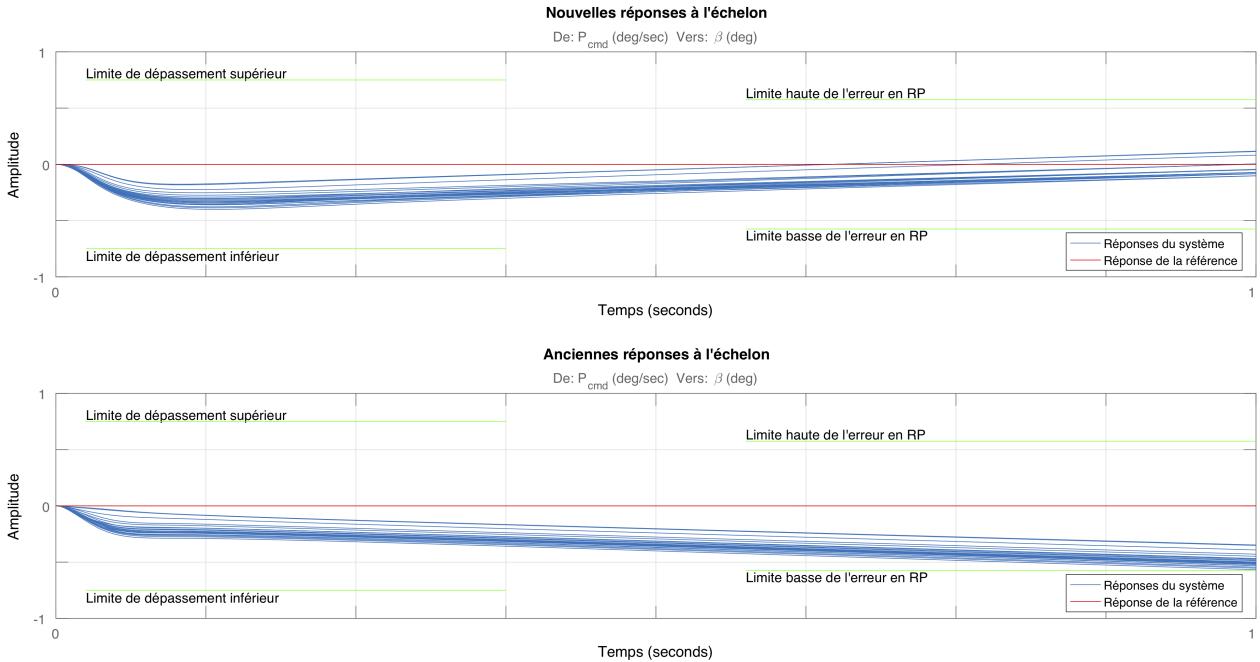


Figure 7.12 Performances du découplage de signaux pour une commande de P_{cmd} vers β

Pour la figure 7.12, les commentaires sont les suivants :

- + : En RP, les nouvelles solutions sont plus proches de la référence, et semblent légèrement moins écartées.
- - : Les nouvelles solutions ont un plus grand dépassement.

Pour la figure 7.13, les commentaires sont les suivants :

- = : Les deux solutions ont sensiblement la même erreur en RP.
- - : Les nouvelles solutions ont un plus grand dépassement inférieur.

Pour la figure 7.14, les commentaires sont les suivants :

- + : Les nouvelles réponses sont plus centrées vers la référence en RP.
- - : Les nouvelles réponses ont un dépassement inférieur plus grand, et des valeurs en RP plus espacées.

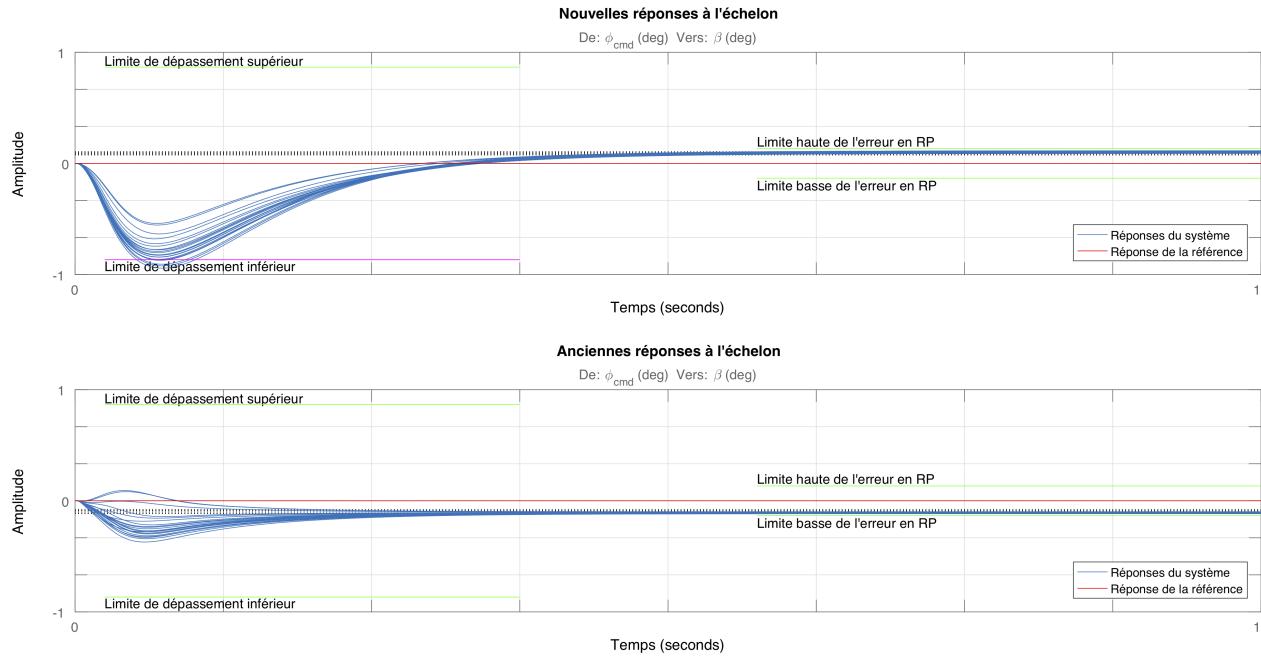


Figure 7.13 Performances du découplage de signaux pour une commande de ϕ_{cmd} vers β

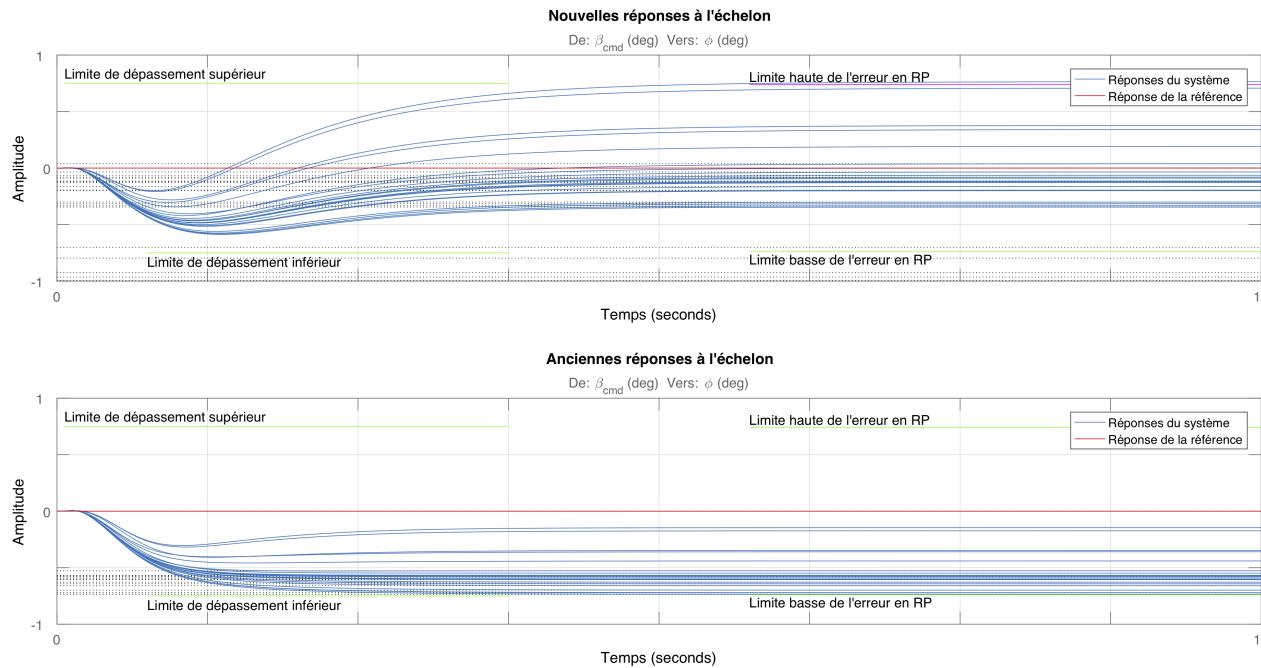


Figure 7.14 Performances du découplage de signaux pour une commande de β_{cmd} vers ϕ

7.5.4 Position des pôles de la boucle fermée

Les deux figures 7.15 et 7.16 nous donne le placement des pôles en BF dans le plan complexe. La première ajoute le gabarit requis pour situer les bons et moins bons positionnement. La seconde ajoute la position des pôles en BO ainsi que leur déplacement lors de la fermeture de boucle.

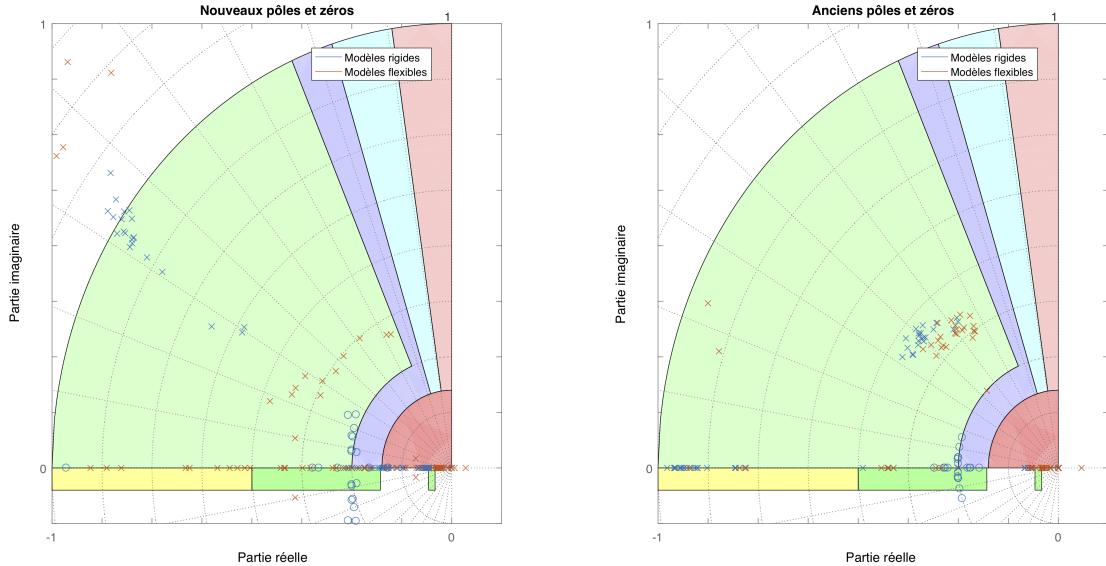


Figure 7.15 Positionnement des pôles et des zéros de la BF

Pour la figure 7.15, les commentaires sont les suivants :

- + : Les pôles du mode Roulis Pur sont de manière générale plus rapides dans la nouvelle solution.
- = : Les modes roulis hollandais sont tous assignés à la zone verte du gabarit. Le placement du mode Spiral est à peu près équivalent dans les deux solutions.
- - : Les anciens pôles du mode Roulis Hollandais sont plus recentrés autour d'une seule dynamique.

Pour la figure 7.16, les commentaires sont les suivants :

- + : Les nouveaux pôles en BF décalés plus sensiblement sur la gauche implique que la nouvelle solution propose de manière générale des modes flexibles plus amortis en BF.

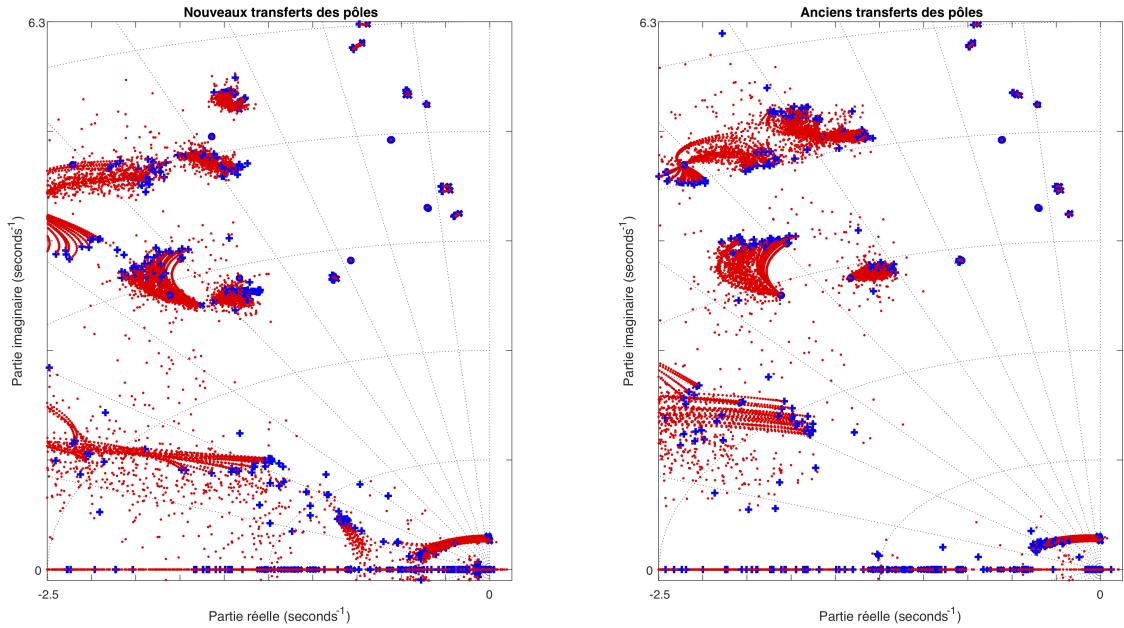


Figure 7.16 Positionnement des pôles avant et après fermeture de la boucle

7.5.5 Marges de stabilité

Jugeons du respect des contraintes de stabilité imposées aux modèles rigides et flexible avec les quatres figures suivantes (Fig. 7.17 à Fig. 7.20). On présente d'abord les diagrammes de Nichols et les marges de phase et de gain, puis on affichera les marges d'amplitude avec les diagrammes de Bode. Pour la figure 7.17, les commentaires sont les suivants :

- + : Les nouvelles réponses fréquentielles sont plus ordonnées et interceptent moins souvent les ellipses limites. Les nouvelles allures fréquentielles semblent aussi plus ordonnées.

Pour la figure 7.18, les commentaires sont les suivants :

- - : Il n'y a aucune inclusions dans les ellipses limites pour l'ancienne solution. Par contre pour la nouvelle solution, on compte deux intersections dont un modèle qui vient mal tangenter la deuxième ellipse de droite. Ce défaut est dû à une optimisation de la solution proposée qui ne satisfait plus les contraintes avec le passage à l'ordre complet. Les intersections restent très proche de l'ellipse limite est ne constituent pas une grande perte en marge de gain et de phase.

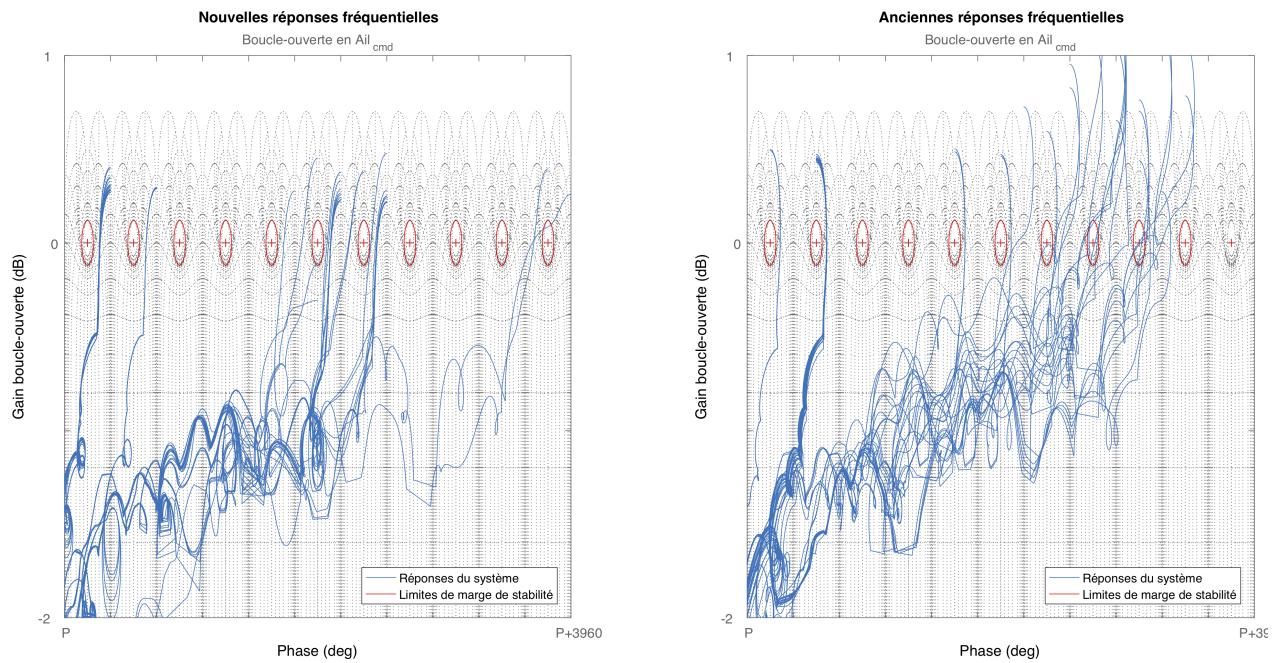


Figure 7.17 Réponses fréquentielles et diagramme de Nichols de la BO de l'aileron

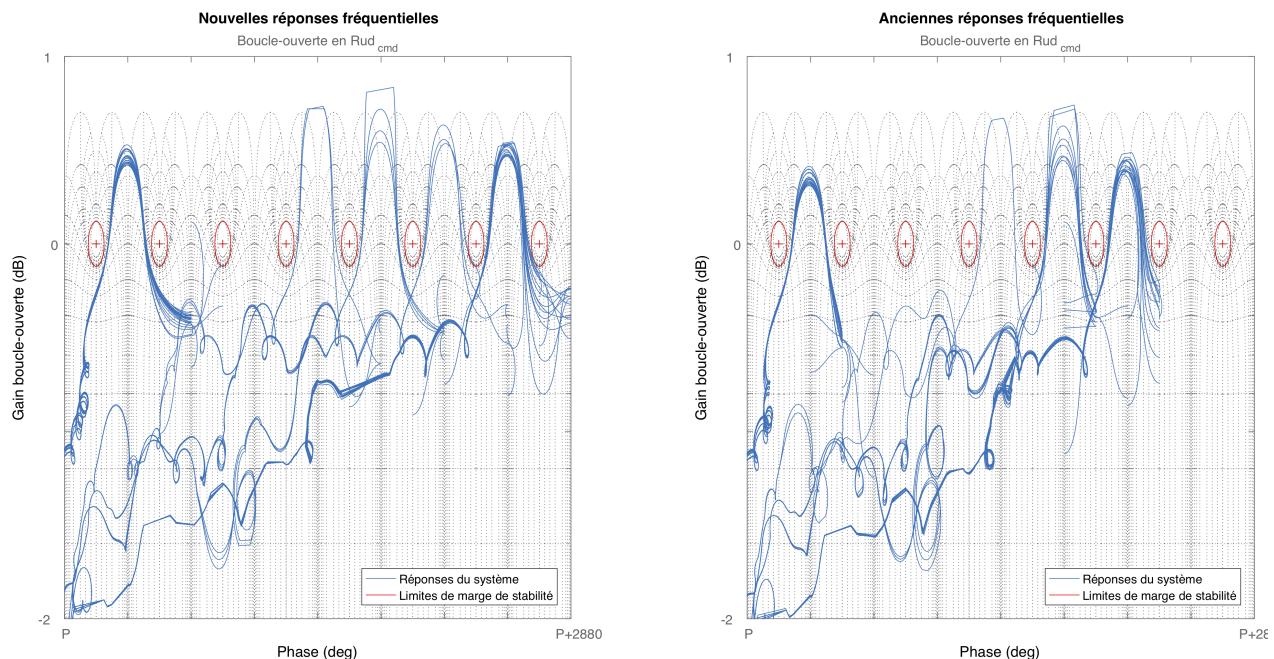


Figure 7.18 Réponses fréquentielles et diagramme de Nichols de la BO du rudder

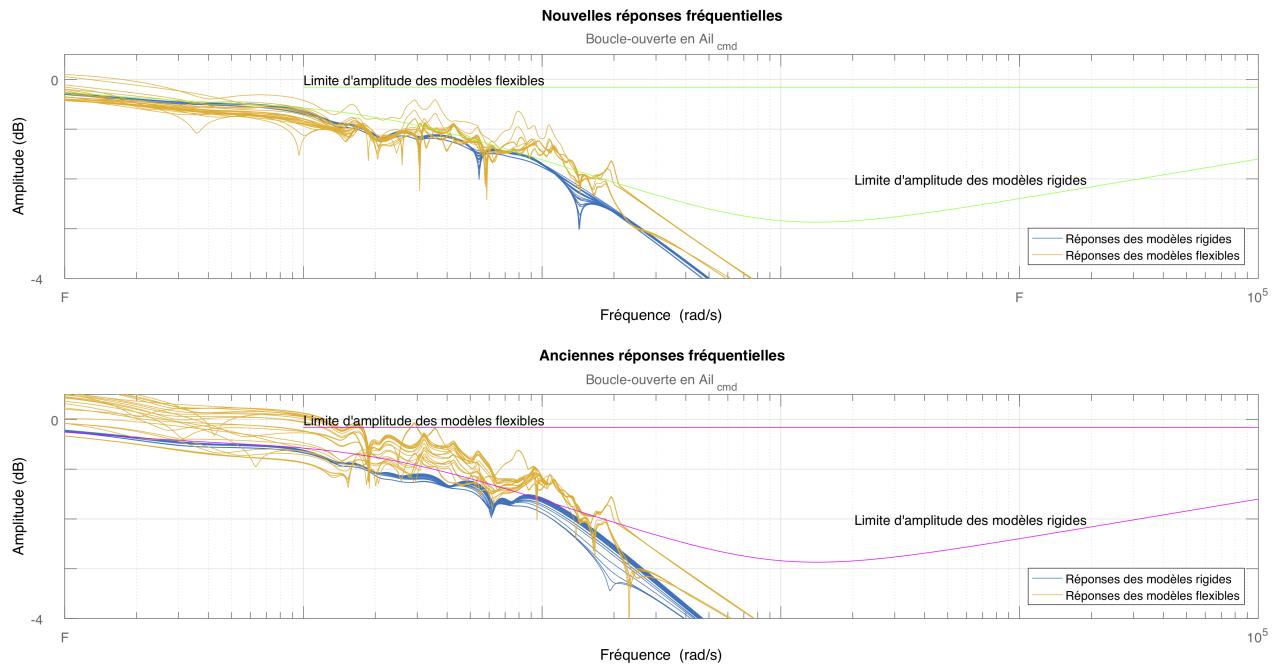


Figure 7.19 Réponses fréquentielles et marges de stabilité pour les modèles

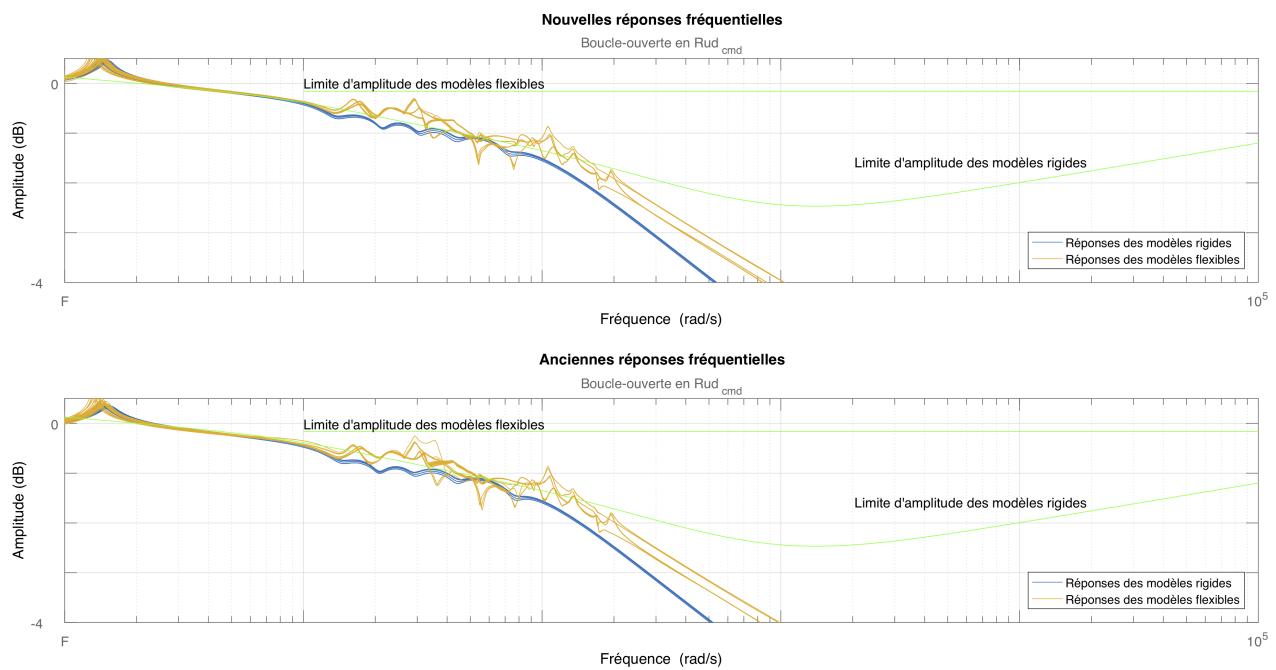


Figure 7.20 Réponses fréquentielles et marges de stabilité pour les modèles

Pour la figure 7.19, les commentaires sont les suivants :

- + : Toutes les nouvelles réponses fréquentielles respectent les limites d'amplitude (autant flexible que rigide). On voit que la contrainte est bien prise en compte.

Pour la figure 7.20, les commentaires sont les suivants :

- = : Les allures fréquentielles des deux solutions se ressemblent. Toutes deux vérifient les marges d'amplitude imposées.

7.5.6 Efforts des actionneurs

Finalement la dernière figure 7.21 propose les réponses temporelles des efforts de commande demandés pour des échelons de P_{cmd} , ϕ_{cmd} et β_{cmd} .

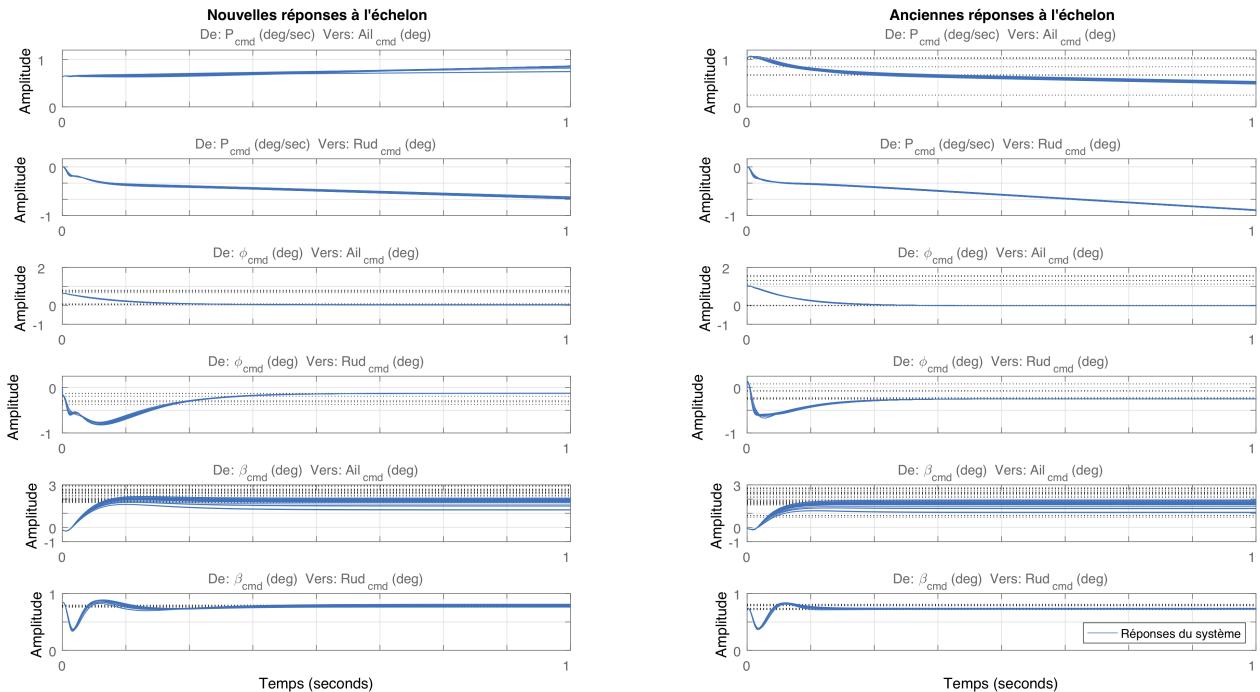


Figure 7.21 Efforts des actionneurs pour des commandes de P_{cmd} , ϕ_{cmd} et β_{cmd}

Pour la figure 7.21, les commentaires généraux sont les suivants :

- + : L'effort de l'aileron semble légèrement moindre pour la nouvelle solution.
- = : Les allures et amplitudes des efforts des actionneurs sont plutôt semblables.
- - : Pour la nouvelle solution, la commande du rudder semble avoir des dépassements inférieurs plus grands.

7.6 Conclusion

Après les deux premières applications, on a présenté ici une augmentation de la méthode de synthèse au réglage des filtres ASE. Les gains obtenus et les graphiques des performances (stabilité, suivi de consigne, etc.) ont été donnés pour juger des résultats synthétisés.

La solution proposée est globalement meilleure que la solution de référence. Les contraintes de stabilité sont mieux respectées et les performances temporelles proposent un bon compromis entre suivi et découplage. Les nouveaux filtres ASE sont proches du réglage initial mais pas identiques, leur réglage est bien pris en compte dans la synthèse.

Néanmoins, il est bon de noter que bien que les performances soient meilleures, le système ne subit pas la contrainte du séquencement. En effet, les gains de la solution de BA dépendent de leur voisins. Finalement, l'importance du réglage de ces filtres avec le contrôleur ne peut être conclue à la suite de cette application. L'ensemble considéré ici se limite à un seul point de vol, non à l'enveloppe complète. On statue cependant que le réglage des filtres ASE peut représenter une ouverture très intéressante (réduction des coûts, amélioration du réglage) qui mérite d'être développée.

Pour conclure, la solution synthétisée est vraiment intéressante car performante et obtenue dans un temps relativement court.

Le réglage de cette synthèse a été faite en quatre itérations au total. Cette information permet de statuer sur deux faits :

- il ne faut donc pas beaucoup de travail pour obtenir des résultats très convaincants,
- et, si il existe quelques imperfections dans les résultats présentés, ils pourront être améliorés par quelques itérations supplémentaires et nouveaux compromis.

L'expérience démontre une nouvelle étendue des possibilités de la méthode qui ne pouvaient être entreprises avec l'ancien outil de BA. Les performances sont satisfaisantes. Quelques améliorations pourraient être faites dans le placement des pôles du roulis hollandais. On a noté aussi que les marges de stabilité ne sont parfois pas validées sur le modèle complet. On pourrait essayer de rassembler les dynamiques réduites dans un modèle incertain comme dans le Document de Référence (DR) "Computing Optimal Uncertainty Models from Frequency Domain Data" (Hindi et al. (2002)).

On pourrait alors étendre encore cette application au réglage de la boucle interne avec les filtres ASE sur l'ensemble de l'enveloppe de vol.

CHAPITRE 8 CONCLUSION

Ce chapitre est le dernier du mémoire de maîtrise. Premièrement, il apporte des conclusions sur l'ensemble des travaux (Section 8.1). En seconde section 8.2, on liste les limitations rencontrées durant l'étude et ses applications. Finalement la dernière section 8.3 ouvre sur des améliorations futures.

8.1 Synthèse des travaux

Les travaux menés pendant cette étude démontre que la synthèse H_∞ structurée et les techniques de résolution associées sont aujourd’hui suffisamment avancées pour aborder des problèmes industriels complexes. Des nouvelles solutions stables, performantes et respectant les contraintes de structures, peuvent être produites rapidement.

De plus, les travaux prouvent que la méthode a la capacité de prendre en compte de nouveaux requis si ils peuvent être exprimés par des canaux de performances à optimiser. Le réglage de la loi de commande peut intégrer les dynamiques flexibles et assurer une marge de stabilité sur ces modèles à risques. Des modèles rigides et flexibles sont utilisés simultanément dans une synthèse unique pour éviter des itérations de réglages supplémentaires si les perturbations souples ne sont considérées qu'a posteriori.

L'étendue des capacités de la technique est démontrée dans la dernière application, quand la synthèse optimise aussi le placement des filtres ASE pour couper les fréquences des modes flexibles les plus contraignants, en plus du réglage de la loi de commande interne.

Les expériences menées dans ce mémoire sont la preuve que la méthode peut produire des lois de commandes certifiables pour un avion civil, et que l'exploration de ces capacités par les industries peut mener à une plus-value très intéressante en terme de performances atteintes et garanties, en terme de temps alloué et donc de réduction des coûts.

8.2 Limitations de la solution proposée

Les limitations de la solution concernent l'ordre du modèle augmenté, donc :

- plus il y a de canaux de performances, plus il y a de contraintes,
- plus il y a de modèles, plus il y a de variations de dynamiques sur l'enveloppe de vol,

- plus les ordres des modèles sont grands, plus il y a de dynamiques secondaires,
- plus la boucle interne à régler est grande, plus il y a de gains réglables.

Cependant toutes ces améliorations augmentent l'ordre du modèle manipulé par la synthèse et allonge considérablement le temps nécessaire pour résoudre l'optimisation. Finalement, un compromis doit être fait entre le temps à allouer et les objectifs à atteindre.

8.3 Améliorations futures

Lorsque les modèles flexibles seront disponibles sur la totalité de l'enveloppe de vol, il sera intéressant de réitérer l'expérience du chapitre 6 afin de voir si le réglage est vraiment différent.

Si les outils de résolution numérique de Matlab étaient plus rapides, on pourrait aussi imaginer procéder au réglage des gains et des filtres ASE sur l'ensemble des conditions de vol. À moins de réduire suffisamment l'ordre des modèles considérés, cette synthèse nécessiterait aujourd'hui plusieurs jours à s'exécuter.

Une dernière piste d'amélioration possible serait d'implémenter un outil de réglage de la synthèse en temps réel. Un outil similaire est proposé par une librairie Matlab récente Model Predictive Control (MPC)¹ bar le biais d'une fonction de réglage des fonctions de performances en temps réel². Pour une commande prédictive, il permet d'ajuster le réglage en modifiant les fonctions de pondérations et les contraintes pendant l'exécution de la synthèse. Il pourrait être intéressant de développer un outil pour la synthèse H_∞ , qui serait plus interactif pour régler les fonctions de pondérations et voir en temps réel l'impact du changement de design. Étant donné la taille des modèles, l'aspect "temps réel" pourrait ne pas fonctionner mais la librairie pourrait être utile pour faire le réglage de quelques modèles pour l'allure générale des contraintes à appliquer.

1. Documentation consultée le 23/11/2018 à l'adresse url suivante : <https://www.mathworks.com/help/mpc/>.

2. Documentation consultée le 23/11/2018 à l'adresse url suivante : <https://www.mathworks.com/help/mpc/ug/run-time-weight-tuning.html>.

RÉFÉRENCES

- D. Alazard, A. Bucharles, G. Ferreres, J. Magni, et S. Prudhomme, “An integrated methodology for flexible aircraft control design”, dans *Structural Aspects of Flexible Aircraft Control (RTO MP-36)*, R. N. 2000, éd., Oct. 1999.
- P. Apkarian et D. Noll, “Non-smooth H_∞ synthesis”, *IEEE Transactions on Automatic Control*, vol. 51, pp. 229–244, 2006.
- , “The H_∞ control problem is solved”, *AerospaceLab Journal, ONERA*, pp. 1–11, 2017.
- P. Apkarian, P. Gahinet, et C. Buhr, “Multi-model, multi-objective tuning of fixed-structure controllers”, dans *2014 European Control Conference (ECC)*, 2014, pp. 856–861.
- D. Arzelier et D. Peaucelle, “Robust impulse-to-peak synthesis : Application to the control of an aerospace launcher”, dans *16th IFAC Symposium on Automatic Control in Aerospace 2004, Saint-Petersburg, Russia, 14-18 June 2004*, série IFAC Proceedings Volumes, vol. 37, no. 6, 2004, pp. 967–972.
- G. Balas, R. Chiang, A. Packard, et M. Safonov, “Robust control toolbox reference R2014a, in the MathWorks Inc.” 2014.
- D. Bates et I. Postlethwaite, *Robust Multivariable Control of Aerospace Systems*, série Control and Simulation. Delft University Press, 2002.
- S. Bennani et F. Ankersen, “Robust attitude control design for the BIOMASS satellite (earth explorer core mission candidate)”, dans *IFAC Proceedings Volumes*, vol. 44, no. 1, 2011, pp. 5130–5135.
- J.-M. Biannic, “Commande modale : Application au pilotage d’un avion”, *Techniques de l’ingénieur Régulation et commande des systèmes asservis*, 2013.
- M. Chilali et P. Gahinet, “ H_∞ design with pole placement constraints : an lmi approach”, *IEEE Transactions on Automatic Control*, vol. 41, pp. 358–367, 1996.
- F. H. Clarke, *Optimization and Nonsmooth Analysis*. Society for Industrial & Applied Mathematics, U.S, 1987.
- R. da Silva de Aguiar, P. Apkarian, et D. Noll, “Structured robust control against mixed uncertainty”, *IEEE Transactions on Control Systems Technology*, vol. 26, pp. 1771–1781,

2018.

G. de Abreu, V. L. Jr., et M. J. Brennan, “Mixed H_2/H_∞ control of a two-floors building model using the linear matrix inequalities approach”, dans *20th International Congress of Mechanical Engineering - COBEM 2009*, 2009.

A. Falcoz, M. Watt, M. Yu, A. Kron, P. Prathyush, D. Bates, F. Ankersen, et L. Massotti, “Intergrated control and structure design framework for spacecraft applied to BIOMASS satellite”, dans *IFAC Proceedings*, vol. 46, no. 19, 2013, pp. 13–18.

C. Favre, “Fly-by wire for commercial aircraft : the airbus experience”, *International Journal of Control*, vol. 59, no. 1, pp. 139–157, 1994.

Aircraft Weight and Balance Handbook, Federal Aviation Administration, 2016, no. FAA-H-8083-1B.

G. Flandin. (consulté le 23.11.18) M2HTML : Documentation system for matlab in HTML. En ligne : <https://www.artefact.tk/software/matlab/m2html/>

P. Gahinet et P. Apkarian, “A linear matrix inequality approach to H_∞ control”, *international Journal of Robust and Nonlinear Control*, vol. 4, pp. 421–448, 1994.

—, “Decentralized and fixed-structure H_∞ control in MATLAB”, dans *Decision and Control and European Control Conference (CDC-ECC), 50th IEEE Conference*, 2011.

—, “Structured H_∞ synthesis in matlab”, *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 1435–1440, 2011, 18th IFAC World Congress.

P. Gahinet et M. A. Turevskiy. (consulté le 23.11.18) Control system tuning in simulink made easy. En ligne : <http://www.codelooker.com/dfilec/9208controlsystemtuninginsimulinkmadeeasy.pdf>

M. Ganet-Schoeller et J. Desmariaux, “Structured H_∞ synthesis for flexible launcher control”, *IFAC-PapersOnLine*, vol. 49, 2016.

H. Gao, X. Yang, et P. Shi, “Multi-objective robust H_∞ control of spacecraft rendezvous”, *IEEE Control Systems Society*, vol. 17, pp. 794–802, 2009.

S. Gumussoy et M. L. Overton, “Fixed-order h_∞ controller design via HIFOO, a specialized nonsmooth optimization package”, dans *2008 American Control Conference*, 2008, pp. 2750–

2754.

S. Gumussoy, D. Henrion, M. Millstone, et M. L. Overton, “Multiobjective robust control with HIFOO 2.0”, dans *Proceedings of the IFAC Symposium on Robust Control Design, Haifa*, 2009.

H. Hindi, C.-Y. Seong, et S. Boyd, “Computing optimal uncertainty models from frequency domain data”, dans *Proceedings of the 41st IEEE Conference on Decision and Control*, 2002, pp. 2898–2905.

S. Joshi et A. Kelkar, “On longitudinal control of high speed aircraft in the presence of aeroelastic modes”, NASA, Rapp. tech., 1996.

M. Knoblauch, D. Saussié, et C. Bérard, “Structured H_∞ control for a launch vehicle”, dans *Proceedings of the American Control Conference*, 2012, pp. 967–972.

R. L. Kosut, H. Salzwedel, et A. Emami-Naeini, “Robust control of flexible spacecraft”, *Journal of Guidance, Control, and Dynamics*, vol. 6, pp. 104–111, 1983.

M. Kozek et A. Schirrer, éds., *Modeling and Control for a Blended Wing Body Aircraft : A Case Study*. Springer, 2015.

A. Kron et J. de Lafontaine, “The generalized (μ,μ) -iteration illustrated by flexible aircraft robust controller reduction”, dans *IFAC Proceedings Volumes*, vol. 37, no. 6, Jun. 2004, pp. 701–706.

A. Kron, J. de Lafontaine, et D. Alazard, “Robust 2-DOF H-Infinity controller for highly flexible aircraft : Design methodology and numerical results”, *Canadian Aeronautics and Space Journal*, vol. 49, no. 1, pp. 19–29, 2003.

A. Kron, V. S. Bilodeau, et F. Ankersen, “Enhanced linear fractional transformation : a matlab toolbox for space system modelling and controller analysis and synthesis”, dans *19 IFAC Symposium on Automatic Control in Aerospace (2003)*, Sept. 2013.

A. Kron, “Conception de lois de commande Fly-By-Wire robustes pour avions de transport civil dont la structure est considérée flexible”, Thèse de doctorat, Université de Sherbrooke, 2004.

L. Lavigne, “Outils d’analyse et de synthèse des lois de commande robuste des systèmes dynamiques plats”, Thèse de doctorat, Université Sciences et Technologies - Bordeaux I, 2003.

H. Lhachemi, D. Saussié, et G. Zhu, “A robust and self-scheduled longitudinal flight control system : a multi-model and structured H_∞ approach”, dans *AIAA Guidance, Navigation and Control Conference*, 2014, p. 601.

—, “A structured H_∞ -based optimization approach for integrated plant and self-scheduled flight control system design”, dans *Aerospace Science and Technology*, 2015, pp. 30–38.

—, “Flutter suppression for a two degree of freedom aeroelastic wing section : a structured H-infinity-based gain-scheduling approach with explicit hidden coupling terms handling”, dans *AIAA Guidance, Navigation, and Control Conference*, 2017, p. 1735.

J.-F. Magni, S. Bennani, et J. Terlouw, “Robust flight control : A design challenge”, dans *Lecture Notes in Control and Information Sciences*, vol. 224. Springer-Verlag Berlin Heidelberg, 1997.

J.-F. Magni, *Robust Modal Control with a Toolbox for Use with MATLAB* ®. Springer US, 2002.

G. Puyou et P. Eezerzere, “Tolerance of aircraft longitudinal control to the loss of scheduling information toward a performance oriented approach”, *IFAC Proceedings Volumes 45.13*, pp. 393–399, 2012.

G. Puyou, G. Ferreres, C. Chiappa, et P. Menard, “A multi-objective method for flight control law design”, dans *Xollection of Technical Papers - AIAA Guidance, Navigation, and Control Conference*, 2004.

M. G. Safonov, K. C. Goh, et J. H. Ly, “Control system synthesis via bilinear matrix inequalities”, *Proceedings of American Control Conference*, pp. 45–49, 1994.

D. Saussié, Q. Barbès, et C. Bérard, “Self-scheduled and structured H_∞ synthesis : A launch vehicle application”, dans *Proceedings of the American Control Conference*, 2013, pp. 1590–1595.

D. D. Vito, A. Kron, J. de Lafontaine, et M. Lovera, “A matlab toolbox for LMI-based analysis and synthesis of LPV/LFT self-scheduled H_∞ control systems”, dans *2010 IEEE International Symposium on Computer-Aided Control System Design*, 2010, pp. 1397–1402.

F. Yang, M. Gani, et D. Henrion, “Fixed-order robust H_∞ controller design with regional pole assignment”, *IEEE Transactions on Automatic Control*, vol. 52, no. 10, pp. 1959–1963, 2007.

G. Zames, “Feedback and optimal sensitivity : Model reference transformations, multiplicative seminorms, and approximate inverses”, *IEEE Transactions on Automatic Control*, vol. AC-26, no. 2, pp. 301–320, 1981.

Zhou, Kemin, J. C. Doyle, et K. Glover, *Essentials of robust control*, Vol. 40. Prentice Hall, New Jersey, 1996.

Zhou, Kemin, et J. C. Doyle, *Essentials of robust control*, Vol. 104. Prentice Hall, New Jersey, 1998.

ANNEXE A RAPPELS MATHÉMATIQUES

Notations utilisés :

- i ou j : l'imaginaire pur
- I : la matrice identité (utilisation d'un indice pour préciser l'ordre)
- ω : une pulsation naturelle en radian/seconde
- s : variable complexe définie par $s = i\omega$
- $P(s), G(s) \dots$: la fonction de transfert du système LTI P, G, \dots
- λ_i : i ème valeur propre
- σ : une valeur singulière. $\bar{\sigma}$ représente la valeur singulière maximale.
- M : une matrice (en général)
- \bar{M} : le conjugué de la matrice M défini par : pour tous les (i, j) $\bar{m}_{i,j} = \bar{m}_{i,j}$
- M^T : la transposée de la matrice M définie par : pour tous les (i, j) $m_{i,j}^T = m_{j,i}$
- M^* : est le conjugué transposé de M , $M^* = \bar{M}^T$. Le produit M^*M est appelé produit hermitien de M .

Fonction sup (borne supérieure) : Soit un ensemble partiellement ordonné E , soit P une partie de E , la borne supérieure de P , notée $\sup(P)$ est, s'il existe, le plus petit des majorants de P . On dira que : $M = \sup(F)$ si $x \leq M$ pour tout x de P , et si pour tout y de E et x de P , tel que $x \leq y$, alors $M \leq y$.

Racine des Moindres Carrés (RMC) : La RMC est une évaluation de la moyenne quadratique de la différence entre les mesures y_i et les prédictions des sorties f_i . On a :

$$RMC = \sqrt{\sum_{i=1}^N (y_i - f_i)^2} = \sqrt{\sum_{i=1}^N (r_i)^2}$$

Norme H_∞ : La norme H_∞ de la LFT $F(P, K)$ (P le système et K le contrôleur) est définie par : $\|F(P, K)\|_\infty = \sup_{\omega}(\bar{\sigma}(F(P, K)(i\omega)))$.

ANNEXE B RAPPELS DE COMMANDE

Représentation d'un système : Une représentation générale d'un boucle de commande peut-être comme la figure¹ B.1.

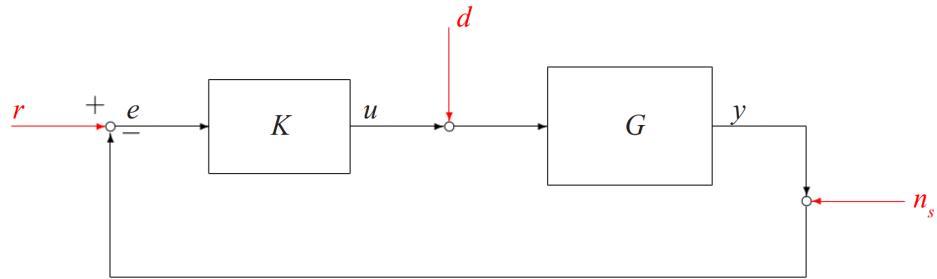


Figure B.1 Représentation en schéma-bloc d'une boucle de commande

Dans la figure précédente, G est le système boucle-ouverte commandé, K est le contrôleur, y la sortie mesurée, u l'entrée commandée, et e l'erreur. En rouge, les entrées sont n_s le bruit de mesure, d des perturbations ou bruit du système et r est le signal de référence.

Boucle-ouverte d'un système : Une boucle d'un système (un transfert d'une entrée vers une sortie) pour laquelle la sortie n'a aucune influence sur le signal d'entrée commandée u . La boucle n'applique aucune comparaison entre un signal de sortie et un signal d'entrée. Dans l'exemple précédent, le transfert de e vers y représente une boucle-ouverte.

Boucle fermée d'un système : Une boucle d'un système (un transfert d'une entrée vers une sortie) pour laquelle un système atteint et maintient une condition de sortie à une valeur de référence. La commande est obtenue en comparant la sortie actuelle et la valeur de référence. Le but d'une boucle fermée est d'asservir un système et de le réguler.

Dans l'exemple précédent, le transfert de r vers y représente une boucle fermée.

Représentation d'état : La représentation d'état d'un système à dynamique non-linéaire

1. La figure provient de l'article "The H_∞ control problem is solved" (Apkarian et Noll (2017)). Elle a été modifiée pour ôter les branches de performances.

prend la forme des équations suivantes :

$$\begin{cases} \dot{x} = f(x, u, w) \\ y = h(x, u, w) \end{cases} \quad (\text{B.1})$$

où $x(t) \in \mathbb{R}^n$ est le vecteur d'état, $u(t) \in \mathbb{R}^m$ le vecteur de commande à l'entrée, $w(t) \in \mathbb{R}^q$ le vecteur d'entrées exogènes, et $y(t) \in \mathbb{R}^p$ le vecteur de sortie. Les fonctions $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^q \rightarrow \mathbb{R}^n$ et $h : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^q \rightarrow \mathbb{R}^p$ sont supposées continuement dérivables.

Un point d'équilibre est caractérisé par (x_e, u_e, w_e) lorsque : $f(x_e, u_e, w_e) = 0$.

Représentation d'état d'un système LTI : La représentation d'état d'un système à dynamique linéaire LTI prend la forme des équations suivantes :

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (\text{B.2})$$

où $x(t) \in \mathbb{R}^n$ est le vecteur d'état, $u(t) \in \mathbb{R}^m$ le vecteur de commande à l'entrée, et $y(t) \in \mathbb{R}^p$ le vecteur de sortie. Les matrices A, B, C, D sont définies telles que $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ et $D \in \mathbb{R}^{p \times m}$.

La fonction de transfert équivalente au système d'état ci-dessus est donnée par : $G(s) = C(sI - A)^{-1}B + D$.

ANNEXE C RAPPELS DE GÉNIE LOGICIEL

Programmation orientée objet

La Programmation Orientée Objet (POO), comme son nom l'indique, regroupe les objets manipulés selon leur nature et fonction dans des catégories. Ces objects interagissent ensuite entre-eux par le biais de fonctions. Le vocabulaire nécessaire à la POO est défini ci-après :

- Une **classe** est un regroupement d'entités avec des caractéristiques communes. Une classe peut représenter une catégorie d'items, un patron ou même une action.
- Une **instance** est une réalisation spécifique d'une classe, un élément de celle-ci. On pourra aussi parler d'objet pour parler d'instance.
- Une **propriété** (ou attribut) est commune à toutes les instances de la classe. Ce sont des caractéristiques communes qui par leur valeur permettent de définir un objet spécifique de la classe.
- Une **méthode** est une fonction avec des entrées, des sorties et une marche à suivre. Une méthode est attachée à une classe, elle représente des actions appliquées sur ou par des instances de la classe. Il n'est pas possible d'appliquer une méthode sur des objets d'autres classes.
- Un **constructeur** est une méthode unique à chaque classe. Elle permet de construire des instances. Si la méthode constructeur est définie avec la classe, il est possible d'attribuer des valeurs aux propriétés de l'instance lors de sa création. Si la méthode n'est pas définie, la fonction par défaut est créée : `classObject=className()`. Dans cette situation, tous les propriétés de l'instance sont des champs vides.
- **Privé ou publique** ? Une classe peut-être définie publique ou privée. Une classe privée permet de garder les valeurs des propriétés confidentielles. Seuls les utilisateurs autorisés pourront avoir accès aux informations internes de la classe. Par défaut, toutes les classes sont publiques.
- Les relations d'**association** et d'**héritage** peuvent lier deux (ou n) classes entre-elles. Si deux classes sont associées, il existe au moins un lien entre-elles, l'une des deux (ou les deux) ont pour propriété un objet de l'autre classe. Par exemple, une entreprise qui posséderait des instances d'employés comme propriété. Si une classe hérite d'une autre, les deux classes forment une relation mère-fille. La classe fille possèdera toutes les propriétés et méthodes de la classe mère. À ces propriétés peuvent s'ajouter de nouvelles qui seront spécifiques à la classe fille. La même remarque peut-être faite pour les méthodes. Par exemple, dans une bibliothèque on

range des instances d'une classe mère **œuvre**. Les œuvres rangées sont alors des livres (classe fille de **œuvre**) et des films (classe fille de **œuvre**). Toutes les œuvres ont un titre, un auteur (réalisateur), un genre, un code de rangement. Les films auront aussi des acteurs, une durée de visionnage, tandis que le livre aura un nombre de page.

Diagrammes Unified Modeling Language

Tous les éléments définis précédemment et leurs relations sont représentés à l'aide d'un diagramme UML appelé **diagramme de classe**. Les classes présentent leurs propriétés ainsi que leurs méthodes (avec entrées et sorties seulement). Les associations et relations d'héritages sont représentées par des flèches (la sens de la flèche indique le sens de l'association).

D'autres diagrammes UML existent, chacun se réfère à une étape particulière dans le projet de conception logicielle. Dans ce mémoire, le diagramme de classe, le diagramme des cas ainsi que quelques diagrammes d'activités seront fournis.

Le **diagramme des cas** permet de représenter les principales fonctionnalités du logiciel. Il met notamment en avant les utilisateurs et les cas d'utilisation qu'ils auront du logiciel mais aussi les ressources à disposition.

Le **diagramme d'activités** permet de représenter l'architecture générale des fonctionnalités (ou sous-activités) d'une activité. Elle permet de schématiser le flux logique de l'activité. Les activités sont symbolisés par des bulles de tâches. Ces tâches sont liées par des flèches. Une fois une activité terminée la flèche de sortie indique l'activité suivante qui commence. Lors de la fin d'une activité, l'accomplissement de n activités simultanées est possible, pour cela la flèche se divise en n . Des évènements logiques peuvent structurer le flux logique (si ... alors ..., tant que, pour tous les éléments, etc.).

ANNEXE D COMPLÉMENT DE FIGURES DU CHAPITRE 2

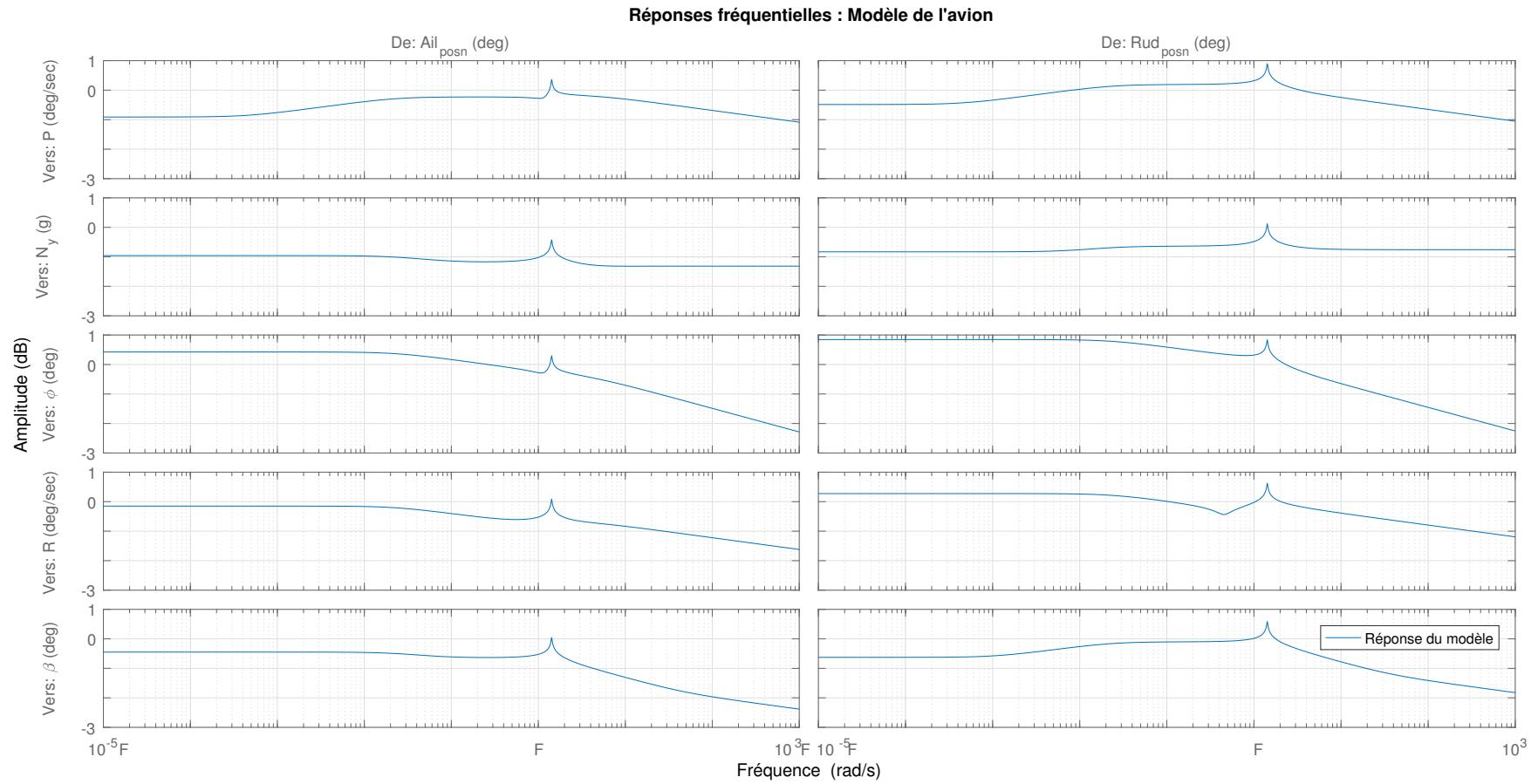


Figure D.1 Allure fréquentielle d'un modèle d'avion rigide

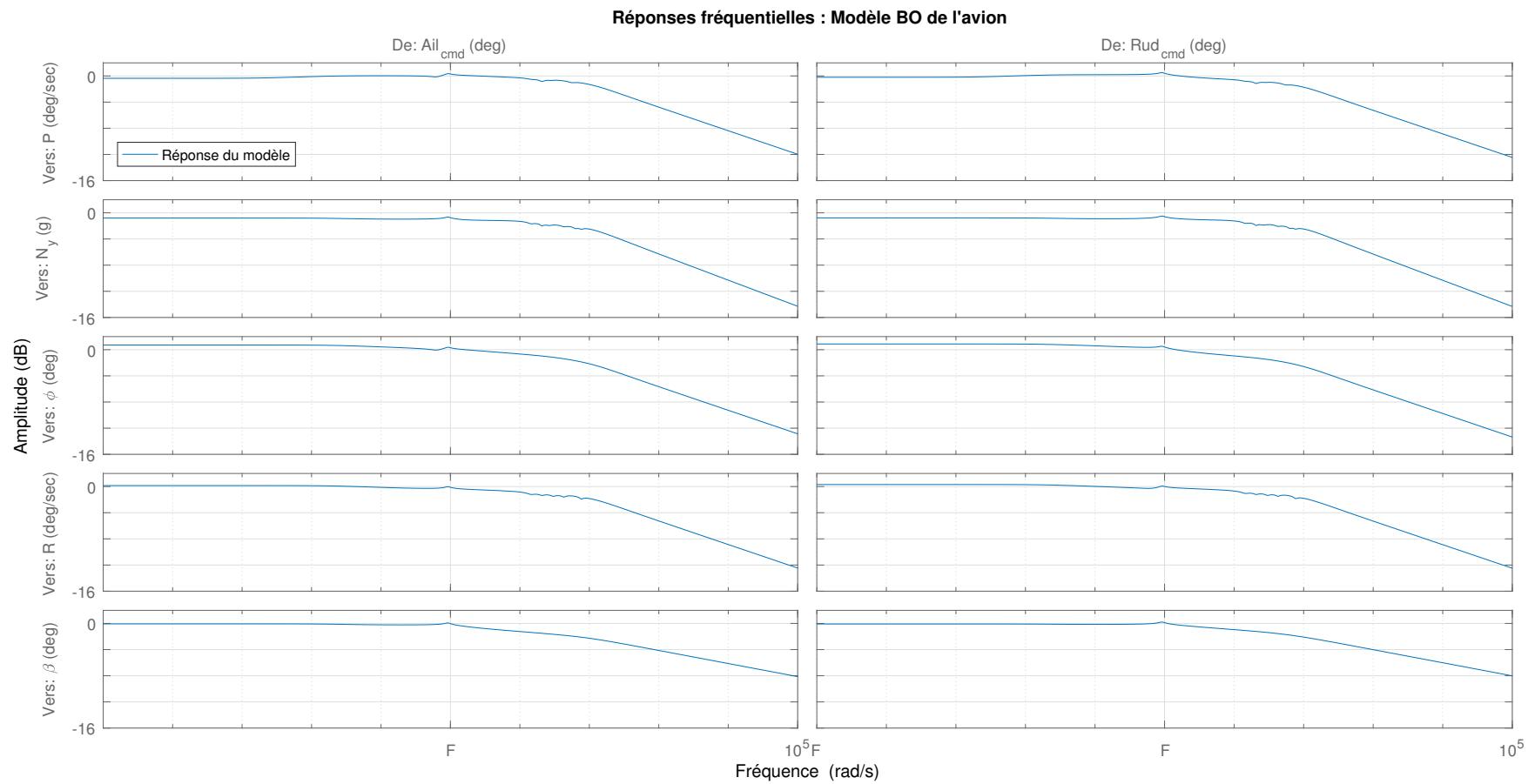


Figure D.2 Allure fréquentielle d'un modèle BO d'avion rigide

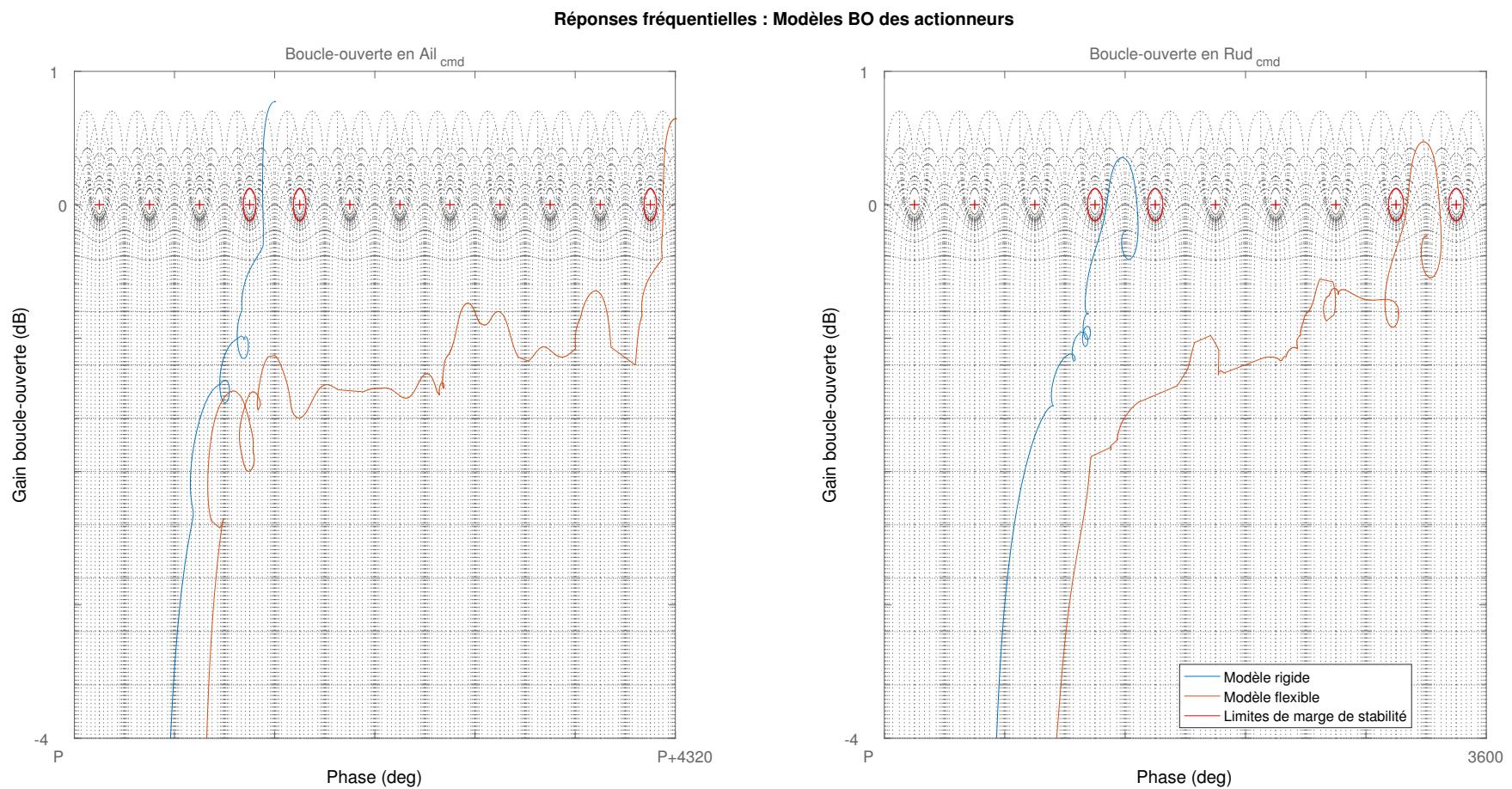


Figure D.3 Allure fréquentielle et marges de stabilité d'un modèle BO rigide d'avion

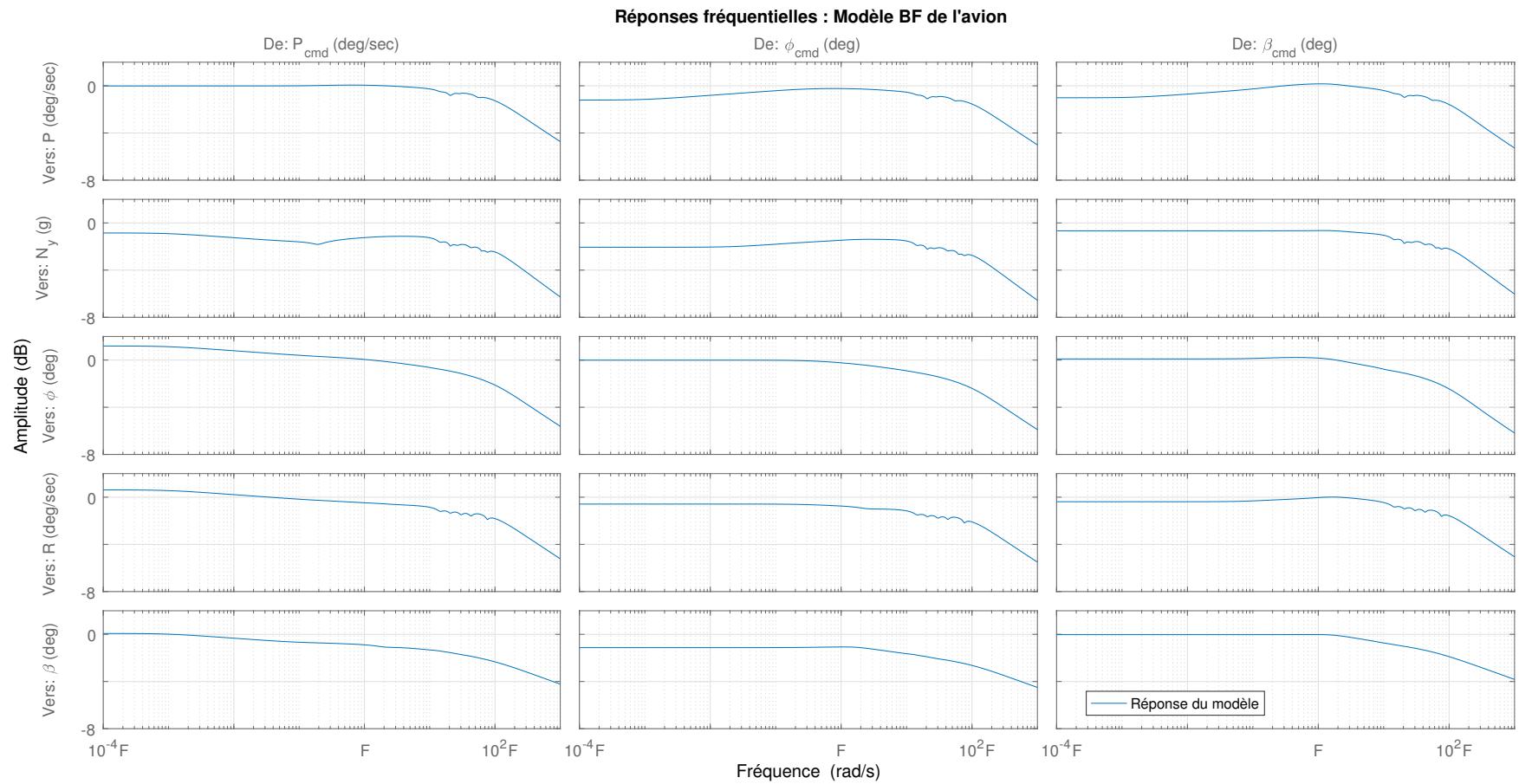


Figure D.4 Allure fréquentielle d'un modèle BF d'avion rigide

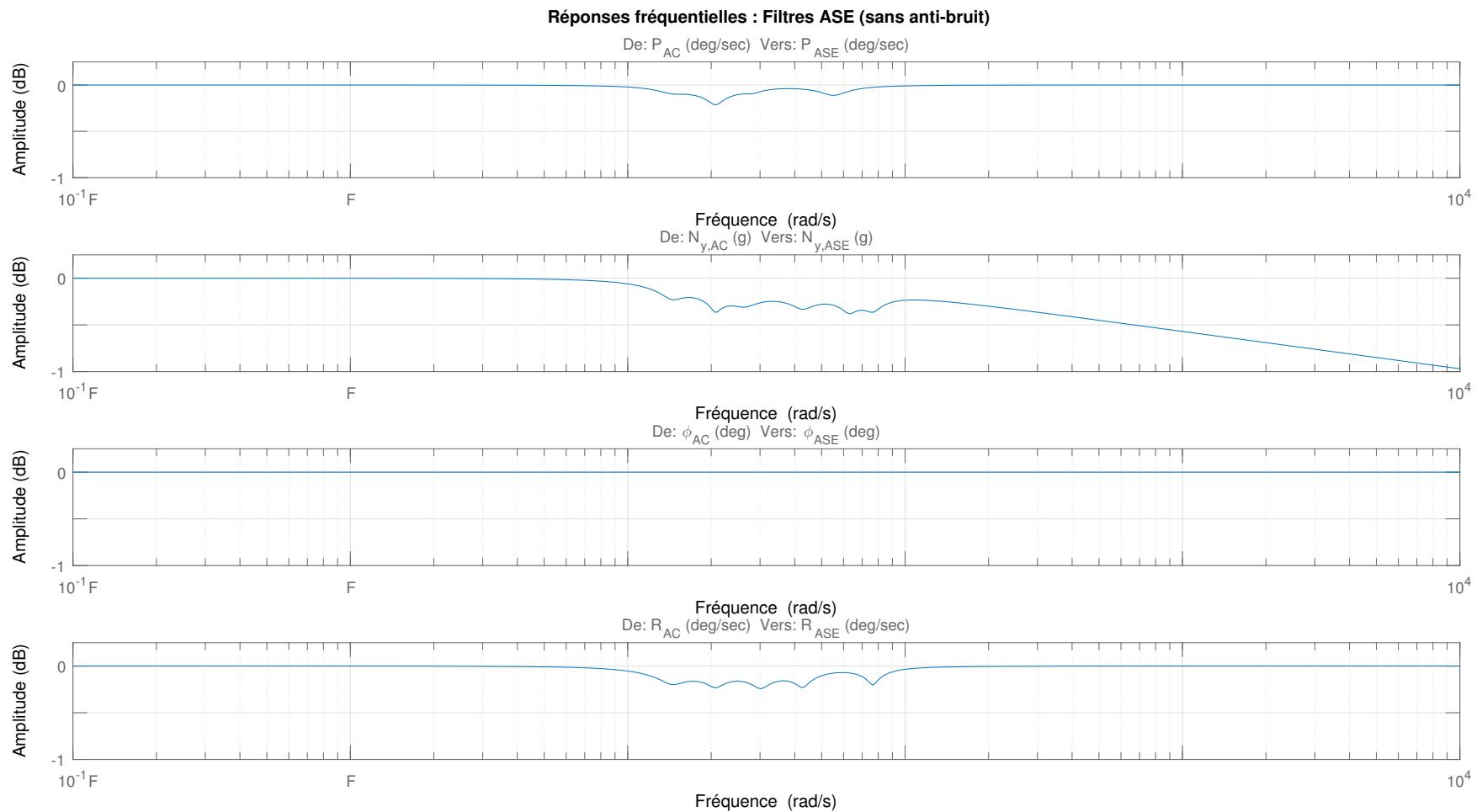


Figure D.5 Allures fréquentielles des filtres ASE sans le filtre Noise

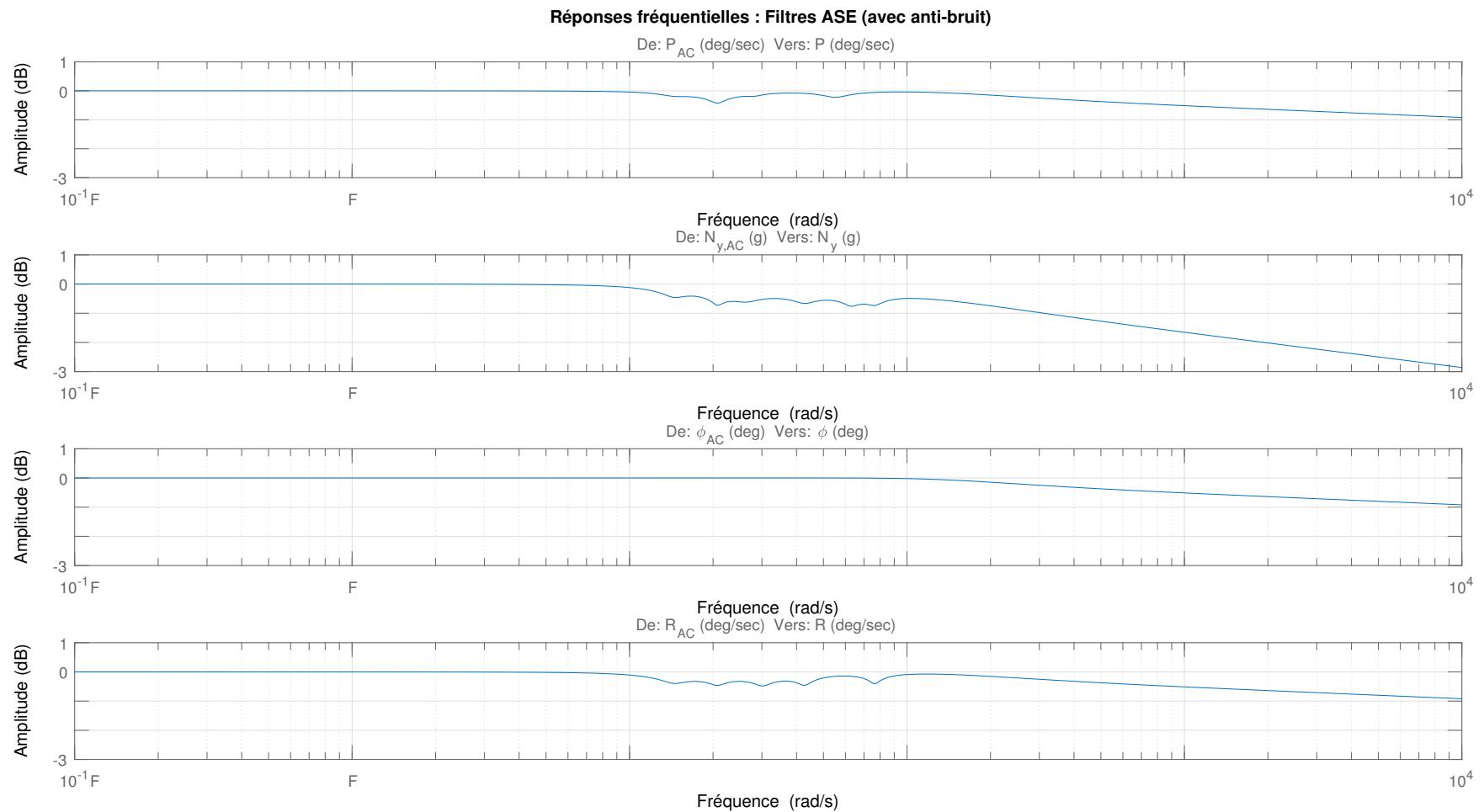


Figure D.6 Allures fréquentielles des filtres ASE avec le filtre Noise

ANNEXE E COMPLÉMENT DE FIGURES DU CHAPITRE 5

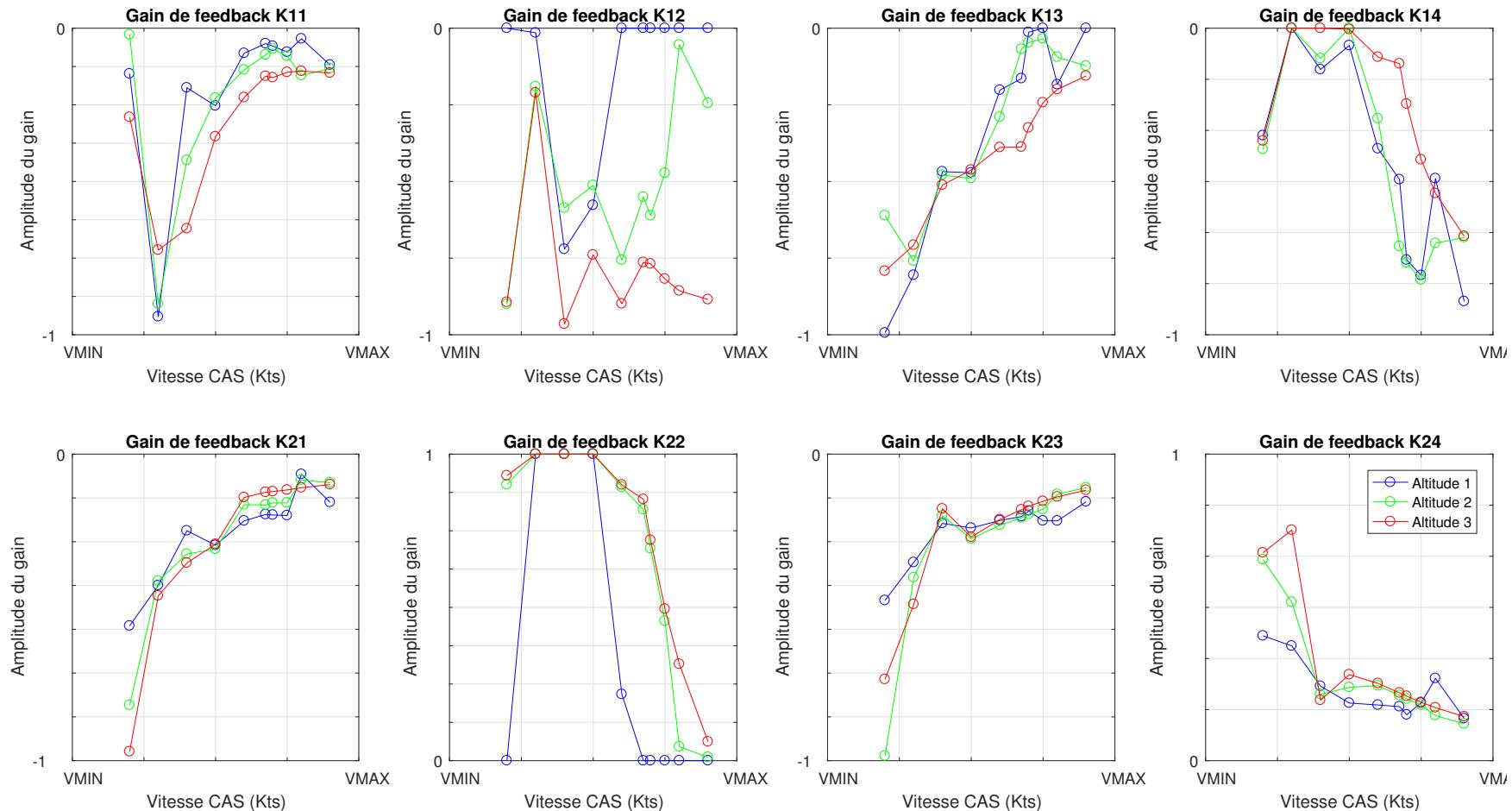


Figure E.1 Évolution des gains de feedback selon l'enveloppe de vol après une synthèse point à point

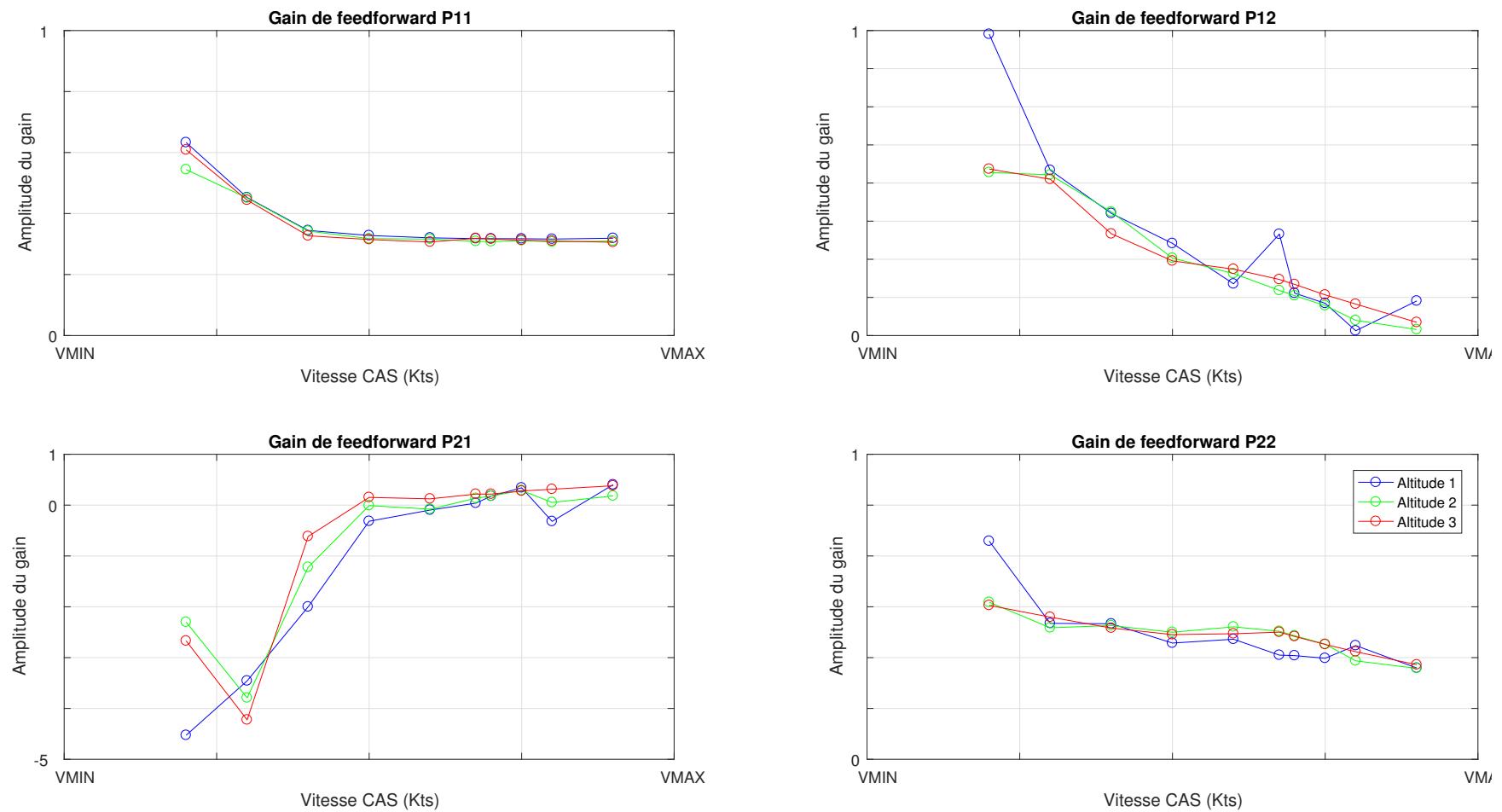


Figure E.2 Évolution des gains de feedforward selon l'enveloppe de vol après une synthèse point à point

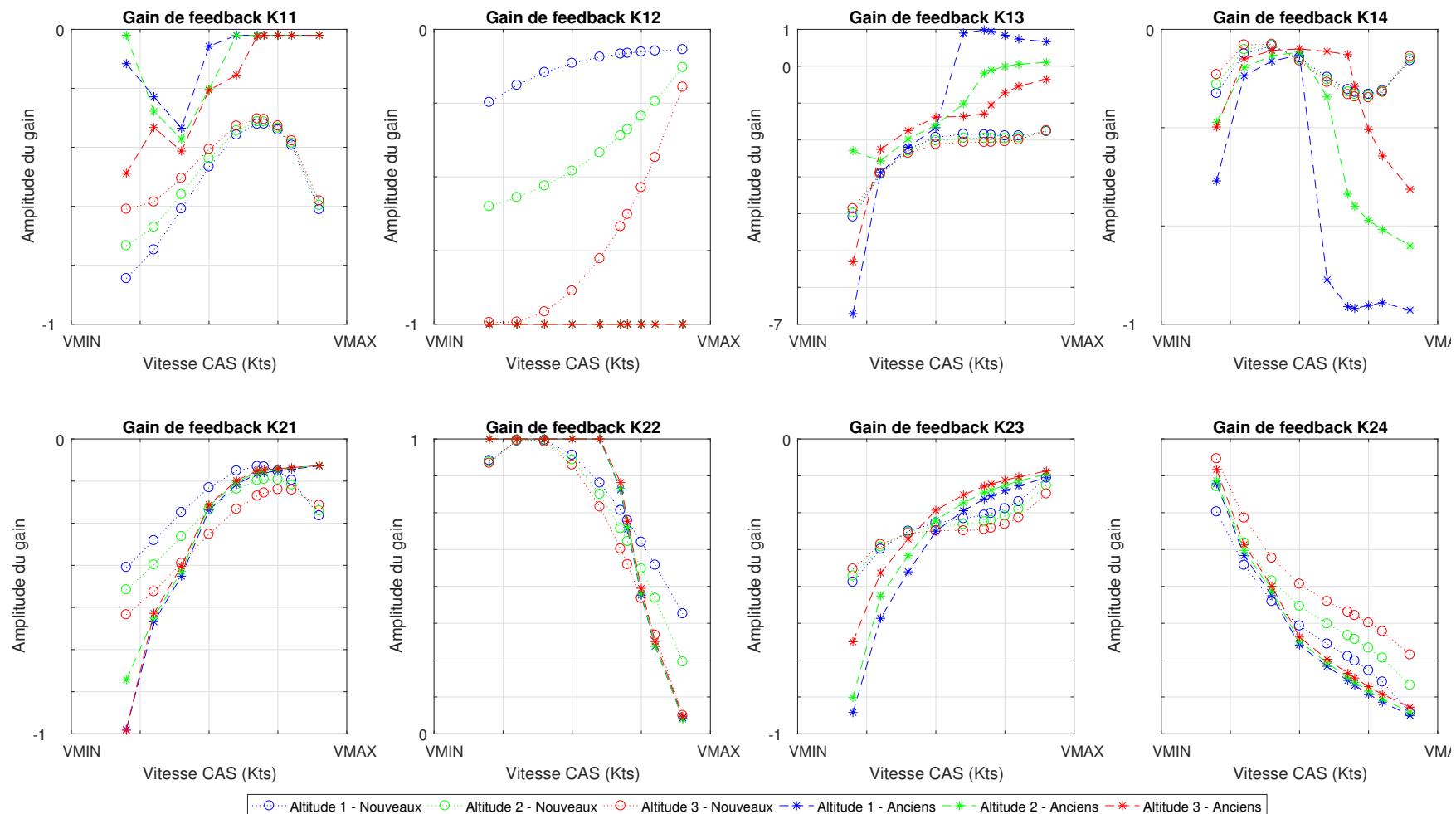


Figure E.3 Évolution des gains de feedback sur l'enveloppe

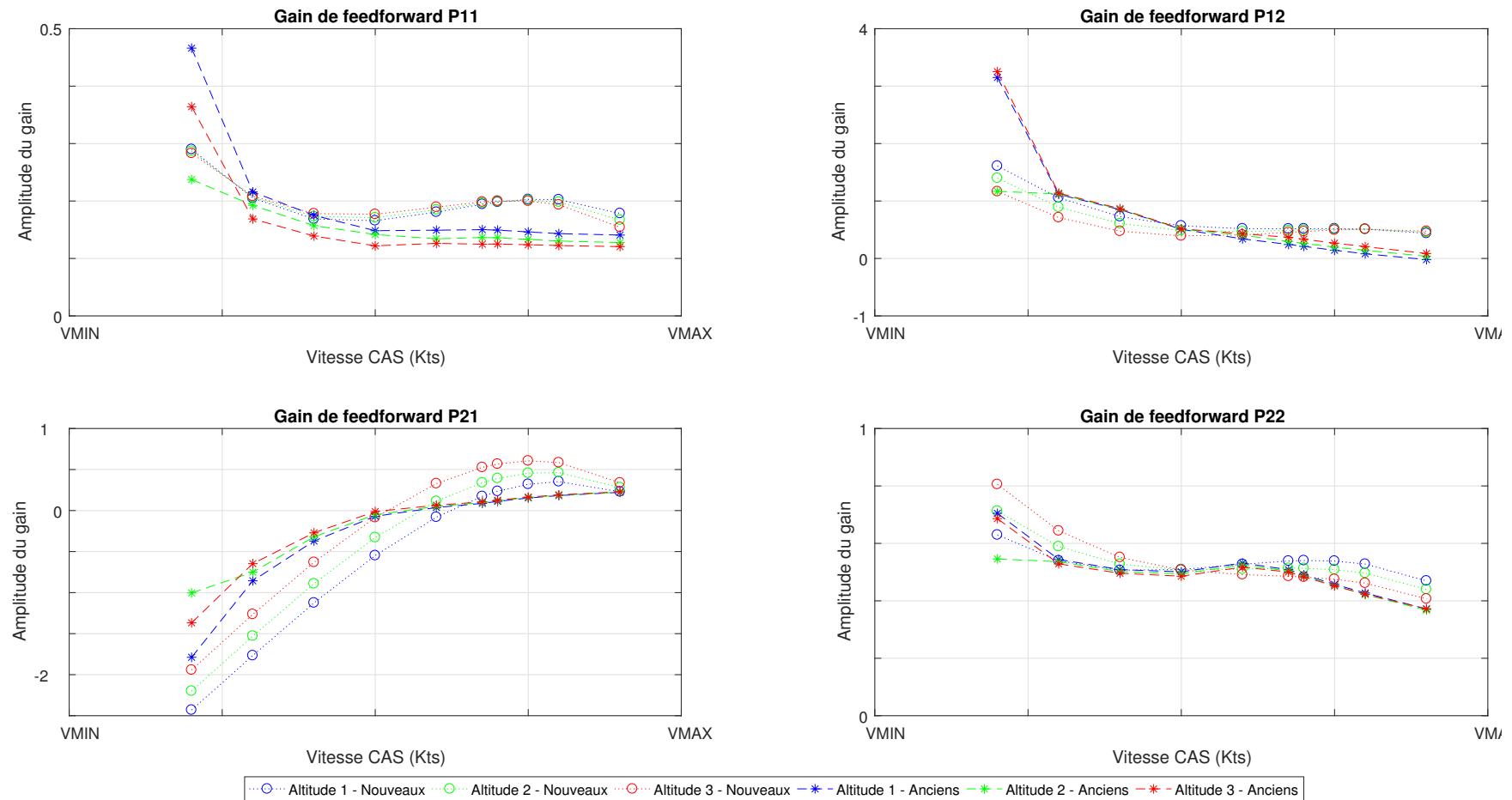


Figure E.4 Évolution des gains de feedforward sur l'enveloppe

ANNEXE F COMPLÉMENT DE FIGURES DU CHAPITRE 6

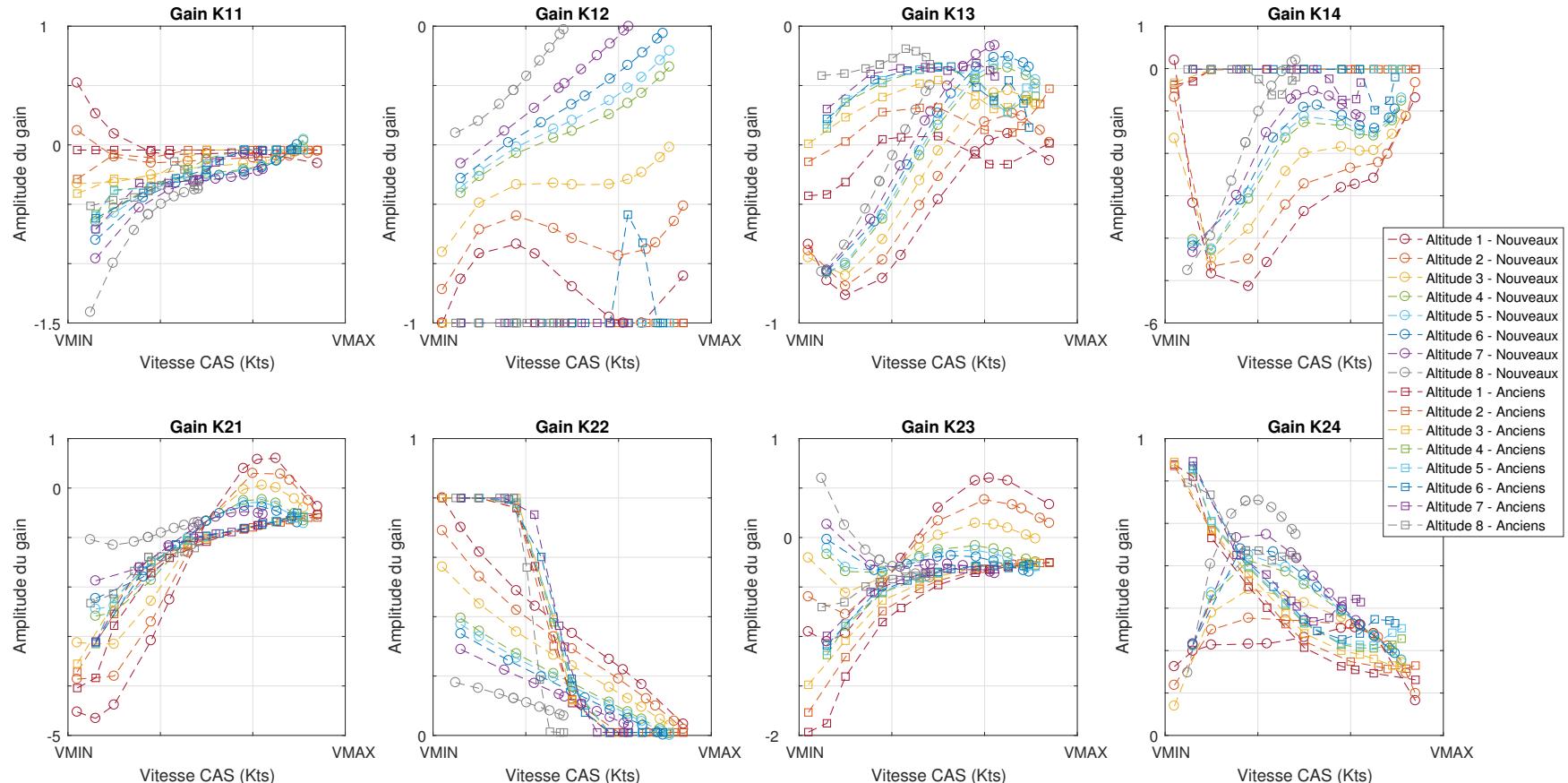


Figure F.1 Évolution des gains de feedback sur l'enveloppe

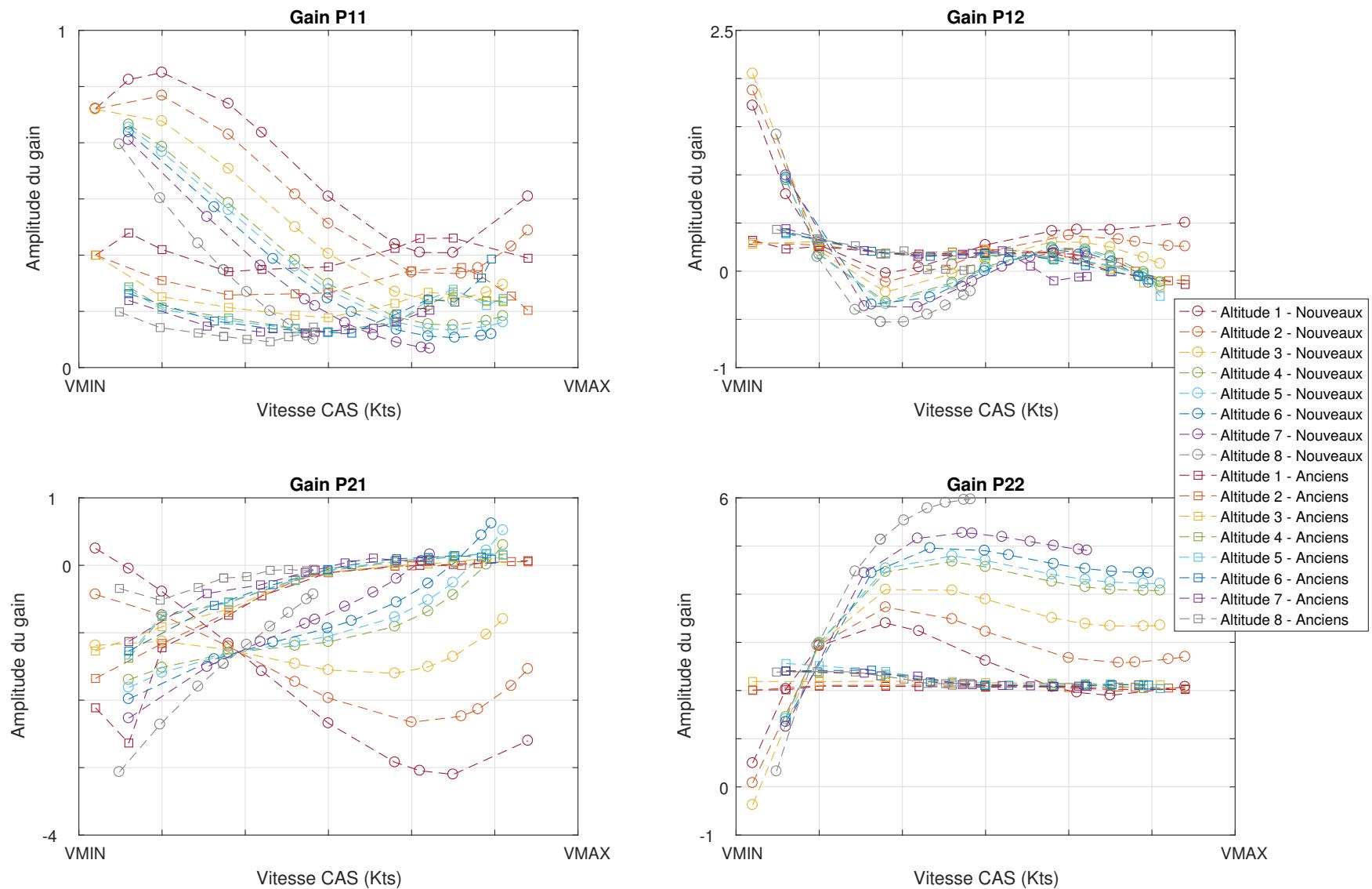


Figure F.2 Évolution des gains de feedforward sur l'enveloppe