HAL
open science

Commande à échantillonnage variable pour les systèmes LPV : application à un sous-marin autonome

Emilie Roche

To cite this version:

Emilie Roche. Commande à échantillonnage variable pour les systèmes LPV : application à un sous-marin autonome. Autre. Université de Grenoble, 2011. Français. NNT : 2011GRENT116 . tel-00721970

HAL Id: tel-00721970

https://theses.hal.science/tel-00721970v1

Submitted on 31 Jul 2012

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

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

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Automatique et Productique**

Arrêté ministériel : 7 août 2006

Présentée par

Emilie ROCHE

Thèse dirigée par **Olivier SENAME** et **Daniel SIMON**

préparée au sein du centre de recherche **INRIA Grenoble Rhône-Alpes**,
du laboratoire **GIPSA-Lab, département automatique**
et de l'école doctorale **EEATS**

Commande à échantillonnage variable pour les systèmes LPV : application à un sous-marin autonome

Thèse soutenue publiquement le **18 octobre 2011**,
devant le jury composé de :

Giuseppe CONTE, Président
Professeur à l'Università Politecnica delle Marche (Ancona, Italie)
Antonio PASCOAL, Rapporteur
Professeur associé à l'Instituto Superior Técnico (Lisboa, Portugal)
Edouard LAROCHE, Rapporteur
Professeur à l'université de Strasbourg (France)
Olivier SENAME, Directeur de thèse
Professeur à Grenoble INP (Grenoble, France)
Daniel SIMON, Co-Directeur de thèse
Chargé de recherche à l'INRIA Rhône-Alpes (Grenoble, France)

Remerciements

Je tiens ici à saluer les personnes qui, de près ou de loin, ont contribué à la concrétisation de ce travail de thèse de doctorat.

Je voudrais tout d'abord remercier Olivier Sename et Daniel Simon qui m'ont encadrée pendant ces trois années, pour l'ambiance de travail agréable qu'ils ont réussi à instaurer, pour leurs conseils, leur disponibilité et leur écoute tout au long de ma thèse. Je n'oublie pas également leur aide précieuse dans la relecture et la correction de mon manuscrit de thèse. Grâce à eux j'ai pu découvrir et apprécier le monde de la recherche.

Je remercie également les membres du jury qui ont accepté de faire le déplacement à Grenoble pour juger cette thèse. Merci à Giuseppe CONTE d'avoir présidé ce jury, et à mes rapporteurs Antonio PASCOAL et Edouard LAROCHE pour l'ensemble de leur commentaires et corrections qui a contribué à améliorer ce manuscrit.

Merci aux membres de l'équipe NeCS qui m'a accueillie pendant ces trois ans, pour les échanges, scientifiques ou amicaux. Merci également aux membres des laboratoires GIPSA-Lab et INRIA Rhône-Alpes pour l'aide logistique et technique. Un merci spécial à Jonathan Dumont pour la jolie vidéo présentée lors de ma soutenance.

Je remercie aussi tout particulièrement mes collègues de bureau qui se sont succédé, et qui sont devenus de vrais amis : Wenjuan, Gabriel, Valentina, Lara, Nicolas. Pour tous les bons moments passés au laboratoire ou en dehors, les pauses thé, les fous rires... qui aident à relâcher la pression dans les périodes plus compliquées. Merci aussi à tous les doctorants et stagiaires du laboratoire, notamment Antoine, Chloé, Lucie, et les membres du GipsADoc.

Et bien sûr merci à tous mes amis qui me supportent depuis plus ou moins longtemps : Jeanne, Florine, Mahmoud, Fanny, et tous les autres !

Et pour finir je tiens bien évidemment à remercier ma famille, à commencer par mes parents qui m'ont toujours soutenue dans ce que j'ai voulu entreprendre (et pour leur aide précieuse dans la réalisation du pot de thèse !) ainsi que ma sœur Frédérique et Philippe Romain et Quentin.

Merci à tous

Abstract

The idea of using a discrete-time controller with variable sampling interval can be interesting in many cases, for example to control the computing power used for the feedback control, or when measurements signals are received none periodically. This variable delay between two measures can be seen as a variable sampling rate for the control law. The discrete-time controller synthesis with variable sampling has already been studied for Linear Time Invariant systems. The purpose of this thesis is to extend these results to Linear Parameter Varying systems (LPV), since that representation allows to keep some important parameters of a non-linear system as varying and known parameters in a linear representation. The controller synthesis relies on the $\mathcal{H}_\infty$method, applied to LPV systems. In particular, two main approaches existing in the literature are considered : the polytopic approach (where the varying parameter evolves in a convex set) and the Linear Fractional Representation (LFR). The proposed method is applied to the altitude control of an Autonomous Underwater Vehicle (AUV), which is difficult to control due to important non-linearity and uncertainty in its model. Moreover in sub-marine environments, the altitude measurement is done using an ultrasonic sensor and the travel of the signal in the water depends on the distance with respect to the sea floor. Simulation results will proved the interest of the proposed method for the altitude control of an AUV, and in particular the improvements when both non-linearity of the system and variable sampling interval are considered, in a unified way, in the LPV model used for the controller synthesis.

Résumé

Dans le contexte des systèmes de contrôle commande embarqués, une meilleure gestion de la puissance de calcul peut être obtenue en utilisant des contrôleurs à intervalle d'échantillonnage variable. Le délai variable entre deux réceptions de mesures, peut alors être vu comme une variation d'intervalle d'échantillonnage pour le contrôleur. La synthèse de lois de commande discrète à intervalle d'échantillonnage variable a déjà été étudiée pour des systèmes stationnaires (LTI). On se propose ici d'étendre cette méthode pour des systèmes Linéaires à Paramètres Variants (LPV), qui permettent entre autre de conserver des certaines non-linéarités (d'un modèle non-linéaire de départ) en temps que paramètres variants et connus d'un système linéaire. Nous développons alors une méthodologie de synthèse de contrôleur à échantillonnage variable, selon l'approche $\mathcal{H}_\infty$appliquée aux systèmes LPV. Dans ce contexte, deux cas distincts sont abordés : celui des systèmes polytopiques (où le paramètre variant évolue dans un volume convexe) et celui des systèmes mis sous forme LFR (Représentation Linéaire Fractionnelle). La méthode

proposée est appliquée au contrôle d'un AUV (Autonomous Underwater Vehicle), qui est système difficile à contrôler du fait d'importantes non-linéarités et incertitudes du modèle. De plus en milieu sous-marin la mesure d'altitude est effectuée avec un capteur à ultrason et la durée du trajet du signal dans l'eau dépend de la distance par rapport au fond. Des résultats de simulations permettent d'illustrer l'intérêt de la méthode pour le contrôle d'altitude d'un AUV, et notamment les améliorations apportées par la prise en compte conjointe dans le modèle utilisé pour la synthèse des contrôleurs de non-linéarités du système et des variations de l'intervalle d'échantillonnage.

Table des matières

Introduction **1**

1 Outils LPV / $\mathcal{H}_\infty$ **7**
1.1 Commande $\mathcal{H}_\infty$ 8
1.1.1 Définition 8
1.1.2 Synthèse de contrôleur $\mathcal{H}_\infty$ 10
1.2 Système Linéaire à Paramètre Variant (LPV) 17
1.3 Approche polytopique pour les systèmes LPV 19
1.3.1 Modélisation sous forme polytopique 19
1.3.2 Synthèse de contrôleur polytopique 21
1.4 Approche LFR pour les systèmes LPV 23
1.4.1 Modélisation LFR 23
1.4.2 Synthèse LFR 25

2 Echantillonnage variable **29**
2.1 État de l'art sur la discrétisation des systèmes LPV 30
2.2 Discrétisation paramétrée 31
2.3 Mise sous forme polytopique discrète du système 33
2.3.1 Représentation discrète polytopique en fonction de δ 34
2.3.2 Mise en forme polytopique discrète pour la commande à intervalle variable 35
2.4 Approche polytopique pour systèmes LPV 37
2.5 Mise sous forme LFR discrète 41
2.5.1 Forme exacte de la LFR discrète considérant la période d'échantillonnage comme paramètre 42
2.5.2 Forme approchée de la LFR discrète considérant la période d'échantillonnage comme paramètre 43
2.6 Extension aux systèmes LPV 44

3 Modélisation AUV **49**
3.1 Modèle non linéaire 50

3.1.1 Equations globales 52
3.1.2 Détails des équations régissant le mouvement du sous-marin 53
3.1.3 Implémentation sous *Matlab/Simulink* 57
3.2 Découplage des différentes directions 58
3.2.1 Réduction de modèle pour l'avance 59
3.2.2 Réduction de modèle au plan horizontal 60
3.2.3 Réduction de modèle au plan vertical 60
3.2.4 Fusion des trois sous-modèles 61
3.3 Plan vertical : structure cascade z/θ 61
3.4 Modèle LPV du sous-marin 63
3.4.1 Modèle LPV polytopique 65
3.4.2 Modèle LPV sous la forme d'une LFR 65

4 Commande LPV de l'AUV 69
4.1 Scénario de simulations 69
4.2 Contrôleur K_u pour la vitesse d'avance 71
4.3 Contrôle d'un système LTI à période variable 73
4.3.1 Contrôleur d'angle d'incidence K_θ discret à période d'échantillonnage constante 74
4.3.2 Synthèse du correcteur d'altitude $K_z(h)$: approche LPV polytopique discrète 76
4.3.3 Synthèse LFR discrète pour le correcteur d'altitude $K_z(h)$. 83
4.4 Contrôle d'un système LPV à période variable 85
4.4.1 Structure cascade polytopique 86
4.4.2 Structure cascade LFR 94
4.5 Conclusion : comparaison des deux approches 101

Conclusion et perspectives 103

Introduction

Contexte

Ce travail de thèse s'inscrit dans le cadre du projet de recherche européen FeedNetBack, qui s'intéresse aux systèmes commandés en réseau (`www.feednetback.eu`).

Ce projet regroupe à la fois des partenaires académiques, comme l'INRIA (Institut National de Recherche en Informatique et en Automatique), coordinateur du projet, les universités ETH de Zürich (Suisse), KTH de Stockholm (Suède), les universités de Séville (Espagne) et de Padova (Italie), mais aussi des partenaires industriels, chargés de mettre en application les innovations technologiques (l'IFREMER (`wwz.ifremer.fr`) pour l'application des sous-marins et VODERA, VIDEOTEC pour celle des caméras).

L'objectif du projet FeedNetBack est d'améliorer l'efficacité et la robustesse des systèmes commandés en réseau (en anglais Networked Control Systems : NCS). En effet les NCS sont composés, en plus du procédé lui même, de capteurs, d'actionneurs et de contrôleurs, communicant via un réseau de communication. Ce type de système présente de nombreux avantages par rapport à des systèmes plus classiques, comme la flexibilité, le faible coût et une maintenance plus facile. Cependant leur particularité conduit à la nécessité d'un contrôle adapté, tenant compte des différentes contraintes, comme les incertitudes temporelles et spatiales, les retards et les limites de communication (bande passante)... afin de garantir stabilité et performances.

Le but est alors de fournir une méthode de conception conjointe, permettant de tenir compte, lors de la conception du contrôle, des contraintes architecturales, de calcul, de communication, de complexité et de gestion d'énergie.

En effet traditionnellement les étapes de conception de lois de commande et de leur implémentation sont réalisées de façon distinctes, avec très peu d'échange de connaissances d'un domaine à l'autre. Ainsi les tâches de contrôle/commande sont souvent considérées comme des systèmes temps-réel "durs", ne supportant aucun retards ou pertes de données. Or la nature même d'un système bouclé par rétroaction apporte une robustesse vis à vis de certaines incertitudes, dont celles liées à l'implémentation, qui permet de relâcher les contraintes temps réel (dans

une certaine limite) (Simon et al., 2009).

Les lois de commandes sont amenées a être implémentées en tant que tâches dans un système d'exploitation temps réel.

Au delà de la solution classique consistant à s'intéresser à l'ordonnancement des tâches sur le système d'exploitation, c'est à dire à l'ordre dans lequel seront exécutées les tâches pour respecter leur contraintes temporelles, il est possible de réguler l'utilisation des ressources informatiques nécessaires à l'exécution de ces tâches de contrôle/commande (Cervin, 2003; Stankovic et al., 1999).

Dans ce contexte, une solution consiste à créer un régulateur d'ordonnancement pour les tâches de commande, qui s'adapte aux variations de ressources de calcul. Ce dispositif estime la charge processeur disponible, la compare à une référence, et modifie la fréquence des tâches périodiques de commande. Ceci amène à la synthèse de lois de commande à échantillonnage variable. Cette démarche a été entamée il y a plusieurs années, avec notamment Sename et al. (2003); Robert (2007). Dans Robert (2007), la synthèse de contrôleurs à période d'échantillonnage variable a été en partie explorée. Cette technique permet d'adapter les ressources (celles du calculateur présent sur le système embarqué) nécessaires au calcul des lois de commandes en agissant sur la période d'échantillonnage. D'autres travaux portant sur l'échantillonnage variable sont disponibles dans la littérature, comme par exemple Sala (2005), où la synthèse d'un contrôleur et d'un observateur discret à période d'échantillonnage variable est réalisée, en utilisant la technique de "gridding" (maillage de l'espace de paramètre). Dans Cuenca et al. (2009), une structure prédicteur/observateur permet l'analyse de stabilité d'un système commandé en réseau soumis à un échantillonnage non-uniforme et des retards variables (l'observateur reconstruit les données perdues et le prédicteur élimine l'influence du retard réseau). Dans Almeida et al. (2011) un détecteur d'évènement permet d'adapter le pas d'échantillonnage ; une structure observateur/contrôleur (gain constant) réalise ensuite le contrôle du système.

Dans le projet FeedNetBack, deux cas d'étude sont étudiés pour le co-design : une flottille de véhicules sous-marins autonomes et un système de surveillance utilisant un réseau de caméras distribuées.

C'est dans le cadre du premier cas d'étude que se situe ce travail. Une flottille de robot sous-marins autonomes et de véhicules de surface constitue un réseau mobile formé de peu d'éléments, mais avec des contraintes de communication importantes. Le but de cette flottille est de réaliser une mission scientifique, par exemple réaliser la cartographie du fond sous-marin, ou encore rechercher une source (de gaz, produit chimique...). L'utilisation d'une flottille plutôt qu'un véhicule seul permet de réduire le temps nécessaire pour la mission, ou de réaliser une tâche plus compliquée qu'un véhicule seul ne pourrait pas accomplir.

Evidemment le contrôle d'une flottille de véhicule nécessite un contrôle local de chacun des véhicules la composant ; c'est à ce niveau que le travail présenté dans cette thèse contribue au projet global.

La commande des véhicules sous-marins autonomes (Autonomous Underwater Vehicle : AUV) constitue un champ de recherche très actif depuis les vingt dernières années, du fait de l'utilisation de plus en plus fréquente de ce type de véhicule pour l'exploration des fonds sous-marins et la récolte de données.

Nous nous intéressons ici au développement de méthodes de commande robuste, en particulier dans le contexte $\mathcal{H}_\infty$, afin de piloter un sous-marin autonome tout en tenant compte de contraintes de communication et/ou de calcul.

La commande $\mathcal{H}_\infty$, qui est à la base de la méthode employée dans la suite de ce manuscrit, a déjà été utilisée pour le commande d'AUV. Citons par exemple Feng and Allen (2004), où un contrôleur $\mathcal{H}_\infty$ d'ordre réduit est utilisé pour synthétiser un pilote automatique pour un AUV (en commandant de façon séparée la vitesse d'avance, la direction (angle de lacet) et la profondeur). Dans Silvestre and Pascoal (2004), le contrôle d'un AUV dans le plan vertical est considéré, en utilisant une technique de gain séquencé non-linéaire. Un ensemble de contrôleurs linéaires robustes, utilisant la méthodologie $\mathcal{H}_\infty$, est synthétisé pour un nombre de points de fonctionnement donné, puis le contrôleur est interpolé en fonction de la valeur de la vitesse d'avance (qui sert de variable de séquencement). La même technique est utilisée dans Silvestre and Pascoal (2007) pour le contrôle de profondeur. Plusieurs autre techniques de contrôle ont été étudiées, comme par exemple la commande par modes glissants dans Healey and Lienard (1993) et Salgado-Jimenez and Jouvencel (2003), et le contrôle par PID amélioré par un anti-windup dans Miyamaoto et al. (2001).

D'autre types de véhicules sous-marins existent et peuvent servir pour des tâches différentes. Par exemple les ROV (Remotely Operated Vehicle) sont des véhicules sous-marins téléguidés, reliés à un navire en surface par un cable d'alimentation, ce qui supprime la contrainte énergétique forte des AUV. Ce type de véhicule est également l'objet de différentes études ; citons par exemple Zanoli and Conte (2003) où le contrôle d'altitude d'un ROV est étudié, en utilisant un PID, puis pour limiter le dépassement en sortie, deux méthodes sont comparées (ajout d'un filtre sur l'entrée de contrôle et un 'discrete fuzzy smoother').

Cependant la plupart des travaux existant pour la commande des AUV sont développés en temps continu, alors que la loi de commande obtenue doit ensuite être implémentée sur un ordinateur embarqué sur le véhicule (donc en temps discret). Or, comme mentionné précédemment, nous désirons considérer, dès la conception du contrôle, les contraintes liées à l'implémentation Dans cet objectif, ce travail de thèse s'inscrit dans la continuité des travaux de thèse de David Robert (Robert,

2007).

Dans le cas d'un sous-marin autonome, la variation de période d'échantillonnage présente un autre intérêt : la mesure de l'altitude du véhicule par rapport au fond est effectuée par un capteur à ultrason. Or ce type de capteur engendre un intervalle de temps variable entre l'envoi du signal de mesure et son retour au capteur, dépendant de la distance par rapport au fond sous-marin. Ainsi même si la mesure est envoyée de façon périodique, son retour lui n'est pas périodique mais décalé d'un temps variable en fonction de l'altitude du véhicule. Nous avons choisi de traiter cette variation dans la réception de la mesure par une variation de période d'échantillonnage du contrôleur, puisqu'il n'est pas nécessaire de mettre à jour la commande sans mise à jour de la mesure. Finalement, dans le but de limiter l'énergie consommée par les agents, il peut être intéressant de limiter le nombre de mesures d'altitude par ultrasons (très consommatrice d'énergie) en augmentant la période d'échantillonnage. Un compromis doit alors être trouvé entre énergie économisée et qualité du contrôle réalisé.

D'autre part un AUV est un système difficile à contrôler, du fait d'importantes non-linéarités dans le modèle, d'un couplage entre certains états du système (altitude et angle de tangage par exemple), de la taille importante du système et de son caractère multivariable (plusieurs capteurs et plusieurs actionneurs)... C'est pourquoi l'approche de commande LPV (Linéaire à Paramètres Variant) a été choisie : elle permet d'une part d'intégrer sous forme de paramètres variants certaines non-linéarités importantes, ou d'autres paramètres physiques du système, mais aussi de profiter de la robustesse apportée par la méthode $\mathcal{H}_\infty$, qui est à la base de la commande LPV.

Les contributions de ce travail de thèse concernent donc l'extension des travaux de Robert (2007) sur les correcteurs à échantillonnage variable, étendu aux systèmes LPV, appliquée à la commande d'un véhicule sous-marin autonome.

Plan de la thèse

La suite de ce manuscrit présente ainsi la synthèse de régulateurs LPV avec un intervalle variable entre l'application de deux commandes à temps discret, appliquée à un sous-marin autonome.

La structure du mémoire se présente comme suit :

Chapitre 1 : Outils LPV / $\mathcal{H}_\infty$

Le chapitre 1 présente les définitions ainsi que les outils de la commande $\mathcal{H}_\infty$, puis l'extension de ces outils pour des systèmes Linéaires à Paramètres Variants (LPV). Ce chapitre n'apporte pas de contributions, mais permet d'apporter les bases théoriques qui seront utilisées par la suite.

Chapitre 2 : Mise en forme d'un système dépendant de la période d'échantillonnage

Dans un premier temps, les travaux présentés dans Robert (2007) sont présentés ; ils permettent l'utilisation de la méthodologie LPV pour intégrer une variation de période d'échantillonnage comme paramètre variant pendant la phase de modélisation puis de synthèse d'un contrôleur à échantillonnage variable. Ces travaux sont ensuite étendus aux systèmes dépendants déjà d'un ou plusieurs paramètres variants en temps continus (ces paramètres peuvent permettre de conserver certaines non-linéarités dans le modèle, comme cela sera développé plus tard).

Chapitre 3 : Véhicule sous-marin autonome

Au cours de ce chapitre, les différents modèles utilisés pour la synthèse des contrôleurs puis pour les simulations sont présentés. Dans un premier temps le modèle non-linéaire du véhicule est défini : il est le plus proche du système réel et est utilisé lors des simulations. Puis un modèle linéarisé utilisé pour la synthèse de contrôleurs est introduit ; en particulier la séparation suivant les différents axes (avance, plan vertical et plan horizontal) est présentée. Finalement une plus grande attention est portée à la modélisation pour le contrôle d'altitude, et une structure en cascade, adaptée à la dépendance entre l'altitude et l'angle de tangage est présentée ; cette structure nécessite la synthèse de deux contrôleurs, et donc deux modèles sont à considérer.

Chapitre 4 : Commandes LPV d'un sous-marin autonome à intervalle d'échantillonnage variable

Dans ce chapitre, le contrôle de l'altitude de l'AUV, utilisant une structure cascade altitude/ angle de tangage est testé en simulation sur le modèle non-linéaire complet du véhicule. Après la description du scénario de simulation, les deux cas considérés tout au long de ce travail sont étudiés. Dans un premier temps, seule la période d'échantillonnage est utilisée comme paramètre variant, les approches polytopique et LFR sont alors comparées. Puis l'extension aux modèles LPV est mise en œuvre, où des paramètres physiques hérités du modèle non-linéaire du véhicule sont ajoutés à la période d'échantillonnage. Là encore, les approches polytopique et LFR seront comparées.

Remarque : Par la suite, il sera souvent fait allusion à une période d'échantillonnage variable, ce qui est bien sûr un abus de langage. Ce raccourci fait référence à l'intervalle entre l'application successive de deux commandes, ce qui dans le cas discret conventionnel représente la période d'échantillonnage.

Publications

- LPV / $\mathcal{H}_\infty$ Control of an Autonomous Underwater Vehicle (AUV), E. Roche, O. Sename et D. Simon, Proceedings of the **European Control Confe-**

rence, Budapest, Hungary, Août 2009
- LPV / $\mathcal{H}_\infty$ varying sampling control for Autonomous Underwater Vehicles, E. Roche, O. Sename et D. Simon, Proceedings of the **IFAC Symposium on System Structure and Control**, Ancona, Italie, Septembre 2010
- Contrôleur LPV/$\mathcal{H}_\infty$discret à échantillonnage variable appliqué à un AUV, E. Roche, O. Sename et D. Simon, présenté aux **Journée Doctorales/Journées Nationales MACS (JD/JN MACS)**, juin 2011, Marseille, France
- A hierarchical Varying Sampling $\mathcal{H}_\infty$Control of an AUV, E. Roche, O. Sename et D. Simon, Proceedings of the **IFAC World Congress**, Milan, Italie, août 2011
- Design of Robust Variable Rate Controllers, E. Roche, O. Sename, A. Seuret, D. Simon, S. Varrier, **FeedNetBack report** (D.04.03), Juillet 2010

Chapitre 1

Outils LPV / $\mathcal{H}_\infty$

Quel que soit le type de loi de commande utilisé, le calcul d'un régulateur nécessite un modèle du système réel, qui présente des incertitudes et des approximations. De plus, le contrôleur doit permettre de satisfaire certaines performances en boucle fermée. Les outils de commande robuste, tels que la commande $\mathcal{H}_\infty$, permettent de garantir les performances en boucle fermée malgré les incertitudes sur la connaissance du modèle de synthèse, ou sur la mesure des signaux. Ainsi la robustesse apportée par la méthode permet de palier le manque de précision du modèle utilisé pour la synthèse. D'autres techniques de commande robuste existent, comme par exemple la commande $\mathcal{H}_2$ ou encore l'association des deux méthodes : la commande multi-objectifs $\mathcal{H}_2/\mathcal{H}_\infty$. Toutefois dans le cadre de cette thèse, on s'intéressera uniquement à la commande $\mathcal{H}_\infty$.

L'extension de la commande $\mathcal{H}_\infty$ aux systèmes Linéaires à Paramètres Variants (LPV) apporte la robustesse de la méthode et permet également au contrôleur de s'adapter à une variation connue (ou estimée) de paramètre. En effet, l'approche LPV, que ce soit pour la modélisation (ou pour la commande), fait intervenir des systèmes (ou régulateurs) sous forme de représentations d'état, dont les matrices dépendent d'un vecteur de paramètres. Ceci permet de faire varier les matrices du système par rapport à un point de fonctionnement, ou à un paramètre important pour le modèle. Comme nous allons le voir par la suite, plusieurs approches ont été développées pour la modélisation des systèmes LPV : l'approche par "gridding" (discrétisation de l'espace des paramètres), l'approche polytopique et la Représentation Linéaire Fractionnelle (LFR pour Linear Fractional Representation). Par exemple dans Poussot-Vassal (2008), le contrôle d'une suspension automobile est considéré, et un modèle LPV polytopique est construit, considérant la raideur des amortisseurs comme paramètres variants. Dans Gauthier et al. (2007), la Représentation Linéaire Fractionnelle (LFR) est utilisée pour construire le modèle LPV du système d'injection “Common Rail” d'un moteur diesel, fonction de divers paramètres importants du moteur.

Ce chapitre n'apporte pas de contribution théorique, mais présente de façon résumée les différents outils qui seront utilisés dans la suite de la thèse. Pour plus de détails sur les différentes méthodes, nous invitons le lecteur à se référer aux références citées au fur et à mesure, notamment Zhou et al. (1996) pour la commande $\mathcal{H}_\infty$, Boyd et al. (1994), Scherer and Wieland (2004) pour la synthèse de contrôleur $\mathcal{H}_\infty$ par la résolution de LMI, Apkarian et al. (1995), Scherer and Wieland (2004) pour l'approche polytopique, et Apkarian and Gahinet (1995) pour l'approche LFR.

La suite du chapitre se présente comme suit : la section 1.1 définit les différentes notations et normes qui seront utilisées par la suite, puis expose la méthodologie de synthèse de contrôleur $\mathcal{H}_\infty$, avec notamment le calcul du régulateur en utilisant les Inégalités Linéaires Matricielles (LMI). Puis une présentation générale des systèmes Linéaires à Paramètres Variants (LPV) est donnée dans la section 1.2. Deux approches de modélisation et commande pour les système LPV sont exposées. Ensuite, la section 1.3 décrit l'approche LPV polytopique et le section 1.4 la Représentation Linéaire Fractionnelle (LFR).

Tout au long du chapitre, ces outils sont développés pour les systèmes continus mais aussi pour les systèmes à temps discret.

1.1 Commande $\mathcal{H}_\infty$

Cette section définit les différentes notations utilisées dans la suite de ce manuscrit, ainsi que la commande robuste $\mathcal{H}_\infty$. Elle a pour but de fournir les outils nécessaires à la compréhension du reste du manuscrit, notamment par la définition des normes de signaux et de systèmes qui sont à la base de l'outil de commande robuste que nous avons sélectionné pour mener nos travaux. La synthèse de ce type de régulateur peut être effectuée de façon efficace par la résolution de problèmes d'optimisation convexe, utilisant des Inégalités Linéaires Matricielles (en anglais Linear Matrix Inequality : LMI), dont nous donnerons également la définition.

1.1.1 Définition

1.1.1.1 Système dynamique linéaire et stationnaire

Les équations d'état d'un système Linéaire stationnaire (LTI pour Linear Time Invariant) continu sont les suivantes :

$$\mathcal{G} : \begin{cases} \dot{x}(t) = Ax(t) + Bw(t) \\ z(t) = Cx(t) + Dw(t) \end{cases} \tag{1.1}$$

où $x(t) \in \mathbb{R}^{n_s}$ est le vecteur d'état, $w(t) \in \mathbb{R}^{n_w}$ le vecteur des entrées, $z(t) \in \mathbb{R}^{n_z}$ le vecteur des sorties et A, B, C, et D sont des matrices réelles aux dimensions

appropriées.

La fonction de transfert correspondante est :

$$\mathcal{G} : T(j\omega) = D + C((j\omega)I - A)^{-1}B \tag{1.2}$$

De façon similaire, un système LTI à temps discret est défini par les équations suivantes, *cf.* Åström and Wittenmark (1997) :

$$\mathcal{G} : \begin{cases} x_{k+1} = \hat{A}x_k + \hat{B}w_k \\ z_k = \hat{C}x_k + \hat{D}w_k \end{cases} \tag{1.3}$$

où les matrices $\hat{A}$, $\hat{B}$, $\hat{C}$ et $\hat{D}$ sont calculées à partir des matrices de la représentation d'état continue, suivant :

$$\begin{aligned} \hat{A}_h = e^{Ah} \quad \hat{B}_h = \int_0^h e^{A\tau} d\tau B \\ \hat{C} = C \qquad \hat{D} = D \end{aligned} \tag{1.4}$$

où h est la période d'échantillonnage.

Remarque : Dans la suite de ce manuscrit, tous les signaux en temps discret seront notés avec un indice k, et toutes les matrices des systèmes discrets avec $\hat{}$, suivant ce qui a été présenté ci-dessus.

1.1.1.2 Norme de signal

La norme $\mathcal{L}_2$ d'un signal $x(t) \in \mathbb{C}$, $x^*(t)$ étant le complexe conjugué de $x(t)$, est définie par :

$$\begin{aligned} \|x(t)\|_2 &= \sqrt{\int_0^{+\infty} x^*(t)x(t)dt} \\ &= \sqrt{\frac{1}{2\pi}\int_{-\infty}^{+\infty} X^*(j\omega)X(j\omega)d\omega} \end{aligned} \tag{1.5}$$

La norme $\mathcal{L}_\infty$ d'un signal x, admettant comme transformée de Laplace $X(p)$ est :

$$\|x(t)\|_\infty = \sup_t |x(t)|, \quad t \in \mathbb{C} \tag{1.6}$$

$$\|X\|_\infty = \sup_{Re(s)\geq 0} \|X(s)\| = \sup_\omega \|X(j\omega)\| \tag{1.7}$$

Finalement la norme H_∞ d'un système G, d'entrée w et de sortie z, est définie par :

$$\begin{aligned} \|G(j\omega)\|_\infty &= \sup_{\omega\in\mathbf{R}} \overline{\sigma}(G(j\omega)) \\ &= \max_{w(t)\in\mathcal{L}_2} \frac{\|z\|_2}{\|w\|_2} \end{aligned} \tag{1.8}$$

avec $\overline{\sigma}$ la valeur singulière maximale. Cette norme représente le gain maximum de la réponse fréquentielle d'un système, quelles que soient les entrées et les sorties.

1.1.1.3 Inégalité linéaires matricielles

Une inégalité linéaire matricielle est définie par une expression de la forme suivante :

$$F(x) = F_0 + \sum_{j=1}^{m} F_j x_j \geq 0 (> 0) \tag{1.9}$$

avec $x \in \mathbb{R}^m$ un vecteur réel, F_0, F_1, ... F_m des matrices réelles symétriques, et $F(x) \geq 0$ (respectivement $F(x) > 0$) signifie que $F(x)$ est une matrice symétrique semi-définie positive (respectivement définie positive) Cette inégalité matricielle linéaire caractérise un ensemble convexe selon x.

Exemple

Un exemple simple d'utilisation des LMIs en automatique est la stabilité quadratique au sens de Lyapunov. Un système dynamique à temps discret de type :

$$x_{k+1} = Ax_k \tag{1.10}$$

est quadratiquement stable si il existe une fonction de Lyapunov quadratique en l'état :

$$V(x_k) = x_k^T P x_k; \; P = P^T > 0 \tag{1.11}$$

telle que :

$$V(x_{k+1}) - V(x_k) < 0, \tag{1.12}$$

Le système (1.10) est alors quadratiquement stable si et seulement si on peut trouver une matrice P telle que :

$$A^T P A - P < 0, \quad P < 0 \tag{1.13}$$

Cette condition peut être réécrite, à l'aide d'un complément de Schur, sous la forme de la LMI suivante :

$$\begin{pmatrix} P & AP \\ PA^T & P \end{pmatrix} > 0 \tag{1.14}$$

1.1.2 Synthèse de contrôleur $\mathcal{H}_\infty$

La norme $\mathcal{H}_\infty$ définie dans la section précédente est utilisée en commande robuste pour la synthèse de régulateur. Le principe repose sur la minimisation de la norme $\mathcal{H}_\infty$ du système bouclé (entre les entrées exogènes $w(t)$ et les sorties contrôlées $z(t)$). Des fonctions de pondérations fréquentielles servent alors à définir les performances désirées en boucle fermée, comme détaillé ci-dessous.

1.1.2.1 Ajout de fonctions de pondérations fréquentielles

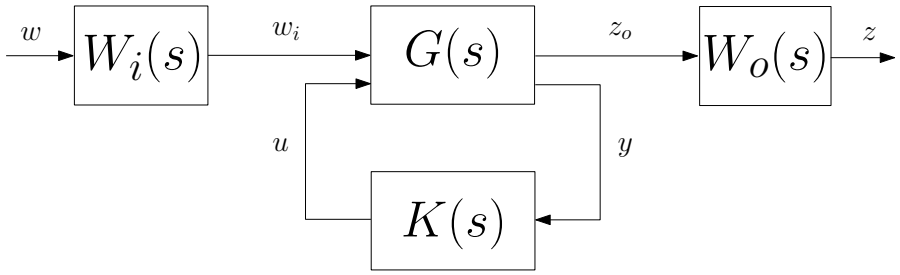

FIGURE 1.1 – Forme générale du problème $\mathcal{H}_\infty$

Des fonctions de pondération, définies dans le domaine fréquentiel, sont utilisées pour modifier les entrées exogènes (pondération $W_i(s)$) et les sorties à contrôler (pondération $W_o(s)$), du schéma de commande, comme le montre la figure 1.1. En modifiant les entrées et sorties, et donc la norme $\mathcal{H}_\infty$ à minimiser (entre w et z), cela permet de spécifier les performances voulues en boucle fermée, par exemple le temps de réponse, la marge de robustesse, et la possibilité de tenir compte des saturations (des actionneurs du système par exemple)(cf Skogestad and Postlethwaite (1996)).

Les fonctions de pondérations sont généralement soit des constantes, soit des filtres du premier ordre, ou d'ordre supérieur. Par exemple, un filtre du premier ordre de la forme (1.15) est souvent utilisée dans un problème d'asservissement pondérant l'erreur de poursuite.

$$W(s) = \frac{\frac{s}{M_s} + w_b}{s + w_b\epsilon} \tag{1.15}$$

Il permet de définir le temps de réponse désiré grâce à la bande passante w_b, la marge de robustesse voulue avec M_s et limiter l'erreur de poursuite avec ϵ.

Cette fonction de pondération peut être réécrite sous forme de représentation d'état, pour faciliter la connexion au système à contrôler, comme suit :

$$\begin{cases} \dot{x}_W = A_W x_W + B_W u_W \\ y_W = C_W x_W + D_W u_W \end{cases} \tag{1.16}$$

avec

$$A_W = -w_b\epsilon \quad B_W = w_b(1 - \frac{\epsilon}{M_s}) \quad C_W = 1 \quad D_W = \frac{1}{M_s} \tag{1.17}$$

où u_W représente l'entrée du filtre $W(s)$ et y_W la sortie.

1.1.2.2 Synthèse

La synthèse de contrôleur $\mathcal{H}_\infty$ utilise le problème standard présenté figure 1.2, où la matrice $P(s)$ représente la connexion du système $G(s)$ avec les fonctions de pondération ($W_i(s)$, $W_o(s)$) :

$$P : \left[\begin{array}{c} \dot{x} \\ \hline z \\ y \end{array}\right] = \left[\begin{array}{c|cc} \mathcal{A} & \mathcal{B}_1 & \mathcal{B}_2 \\ \hline \mathcal{C}_1 & \mathcal{D}_{11} & \mathcal{D}_{12} \\ \mathcal{C}_2 & \mathcal{D}_{21} & \mathcal{D}_{22} \end{array}\right] \left[\begin{array}{c} x \\ \hline w \\ u \end{array}\right] \tag{1.18}$$

avec x regroupant l'état du système et des pondérations $x \in \mathbb{R}^{n'_s}$, $z \in \mathbb{R}^{n_z}$ les sorties à contrôler, $y \in \mathbb{R}^{n_y}$ les mesures disponibles/nécessaires au contrôle, $w \in \mathbb{R}^{n_w}$ les sorties exogènes et $u \in \mathbb{R}^{n_u}$ les entrées de contrôle. $\mathcal{A} \in \mathbb{R}^{n'_s \times n'_s}$, $\mathcal{D}_{11} \in \mathbb{R}^{n_z \times n_w}$ et $\mathcal{D}_{22} \in \mathbb{R}^{n_y \times n_u}$.

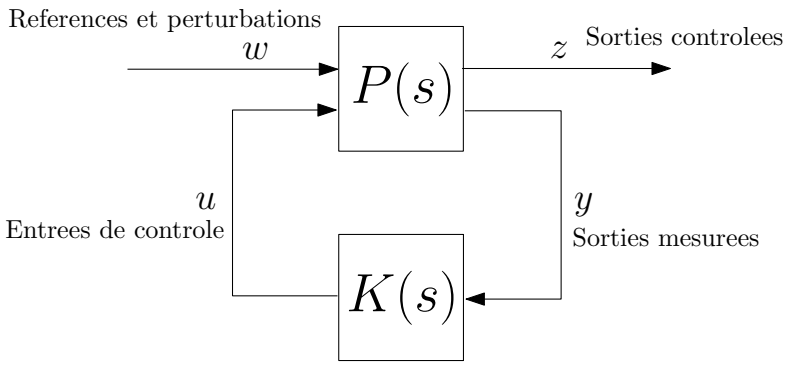

FIGURE 1.2 – Problème standard

Le régulateur que l'on cherche à calculer avec la méthodologie $\mathcal{H}_\infty$ est défini par :

$$K : \begin{cases} \dot{x}_K = A_K x_K + B_K y \\ u = C_K x_K + D_K y \end{cases} \tag{1.19}$$

avec $A_K \in \mathbb{R}^{n_K \times n_K}$.

La connexion de $P(s)$ avec le contrôleur $K(s)$ (selon la figure 1.2 aboutit à un système bouclé $P_{cl}(s)$ défini par l'équation suivante (dont l'état correspond à $\tilde{x} = \binom{x}{x_K}$) :

$$P_{cl}(s) : \left[\begin{array}{c} \dot{\tilde{x}} \\ z \end{array}\right] = \left(\begin{array}{cc} \mathcal{A}_{cl} & \mathcal{B}_{cl} \\ \mathcal{C}_{cl} & \mathcal{D}_{cl} \end{array}\right) \left[\begin{array}{c} \tilde{x} \\ w \end{array}\right] \tag{1.20}$$

où (en considérant la matrice $\mathcal{D}_{22} = 0$ par simplification)

$$\mathcal{A}_{cl} = \begin{pmatrix} \mathcal{A} + \mathcal{B}_2 D_K \mathcal{C}_2 & \mathcal{B}_2 C_K \\ B_K \mathcal{C}_2 & A_K \end{pmatrix} \tag{1.21}$$

$$\mathcal{B}_{cl} = \begin{pmatrix} \mathcal{B}_1 + \mathcal{B}_2 D_K \mathcal{D}_{21} \\ B_K \mathcal{D}_{21} \end{pmatrix} \tag{1.22}$$

$$\mathcal{C}_{cl} = \begin{pmatrix} \mathcal{C}_1 + \mathcal{D}_{12} D_K \mathcal{C}_2 & \mathcal{D}_{11} C_K \end{pmatrix} \tag{1.23}$$

$$\mathcal{D}_{cl} = \mathcal{D}_{11} + \mathcal{D}_{12} D_K \mathcal{D}_{21} \tag{1.24}$$

L'objectif de commande est alors de trouver un contrôleur $K(s)$ qui, à partir des mesures y, calcule les commandes u à appliquer au système, qui limitent les effets de w sur z, c'est à dire tel que la norme $\mathcal{H}_\infty$ de la fonction de transfert entre w et z satisfasse :

$$\|T_{zw}(s)\|_\infty := \left\|\mathcal{C}_{cl}(sI - \mathcal{A}_{cl})^{-1}\mathcal{B}_{cl} + \mathcal{D}_{cl}\right\|_\infty < \gamma_\infty \tag{1.25}$$

La résolution de ce problème par des Inégalités Linéaires Matricielles (LMI) revient à la résolution du Lemme Réel Borné (Scherer and Wieland, 2004) c'est à dire à trouver une matrice définie positive $X = X^T \geq 0$ telle que :

$$\begin{bmatrix} \mathcal{A}_{cl}^T X + X\mathcal{A}_{cl} & X\mathcal{B}_{cl} & \mathcal{C}_{cl}^T \\ \mathcal{B}_{cl}^T X & -\gamma_\infty I & \mathcal{D}_{cl}^T \\ \mathcal{C}_{cl} & \mathcal{D}_{cl} & -\gamma_\infty I \end{bmatrix} \prec 0 \tag{1.26}$$

Dans le cas où le système $P_{cl}(s)$ est connu, les seules inconnues sont X et γ : le Lemme réel borné est alors une LMI, c'est à dire un problème d'optimisation convexe, qui peut être résolu en utilisant les outils appropriés. C'est alors un moyen de calculer le norme $\mathcal{H}_\infty$ du système bouclé.

Dans le cas de la synthèse de contrôleur, le système $P_{cl}(s)$ contient les matrices du contrôleur que l'on cherche à déterminer : le Lemme Réel Borné est alors une Inégalité Bilinéaire Matricielle (BMI) qui ne peut pas être résolue directement.

Plusieurs méthodes existent pour résoudre ce problème, par exemple avec un changement de variable (Scherer et al., 1997), ou en utilisant le Lemme de Projection présenté dans Apkarian and Gahinet (1994) : c'est cette méthode que nous avons choisie.

Solution du Lemme Réel borné :

La synthèse de contrôleur se fait sous les hypothèses suivantes :

- $(\mathcal{A}, \mathcal{B}_2, \mathcal{C}_2)$ est stabilisable et détectable
- $\mathcal{D}_{22} = 0$

En partitionnant la matrice X de la manière suivante :

$$X = \begin{pmatrix} S & N \\ N^T & * \end{pmatrix}; \quad X^{-1} = \begin{pmatrix} R & M \\ M^T & * \end{pmatrix} \tag{1.27}$$

avec R, $S \in \mathbb{R}^{n'_s \times n'_s}$ et M, $N \in \mathbb{R}^{n'_s \times n_K}$, n_K étant l'ordre du contrôleur.

La résolution du lemme réel borné en temps continu est alors équivalent à la résolution des LMI suivantes :

$$\left(\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} \mathcal{A}R + R\mathcal{A}^T & R\mathcal{C}_1^T & \mathcal{B}_1 \\ \mathcal{C}_1 R & -\gamma_\infty I & \mathcal{D}_{11} \\ \hline \mathcal{B}_1^T & \mathcal{D}_{11}^T & -\gamma_\infty I \end{array}\right) \left(\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array}\right) \tag{1.28}$$

$$\left(\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} \mathcal{A}^T S + S\mathcal{A} & S\mathcal{B}_1 & \mathcal{C}_1^T \\ \mathcal{B}_1^T S & -\gamma_\infty I & \mathcal{D}_{11}^T \\ \hline \mathcal{C}_1 & \mathcal{D}_{11} & -\gamma_\infty I \end{array}\right) \left(\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array}\right) \tag{1.29}$$

$$\begin{pmatrix} R & I \\ I & S \end{pmatrix} \geq 0 \tag{1.30}$$

où les matrices $\mathcal{N}_R$ et $\mathcal{N}_S$ sont respectivement les bases du noyau de $(\mathcal{B}_2^T, \mathcal{D}_{12}^T)$ et $(\mathcal{C}_2, \mathcal{D}_{21})$. Les matrices R et S ainsi que le scalaire γ_∞ sont les variables de décision. On peut noter qu'il n'y a plus de couplage (multiplication entre 2 inconnues) donc ces équations sont bien des LMI, qui peuvent être résolues, en utilisant *Matlab*, ou *Yalmip* (Lofberg, 2004) par exemple.

Calcul du contrôleur : Les matrices du contrôleur $K = \begin{pmatrix} A_K & B_K \\ C_K & D_K \end{pmatrix}$ sont ensuite déduites des matrice R et S en suivant la procédure suivante :

- Vérifier que $rang(I - RS) \leq n_s$
- Calculer les deux matrices M et N de l'équation (1.27) suivant :

$$MN^T = I - RS \tag{1.31}$$

- Trouver la variable X solution du Lemme Réel Borné (équation (1.26)) en résolvant l'équation linéaire :

$$\begin{pmatrix} S & I \\ N^T & 0 \end{pmatrix} = X \begin{pmatrix} I & R \\ 0 & M^T \end{pmatrix} \tag{1.32}$$

- Résoudre le Lemme Réel Borné (équation (1.26)) avec cette valeur de X, les variables de décision étant alors les matrices du contrôleur K

Cette méthode permet de trouver un contrôleur continu satisfaisant la propriété (1.25).

Cas discret : La même méthodologie peut être appliquée pour la synthèse de contrôleur en temps discret.

De la même façon que dans le cas continu, des fonctions de pondérations servent à définir des objectifs pour la boucle fermée et sont ensuite connectées au système. La connexion de ce système augmenté avec le contrôleur $\hat{K}$ donne le système $\hat{P}$ suivant :

$$\hat{P} \ : \ \left[\begin{array}{c} x_{k+1} \\ \hline z_k \\ y_k \end{array}\right] = \left[\begin{array}{c|cc} \hat{\mathcal{A}} & \hat{\mathcal{B}}_1 & \hat{\mathcal{B}}_2 \\ \hline \hat{\mathcal{C}}_1 & \hat{\mathcal{D}}_{11} & \hat{\mathcal{D}}_{12} \\ \hat{\mathcal{C}}_2 & \hat{\mathcal{D}}_{21} & \hat{\mathcal{D}}_{22} \end{array}\right] \left[\begin{array}{c} x \\ \hline w \\ u \end{array}\right] \tag{1.33}$$

Dans ce cas, les LMI équivalentes à (1.28)-(1.30) deviennent :

$$\left(\begin{array}{c|c} \hat{\mathcal{N}}_R & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} \hat{\mathcal{A}}\hat{R}\hat{\mathcal{A}}^T - \hat{R} & \hat{\mathcal{A}}\hat{R}\hat{\mathcal{C}}_1^T & \hat{\mathcal{B}}_1 \\ \hat{\mathcal{C}}_1\hat{R}\hat{\mathcal{A}}^T & -\gamma I + \hat{\mathcal{C}}_1\hat{R}\hat{\mathcal{C}}_1^T & \hat{\mathcal{D}}_{11} \\ \hline \hat{\mathcal{B}}_1^T & \hat{\mathcal{D}}_{11}^T & -\gamma I \end{array}\right) \left(\begin{array}{c|c} \hat{\mathcal{N}}_R & 0 \\ \hline 0 & I \end{array}\right) \tag{1.34}$$

$$\left(\begin{array}{c|c} \hat{\mathcal{N}}_S & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} \hat{\mathcal{A}}^T\hat{S}\hat{\mathcal{A}} - \hat{S} & \hat{\mathcal{A}}^T\hat{S}\hat{\mathcal{B}}_1 & \hat{\mathcal{C}}_1^T \\ \hat{\mathcal{B}}_1^T\hat{S}\hat{\mathcal{A}} & -\gamma I + \hat{\mathcal{B}}_1^T\hat{S}\hat{\mathcal{B}}_1 & \hat{\mathcal{D}}_{11}^T \\ \hline \hat{\mathcal{C}}_1 & \hat{\mathcal{D}}_{11} & -\gamma I \end{array}\right) \left(\begin{array}{c|c} \hat{\mathcal{N}}_S & 0 \\ \hline 0 & I \end{array}\right) \tag{1.35}$$

$$\begin{pmatrix} \hat{R} & I \\ I & \hat{S} \end{pmatrix} \geq 0 \tag{1.36}$$

La reconstruction est identique à celle présentée précédemment pour le cas continu, en remplaçant le Lemme Réel Borné de l'équation (1.26) par sa version discrète :

$$\begin{bmatrix} -\hat{X}^{-1} & \hat{\mathcal{A}}_{cl} & \hat{\mathcal{B}}_{cl} & 0 \\ \hat{\mathcal{A}}_{cl}^T & -\hat{X} & 0 & \hat{\mathcal{C}}_{cl}^T \\ \hat{\mathcal{B}}_{cl}^T & 0 & -\gamma I & \hat{\mathcal{D}}_{cl}^T \\ 0 & \hat{\mathcal{C}}_{cl} & \hat{\mathcal{D}}_{cl} & -\gamma I \end{bmatrix} \leq 0 \tag{1.37}$$

Que ce soit dans le cas continu ou dans le cas discret, les fonctions de pondération fréquentielles utilisées pour fixer les objectifs de la commande peuvent agir sur différentes variables de la boucle de contrôle : l'erreur de poursuite, l'action, une des sorties mesurées, une des sorties à contrôler... Dans la prochaine section, nous présentons pour illustrer, un cas particulier : le problème de sensibilité mixte.

1.1.2.3 Problème de sensibilité mixte

Un des problèmes les plus répandus en commande $\mathcal{H}_\infty$ est le problème de sensibilité mixte. Il consiste à ajouter 2 pondérations dans la boucle de contrôle,

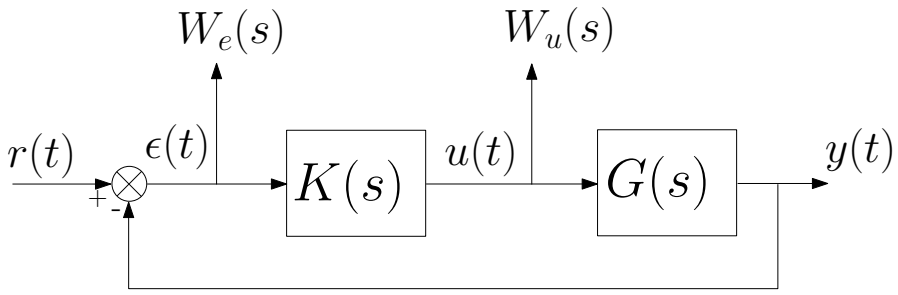

FIGURE 1.3 – Boucle fermée du problème de sensibilité mixte

une pondération $W_e(s)$ sur l'erreur de poursuite $\epsilon(t)$ et une seconde $W_u(s)$ sur l'action $u(t)$ comme sur le schéma de la figure (1.3).

Ces 2 pondérations agissent directement sur les fonctions de sensibilité S concernant l'erreur de poursuite, entre l'erreur $\epsilon(t)$ et la référence $r(t)$ et KS entre l'action $u(t)$ et la référence $r(t)$.

Rappelons que la fonction de sensibilité en sortie dans le cas multivariable S_y est définie par :

$$S_y(s) = (I_p + G(s)K(s))^{-1} \tag{1.38}$$

avec $p =$ le nombre de sorties mesurées.

La fonction de sensibilité KS_y qui permet de tenir compte de la saturation des actionneurs est définie de la façon suivante :

$$K(s)S_y(s) = \frac{U(s)}{R(s)} = K(s)(I_p + K(s)G(s))^{-1} \tag{1.39}$$

où nous notons (par simplicité) $U(s)$ la transformée de Laplace de $u(t)$. La pondération de l'erreur de poursuite $\epsilon(t)$ et de l'action $u(t)$ par les fonctions $W_e(s)$ et $W_u(s)$ permet de définir deux nouvelles sorties à contrôler, $e_1(t)$ et $e_2(t)$:

$$E_1(s) = W_e(s)S_y(s)R(s) \tag{1.40}$$

$$E_2(s) = W_u(s)K(s)S_y(s)R(s) \tag{1.41}$$

Dans ce cas, le système généralisé $P(s)$ devient :

$$P(s) = \left[\begin{array}{c|c} W_e(s) & -W_e(s)G(s) \\ 0 & W_u(s) \\ \hline I_p & -G(s) \end{array}\right] \tag{1.42}$$

La méthodologie présentée précédemment s'applique, et le problème $\mathcal{H}_\infty$ correspondant est de trouver un contrôleur $K(s)$ tel que :

$$\|T_{ew}(s)\|_\infty \leq \gamma \tag{1.43}$$

avec

$$T_{ew}(s) = \begin{bmatrix} W_e(s) \\ 0 \end{bmatrix} + \begin{bmatrix} -W_e(s)G(s) \\ W_u(s) \end{bmatrix} K(I_p + G(s)K(s))^{-1} I = \begin{bmatrix} W_e(s)S_y(s) \\ W_u(s)K(s)S_y(s) \end{bmatrix} \tag{1.44}$$

La méthode et les équations présentées ici dans le cas continu sont directement transposables au cas discret, sans difficultés particulières.

1.2 Système Linéaire à Paramètre Variant (LPV)

Un système Linéaire à Paramètres Variants (LPV) continu est défini par les équations suivantes :

$$\begin{cases} \dot{x}(t) = A(\rho(.))x(t) + B(\rho(.))w(t) \\ z(t) = C(\rho(.))x(t) + D(\rho(.))w(t) \end{cases} \tag{1.45}$$

avec $x(t) \in \mathbb{R}^{n_s}$ l'état du système, $w(t) \in \mathbb{R}^{n_w}$ le vecteur d'entrée et $z(t) \in \mathbb{R}^{n_z}$ le vecteur de sortie. Les matrices A, B, C et D dépendent du vecteur de paramètres $\rho(.)$, qui prend ses valeurs dans l'espace de paramètre $\mathcal{P}_\rho$, qui est l'ensemble convexe suivant :

$$\mathcal{P}_\rho := \left\{ \rho(.) := \begin{bmatrix} \rho_1(.) & \dots & \rho_n(.) \end{bmatrix}^T \in \mathbb{R}^n \ , \ \rho_i(.) \in \begin{bmatrix} \underline{\rho}_i & \overline{\rho}_i \end{bmatrix} \forall i = 1, \dots, n \right\} \tag{1.46}$$

où $\underline{\rho}_i$ et $\overline{\rho}_i$ indiquent respectivement les bornes minimales et maximales du paramètre ρ_i.

Les paramètres $\rho_i(.)$ peuvent dépendre du temps, mais aussi de l'état du système (on parle alors de système quasi-LPV). Pour ne pas alourdir l'écriture, le vecteur de paramètre sera noté ρ par la suite.

On remarque que si le vecteur de paramètres est constant, on retrouve un système LTI classique.

De la même façon, un système LPV à temps discret est défini par :

$$\begin{cases} x_{k+1} = \hat{A}(\rho_k)x_k + \hat{B}(\rho_k)w_k \\ z_k = \hat{C}(\rho_k)x_k + \hat{D}(\rho_k)w_k \end{cases} \tag{1.47}$$

avec le vecteur de paramètres ρ_k défini au temps k.

$$\rho_k = \begin{bmatrix} \rho_{k_1} & \dots & \rho_{k_n} \end{bmatrix}^T \in \mathbb{R}^n \ , \ \rho_{k_i} \in \begin{bmatrix} \underline{\rho}_{k_i} & \overline{\rho}_{k_i} \end{bmatrix} \forall i = 1, \dots, n \tag{1.48}$$

La synthèse de contrôleurs LPV adaptés à ce type de système ne peut pas se faire directement à partir de la méthodologie de synthèse de contrôleur $\mathcal{H}_\infty$. En effet une méthode simple pour calculer un correcteur en utilisant la méthodologie $\mathcal{H}_\infty$ présentée auparavant (voir la section 1.1.2) serait de choisir une variable de décision $\hat{X}$ (dépendant ou non du paramètre ρ) et d'appliquer le Lemme Réel Borné de l'équation (1.37) à un système LPV.

Pour un système en temps discret, et pour une variable de décision indépendante du paramètre, cela revient à résoudre la LMI suivante :

$$\begin{bmatrix} -\hat{X}^{-1} & \hat{\mathcal{A}}_{cl}(\rho) & \hat{\mathcal{B}}_{cl}(\rho) & 0 \\ \hat{\mathcal{A}}_{cl}(\rho)^T & -\hat{X} & 0 & \hat{\mathcal{C}}_{cl}(\rho)^T \\ \hat{\mathcal{B}}_{cl}(\rho)^T & 0 & -\gamma I & \hat{\mathcal{D}}_{cl}(\rho)^T \\ 0 & \hat{\mathcal{C}}_{cl}(\rho) & \hat{\mathcal{D}}_{cl}(\rho) & -\gamma I \end{bmatrix} \leq 0 \tag{1.49}$$

Or le vecteur de paramètre prend une infinité de valeur à l'intérieur de $\mathcal{P}_\rho$, ce qui entraîne une infinité de LMI à résoudre et rend cette approche impossible. Pour résoudre ce problème, il faut donc utiliser une méthode qui va permettre de se ramener à un problème de dimension finie. Il en existe 3 principales :

- le "gridding" ou "maillage" (voir par exemple Apkarian and Adams (2008)). Cette approche consiste à discrétiser l'espace des paramètres $\mathcal{P}_\rho$ (en un nombre fini de points) pour faire la synthèse du contrôleur à chaque point du maillage. Ensuite la stabilité de la boucle fermée est vérifiée sur un maillage plus fin de façon à "garantir" la propriété sur l'ensemble de l'espace $\mathcal{P}_\rho$.
- l'approche polytopique : cette approche est utilisée lorsque la dépendance du système est affine par rapport au vecteur de paramètres, qui varie alors à l'intérieur d'un polytope. Le système LPV est représenté comme la combinaison convexe des systèmes définis aux sommets du polytope, ces sommets étant les bornes de chaque paramètre $\begin{bmatrix} \underline{\rho}_i & \overline{\rho}_i \end{bmatrix}$ (voir Apkarian et al. (1995),Scherer and Wieland (2004)).
- la Representation Fractionnelle Linéaire (LFR). Cette méthode a été développée et est très employée pour l'étude de la robustesse en stabilité et en performances des systèmes linéaires. En outre, elle s'avère également très intéressante dans le cadre des systèmes LPV (Apkarian and Gahinet, 1995) comme nous allons le voir par la suite. La principe de la méthode est de diviser le système LPV en deux parties distinctes : l'une contenant toute la dépendance du système vis-à-vis des paramètres ρ_i, et l'autre la dynamique du système sous la forme d'un système LTI (indépendante du vecteur de paramètre) (*cf.* Apkarian and Gahinet (1995)).

L'approche dite par "gridding" ne sera pas étudié dans cette thèse. Les approches polytopique et LFR seront décrites et comparées dans le cas de la synthèse de contrôleur à période d'échantillonnage variable. Les 2 prochaines sections pré-

sentent ces deux approches de façon plus détaillée, mais dans un contexte général.

Dans la suite de cette thèse, le but est de synthétiser un correcteur LPV discret à période d'échantillonnage variable, c'est pourquoi l'approche discrète sera privilégiée.

Remarque : Dans le cas général d'un système LPV, on peut être amené à considérer des bornes sur les dérivées des paramètres ($\dot{\rho}_i$). Toutefois, les approches considérées dans ce travail s'inscrivant dans le contexte de la stabilité quadratique (Scherer and Wieland, 2004; Apkarian and Gahinet, 1995), ceci n'est pas nécessaire, ce qui revient à dire que la vitesse de variation des paramètres n'est pas contrainte.

1.3 Approche polytopique pour les systèmes LPV

Pour garantir la stabilité d'un système LPV en boucle fermée, l'approche polytopique consiste à appliquer les outils de la commande robuste $\mathcal{H}_\infty$ à un système dont les paramètres variants évoluent à l'intérieur d'un polytope. La propriété de convexité apportée par le polytope permet alors de limiter l'analyse de stabilité aux systèmes définis aux sommets du polytope, ce qui garantit tout de même la stabilité de tout système à l'intérieur du polytope.

De la même façon, pour la synthèse de contrôleur, l'ensemble du calcul est réduit à calculer un contrôleur pour chaque sommet du polytope, ce qui réduit le problème à une dimension finie. Cette section donne une version résumée des résultats en temps continu et discret ; pour plus de détails, le lecteur peut se référer en particulier aux articles suivants : Apkarian et al. (1995), Scherer and Wieland (2004).

1.3.1 Modélisation sous forme polytopique

A partir de l'équation d'un système LPV discret (1.47), un système LPV polytopique discret peut être déduit facilement si la dépendance des matrices d'état ($\hat{A}$, $\hat{B}$, $\hat{C}$ et $\hat{D}$) vis-à-vis du vecteur de paramètres ρ_k est affine.

Comme chaque paramètre est supposé borné : $\rho_{k_i} \in \begin{bmatrix} \underline{\rho}_{k_i} & \overline{\rho}_{k_i} \end{bmatrix}$, le vecteur de paramètres ρ_k évolue à l'intérieur d'un polytope délimité par $N = 2^n$ sommets ω_i (n étant le nombre de paramètres), suivant :

$$\rho_k \in \mathbf{Co}\{\omega_1, \ldots, \omega_N\} \tag{1.50}$$

Le vecteur de paramètre ρ_k s'écrit comme la combinaison convexe suivante des sommets ω_i :

$$\rho_k = \sum_{i=1}^{N} \alpha_i(k)\omega_i, \;\; \alpha_i(k) \geq 0, \;\; \sum_{i=1}^{N} \alpha_i(k) = 1 \tag{1.51}$$

Ces sommets sont définis par un vecteur $\omega_i = [\nu_{i1}, \ldots, \nu_{in}]$ avec ν_{ij} égal à $\underline{\rho_j}$ ou $\overline{\rho_j}$.

Dans ce cas, le système est défini comme la combinaison convexe des systèmes aux sommets du polytope ω_i :

$$\begin{pmatrix} A(\rho_k) & B(\rho_k) \\ C(\rho_k) & D(\rho_k) \end{pmatrix} = \sum_{i=1}^{N} \alpha_i(k) \begin{pmatrix} A(\omega_i) & B(\omega_i) \\ C(\omega_i) & D(\omega_i) \end{pmatrix} \tag{1.52}$$

$$= \sum_{i=1}^{N} \alpha_i(k) \begin{pmatrix} A_i & B_i \\ C_i & D_i \end{pmatrix} \tag{1.53}$$

$$\in \mathbf{Co}\left\{ \begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix}, \ldots, \begin{pmatrix} A_N & B_N \\ C_N & D_N \end{pmatrix} \right\} \tag{1.54}$$

Exemple

On considère le système discrèt suivant, avec 2 variables d'état x_1 et x_2 ainsi que 2 paramètres variants ρ_1 et ρ_2

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_{k+1} = \begin{pmatrix} x_1 + \rho_1 x_2 \\ \rho_2 x_1 + x_2 \end{pmatrix}_k \tag{1.55}$$

$$= \begin{pmatrix} 1 & \rho_1 \\ \rho_2 & 1 \end{pmatrix}_k \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_k \tag{1.56}$$

$$= \hat{A}(\rho_{1,2}) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_k \tag{1.57}$$

Ce système est composé de 2 paramètres, bornés par $\begin{bmatrix} \underline{\rho}_{1,2} & \overline{\rho}_{1,2} \end{bmatrix}$ donc l'espace de paramètres est défini par un polytope (rectangle) à 4 sommets définis comme suit :

$$\mathcal{P}_\rho = \left\{ (\underline{\rho}_1, \underline{\rho}_2), (\underline{\rho}_1, \overline{\rho}_2), (\overline{\rho}_1, \underline{\rho}_2), (\overline{\rho}_1, \overline{\rho}_2) \right\} \tag{1.58}$$

Les coordonnées polytopiques (α_i de l'équation (1.52)) sont ensuite obtenues de la manière suivante, pour chaque sommet du polytope ω_i, en fonction de ρ_1 et ρ_2 à l'instant k :

$$
\begin{aligned}
\omega_1 &= (\underline{\rho}_1, \underline{\rho}_2), & \alpha_1^{(k)} &= \left(\frac{\overline{\rho_1} - \rho_1^{(k)}}{\overline{\rho_1} - \underline{\rho_1}}\right) \times \left(\frac{\overline{\rho_2} - \rho_2^{(k)}}{\overline{\rho_2} - \underline{\rho_2}}\right) \\
\omega_2 &= (\underline{\rho}_1, \overline{\rho}_2), & \alpha_2^{(k)} &= \left(\frac{\overline{\rho_1} - \rho_1^{(k)}}{\overline{\rho_1} - \underline{\rho_1}}\right) \times \left(\frac{\rho_2^{(k)} - \underline{\rho_2}}{\overline{\rho_2} - \underline{\rho_2}}\right) \\
\omega_3 &= (\overline{\rho}_1, \underline{\rho}_2), & \alpha_3^{(k)} &= \left(\frac{\rho_1^{(k)} - \underline{\rho_1}}{\overline{\rho_1} - \underline{\rho_1}}\right) \times \left(\frac{\overline{\rho_2} - \rho_2^{(k)}}{\overline{\rho_2} - \underline{\rho_2}}\right) \\
\omega_4 &= (\overline{\rho}_1, \overline{\rho}_2), & \alpha_4^{(k)} &= \left(\frac{\rho_1^{(k)} - \underline{\rho_1}}{\overline{\rho_1} - \underline{\rho_1}}\right) \times \left(\frac{\rho_2^{(k)} - \underline{\rho_2}}{\overline{\rho_2} - \underline{\rho_2}}\right)
\end{aligned}
\tag{1.59}
$$

Finalement, le système LPV de l'équation (1.55) mis sous forme polytopique est décrit par :

$$
\begin{aligned}
\begin{pmatrix} A(\rho_{1,2}) & B(\rho_{1,2}) \\ C(\rho_{1,2}) & D(\rho_{1,2}) \end{pmatrix} &= \alpha_1^{(k)} \begin{pmatrix} A(\omega_1) & B(\omega_1) \\ C(\omega_1) & D(\omega_1) \end{pmatrix} + \alpha_2^{(k)} \begin{pmatrix} A(\omega_2) & B(\omega_2) \\ C(\omega_2) & D(\omega_2) \end{pmatrix} \\
&+ \alpha_3^{(k)} \begin{pmatrix} A(\omega_3) & B(\omega_3) \\ C(\omega_3) & D(\omega_3) \end{pmatrix} + \alpha_4^{(k)} \begin{pmatrix} A(\omega_4) & B(\omega_4) \\ C(\omega_4) & D(\omega_4) \end{pmatrix}
\end{aligned}
\tag{1.60}
$$

1.3.2 Synthèse de contrôleur polytopique

Après l'obtention d'un modèle polytopique, un correcteur polytopique peut être calculé en suivant la méthodologie présentée ci-dessous. Ce contrôleur n'est pas uniquement robuste à la variation de paramètre, il s'adapte à la valeur courante (mesurée ou estimée) du paramètre. Les matrices du contrôleur sont ainsi recalculées à chaque instant, en fonction de la valeur du paramètre.

La méthode choisie est basée sur la synthèse $\mathcal{H}_\infty$, donc la première étape consiste à spécifier les performances désirées en boucle fermée grâce à des fonctions de pondérations fréquentielles. Ces pondérations sont ensuite connectées au système pour former le modèle augmenté P de la figure 1.2.

Pour garantir la stabilité quadratique de la boucle fermée, ainsi que les performances au sens $\mathcal{H}_\infty$, et ce quelle que soit la valeur du vecteur de paramètres $\rho_k \in \mathcal{P}_\rho$, une seule variable de décision X doit vérifier la propriété pour chaque LMI (pour chacun des systèmes sommets (A_i, B_i, C_i, D_i)).

La synthèse d'un contrôleur polytopique à partir du modèle polytopique de l'équation (1.52), sous les hypothèses :

– $\hat{\mathcal{D}}_{22}(\rho_k) = 0$, c'est à dire $\hat{\mathcal{D}}_{22_i} = 0$ pour tout $i = 1,\ldots,N$

- $\hat{\mathcal{B}}_2(\rho_k)$, $\hat{\mathcal{C}}_2(\rho_k)$, $\hat{\mathcal{D}}_{12}(\rho_k)$ et $\hat{\mathcal{D}}_{21}(\rho_k)$ sont indépendants du vecteur de paramètres ρ_k, c'est à dire $\hat{\mathcal{B}}_2(\rho_k) = \hat{\mathcal{B}}_2$, $\hat{\mathcal{C}}_2(\rho_k) = \hat{\mathcal{C}}_2$, $\hat{\mathcal{D}}_{12}(\rho_k) = \hat{\mathcal{D}}_{12}$ et $\hat{\mathcal{D}}_{21}(\rho_k) = \hat{\mathcal{D}}_{21}$
- les paires $(\hat{\mathcal{A}}(\rho_k), \hat{\mathcal{B}}_2)$ et $(\hat{\mathcal{A}}(\rho_k), \hat{\mathcal{C}}_2)$ sont respectivement quadratiquement stabilisable et detectable sur $\mathcal{P}_\rho$

revient donc à résoudre les LMI suivantes pour chaque sommet du polytope, c'est à dire $N = 2^n$ fois :

$$\left(\begin{array}{c|c} \hat{\mathcal{N}}_R & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} \hat{\mathcal{A}}_i\hat{R}\hat{\mathcal{A}}_i^T - \hat{R} & \hat{\mathcal{A}}_i\hat{R}\hat{\mathcal{C}}_{1i}^T & \hat{\mathcal{B}}_{1i} \\ \hat{\mathcal{C}}_{1i}\hat{R}\hat{\mathcal{A}}_i^T & -\gamma I + \hat{\mathcal{C}}_{1i}\hat{R}\hat{\mathcal{C}}_{1i}^T & \hat{\mathcal{D}}_{11i} \\ \hline \hat{\mathcal{B}}_{1i}^T & \hat{\mathcal{D}}_{11i}^T & -\gamma I \end{array}\right) \left(\begin{array}{c|c} \hat{\mathcal{N}}_R & 0 \\ \hline 0 & I \end{array}\right) < 0, \quad i = 1 \dots N \tag{1.61}$$

$$\left(\begin{array}{c|c} \hat{\mathcal{N}}_S & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} \hat{\mathcal{A}}_i^T\hat{S}\hat{\mathcal{A}}_i - \hat{S} & \hat{\mathcal{A}}_i^T\hat{S}\hat{\mathcal{B}}_{1i} & \hat{\mathcal{C}}_{1i}^T \\ \hat{\mathcal{B}}_{1i}^T\hat{S}\hat{\mathcal{A}}_i & -\gamma I + \hat{\mathcal{B}}_{1i}^T\hat{S}\hat{\mathcal{B}}_{1i} & \hat{\mathcal{D}}_{11i}^T \\ \hline \hat{\mathcal{C}}_{1i} & \hat{\mathcal{D}}_{11i} & -\gamma I \end{array}\right) \left(\begin{array}{c|c} \hat{\mathcal{N}}_S & 0 \\ \hline 0 & I \end{array}\right) < 0, \quad i = 1 \dots N \tag{1.62}$$

$$\begin{pmatrix} \hat{R} & I \\ I & \hat{S} \end{pmatrix} \geq 0 \tag{1.63}$$

avec $\hat{\mathcal{N}}_R$ et $\hat{\mathcal{N}}_S$ respectivement les bases de l'espace nul de $(\hat{\mathcal{B}}_2^T, \hat{\mathcal{D}}_{12}^T)$ et $(\hat{\mathcal{C}}_2, \hat{\mathcal{D}}_{21})$.

Le calcul des matrices du correcteur polytopique discret $(\hat{A}_{K_i}, \hat{B}_{K_i}, \hat{C}_{K_i}, \hat{D}_{K_i})$ à chaque sommet du polytope se fait de manière équivalente au cas LTI, c'est à dire en calculant la matrice de décision $\hat{X}$ à partir de $\hat{R}$ et $\hat{S}$, puis en résolvant N fois le Lemme Réel Borné de l'équation (1.37) aux sommets.

Finalement, les matrices du contrôleur pour un point de fonctionnement ρ_k fixé sont obtenues par une combinaison convexe des correcteurs sommets, suivant :

$$\begin{pmatrix} \hat{A}_K(\rho_k) & \hat{B}_K(\rho_k) \\ \hat{C}_K(\rho_k) & \hat{D}_K(\rho_k) \end{pmatrix} = \sum_{i=1}^{N} \alpha_i(k) \begin{pmatrix} \hat{A}_{K_i} & \hat{B}_{K_i} \\ \hat{C}_{K_i} & \hat{D}_{K_i} \end{pmatrix} \tag{1.64}$$

Remarque : le cas continu La même méthode peut être appliquée pour la synthèse de contrôleur polytopique en temps continu (voir Apkarian and Gahinet

(1994) pour plus de détails). Les LMI (1.61)-(1.63) deviennent alors :

$$\left(\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} A_iR + RA_i^T & RC_{1i}^T & B_{1i} \\ C_{1i}R & -\gamma I & D_{11i} \\ \hline B_{1i}^T & D_{11i}^T & -\gamma I \end{array}\right) \left(\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array}\right)$$
$$< 0, \quad i = 1 \dots N \tag{1.65}$$

$$\left(\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array}\right)^T \left(\begin{array}{cc|c} A_i^TS + SA_i & SB_{1i} & C_{1i}^T \\ B_{1i}^TS & -\gamma I & D_{11i}^T \\ \hline C_{1i} & D_{11i} & -\gamma I \end{array}\right) \left(\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array}\right)$$
$$< 0, \quad i = 1 \dots N \tag{1.66}$$

$$\begin{pmatrix} R & I \\ I & S \end{pmatrix} \geq 0 \tag{1.67}$$

avec les matrices $\mathcal{N}_R$ et $\mathcal{N}_S$ définies comme précédemment. La reconstruction du contrôleur se fait de façon analogue au cas discret, en remplaçant le Lemme Réel Borné par son expression continue.

1.4 Approche LFR pour les systèmes LPV

La Representation Linéaire Fractionnelle (LFR) est principalement utilisée en analyse robuste, pour étudier l'influence d'incertitudes sur la stabilité et les performances d'un système en boucle fermée (cf Skogestad and Postlethwaite (1996).

Les LFR peuvent également servir à la construction de modèles dynamiques dépendants de paramètres variants, en supposant que le paramètre est connu (mesurable) ou qu'il peut être estimé (Biannic, 1996; Gauthier, 2007; Wu and Dong, 2006).

1.4.1 Modélisation LFR

La modélisation d'un système LPV sous la forme d'une Représentation Linéaire Fractionnelle (LFR) consiste à séparer le système en deux parties distinctes : le système LTI $P(z)$ d'une part, indépendant des paramètres variants, et la partie variante $\Theta(\rho)$ d'autre part, comme le montre la figure 1.4.

La matrice $\Theta(\rho)$, supposée connue, décrit la façon dont le vecteur de paramètres influence le système. Cette matrice est donnée sous la forme d'une matrice diagonale, faisant intervenir des scalaires répétés, et/ou des matrices pleines (comme lorsque la forme LFR est utilisée pour l'étude de l'influence des incertitudes).

Le système LPV de l'équation (1.47) peut être représenté sous la forme d'une

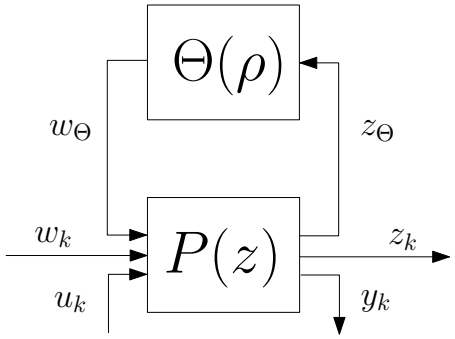

FIGURE 1.4 – Système sous forme LFR

LFR comme suit :

$$\left[\begin{array}{c} x_{k+1} \\ \hline z_{\Theta_k} \\ z_k \\ y_k \end{array}\right] = \left[\begin{array}{c|ccc} \hat{A} & \hat{B}_\Theta & \hat{B}_1 & \hat{B}_2 \\ \hline \hat{C}_\Theta & \hat{D}_{\Theta\Theta} & \hat{D}_{\Theta 1} & \hat{D}_{\Theta 2} \\ \hat{C}_1 & \hat{D}_{1\Theta} & \hat{D}_{11} & \hat{D}_{12} \\ \hat{C}_2 & \hat{D}_{2\Theta} & \hat{D}_{21} & \hat{D}_{22} \end{array}\right] \left[\begin{array}{c} x_k \\ \hline w_{\Theta_k} \\ w_k \\ u_k \end{array}\right] \tag{1.68}$$

et on définit pour la suite : $\tilde{A} = \hat{A}$, $\tilde{B} = \left[\begin{array}{ccc} \hat{B}_\Theta & \hat{B}_1 & \hat{B}_2 \end{array}\right]$, $\tilde{C} = \left[\begin{array}{ccc} \hat{C}_\Theta & \hat{C}_1 & \hat{C}_2 \end{array}\right]^T$ et $\tilde{D} = \left[\begin{array}{ccc} \hat{D}_{\Theta\Theta} & \hat{D}_{\Theta 1} & \hat{D}_{\Theta 2} \\ \hat{D}_{1\Theta} & \hat{D}_{11} & \hat{D}_{12} \\ \hat{D}_{2\Theta} & \hat{D}_{21} & \hat{D}_{22} \end{array}\right]$

La Transformation Linéaire Fractionnelle (LFT) supérieure (Zhou et al., 1996) définissant l'interconnexion de la matrice Θ et du système $P(z)$ est donnée par :

$$F_u(P, \Theta) = \tilde{D} + \tilde{C}\Theta(I - \tilde{A}\Theta)^{-1}\tilde{B} \tag{1.69}$$

et représente le système d'équations :

$$\left[\begin{array}{c} x_{k+1} \\ z_k \\ y_k \end{array}\right] = F_u(P, \Theta) \left[\begin{array}{c} x_k \\ w_k \\ u_k \end{array}\right] \tag{1.70}$$

Exemple

On reprend ici le système LPV discret donné en exemple dans la section polytopique pour en donner la modélisation sous la forme d'une LFR.

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_{k+1} = \begin{pmatrix} x_1 + \rho_1 x_2 \\ \rho_2 x_1 + x_2 \end{pmatrix}_k \tag{1.71}$$

On définit la matrice de paramètres Θ par une matrice diagonale qui contient

les 2 paramètres :

$$\Theta_k = \begin{pmatrix} \rho_{1_k} & 0 \\ 0 & \rho_{2_k} \end{pmatrix} \tag{1.72}$$

Le système est placé sous la LFR décrite figure 1.4, avec

$$z_\Theta = \begin{pmatrix} x_{2_k} \\ x_{1_k} \end{pmatrix}; \quad w_\Theta = \Theta_k \times z_\Theta \tag{1.73}$$

et P tel que :

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_{k+1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_k + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} w_\Theta \tag{1.74}$$

1.4.2 Synthèse LFR

Utilisant un modèle sous forme LFR, cette section décrit la méthodologie de calcul d'un correcteur LFR. Comme pour le contrôleur polytopique, un contrôleur LFR s'adaptera à la valeur mesurée ou estimée des paramètres. Dans le cas LFR, le contrôleur est défini par la LFT inférieure de la matrice LTI du contrôleur $K(z)$ avec la matrice des paramètres Θ, suivant :

$$F_l(K, \Theta) := K_{11} + K_{12}\Theta(I - K_{22}\Theta)^{-1}K_{21} \tag{1.75}$$

où le correcteur K est divisé en K_{11}, K_{12}, K_{21} et K_{22} (en accord avec la figure 1.5 suivant :

$$\begin{pmatrix} u \\ \tilde{z}_\Theta \end{pmatrix} = \left[\begin{array}{c|c} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right] \begin{pmatrix} y \\ \tilde{w}_\Theta \end{pmatrix} \tag{1.76}$$

Le schéma de contrôle sous forme LFR est représenté sur la figure 1.5, où l'on voit apparaître la dépendance entre la matrice de paramètres Θ, le modèle généralisé du système $P(z)$ et le contrôleur $K(z)$. Il faut noter que même si la figure 1.5 représente le cas où la même matrice Θ représente le séquencement du système et du contrôleur, on peut également considérer pour le contrôleur un sous-ensemble seulement de Θ.

Le problème consiste alors à déterminer le contrôleur discret $K(z)$ qui stabilise le système et respecte les spécifications définies par des fonctions de pondérations incluses dans la boucle de contrôle (suivant la méthode $\mathcal{H}_\infty$, le modèle généralisé $P(z)$ contient le système et les fonctions de pondérations, comme décrit dans la section 1.1.2).

On donne ici les grandes lignes de la méthode à employer ; pour plus de détails, le lecteur peut se référer à Apkarian and Gahinet (1995).

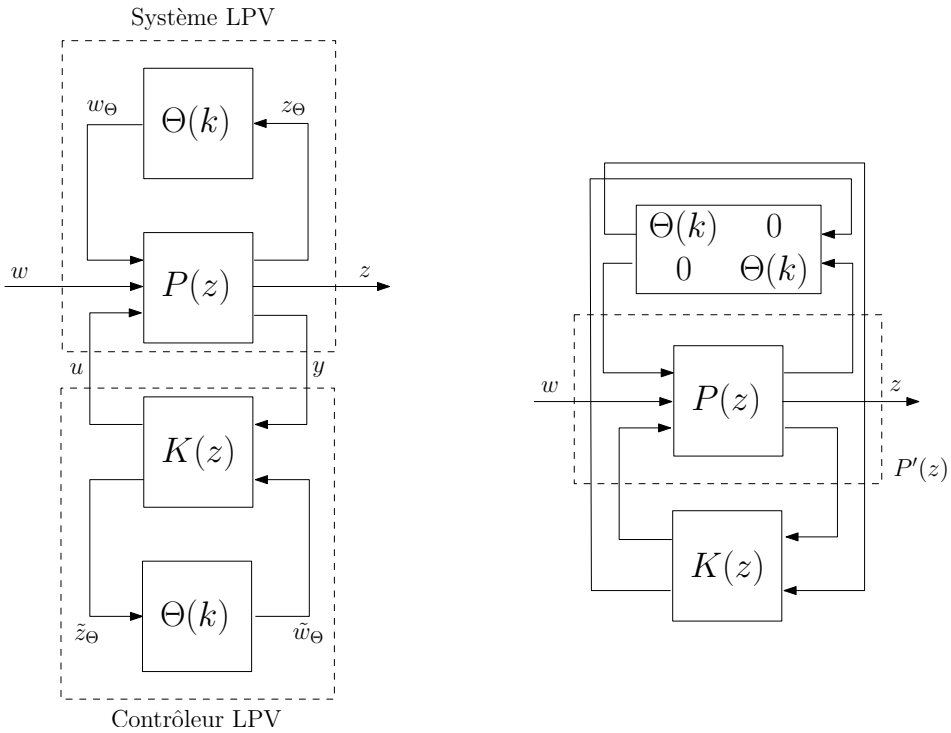

FIGURE 1.5 – Structure de commande LFR

De la même façon que pour l'approche polytopique, le calcul du correcteur se fait par l'application du Lemme Réel Borné. Pour réduire le conservatisme dû à la présence du bloc "d'incertitude" Θ (qui contient les paramètres), le problème est reformulé en utilisant des matrices de mise à l'échelle L_3 et J_3 (tel que L_3, $J_3 > 0$: $L_3\Theta = \Theta L_3$, $J_3\Theta = \Theta J_3$, $\forall\Theta \in \Delta$, Δ étant la structure de l'incertitude). Sous les hypothèses :

- les paires $(\hat{A}, \hat{B}_2)$ et $(\hat{A}, \hat{C}_2)$ sont respectivement stabilisable et détectable
- $\hat{D}_{22} = 0$

Le calcul du correcteur se fait alors grâce à la résolution des LMI suivantes :

$$\tilde{\mathcal{N}}_r^T \begin{pmatrix} \hat{A}\hat{R}\hat{A}^T - \hat{R} + \hat{B}_\theta \hat{J}_3 \hat{B}_\theta^T & \star & \star & \star \\ \hat{C}_\theta \hat{R}\hat{A}^T + \hat{D}_{\theta\theta}\hat{J}_3\hat{B}_\theta^T & \hat{C}_\theta\hat{R}\hat{C}_\theta^T + \hat{D}_{\theta\theta}\hat{J}_3\hat{D}_{\theta\theta}^T - \hat{J}_3 & \star & \star \\ \hat{C}_1\hat{R}\hat{A}^T + \hat{D}_{1\theta}\hat{J}_3\hat{B}_\theta^T & \hat{C}_1\hat{R}\hat{C}_\theta^T + \hat{D}_{1\theta}\hat{J}_3\hat{D}_{\theta\theta}^T & \hat{C}_1\hat{R}\hat{C}_1^T + \hat{D}_{1\theta}\hat{J}_3\hat{D}_{1\theta}^T - \gamma I & \star \\ \hat{B}_1^T & \hat{D}_{\theta 1}^T & \hat{D}_{11}^T & -\gamma I \end{pmatrix} \tilde{\mathcal{N}}_r < 0 \tag{1.77}$$

$$\tilde{\mathcal{N}}_s \begin{pmatrix} \hat{A}^T\hat{S}\hat{A} - \hat{S} + \hat{C}_\theta^T\hat{L}_3\hat{C}_\theta & \star & \star & \star \\ \hat{B}_\theta^T\hat{S}\hat{A} + \hat{D}_{\theta\theta}^T\hat{L}_3\hat{C}_\theta & \hat{B}_\theta^T\hat{S}\hat{B}_\theta + \hat{D}_{\theta\theta}^T\hat{L}_3\hat{D}_{\theta\theta} - \hat{L}_3 & \star & \star \\ \hat{B}_1^T\hat{S}\hat{A} + \hat{D}_{\theta 1}^T\hat{L}_3\hat{C}_\theta & \hat{B}_1^T\hat{S}\hat{B}_\theta + \hat{D}_{\theta 1}^T\hat{L}_3\hat{D}_{\theta\theta} & \hat{B}_1^T\hat{S}\hat{B}_1 + \hat{D}_{\theta 1}^T\hat{L}_3\hat{D}_{\theta 1} - \gamma I & \star \\ \hat{C}_1 & \hat{D}_{1\theta} & \hat{D}_{11} & -\gamma I \end{pmatrix} \tilde{\mathcal{N}}_s < 0 \tag{1.78}$$

$$\begin{pmatrix} \hat{R} & I \\ I & \hat{S} \end{pmatrix} > 0 \tag{1.79}$$

$$\begin{pmatrix} \hat{L}_3 & I \\ I & \hat{J}_3 \end{pmatrix} > 0 \tag{1.80}$$

avec $\tilde{\mathcal{N}}_r$ et $\tilde{\mathcal{N}}_s$ respectivement bases du noyau de $(\hat{B}_2^T, \hat{D}_{\Theta 2}^T, \hat{D}_{12}^T, 0)$ et $(\hat{C}_2^T, \hat{D}_{2\Theta}^T, \hat{D}_{21}^T, 0)$

A partir des matrices de décision obtenues par la résolution des LMIs précédentes ($\hat{R}$, $\hat{S}$, $\hat{L}_3$ et $\hat{J}_3$), les matrices du contrôleur $\hat{\Omega}$ sont calculées sous la forme :

$$\hat{\Omega} = \begin{bmatrix} \hat{A}_K & \hat{B}_{K1} & \hat{B}_{K\theta} \\ \hat{C}_{K1} & \hat{D}_{K11} & \hat{D}_{K1\theta} \\ \hat{C}_{K\theta} & \hat{D}_{K\theta 1} & \hat{D}_{K\theta\theta} \end{bmatrix} \in \mathbb{R}^{(k+m_2+r)\times(k+p_2+r)} \tag{1.81}$$

La reconstruction du correcteur est décrite en détails dans cet article référence Apkarian and Gahinet (1995), on donne ici les étapes importantes du calcul.

- A partir des matrices de décisions $\hat{R}$ et $\hat{S}$ obtenues par la résolution des LMI précédentes, calculer la matrice $\hat{X}_{cl}$ solution de Lemme Réel Borné Cette étape est analogue au cas continu présenté dans les equations 1.31 et 1.32.
- Reconstruire les matrices de mise à l'échelle complète à partir de $\hat{L}_3$ et $\hat{J}_3$:

$$\hat{L} = \begin{pmatrix} \hat{L}_1 & \hat{L}_2 \\ \hat{L}_2^T & \hat{L}_3 \end{pmatrix} > 0 \ , \quad \hat{L}^{-1} = \begin{pmatrix} * & * \\ * & \hat{J}_3 \end{pmatrix} \tag{1.82}$$

  La matrice $\hat{L}$ peut être reconstruite de la même façon que $\hat{X}_{cl}$ (avec une décomposition en valeur singulière sur chaque bloc diagonal de $I - \hat{L}_3\hat{J}_3$). Puis à partir de $\hat{L}$ calculer les matrices $\hat{\mathcal{L}}$ et $\hat{\mathcal{J}}$ suivantes :

$$\hat{\mathcal{L}} = \begin{pmatrix} \hat{L} & 0 \\ 0 & \mathcal{I}_{p_1} \end{pmatrix} \ , \quad \hat{\mathcal{J}} = \hat{\mathcal{L}}^{-1} \tag{1.83}$$

– finalement résoudre la LMI suivante, dont la variable de décision est le groupement $\hat{\Omega}$ des matrices de la représentation d'état discrète du contrôleur :

$$\hat{\Psi} + \hat{Q}^T \hat{\Omega}^T \hat{P} + \hat{P}^T \hat{\Omega} \hat{Q} < 0 \tag{1.84}$$

avec :

$$\hat{\Psi} = \begin{bmatrix} -\hat{X}_{cl}^{-1} & \hat{A}_0 & \hat{B}_0 & 0 \\ \hat{A}_0^T & -\hat{X}_{cl} & 0 & \hat{C}_0^T \\ \hat{B}_0^T & 0 & -\gamma\hat{\mathcal{L}} & \hat{D}_{11}^T \\ 0 & \hat{C}_0 & \hat{D}_{11} & -\gamma\hat{\mathcal{J}} \end{bmatrix} \tag{1.85}$$

$$\hat{P} = \begin{bmatrix} \hat{\mathcal{B}}^T & 0 & 0 & \hat{\mathcal{D}}_{12}^T \end{bmatrix} \ , \ \hat{Q} = \begin{bmatrix} 0 & \hat{\mathcal{C}} & \hat{\mathcal{D}}_{21} & 0 \end{bmatrix} \tag{1.86}$$

Remarque : Que ce soit dans le cas polytopique ou dans le cas LFR, le nombre de paramètres a une influence sur les LMIs à résoudre. Pour l'approche polytopique, c'est le nombre de LMI qui augmente avec le nombre de paramètres ($2^n + 1$ LMIs avec n le nombre de paramètres), ce qui peut poser problème pour un trop grand nombre de paramètres (au delà de 3 ou 4 paramètres). Pour l'approche LFR, le nombre de LMI reste constant (4 LMIs à résoudre), c'est la taille des matrices de la LMI qui augmente ; il en découle un problème de plus en plus difficile à résoudre avec l'augmentation du nombre de paramètres.

1.4.2.1 Conclusion

Ce chapitre a présenté de manière succincte les différents outils de modélisation de systèmes et de synthèse de contrôleurs, pour les systèmes LTI puis pour les systèmes LPV. Les approches polytopique et LFR ont été présentées. Elle sont utilisée dans le chapitre suivant pour calculer un correcteur à période d'échantillonnage variable.

Chapitre 2

Mise en forme d'un système dépendant de la période d'échantillonnage

L'utilisation de correcteur discret à période d'échantillonnage variable peut être intéressante dans plusieurs cas. Par exemple lorsque la mesure, bien qu'envoyée de façon périodique, est reçue à intervalle variable : c'est le cas en milieu marin lorsque la mesure d'altitude est effectuée avec un capteur à ultrason (la durée du trajet du signal dans l'eau dépend de la distance par rapport au fond sous-marin). Le délai variable entre deux réceptions de mesures, correspondant à une distance variable par rapport au fond, peut être vu comme une variation de période d'échantillonnage pour le contrôleur. En effet, il est inutile de recalculer la commande à appliquer au système tant que le mesure n'est pas arrivée, ce qui induit un instant de calcul de la commande variable donc une période d'échantillonnage variable.

Nous pouvons également citer le travail de Robert et al. (2010) concernant la commande en boucle fermée de l'ordonnancement d'un ensemble de tâche exécutée par un processeur temps-réel. Dans ce cas, il peut être intéressant d'augmenter la période d'échantillonnage d'une tâche de contrôle afin de réduire son utilisation des ressources informatiques, en particulier si le processeur arrive en limite de ses capacités.

L'approche développée ici poursuit les travaux de thèse de Robert (2007), qui a développé une méthode permettant d'obtenir, à partir d'une représentation d'état continue d'un système dynamique, une représentation d'état à temps discret à paramètres variants, sous forme polytopique (ou sous la forme d'une LFR), fonction de la période d'échantillonnage.

Or plusieurs travaux ont montré l'intérêt des systèmes LPV, pour conserver certaines non-linéarités d'un système en tant que paramètre variant d'un modèle

linéaire, par exemple Bruzelius (2004), Gauthier (2007).

Nous proposons ici de combiner ces deux objectifs, en étendant les travaux de David Robert aux systèmes LPV, c'est à dire aux systèmes dont la représentation d'état continue dépend d'un ensemble de paramètres. Cette méthode sera ensuite appliquée au cas du contrôle de l'altitude d'un véhicule sous-marin autonome. Plusieurs travaux ont été mené dans ce sens : dans Roche et al. (2010) la synthèse d'un contrôleur à période d'échantillonnage variable pour un système continu LTI, utilisant l'approche LFR, est mise en œuvre pour le contrôle d'altitude d'un AUV. Dans Roche et al. (2011) ; une structure cascade est mise en place, tenant compte du lien géométrique entre l'altitude et l'angle de tangage ; l'approche polytopique est utilisée pour synthétiser un contrôleur d'altitude séquencé par la période d'échantillonnage (alors que le régulateur de l'angle de tangage est à période d'échantillonnage constante). Dans Roche et al. (2012) la synthèse de contrôleur à période d'échantillonnage variable est étendue au cas LPV puis appliquée au contrôle de l'angle de tangage d'un AUV.

Le chapitre se présente de la façon suivante : la section 2.1 présente l'état de l'art concernant la discrétisation des systèmes sous forme LPV, et l'utilisation de ces techniques à la modélisation de système en vue de la synthèse d'un contrôleur. Puis la section 2.2 rappelle la technique de discrétisation d'un système LTI, et montre comment conserver la période d'échantillonnage en tant que paramètre variant dans les matrices d'état du modèle discret. La section 2.3 explique comment transformer le modèle paramétré obtenu en un système polytopique, avec la période d'échantillonnage comme unique paramètre variant. Puis la section 2.4 discute de la possibilité d'étendre cette méthode aux systèmes dont la représentation d'état continue est déjà sous forme LPV. Finalement les sections 2.5 et 2.6 montrent comment obtenir un modèle LFR avec la période d'échantillonnage comme seul paramètre variant à partir d'une système LTI (section 2.5), puis ces résultats sont étendus aux systèmes LPV (section 2.6).

2.1 État de l'art sur la discrétisation des systèmes LPV

Bien qu'assez peu étudiée, la discrétisation des systèmes Linéaires à Paramètres Variants (LPV) a fait l'objet de quelques études, en particulier Apkarian (1997), Tóth (2010), Tóth et al. (2010), Tóth et al. (2009).

Apkarian (1997) est le tout premier article à s'intéresser au problème de la discrétisation d'un système sous forme LPV générale (il n'est pas nécessaire de se placer dans le cas polytopique ou LFR). Cet article discute de la discrétisation d'un contrôleur LPV obtenu après une synthèse en continu. L'approche complète décrit

l'évolution de l'état discret par l'intégrale de la fonction représentant l'évolution de l'état continu, sur une période d'échantillonnage, mais le calcul en ligne de cette exponentielle est souvent trop coûteux pour être implémenté en temps réel. C'est pourquoi l'approximation de l'exponentielle par la méthode trapézoïdale est développée. De plus un système de "rafraichissement" est proposé pour ne mettre à jour les matrices du contrôleur LPV avec la nouvelle valeur du vecteur de paramètre que pour un multiple de la période d'échantillonnage (et non à tous les échantillons) et ainsi faire des économies en termes de ressources de calcul. L'ensemble de la méthode est testée sur un robot à deux segment et montre son efficacité.

Dans Tóth et al. (2010) le système LPV est présenté sous une formulation générale, et plusieurs approches de discrétisation sont comparées. L'approche complète décrit l'évolution de l'état par l'intégrale de la fonction représentant la variation de l'état, sur une période d'échantillonnage, puis plusieurs approximations de cette intégrale sont présentées : rectangulaire, polynomiale, trapézoïdale... Les différentes approches sont ensuite comparées, en termes d'erreur de discrétisation et de vitesse de convergence.

Toutefois dans notre approche, on cherche bien sur à obtenir un modèle discret le plus juste possible, mais surtout dont la modélisation permet d'appliquer les outils de synthèse de contrôleur LPV. En particulier, il est primordial de pouvoir transformer le système discrétisé, soit sous forme polytopique, soit sous la forme d'une LFR ; il faut donc que l'étape de discrétisation aboutisse à un modèle dépendant de façon assez simple de la période d'échantillonnage (affine pour le cas polytopique, sous forme de variation additive ou multiplicative dans le cas d'une LFR). Notre approche est donc différente et ne peut pas utiliser les résultats cités comme point de départ.

Il est aussi important de noter que nos travaux poursuivent ceux initiés antérieurement lors de la thèse de D/ Robert (Robert et al., 2006; Robert, 2007; Robert et al., 2007)

2.2 Discrétisation paramétrée en fonction de la période d'échantillonnage

Cette section, issue des résultats présentés dans Robert (2007), décrit comment obtenir une représentation d'état d'un système discret dépendant de la période d'échantillonnage à partie d'une représentation d'état continue d'un système LTI.

A partir de la représentation d'état d'un système linéaire à temps continu :

$$\mathcal{G} : \begin{cases} \dot{x}(t) = Ax(t) + Bw(t) \\ z(t) = Cx(t) + Dw(t) \end{cases} \tag{2.1}$$

la discrétisation exacte de ce système précédé d'un bloqueur d'ordre zéro permet d'aboutir à un système discret de la forme Åström and Wittenmark (1997) :

$$\hat{\mathcal{G}} : \begin{cases} x_{k+1} = \hat{A}x_k + \hat{B}w_k \\ z_k = \hat{C}x_k + \hat{D}w_k \end{cases} \tag{2.2}$$

avec les matrices $\hat{A}$, $\hat{B}$, $\hat{C}$ et $\hat{D}$ définies comme dans (2.3), h étant la période d'échantillonnage.

$$\begin{aligned} \hat{A}_h = e^{Ah} \quad \hat{B}_h = \int_0^h e^{A\tau} d\tau B \\ \hat{C} = C \qquad \hat{D} = D \end{aligned} \tag{2.3}$$

Dans le cas où la matrice A du système continu est inversible, $\hat{B}_h$ peut aussi s'exprimer de la façon suivante :

$$\hat{B}_h = A^{-1}(\hat{A}_h - I)B \tag{2.4}$$

La méthode numérique habituelle pour calculer les matrices de la représentation d'état à temps discret utilise l'exponentielle de la matrice M :

$$M = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} \tag{2.5}$$

de la manière suivante :

$$\begin{pmatrix} \hat{A}_h & \hat{B}_h \\ 0 & I \end{pmatrix} = e^{Mh} = exp\left(\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} h\right) \tag{2.6}$$

Un modèle affine en la période d'échantillonnage peut être obtenu en utilisant un développement en série de Taylor de l'exponentielle (2.6) à l'ordre l, ce qui conduit à :

$$\hat{A}_h = I + \sum_{i=1}^{l} \frac{A^i}{i!} h^i + o(h^{l+1}) \tag{2.7}$$

$$\hat{B}_h = \sum_{i=1}^{l} \frac{A^{i-1}B}{i!} h^i + o(h^{l+1}) \tag{2.8}$$

Nous considérons dans ce qui suit que la période d'échantillonnage évolue entre h_{min} et h_{max} avec $h_{min} > 0$. Or le développement en série de Taylor de l'équation (2.7) et (2.8) est situé au voisinage de $h = 0$. Afin de prendre en compte les

variations de la période autour d'une période nominale, elle est décomposée en une partie constante h_0 et une partie variable δ tel que :

$$h = h_0 + \delta \qquad \text{avec} \qquad h_{min} - h_0 \leq \delta \leq h_{max} - h_0 \tag{2.9}$$

Dans ce cas l'équation (2.6) devient :

$$\begin{pmatrix} \hat{A}_h & \hat{B}_h \\ 0 & I \end{pmatrix} = \begin{pmatrix} \hat{A}_{h_0} & \hat{B}_{h_0} \\ 0 & I \end{pmatrix} \begin{pmatrix} \hat{A}_\delta & \hat{B}_\delta \\ 0 & I \end{pmatrix} \tag{2.10}$$

$$= exp\left(\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} h_0\right) exp\left(\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} \delta\right) \tag{2.11}$$

Finalement les matrices de la représentation d'état discrète du système (2.2) s'écrivent comme suit, avec $\hat{A}_\delta$, $\hat{B}_\delta$ les matrices variables en fonction de δ et $\hat{A}_{h_0}$, $\hat{B}_{h_0}$ constantes (puisque dépendant de la partie constante de la période d'échantillonnage).

$$\begin{aligned} \hat{A}_h &= \hat{A}_{h_0}\hat{A}_\delta \\ \hat{B}_h &= \hat{B}_{h_0} + \hat{A}_{h_0}\hat{B}_\delta \end{aligned} \tag{2.12}$$

Le paramètre δ étant considéré petit, nous pouvons maintenant remplacer $\hat{A}_\delta$ et $\hat{B}_\delta$ par les développements en série de Taylor des équations (2.7), (2.8) (en δ). On obtient alors les matrices de la représentation d'état discrète du système, parametrées par la variation de période d'échantillonnage δ autour de la valeur nominale h_0.

$$\begin{aligned} \hat{A}_h &= \hat{A}_{h_0}\left(I + \sum_{i=1}^{l} \frac{A^i}{i!}\,\delta^i\right) + o(\delta^{l+1}) \\ \hat{B}_h &= \hat{B}_{h_0} + \hat{A}_{h_0}\left(\sum_{i=1}^{l} \frac{A^{i-1}B}{i!}\,\delta^i\right) + o(\delta^{l+1}) \end{aligned} \tag{2.13}$$

Le système discret obtenu est affine par rapport aux puissances successives de δ. δ étant borné, nous pourrons construire aisément un modèle polytopique comme présenté dans la prochaine section.

2.3 Mise sous forme polytopique discrète du système

La méthodologie complète d'obtention d'un modèle polytopique avec la période d'échantillonnage comme paramètre variant est décrite dans Robert (2007); Robert et al. (2010). Dans cette section, seul un modèle polytopique de complexité réduite, en considérant la dépendance entre les paramètres (ici les puissances successives de δ), est présenté.

2.3.1 Représentation discrète polytopique en fonction de δ

Le modèle affine obtenu dans la section précédente (2.13) dépend du vecteur de paramètre $\rho = [\delta, \delta^2, \ldots, \delta^l]$. Chaque élément de ce vecteur est borné (car la période d'échantillonnage $h = h_0 + \delta$ est bornée) donc ρ évolue à l'intérieur d'un polytope à 2^l sommets. Les sommets sont définis par un vecteur $[\nu_1, \ldots, \nu_N]$, avec $N = 2^l$ où chaque élément ν_j est la borne minimale ou maximale de δ^j.

Un modèle polytopique classique, ne tenant pas compte de la dépendance entre les éléments du vecteur de paramètres, serait donc formé de 2^l sommets. Or le vecteur de paramètres étant formé des puissances successives d'une même variable δ, l'ensemble des valeurs possibles pour ρ est très réduit : c'est une parabole dans le cas $l = 2$, qui serait représenté par un polytope à $2^2 = 4$ sommets (rectangle).

En gardant la structure polytopique qui est nécessaire pour l'application de la méthode, il est toutefois possible de réduire la taille du polytope, comme le montre la figure 2.1 pour $l = 2$ et $l = 3$ (voir Robert (2007) et Robert et al. (2010) pour plus de détails).

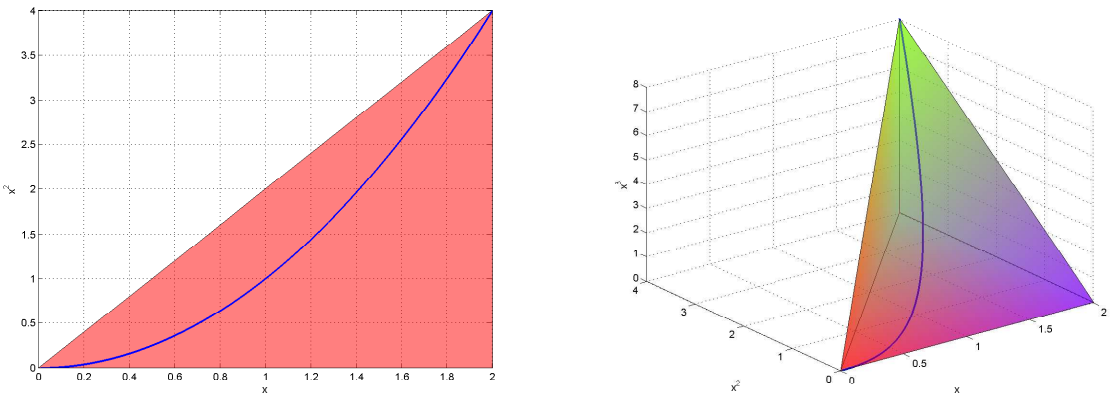

FIGURE 2.1 – Réduction du polytope pour $l = 2$ et $l = 3$

En effet, selon la forme particulière du vecteur de paramètre ρ (puissances successives de δ), une partie des vecteurs de paramètres sommets sont inatteignables. Ne sont conservés que les sommets définis par un vecteur $[\nu_1, \ldots, \nu_{l+1}]$, $\nu_i = 0$ ou δ^i_{max}, qui vérifient :

$$\nu_{n+1} \leq \delta_{max} \nu_n \tag{2.14}$$

Par exemple pour un développement en série de Taylor à l'ordre $l = 3$ (cf

equation 2.13), les sommets admissibles sont :

$$\begin{aligned}
&(0,\ 0,\ 0)\\
&(\delta_{max},\ 0,\ 0)\\
&(\delta_{max},\ \delta^2_{max},\ 0)\\
&(\delta_{max},\ \delta^2_{max},\ \delta^3_{max})
\end{aligned} \tag{2.15}$$

La réduction du nombre de sommets du polytope, qui passe de 2^l à $l+1$, permet ainsi de diminuer le conservatisme et la complexité de l'approche polytopique.

Il faut noter que cette réduction n'est valable que pour $\delta_{min} = 0$, c'est à dire en choisissant comme période nominale la borne inférieure de l'intervalle de variation de la période $h_{min} = h_0$.

On obtient finalement un modèle polytopique défini de la façon suivante :

$$x_{k+1}(t) = A(\rho)x_k + B(\rho)w_k \tag{2.16}$$

$$z_k = C(\rho)x_k + D(\rho)w_k \tag{2.17}$$

$$\tag{2.18}$$

avec $\rho = [\delta,\ \delta^2,\ \ldots]$

2.3.2 Mise en forme polytopique discrète pour la commande à intervalle variable

La méthode de synthèse de contrôleur LPV polytopique est basée sur la commande $\mathcal{H}_\infty$, les étapes à suivre sont donc les mêmes (celles définies dans la section 1.1.2).

Le première étape consiste à définir les performances souhaitées en boucle fermée grâce à des fonctions de pondérations fréquentielles. Dans le cas du contrôle à période d'échantillonnage variable, ces fonctions peuvent être choisies dépendantes de l'inverse de la période d'échantillonnage ($f = \frac{1}{h}$). En effet il a été montré dans Robert et al. (2010) qu'un fonction de pondération dépendante de h permet de gérer la dégradation des performances de la boucle fermée, lorsque la période d'échantillonnage augmente. La stabilité est ainsi préservée, au prix d'une dégradation (maîtrisée) des performances.

Le cas du problème de sensibilité mixte présenté dans la section 1.1.2.3 est étudié dans la suite de ce manuscrit. On choisit de définir une fonction de pondération sur l'erreur de poursuite fonction de l'inverse de la période d'échantillonnage f suivant :

$$W_e(h) = \frac{\frac{s}{M_{S_e}} + w_{S_e}(f)}{s + w_{S_e}(f)\epsilon_{S_e}} \tag{2.19}$$

où M_{S_e}, w_{S_e} et ϵ_{S_e} permettent de spécifier les performances désirées en boucle fermée.

Cette pondération est réécrite sous la forme canonique commandable, de la façon suivante :

$$\begin{cases} \dot{x} = (a \times f)x + (a \times f - b \times f)u \\ y = x + u \end{cases} \tag{2.20}$$

Elle est ensuite discrétisée, suivant les équations suivantes :

$$W_e(z) : \begin{cases} x_{k+1} = \hat{A}_{W_e}\, x_k + \hat{B}_{W_e}\, u_k \\ y_k = x_k + u_k \end{cases} \tag{2.21}$$

$$\begin{cases} \hat{A}_{W_e} = e^{af\,h} = e^{a} \\ \hat{B}_{W_e} = (af)^{-1}(\hat{A}_{W_e} - I)bf = a^{-1}(\hat{A}_{W_e} - I)b \end{cases} \tag{2.22}$$

A la suite de la discrétisation, la fonction de pondération ne dépend plus explicitement de la fréquence d'échantillonnage, suite à une simplification avec la période. Cette simplification permet de connecter directement la pondération à la représentation LPV polytopique du système à contrôler (dans le cas contraire, il faudrait prêter attention aux produits de paramètres éventuels) pour aboutir au modèle augmenté $P(z, \delta)$ de la figure 2.2.

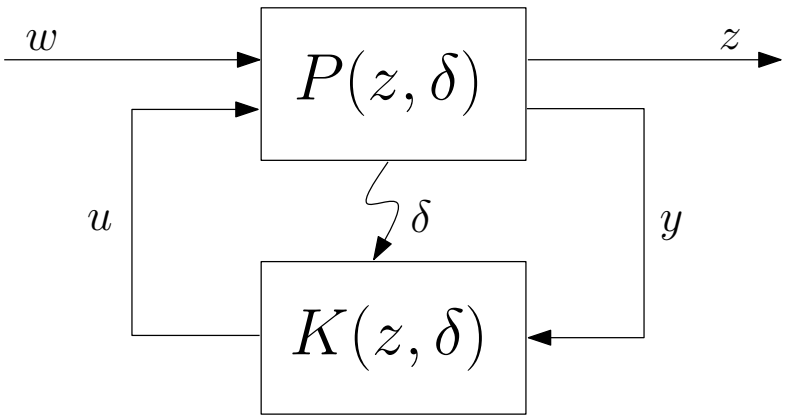

FIGURE 2.2 – Modèle augmenté et contrôleur séquencé par la variation de période d'échantillonnage

A partir de ce modèle polytopique ($P(z, \delta)$) ayant comme paramètre variant la variation de période d'échantillonnage δ, la méthodologie présentée paragraphe 1.3.2 peut être directement appliquée pour la synthèse d'un contrôleur polytopique.

2.4 Extension de l'approche polytopique aux systèmes LPV

La méthode présentée dans le paragraphe précédent s'applique pour des systèmes stationnaires. Ce paragraphe discute de son extension pour des systèmes LPV, c'est à dire dont la représentation d'état continue dépendant déjà d'un ou plusieurs paramètres, est donnée par :

$$G(s,\rho)\begin{cases} \dot{x}(t) = A(\rho(.))x(t) + B(\rho(.))w(t) \\ z(t) = C(\rho(.))x(t) + D(\rho(.))w(t) \end{cases} \tag{2.23}$$

Comme pour le cas LTI, la période d'échantillonnage doit être intégrée au système durant la phase de discrétisation. Mais il s'agit ici de discrétiser un système LPV dont les matrices de la représentation d'état dépendent d'un vecteur de paramètres ρ. Pour pouvoir ajouter la période d'échantillonnage à l'ensemble de paramètres déjà existant, on supposera que le système LPV dépend de façon affine du vecteur de paramètres.

L'équation (2.3) dans le cas LPV devient donc :

$$\begin{aligned} \hat{A} &= e^{A(\rho)h} \quad \hat{B} = \int_0^h e^{A(\rho)\tau} d\tau B(\rho) \\ \hat{C} &= C(\rho) \quad \hat{D} = D(\rho) \end{aligned} \tag{2.24}$$

Remarque : Les paramètre ρ et h sont à normalement définis à chaque instant d'échantillonnage et devrait donc être notés ρ_k et h_k. Ils sont notés ici sans indices par abus de notation pour simplifier l'écriture.

Le produit de la matrice $A(\rho)$ du système continu LPV avec la période d'échantillonnage h, que l'on veut garder comme paramètre, rend le système non affine par rapport au nouveau vecteur de paramètres $[\rho;\ h]$, ce qui pose problème dans le contexte de la synthèse par approche polytopique. De plus l'approximation en série de Taylor qui permet d'avoir un système affine par rapport à h dans le cas continu (equation 2.13) fait apparaître les puissances successives de A. Cela ne pose pas de problème dans le cas LTI (où cette matrice est constante) mais dans le cas LPV, comme la matrice A dépend du vecteur de paramètre $(A(\rho))$, des produits de paramètres vont apparaître. Une méthode simple pour contourner ce problème serait de définir un nouveau paramètre pour chaque produit, mais ceci augmenterait rapidement le nombre de paramètres à considérer. Or le nombre de paramètres est un aspect crucial de la modélisation et de la synthèse polytopique, comme expliqué au paragraphe 1.3.

Une autre approche, développée ici, consiste à faire la synthèse en continu sur le système polytopique dépendant de ρ puis de faire la discrétisation en ligne du contrôleur LPV continu obtenu.

Le calcul d'un contrôleur polytopique pour un système LPV, en suivant cette méthode, se décompose en 4 étapes présentées ci-dessous.

1- Spécification des performances

Des fonctions de pondération sont utilisées pour spécifier les performances désirées en boucle fermée (suivant la méthodologie de synthèse $\mathcal{H}_\infty$ présentée au paragraphe 1.1). Comme expliqué dans Robert (2007); Robert et al. (2010) pour une synthèse discrète, ces fonctions de pondérations peuvent être paramétrées par la période d'échantillonnage de façon à adapter les performances du contrôle à la valeur courante de la période.

Remarque : Cette adaptation est particulièrement intéressante dans le contexte temps-réel car elle permet de conserver la stabilité pratique en autorisant une dégradation maîtrisée des performances en boucle fermée en cas de sous-échantillonnage.

Dans notre cas, bien que la synthèse du contrôleur soit réalisée en temps continu, des fonctions de pondérations $W(\tilde{h})$ dépendantes d'un paramètre $\tilde{h}$ sont utilisées afin de permettre l'adaptation des performances en boucle fermée en fonction de $\tilde{h}$ (dans notre cas, $\tilde{h}$ sera ensuite choisi en cohérence avec la période d'échantillonnage du régulateur discrétisé). La bande passante de la fonction de pondération est choisie dépendante du paramètre $\tilde{h}$ ce qui permet d'adapter le temps de réponse en boucle fermée à la valeur mesurée de $\tilde{h}$. Par exemple pour un filtre du premier ordre, l'expression suivante est considérée pour l'erreur de poursuite :

$$W(\tilde{h}) = \frac{\frac{s}{M_s} + w_b\tilde{h}}{s + w_b\tilde{h}\epsilon} \tag{2.25}$$

avec M_s la marge de module désirée, $w_b\tilde{h}$ la bande passante souhaitée en boucle fermée, et ϵ l'erreur de poursuite maximale en régime permanent.

Cette fonction de pondération peut être réécrite sous forme de représentation d'état comme suit :

$$\begin{cases} \dot{x}_W = A_W(\tilde{h})x_W + B_W(\tilde{h})u_W \\ y_W = C_W x_W + D_W u_W \end{cases} \tag{2.26}$$

avec

$$A_W = -w_b\epsilon\tilde{h} \;\; B_W = w_b(1 - \frac{\epsilon}{M_s})\tilde{h} \tag{2.27}$$

$$C_W = 1 \;\; D_W = \frac{1}{M_s} \tag{2.28}$$

On obtient finalement une fonction de pondération sous forme de représentation d'état qui dépend du paramètre $\tilde{h}$ qui doit finalement être ajouté au vecteur de paramètres existant ρ dont dépend déjà le système LPV polytopique continu.

2- Connexion du système et des pondérations : obtention d'un modèle polytopique continu

En considérant que le paramètre $\tilde{h}$ est borné par $[\underline{\tilde{h}},\ \overline{\tilde{h}}]$, il est alors facile d'obtenir une représentation continue polytopique du système augmenté (défini par la connexion du système et des fonctions de pondération).

Il suffit d'ajouter le paramètre $\tilde{h}$ à l'ensemble des paramètres déjà existant dans le modèle polytopique de départ et de définir : $\rho' = [\rho, \tilde{h}]$. Puis chaque sommet du polytope de départ $\omega_i = [\nu_{i1}, \ldots, \nu_{in}]$ est utilisé pour créer 2 nouveaux sommets en ajoutant la borne minimale ou maximale de $\tilde{h}$:

$$\omega_i' = [\nu_{i1}, \ldots, \nu_{in}, \underline{\tilde{h}}] \ \text{ et } \ \omega_i'' = [\nu_{i1}, \ldots, \nu_{in}, \overline{\tilde{h}}] \tag{2.29}$$

Le nombre de paramètres passe alors de n à $N = 2n$.

Finalement, pour chaque sommet du nouveau polytope, la pondération est connectée au système, suivant le schéma de contrôle envisagé .

Par exemple, le problème de sensibilité mixte peut être traité (voir 1.1.2), avec une fonction de pondération sur l'erreur de poursuite dépendante du paramètre $\tilde{h}$, et une pondération constante sur la commande. Le schéma de contrôle est rappelé sur la figure 2.3.

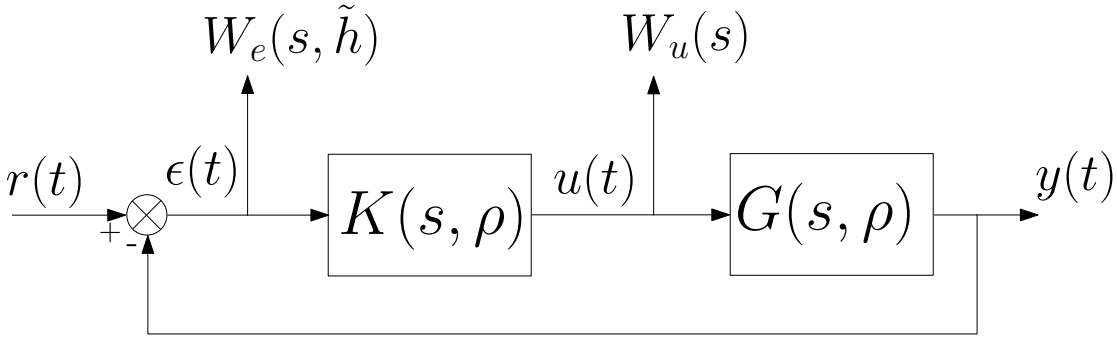

FIGURE 2.3 – Boucle fermée du problème de sensibilité mixte dans le cas LPV

La connexion des fonction de pondération et du système mène à la forme générale présentée sur la figure 2.4.

Pour chaque sommet ω_i' du nouveau polytope (défini par l'ensemble de paramètres $[\rho, \tilde{h}]$), le système augmenté $P_{\omega_i'}(s)$ est calculé à partir des pondérations et du système $G(s, \rho)$.

A la fin de cette étape, on obtient un ensemble de systèmes sommets $P_{\omega_i'}$ qui définit la nouvelle représentation d'état polytopique continue du système. Ce système est utilisé pour la synthèse du contrôleur lors de la prochaine étape.

3- Synthèse du contrôleur continu

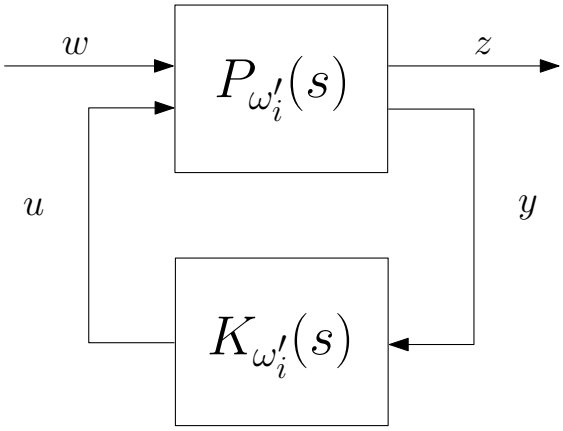

FIGURE 2.4 – Interconnexion du système et du contrôleur LPV

En utilisant le système obtenu à l'étape précédente, un contrôleur LPV polytopique est obtenu directement en appliquant la méthode présentée au chapitre précédent, section 1.3, pour les systèmes à temps continu. Le contrôleur dépend du même ensemble de paramètres que le système, c'est-à-dire $[\rho, \tilde{h}]$.

On obtient alors un ensemble de contrôleurs définis aux sommets du polytope ω_i'. Pour une valeur précise du vecteur de paramètres ρ, le contrôleur correspondant est calculé par la combinaison convexe des N contrôleurs sommets, suivant :

$$\begin{pmatrix} A_K(\rho,\tilde{h}) & B_K(\rho,\tilde{h}) \\ C_K(\rho,\tilde{h}) & D_K(\rho,\tilde{h}) \end{pmatrix} = \sum_{i=1}^{N'} \alpha_i(t) \begin{pmatrix} A_K(\omega_i') & B_K(\omega_i') \\ C_K(\omega_i') & D_K(\omega_i') \end{pmatrix} \tag{2.30}$$

$N = 2n$ étant le nombre de sommets du nouveau polytope.

Le contrôleur obtenu est défini en temps continu, et peut s'adapter à une variation de paramètres (à l'intérieur de leur intervalle de variation considéré pour la synthèse) en garantissant la stabilité et les performances souhaitées en boucle fermée.

4- Discrétisation (exacte ou approchée) du régulateur LPV

La dernière étape concerne la discrétisation du contrôleur obtenu. Parmi les méthodes décrites précédemment on utilise ici la méthode numérique classique introduite dans la section 2.2, c'est-à-dire utilisant l'exponentielle de la matrice :

$$M_K = \begin{pmatrix} A_K(\rho,\tilde{h}) & B_K(\rho,\tilde{h}) \\ 0 & 0 \end{pmatrix} \tag{2.31}$$

La représentation d'état du contrôleur discret est donc donnée par l'équation

suivante :

$$\mathcal{K}_d : \begin{cases} x_{K_{k+1}} = \hat{A}_K x_k + \hat{B}_K w_k \\ z_{K_k} = \hat{C}_K x_k + \hat{D}_K w_k \end{cases} \tag{2.32}$$

où les matrices $\hat{A}_K$ et $\hat{B}_K$ sont données par :

$$\begin{pmatrix} \hat{A}_K & \hat{B}_K \\ 0 & I \end{pmatrix} = e^{M_K \tilde{h}} = exp\left(\begin{pmatrix} A_K(\rho, \tilde{h}) & B_K(\rho, \tilde{h}) \\ 0 & 0 \end{pmatrix} \tilde{h}\right) \tag{2.33}$$

et $\hat{C}_K = C_K$ et $\hat{D}_K = D_K$.

La discrétisation du contrôleur LPV polytopique doit être effectuée en ligne, à la suite de la combinaison convexe permettant d'adapter le contrôleur au point de fonctionnement considéré.

Remarque : On rappelle que le paramètre $\tilde{h}$ est défini en continu et est choisi en cohérence avec la période d'échantillonnage.

Remarque : Dans le cas de systèmes embarqués, ou pour tout autre système où la puissance de calcul est un enjeu crucial, le calcul en ligne de l'exponentiel de matrices d'état, pour obtenir la representation discrète exacte, peut être une source supplémentaire de dépenses d'énergie non souhaitable. Dans ce cas, l'approximation en série de Taylor de l'exponentielle à l'ordre l peut réduire l'énergie nécessaire au calcul, et donc supprimer cet inconvenient. Un compromis doit être fait pour le choix de la valeur de l, entre la précision du résultat et le temps de calcul.

2.5 Mise sous forme LFR discrète

Une autre formulation permettant de garder la période d'échantillonnage comme paramètre dans le modèle est la représentation linéaire fractionnelle (LFR) présentée dans le chapitre 1.4.

L'objectif est de transformer le système discrétisé dépendant de la période d'échantillonnage obtenu section 2.2 sous la forme LFR présentée sur la figure 2.5. Deux approches différentes sont présentées, l'une exacte et l'autre approchée, tenant compte de la forme particulière dont le système discret dépend du paramètre (la variation δ de période d'échantillonnage). Là encore la méthodologie complète pour l'obtention d'une LFR est détaillée dans Robert (2007), on ne donne ici que les étapes importantes de la modélisation.

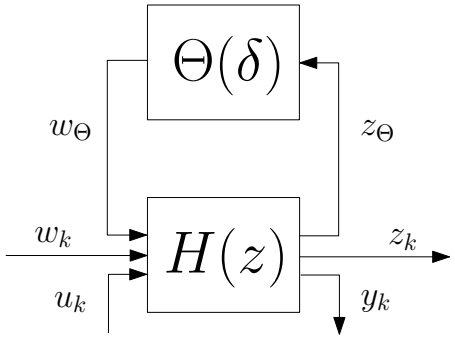

FIGURE 2.5 – Système sous forme LFR

2.5.1 Forme exacte de la LFR discrète considérant la période d'échantillonnage comme paramètre

La forme exacte des matrices de la représentation d'état discrète dépendant de la période d'échantillonnage est rappelée ici :

$$\hat{A}_h = \hat{A}_{h_0}\hat{A}_\delta$$
$$\hat{B}_h = \hat{B}_{h_0} + \hat{A}_{h_0}\hat{B}_\delta$$

A partir de ces expressions, deux incertitudes sont calculées, en considérant la variation des matrices $\hat{A}_h$ et $\hat{B}_h$ par rapport aux matrices définies à la période d'échantillonnage nominale h_0, suivant :

$$\hat{A}_h - \hat{A}_{h_0} = \hat{A}_{h_0}(\hat{A}_\delta - I) = \hat{A}_{h_0} \times \Delta_1(\delta)$$
$$\hat{B}_h - \hat{B}_{h_0} = \hat{A}_{h_0}\hat{B}_\delta = \Delta_2(\delta)$$

Ces deux incertitudes sont placées sur la diagonale de la matrice des paramètres Θ. Pour réécrire le système sous la forme LFR de la figure 2.5, on définit les vecteurs d'entrée et de sortie du bloc des paramètres suivants :

$$\begin{bmatrix} w_{\Theta_1} \\ w_{\Theta_2} \end{bmatrix} = \begin{pmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{pmatrix} \begin{bmatrix} z_{\Theta_1} \\ z_{\Theta_2} \end{bmatrix} \tag{2.34}$$

Finalement, à partir du système continu LTI de l'équation (2.1), la forme LFR de la figure 2.5, en considérant la période d'échantillonnage d'un système discrétisé comme paramètre variant, est définie de la façon suivante :

$$\left[\begin{array}{c} \dot{x} \\ \hline z_{\Theta_1} \\ z_{\Theta_2} \\ y \end{array}\right] = \begin{pmatrix} \hat{\mathcal{A}} & \hat{\mathcal{B}} \\ \hat{\mathcal{C}} & \hat{\mathcal{D}} \end{pmatrix} \left[\begin{array}{c} x \\ \hline w_{\Theta_1} \\ w_{\Theta_2} \\ u \end{array}\right] \tag{2.35}$$

où :

$$\begin{aligned}
\Theta(\delta) &= \begin{pmatrix} \Delta_1(\delta) & 0 \\ 0 & \Delta_2(\delta) \end{pmatrix} \\
\hat{\mathcal{A}} = \hat{A}_{h_0} \quad \hat{\mathcal{B}} &= \begin{pmatrix} \hat{A}_{h_0} & I & \hat{B}_{h_0} \end{pmatrix} \\
\hat{\mathcal{C}} = \begin{pmatrix} I \\ 0 \\ \hat{C} \end{pmatrix} \quad \hat{\mathcal{D}} &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & I \\ 0 & 0 & \hat{D} \end{pmatrix}
\end{aligned} \tag{2.36}$$

où $\hat{\mathcal{A}}$, $\hat{\mathcal{B}}$, $\hat{\mathcal{C}}$ et $\hat{\mathcal{D}}$ sont les matrices définissant la représentation d'état du système $P(z)$ de la figure 2.5.

2.5.2 Forme approchée de la LFR discrète considérant la période d'échantillonnage comme paramètre

Dans le cas précédent, la matrice des paramètres est de forme bloc diagonale, où $\Delta_1(\delta)$ et $\Delta_2(\delta)$ sont des matrices pleines. Or ces deux matrices ne dépendent que d'un seul paramètre variant, la variation de période d'échantillonnage δ. Ainsi nous allons chercher ici à représenter la matrice des paramètres sous la forme d'un scalaire répété $\Theta = \delta \times I_r$. En utilisant la nature particulière du vecteur de paramètres pour simplifier la forme de la matrice $\Theta(\rho)$ cela permet de réduire le conservatisme de l'approche.

La forme approchée consiste à réduire le conservatisme en utilisant les développement en série de Taylor des blocs Δ_1 et Δ_2, pour faire apparaître la dépendance en δ, suivant :

$$\Delta_1 = \hat{A}_\delta - I \approx \sum_{i=1}^{l} \frac{A^i}{i!}\ \delta^i \tag{2.37}$$

$$\Delta_2 = \hat{A}_{h_0}\hat{B}_\delta \approx \hat{A}_{h_0} \sum_{i=1}^{l} \frac{A^i B}{i!}\ \delta^i \tag{2.38}$$

En utilisant cette approximation, une nouvelle forme LFR approchée peut être définie par :

$$\begin{aligned}
\Theta &= \delta I_{2\times l\times n_s} \\
\hat{\mathcal{A}} = \hat{A}_{h_0} \quad \hat{\mathcal{B}} &= \begin{pmatrix} \hat{\mathcal{B}}_1 & \hat{\mathcal{B}}_2 & \hat{B}_{h_0} \end{pmatrix}
\end{aligned} \tag{2.39}$$

$$\hat{\mathcal{C}} = \begin{pmatrix} \hat{\mathcal{C}}_1 \\ \hat{\mathcal{C}}_2 \\ \hat{C}_d \end{pmatrix} \quad \hat{\mathcal{D}} = \begin{pmatrix} \bar{\mathcal{D}} & 0 & 0 \\ 0 & \bar{\mathcal{D}} & \hat{\mathcal{D}}_{2u} \\ 0 & 0 & \hat{D}_d \end{pmatrix} \tag{2.40}$$

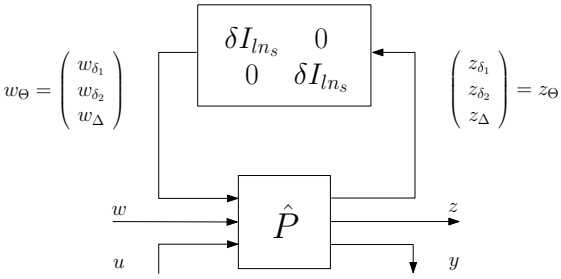

FIGURE 2.6 – Modèle LFR dépendant de la variation de période d'échantillonnage δ : forme approchée

avec n_s le nombre d'état et

$$\hat{\mathcal{B}}_1 = \begin{pmatrix} \hat{A}_{h_0} A & 0_{n_s 1} & \dots & \dots 0_{n_s(l-1)} \end{pmatrix} \quad \hat{\mathcal{B}}_2 = \begin{pmatrix} \hat{A}_{h_0} & 0_{n_s 1} & \dots & \dots & 0_{n_s(l-1)} \end{pmatrix} \tag{2.41}$$

$$\hat{\mathcal{C}}_1 = \begin{pmatrix} I_{n_s 1} \\ \vdots \\ \vdots \\ I_{n_s l} \end{pmatrix} \quad \hat{\mathcal{C}}_2 = \begin{pmatrix} 0_{n_s 1} \\ \vdots \\ \vdots \\ 0_{n_s l} \end{pmatrix} \quad \bar{\mathcal{D}} = \begin{pmatrix} 0_{n_s} & \frac{A}{2} & \dots & 0_{n_s} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \frac{A}{l} \\ 0_{n_s} & \dots & \dots & 0_{n_s} \end{pmatrix} \quad \hat{\mathcal{D}}_{2u} = \begin{pmatrix} B \\ \vdots \\ \vdots \\ B \end{pmatrix} \tag{2.42}$$

Cette deuxième formulation permet de réduire le conservatisme dû à la matrice d'incertitude $\Theta(\delta)$ maintenant de type diagonale, mais la taille de cette matrice est augmentée selon l'ordre du développement l (taille totale de la matrice $2 \times l \times n_s$). Il y a donc un compromis à trouver, entre la précision du développement en série de Taylor et la complexité.

2.6 Extension aux systèmes LPV

Comme pour le cas polytopique, on souhaite adapter la méthode existante pour les systèmes LTI aux systèmes LPV, déjà représentés sous forme LFR en temps continu.

Comme nous l'avons souligné auparavant, l'intérêt de la formulation d'un système sous forme LFR est la séparation de la partie variable dépendant des paramètres (Δ sur la figure 2.7) de la dynamique du système (notée P sur cette même figure).

Ainsi une LFR peut être discrétisée, en supposant que les entrées w_Δ et sorties z_Δ du bloc Δ sont échantillonnées et maintenues constantes entre deux pas d'échantillonnage (par un bloqueur d'ordre zéro). La méthodologie présentée précédemment pour le cas LTI peut donc être étendue aux systèmes LPV, afin d'obtenir

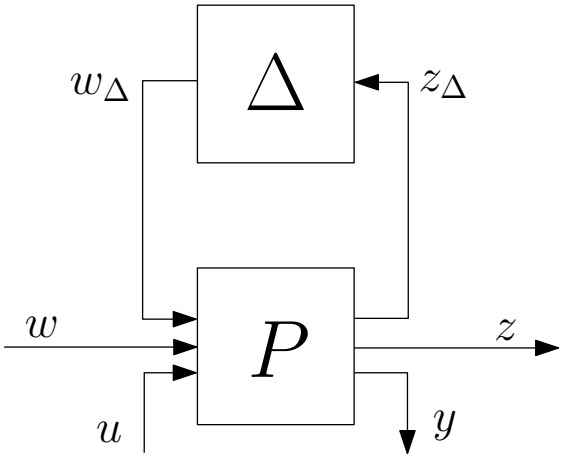

FIGURE 2.7 – Système sous forme LFR

un modèle LFR discret paramétré par l'ensemble de paramètres de départ et la période d'échantillonnage.

Le système sous forme LFR de la figure 2.7 est supposé décrit par les équations suivantes :

$$\begin{bmatrix} \dot{x} \\ \hline z_\Delta \\ z \\ y \end{bmatrix} = \left[\begin{array}{c|ccc} A & B_\Delta & B_1 & B_2 \\ \hline C_\Delta & D_{\Delta\Delta} & D_{\Delta 1} & D_{\Delta 2} \\ C_1 & D_{1\Delta} & D_{11} & D_{12} \\ C_2 & D_{2\Delta} & D_{21} & D_{22} \end{array}\right] \begin{bmatrix} x \\ \hline w_\Delta \\ w \\ u \end{bmatrix} \tag{2.43}$$

En regroupant les différentes matrices et les signaux d'entrée-sortie, on obtient une représentation d'état continue sous la forme classique :

$$\begin{aligned} \dot{x}(t) &= \mathcal{A}x(t) + \mathcal{B}\tilde{w}(t) \\ \tilde{z}(t) &= \mathcal{C}x(t) + \mathcal{D}\tilde{w}(t) \end{aligned} \tag{2.44}$$

avec

$$\begin{aligned} \mathcal{A} &= A & \mathcal{B} &= \begin{bmatrix} B_\Delta & B_1 & B_2 \end{bmatrix} \\ \mathcal{C} &= \begin{bmatrix} C_\Delta \\ C_1 \\ C_2 \end{bmatrix} & \mathcal{D} &= \begin{bmatrix} D_{\Delta\Delta} & D_{\Delta 1} & D_{\Delta 2} \\ D_{1\Delta} & D_{11} & D_{12} \\ D_{2\Delta} & D_{21} & D_{22} \end{bmatrix} \end{aligned} \tag{2.45}$$

Les entrées et sorties sont également groupées :

$$\tilde{w} = \begin{bmatrix} w_\Delta \\ w \\ u \end{bmatrix} \; et \; \tilde{z} = \begin{bmatrix} z_\Delta \\ z \\ y \end{bmatrix} \tag{2.46}$$

La méthode présentée dans la section précédente pour les systèmes LTI peut alors être appliquée sur le système représenté par l'équation 2.44. On obtient ainsi

un système discret paramétré par la variation de période d'échantillonnage δ (h_0 étant la partie nominale constante de la période : $h = h_0 + \delta$), suivant :

$$\hat{A}_h = \hat{\mathcal{A}}_{h_0}\hat{\mathcal{A}}_\delta \tag{2.47}$$

$$\hat{B}_h = \hat{\mathcal{B}}_{h_0} + \hat{\mathcal{A}}_{h_0}\hat{\mathcal{B}}_\delta \tag{2.48}$$

où $\hat{\mathcal{A}}_{h_0}$ et $\hat{\mathcal{B}}_{h_0}$ sont définit par l'équation 2.6 en h_0.

En appliquant le développement en série de Taylor de l'equation 2.7 en δ (suivant la section 2.5.2), on obtient la représentation d'état suivante pour le système discret dépendant des paramètres initiaux Δ et de la variation de la période d'échantillonnage δ :

$$\hat{P} \ : \ \begin{cases} x_{k+1} = \hat{\mathcal{A}}x_k + \hat{\mathcal{B}}\begin{bmatrix} w_{\delta_1} \\ w_{\delta_2} \\ w_\Delta \\ w \\ u \end{bmatrix}_k \\ \begin{bmatrix} z_{\delta_1} \\ z_{\delta_2} \\ z_\Delta \\ z \\ y \end{bmatrix}_k = \hat{\mathcal{C}}x_k + \hat{\mathcal{D}}\begin{bmatrix} w_{\delta_1} \\ w_{\delta_2} \\ w_\Delta \\ w \\ u \end{bmatrix}_k \end{cases} \tag{2.49}$$

avec

$$\Theta = \begin{pmatrix} \delta I_{2\times l\times n_s} & 0 \\ 0 & \Delta \end{pmatrix} \tag{2.50}$$

$$\hat{\mathcal{A}} = \hat{\mathcal{A}}_{h_0} \quad \hat{\mathcal{B}} = \begin{pmatrix} \hat{\mathcal{B}}_1 & \hat{\mathcal{B}}_2 & \hat{\mathcal{B}}_{h_0} \end{pmatrix} \tag{2.51}$$

$$\hat{\mathcal{C}} = \begin{pmatrix} \hat{\mathcal{C}}_1 \\ \hat{\mathcal{C}}_2 \\ \hat{\mathcal{C}} \end{pmatrix} \quad \mathcal{D} = \begin{pmatrix} \hat{\hat{D}} & 0 & 0 \\ 0 & \hat{\hat{D}} & \hat{\mathcal{D}}_{2u} \\ 0 & 0 & \hat{\mathcal{D}} \end{pmatrix} \tag{2.52}$$

où les matrices $\hat{\mathcal{B}}_1$, $\hat{\mathcal{B}}_2$, $\hat{\mathcal{C}}_1$, $\hat{\mathcal{C}}_1$ $\hat{\hat{D}}$ et $\hat{\mathcal{D}}_{2u}$ sont définies par les équations (2.41) et (2.42)

Le schéma de la figure 2.8 présente la forme LFR finalement obtenue.

On obtient bien une forme LFR du système LPV discrétisé, paramétrée par l'ensemble de paramètres de départ Δ, ainsi que par la variation de période d'échantillonnage δ.

La méthode de synthèse de contôleur LFR présenté dans la section 1.4 peut alors être directement appliquée sur le système $\hat{P}$

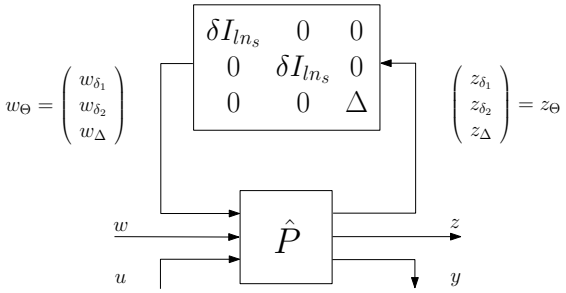

FIGURE 2.8 – Modèle LFR dépendant des paramètres initiaux Δ et de la variation de période d'échantillonnage δ

2.6.0.1 Conclusion

Dans ce chapitre, l'ensemble des outils nécessaires à la synthèse de contrôleurs discrets dépendants de la période d'échantillonnage ont été présentés.

Dans un premier temps, nous avons présenté comment obtenir, à partir d'une représentation d'état d'un système LTI continu, un système discret dont la représentation d'état dépend de la période d'échantillonnage.

Ce système a ensuite été représenté, soit sous une forme polytopique, soit sous une forme LFR, de façon a pouvoir utiliser les outils de synthèse de contrôleur LPV présentés dans le chapitre 1.

La méthode a ensuite été étendue pour des systèmes LPV continus, représentant une contribution originale de ce travail. Dans le cas polytopique, l'application directe de la méthode n'est pas possible ; nous avons donc proposé une approche différente, en synthétisant le contrôleur en continu, puis en faisant la discrétisation en ligne.

Dans le cas LFR, grâce à la séparation de la partie variable du système, dépendante des paramètres, du reste de la dynamique, la méthode peut être directement appliquée, en ajoutant la variation de période d'échantillonnage à l'ensemble de paramètres déjà existant.

Chapitre 3

Véhicule sous-marin autonome : modélisation pour la commande

Le véhicule sous-marin autonome (en anglais, Autonomous Underwater Vehicle : AUV) considéré comme application dans ce manuscrit est l'$Aster^X$ (présenté sur la figure 3.1) appartenant à l'IFREMER : Institut français de recherche pour l'exploitation de la mer (voir leur site web : *http ://www.ifremer.fr/institut*). L'$Aster^X$ est un véhicule sous-marin de type torpille, de $4,5$ mètres de long et de 793 kg, d'un autonomie de 11 heures en mission. il est équipé de plusieurs instruments de mesures (Centrale inertielle, loch Doppler..) qui permettent en théorie la connaissance de l'ensemble de l'état (six vitesses et six positions, comme il sera détaillé par la suite).

Le type de mission considéré dans cette étude est le suivi de fond sous-marin, en vue de sa cartographie par exemple. Pour cela, le véhicule doit maintenir une altitude constante par rapport au fond, ainsi qu'une vitesse d'avance constante.

L'analyse détaillée de la modélisation d'un sous-marin du même type que celui étudié est un problème classique (Fossen, 1994; Santos, 1995). La modélisation du sous-marin ayant pour but une implémentation en vue de simulation, on ne cherchera pas seulement une équation donnant le comportement global du sous-marin, mais également une description de chacun des comportements (dynamique, hydrodynamique...) pris séparément, permettant d'aboutir à un modèle d'état non-linéaire du véhicule. On fait alors l'hypothèse que le comportement global est la résultante de chacune des composantes prises séparément.

La suite du chapitre se présente comme suit : la section 3.1 présente le modèle non-linéaire du sous-marin, en donnant l'équation globale régissant son mouvement, puis en détaillant les différentes actions dues à chacunes des forces s'exerçant sur le véhicule. Le modèle du sous marin étant de taille importante et présentant de nombreux couplages, la section 3.2 présente comment diviser le modèle global en plusieurs sous-modèles, un pour l'avance suivant l'axe Ox, obtenu par linéarisa-

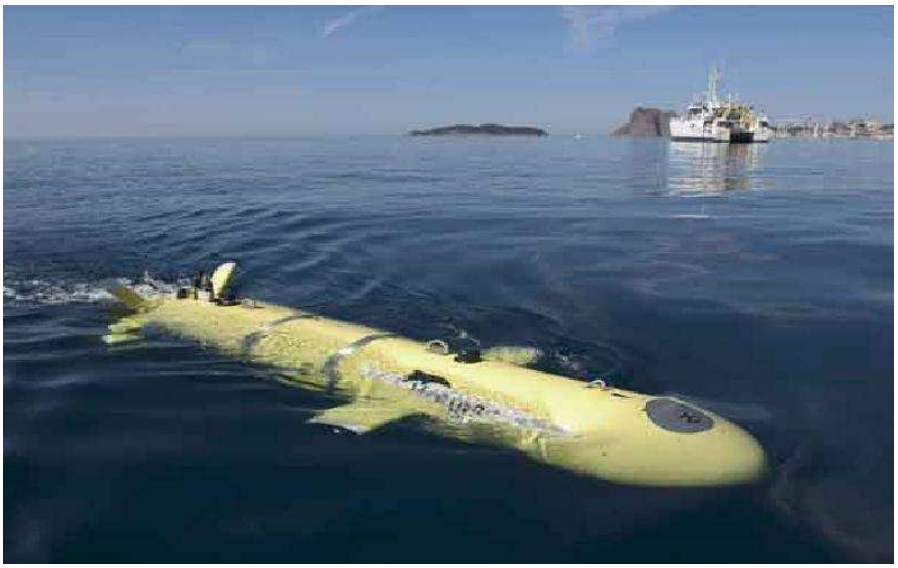

FIGURE 3.1 – Véhicule Aster^X

tion tangentielle du modèle non-linéaire, et deux pour chacun des plans horizontal et vertical. Pour le contrôle de l'altitude, le lien fort entre l'altitude et l'angle de tangage est exploité dans la section 3.3, où une structure cascade est développée. Finalement la section 3.4 présente 2 modèles LPV du sous marin, l'un utilisant la formulation polytopique, le second l'approche LFR, afin de conserver une non-linéarité importante dans le modèle.

3.1 Modèle non linéaire

Pour la modélisation du sous-marin, deux repères sont à considérer :

- le repère lié au véhicule : $\Re(C, X, Y, Z)$ dont l'origine C est le centre de carène (centre du volume où s'applique la poussée d'Archimède) peut être différente du centre de gravité de véhicule.
- le repère inertiel : $\Re_0(O, X_0, Y_0, Z_0)$ qui dans le cas d'un sous-marin se déplaçant à faible vitesse peut être pris comme étant lié à la Terre.

Pour décrire le comportement du véhicule, on utilisera un vecteur d'état à 12 variables $\begin{bmatrix} \eta \\ \nu \end{bmatrix}$.

- $\eta \in \mathbb{R}^6$ représente la situation du véhicule exprimée dans le référentiel inertiel $\Re_0$: $\eta = \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix}$ avec
  - η_1 la position du véhicule : $\eta_1 = \begin{bmatrix} x & y & z \end{bmatrix}^T$ (à noter que suivant l'usage

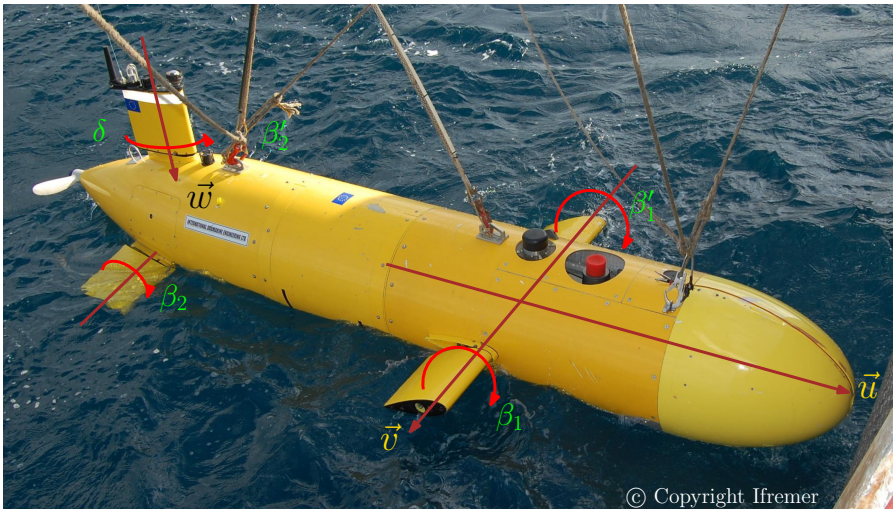

FIGURE 3.2 – Axes et actionneurs

dans la communauté sous-marine, l'axe Oz est dirigé vers le bas)

– η_2 l'orientation du véhicule : $\eta_2 = \begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^T$

Les angles ϕ, θ et ψ représentent respectivement le roulis, le tangage et le lacet du véhicule, comme le montre la figure suivante :

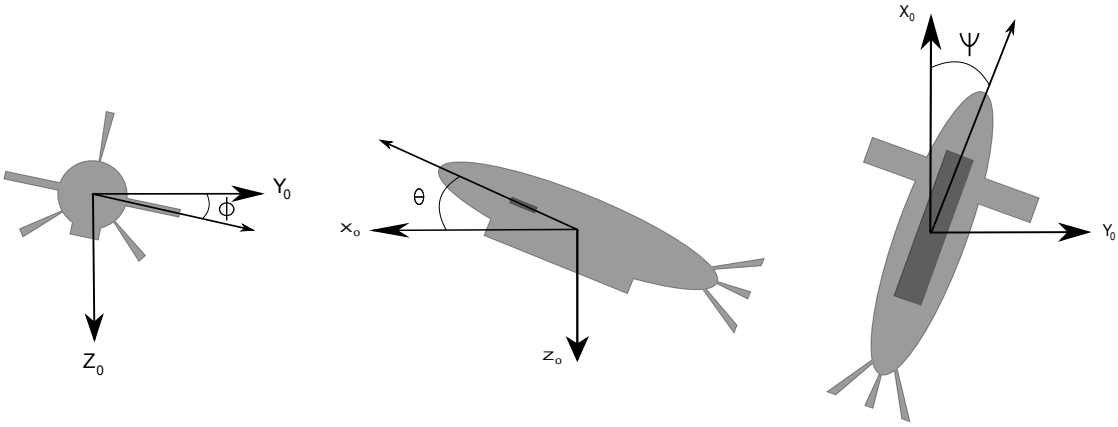

FIGURE 3.3 – Définition des angles ϕ (roulis), θ (tangage), et Ψ (lacet)

L'orientation du sous-marin est définie par une rotation d'un angle ϕ autour de l'axe X_0, puis une rotation d'un angle θ autour de l'axe Y_0 et finalement une rotation d'angle ψ autour de l'axe Z_0.

– $\nu \in \mathbb{R}^6$ représente le torseur cinématique : $\nu = \begin{bmatrix}\nu_1 \\ \nu_2\end{bmatrix}$ exprimé dans le repère local du véhicule $\Re$. Les variables ν_1 et ν_2 sont respectivement les vitesses

linéaires et les vitesses angulaires (dérivées premières des vecteurs η_1 et η_2 au changement de repère près) et sont notées : $\nu_1 = \begin{bmatrix} u & v & w \end{bmatrix}^T$ et $\nu_2 = \begin{bmatrix} p & q & r \end{bmatrix}^T$

Remarque : La paramétrisation utilisée (roulis-tangage-lacet) a l'inconvenient de présenter une singularité pour un angle de tangage θ égal à 90°. Cette singularité n'a pas d'incidence sur le modèle du sous-marin, car cette configuration ($\theta = 90^\circ$) n'est pas atteignable avec ce type de véhicule.

3.1.1 Equations globales

L'étude complète permet d'aboutir au système dynamique suivant (extraite par exemple de Fossen (1994); Santos (1995); Jalving and Storkersen (1994)) :

$$M\dot{\nu} = G(\nu)\nu + D(\nu)\nu + \Gamma_g + \Gamma_u \tag{3.1}$$

$$\dot{\eta} = J_c(\eta_2)\nu \tag{3.2}$$

où les différents paramètres correspondent à :

- M est la matrice d'inertie. Elle est composée de 2 termes : M_d, la matrice d'inertie due à la masse du véhicule et M_a, la matrice d'inertie d'eau ajoutée, dont les paramètres sont issus de modèles semi-empiriques et sont connus avec une faible imprécision,
- $G(\nu)$ représente l'action des forces de Coriolis et des forces centrifuges. Il est également composé de deux termes : G_d dû aux forces dynamiques, et G_a dû aux forces hydrodynamiques,
- $D(\nu)$ est la matrice des coefficients d'amortissement hydrodynamique (également issue d'estimation et d'identification),
- Γ_g est le vecteur des force de gravité et de la poussée d'Archimède,
- $J_c(\eta_2)$ est la matrice de changement de référentiel, pour passer du référentiel lié au véhicule $\mathcal{R}(C, X, Y, Z)$ au référentiel inertiel $\mathcal{R}_0(O, X_0Y_0Z_0)$,
- Γ_u décrit les forces et moments générés par les actionneurs de l'AUV. Le véhicule considéré dans cette thèse (voir Figure 3.1) possède un propulseur à hélice pour le déplacement suivant l'axe Ox (poussée Q_c) ainsi que 5 ailerons indépendants :
  - 2 ailerons horizontaux à l'avant du véhicule dont l'action est représentée par les angles β_1 et β_1'.
  - 1 aileron vertical à la queue du véhicule dont l'action est représentée par un angle δ.
  - 2 ailerons inclinés à la queue du véhicule, dont l'action est représentée par des angles β_2 et β_2' (ailerons inclinés de $\pm\pi/3$ par rapport à l'aileron vertical.

Remarque : La modélisation des gouvernes ainsi que des efforts hydrodynamiques n'est valable que pour un angle d'incidence (différence entre l'axe du véhicule et la vitesse du fluide par rapport à l'aile, qui représente l'inclinaison du sous-marin dans le fluide en mouvement) inférieur à 25 degrés environ. Au delà la force de portance s'écroule, il y a "décrochag" et les modèles utilisés ne sont plus valides.

3.1.2 Détails des équations régissant le mouvement du sous-marin

Pour décrire le comportement du système, on sépare l'analyse en deux parties : la cinématique (aspects géométriques du mouvement) et la dynamique (effets des forces et des moments sur le mouvement).

3.1.2.1 Etude cinématique

La trajectoire du véhicule est donnée par la relation suivante :

$$\dot{\eta}_1 = J_{c_1}(\eta_2)\nu_1 \tag{3.3}$$

où $J_{c_1}(\eta_2)$ correspond à la matrice de passage du référentiel local (où sont exprimées les vitesses linéaires u, v et w) au référentiel inertiel (où sont exprimées les positions x, y et z) :

$$J_{c_1}(\eta_2) = \begin{bmatrix} \cos\theta\cos\psi & \sin\theta\sin\phi\cos\psi - \sin\psi\cos\phi & \sin\theta\cos\phi\cos\psi + \sin\psi\sin\phi \\ \cos\theta\sin\psi & \sin\theta\sin\phi\sin\psi + \cos\psi\cos\phi & \sin\theta\cos\phi\sin\psi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{3.4}$$

De la même façon, pour les vitesses angulaires :

$$\dot{\eta}_2 = J_{c_2}(\eta_2)\nu_2 \tag{3.5}$$

avec

$$J_{c_2}(\eta_2) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \tag{3.6}$$

3.1.2.2 Etude dynamique

• Dynamique d'un corps rigide

On applique le premier principe de la dynamique au centre de gravité G du sous-marin :

$$\Gamma_1 = m\dot{V}_{G/\Re_0} \tag{3.7}$$

Γ_1 étant la somme des forces appliquées au sous-marin et m représente la somme de la masse du véhicule et de la masse d'eau ajoutée.

On applique également les lois de la mécanique des solides :

$$\Gamma_2 = \dot{H}(C)_{/\Re_0} + V_C \wedge (mV_G) \tag{3.8}$$

avec $H(C) = m(\overrightarrow{CG} \wedge V_C) + I_C\nu_2$ le moment cinétique au point C.

Le développement des calculs permet d'aboutir à une relation du type :

$$M_d\dot{\nu} = G_d(\nu)\nu \tag{3.9}$$

avec : M_d la matrice d'inertie due à la dynamique du système :

$$M_d = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & -mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_{xx} & -I_{xy} & -I_{xz} \\ mz_G & 0 & -mx_G & -I_{xy} & I_{yy} & -I_{yz} \\ -my_G & mx_G & 0 & I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}$$

et $G_d(\nu)\nu$ le vecteur des forces de Coriolis et des forces centrifuges appliquées au véhicule :

$$G_d(\nu) = \begin{bmatrix} \mathbf{0_{3x3}} & mS\nu_\mathbf{1} + \nu_\mathbf{2} \wedge \overrightarrow{CG} \\ mS\nu_\mathbf{1} + \nu_\mathbf{2} \wedge \overrightarrow{CG} & S(\mathbf{I_C}\nu_\mathbf{2}) \end{bmatrix}$$

$$G_d(\nu) = \begin{bmatrix} 0 & 0 & 0 & \dots & \dots & 0 & -m(w+py_G-qx_G) & m(v-pz_G+rx_G) \\ 0 & 0 & 0 & \dots & \dots & m(w+py_G-qx_G) & 0 & -m(u+qz_G-ry_G) \\ 0 & 0 & 0 & \dots & \dots & -m(v-pz_G+rx_G & m(u+qz_G-ry_G) & 0 \\ 0 & -m(w+py_G-qx_G) & m(v-pz_G+rx_G) & \dots & \dots & 0 & I_{xz}p+I_{yz}q-I_{zz}r & -I_{xy}p+I_{yy}q-I_{yz}r \\ m(w+py_G-qx_G) & 0 & -m(u+qz_G-ry_G) & \dots & \dots & -I_{xz}p-I_{yz}q+I_{zz}r & 0 & -I_{xx}p+I_{xy}q+I_{xz}r \\ -m(v-pz_G+rx_G) & m(u+qz_G-ry_G) & 0 & \dots & \dots & I_{xy}p-I_{yy}q+I_{yz}r & I_{xx}p-I_{xy}q-I_{xz}r & 0 \end{bmatrix} \tag{3.10}$$

- **Poids et poussée d'Archimède**

En ajoutant la contribution du poids ($F = mg$) et celle de la poussée d'Archimède ($F_A = \mu V g$, V étant le volume immergé, μ la masse volumique du fluide et g la constante de gravité) et en projetant dans le repère local lié au véhicule, on obtient le résultat suivant :

$$\Gamma_g(\nu) = g \begin{bmatrix} -(m-\mu V)\sin\theta \\ (m-\mu V)\sin\phi\cos\theta \\ (m-\mu V)\cos\phi\cos\theta \\ (y_g m - y_f \mu V)\cos\theta\cos\phi - (z_g m - z_f \mu V)\cos\theta\sin\phi \\ -(z_g m - z_f \mu V)\sin\theta - (x_g m - x_f \mu V)\cos\theta\cos\phi \\ (x_g m - x_f \mu V)\cos\theta\sin\phi + (y_g m - y_f \mu V)\sin\theta \end{bmatrix} \tag{3.11}$$

- **Efforts d'amortissement hydrodynamique**

Ces forces et moments agissent sur tout corps en mouvement dans un fluide supposé visqueux. Ils sont dus à l'action de la masse d'eau ajoutée ainsi qu'aux frottements visqueux du fluide sur le corps en mouvement. En pratique, ils sont issus de méthodes semi-empiriques, donc avec une incertitude parfois importante sur leur valeur. D'une façon générale, on peut formuler les efforts hydrodynamiques de la manière suivante :

Force axiale : $X_{u|u|}u|u|$

Force latérale : $Y_{r|v|}r|v| + Y_{v|v|}v|v|$

Force verticale : $Z_{w|w|}w|w| + Z_{q|w|}q|w|$

Moment de roulis : $K_{p|p|}p|p|$

Moment de tangage : $M_{q|q|}q|q| + M_{w|q|}w|q|$

Moment de lacet : $N_{r|r|}r|r| + N_{v|r|}v|r|$

avec les constantes suivantes :

⋆ X_{vr} X_{wq} Y_{wp} Y_{uv} Z_{uw} Z_{uq} des termes de nature inertielle : ils correspondent à la prise en compte de la masse d'eau ajoutée lors de la dérivation de la vitesse.

⋆ $X_{u|u|}$ $Y_{v|v|}$ $Z_{w|w|}$ $Y_{r|v|}$ $Y_{v|r|}$ $Z_{w|q|}$ $Z_{q|w|}$ $N_{v|v|}$ $M_{w|w|}$ $Y_{r|r|}$ $Z_{q|q|}$ $N_{v|r|}$ $N_{r|v|}$ $M_{w|q|}$ $M_{q|w|}$ $N_{r|r|}$ $M_{q|q|}$ des termes de trainée.

⋆ Y_{ur} Y_{uv} Y_{up} Z_{uw} Z_{uq} L_{uv} L_{ur} L_{up} M_{uw} M_{uq} N_{uv} N_{ur} N_{up} des termes de portance.

⋆ $L_{p|p|}$ est un terme d'amortissement en roulis.

- **Propulsion par l'hélice**

La propulsion principale de l'engin est assurée par une hélice placée dans l'axe du véhicule. Cette hélice a pour effet une poussée Q_c, à partir de laquelle est calculée la vitesse de rotation des pales de l'hélice : $N = 2.24 \times \sqrt{Q_c/P_x}$ (P_x étant la poussée maximale).

Remarque L'ensemble des paramètres présents dans cette section ont été fourni par l'IFREMER.

Cette vitesse de rotation permet de calculer le coefficient d'avance : $J = \frac{V_a}{ND_H}$ avec V_a la vitesse d'avance ($V_a = u + q * z_g$). On peut alors calculer le coefficient de poussée : $K_t = \alpha_0 - \alpha_1 \times J + \alpha_2 \times J^2 - \alpha_3 \times J^3$.

L'expression des efforts de propulsion est alors donnée par la relation suivante :

$$\Gamma_u = 0.25 \times K_t \mu N^2 \frac{D_h^4}{\pi^2} \tag{3.12}$$

où D_h est un paramètre de l'hélice.

Remarque : Le couple de réaction de l'hélice, qui induit un angle de roulis lors d'une variation de poussée n'est pas modélisé ici ; seule la force suivant l'axe Ox est considérée.

- **Forces dues aux gouvernes**

Les gouvernes exercent deux forces : la force de portance f_n normale à la surface de l'aileron et la force de trainée f_t, tangente à la surface dont les expressions sont les suivantes :

$$f_n = k_n(\beta - \alpha)v_r \tag{3.13}$$

et

$$f_t = (k_{t_1}(\beta - \alpha)^2 + k_{t_2})v_r \tag{3.14}$$

avec :

⋆ β : l'angle de braquage de la gouverne par rapport au sous-marin.

⋆ α : l'angle d'incidence du sous-marin dans la direction considérée.

⋆ v_r : la vitesse de l'eau sur l'aileron : $v_r = u^2 + v^2 + w^2$.

⋆ k_n, k_{t_1} et k_{t_2} : coefficients dépendant de la surface et de la forme des gouvernes, ainsi que de la masse volumique du fluide.

Il suffit ensuite de projeter ces forces sur les axes Ox et Oz pour les gouvernes horizontales, Ox et Oy pour la gouverne verticale et Ox, Oy et Oz pour la paire d'ailerons inclinés.

Nous obtenons au final un modèle non-linéaire de la forme :

$$\dot{X} = f(X,\, U) \tag{3.15}$$

$$Y = g(X, U) \tag{3.16}$$

où $X = (\eta,\ \nu)^T$ est l'état, U les commandes (β_1, β_1', δ, β_2, β_2',), et Y les mesures (ici les 12 variables d'état).

3.1.3 Implémentation sous *Matlab/Simulink*

Le modèle non linéaire complet est implémenté sous Matlab, en vue de simulations. Les différents blocs du schéma simulink correspondent aux fonctions

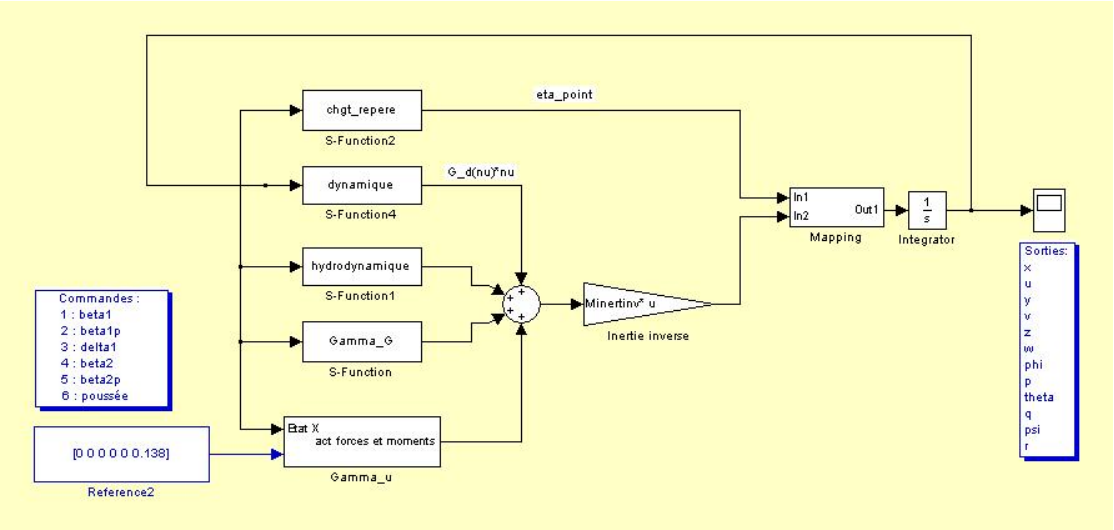

FIGURE 3.4 – Schéma Simulink de simulation du modèle non-linéaire du sous-marin

suivantes :

- *chgt_repere* : permet de passer du référentiel lié au véhicule, où sont exprimées les vitesses, au référentiel terrestre pour l'expression des positions (terme $J_c(\eta_2)$ de l'équation (3.2))
- dynamique : contient la participation du terme $G(\nu)\nu$ de l'équation (3.1) regroupant l'action des forces centrifuges et de Corriolis (équation 3.9)
- *hydrodynamique* : décrit les amortissements hydrodynamiques identifiés s'applicant au sous-marin
- *Gamma_G* : regroupe la force de gravité et la poussée d'archimède (de l'equation 3.11)
- *act forces et moments* : regroupe tous les efforts liés aux actionneurs (Γ_u dans l'équation (3.1)), c'est à dire les 5 ailerons répartis sur le véhicule (equation 3.13 et 3.14 après projection sur les axes) et la poussée de l'hélice (equation 3.12).
- *Minertinv* : correspond à l'inverse de la matrice d'inertie, regroupant la masse du véhicule et les termes de masses ajoutées.
- *Mapping* : alors que les vitesses et les positions sont calculées séparement, ce bloc permet la réorganisation des états x_i du système sous une forme plus classique $[x_1,\ \dot{x}_1,\ \ldots,\ x_n,\ \dot{x}_n]$. Cette étape permet notamment d'avoir après linéarisation une meilleure structure pour la matrice A du système (la matrice A contient un grand nombre de 0)
- *Integrator* : à partir de la dérivée de la position et de la vitesse, l'intégrateur permet d'obtenir l'état X du système au temps t

Dans la suite de ce manuscrit, nous avons choisi d'appliquer des outils de commande linéaire à paramètres variants, dont la synthèse est basé sur un modèle linéarisé du système, dont l'obtention est détaillée dans la suite de ce chapitre.

3.2 Découplage des différentes directions

Le contrôle de l'ensemble du véhicule sous-marin par un seul contrôleur est un problème difficile à cause de couplages, d'incertitudes, du sous-actionnement du véhicule, mais également de la taille importante du système (on rappelle que le modèle complet est représenté par 12 variables d'état et 6 commandes). Une solution habituelle (Feng and Allen, 2004) est de diviser le modèle en 3 sous-modèles de taille réduite. Cette solution permet de découpler le contrôle suivant 3 sous-modèles : un pour l'avance suivant l'axe Ox, puis 2 pour chacun des plans horizontal (Oxy) et vertical (Oxz).

3.2.1 Réduction de modèle pour l'avance

Pour le contrôle de la vitesse longitudinale, suivant la direction Ox, un modèle contenant toutes les variables d'état est construit, de façon à prendre en compte les perturbations induites par les termes de couplage, mais avec pour seul actionneur la poussée de l'hélice Q_c, qui est la seule à avoir une influence sur la vitesse u.

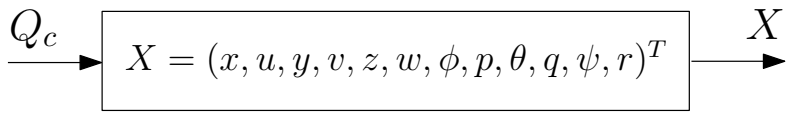

FIGURE 3.5 – Modèle réduit pour l'avance du véhicule

L'exemple de mission choisi comme application est le suivi de fond sous-marin en vue de sa cartographie, ce qui nécessite une vitesse d'avance constante ; on choisit donc le point d'équilibre suivant $\nu_e = [u_e\ v_e\ w_e\ p_e\ q_e\ r_e] = [1\ 0\ 0\ 0\ 0\ 0]$: toutes les vitesses sont égales à zéro sauf la vitesse longitudinale égale à $1m/s$ (vitesse d'avance choisie pour les besoins de la mission).

La linéarisation tangentielle autour du point d'équilibre ν_e permet d'obtenir un modèle de la forme :

$$\begin{cases} \dot{\tilde{X}} = A\tilde{X}(t) + B\tilde{U}(t) \\ \tilde{Y} = C\tilde{X}(t) + D\tilde{U}(t) \end{cases}$$

avec

- $\tilde{X}$ l'état du système : $\tilde{X} = [x\ u\ y\ v\ z\ w\ \phi\ p\ \theta\ q\ \psi\ r]^T$ (le système étant linéarisé, il s'agit évidement de variations autour du point d'équilibre ; les notations précédentes sont conservées pour simplifier l'écriture).
- $\tilde{U}$ les commandes $\tilde{U} = Q_c$
- $\tilde{Y}$ les sorties mesurées : tout l'état est mesuré.

Toutes les matrices du modèle linéaire A, B, C et D dépendent des paramètres : hydrodynamiques, masse du véhicule, dimension des ailerons... et ont la structure suivante :

$$A = \begin{bmatrix} 0 & a_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_2 & 0 & 0 & 0 & 0 & 0 & 0 & a_3 & a_4 & 0 & 0 \\ 0 & 0 & 0 & a_5 & 0 & 0 & 0 & 0 & 0 & 0 & a_6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_7 & a_7 & 0 & 0 & 0 & a_9 \\ 0 & 0 & 0 & 0 & 0 & a_{10} & 0 & 0 & a_{11} & 0 & 0 & 0 \\ 0 & a_{12} & 0 & 0 & 0 & a_{13} & 0 & 0 & a_{14} & a_{15} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{16} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_{17} & a_{18} & 0 & 0 & 0 & a_{19} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{20} & 0 & 0 \\ 0 & a_{21} & 0 & 0 & 0 & a_{22} & 0 & 0 & a_{23} & a_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{25} \\ 0 & 0 & 0 & a_{26} & 0 & 0 & a_{27} & 0 & 0 & 0 & 0 & a_{28} \end{bmatrix} \tag{3.17}$$

$$B = \begin{bmatrix} 0 & b_1 & 0 & 0 & 0 & b_2 & 0 & 0 & 0 & b_3 & 0 & 0 \end{bmatrix}^T \tag{3.18}$$

$$C = \mathbb{I}_{12} \tag{3.19}$$

$$D = 0_{12\times 6} \tag{3.20}$$

Se modèle sera utilisé dans la section 4.2 pour la synthèse du contrôleur de la vitesse d'avance.

3.2.2 Réduction de modèle au plan horizontal

Dans le plan Oxy, le contrôle du sous-marin est réduit au contrôle de l'angle de lacet ψ (pour ce modèle, on suppose que la vitesse d'avance u est constante). Les variables d'état conservées dans le modèle réduit sont v, ψ et r (vitesse suivant O_y, angle de lacet et vitesse correspondant). Les 3 ailerons à l'arrière du véhicule (angles δ, β_2 et β_2') sont également conservés dans ce modèle pour le plan vertical. En effet ces 3 ailerons ont une force de portance non nulle dans le plan vertical (un aileron vertical et deux ailerons inclinés de $\pm\frac{\pi}{3}$)

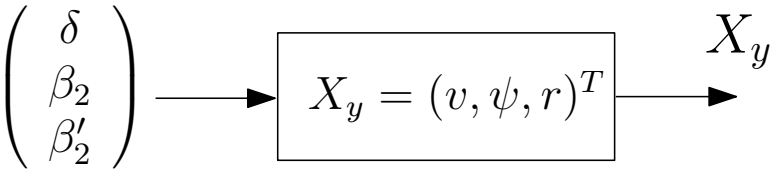

FIGURE 3.6 – Modèle réduit pour le plan horizontal

Un modèle linéaire, qui ne sera pas détaillé dans cette thèse, peut être obtenu par linéarisation tangentielle comme dans la section précédente.

3.2.3 Réduction de modèle au plan vertical

Pour le contrôle de l'altitude z, le modèle est réduit à 4 variables d'état : z et θ ainsi que les vitesses correspondantes w et q (on suppose là encore que la vitesse d'avance u est constante). En ce qui concerne les actionneurs, seuls 4 ailerons sont nécessaires : les 2 ailerons horizontaux à l'avant du véhicule (β_1 et β_1') et les 2 ailerons inclinés à l'arrière (β_2 et β_2'). Comme l'AUV doit rester dans le plan vertical (mode suivi de fond) les ailerons d'une même paire doivent être commandés avec le même angle, c'est à dire $\beta_1 = \beta_1'$ et $\beta_2 = \beta_2'$. Le modèle réduit du sous-marin ne possède donc plus que 2 commandes.

Ce modèle pourrait également être obtenu par linéarisation tangentielle autour d'un point d'équilibre. Par la suite, nous avons choisi de mettre en place une

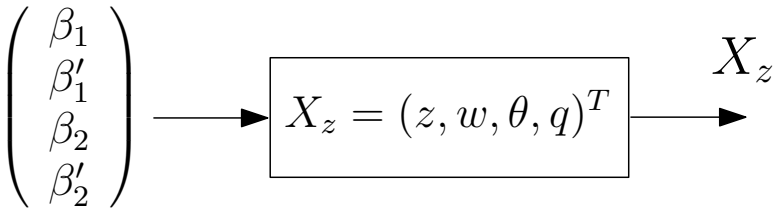

FIGURE 3.7 – Modèle réduit pour le plan vertical

structure cascade pour le contrôle de l'altitude, qui sera présentée dans la section 3.3.

3.2.4 Fusion des trois sous-modèles

Le contrôle de l'ensemble du véhicule peut alors se faire en synthétisant un correcteur pour chacun des sous-modèles précédent, puis en ajoutant les commandes issues de chaque contrôleur. Plusieurs précautions sont cependant à prendre : tout d'abord concernant la saturation des actionneurs utilisés par plusieurs contrôleurs (si la somme de 2 actions dépasse la saturation, qu'en est-il de la stabilité du système global, et de ses performances ?). De plus la robustesse apportée par la méthode de synthèse ($\mathcal{H}_\infty$ par exemple) est-elle suffisante pour rejeter les perturbations provoquées par les variations des autres états que ceux compris dans le modèle de synthèse (par exemple lors du contrôle global, le contrôleur pour l'altitude est-il robuste à une variation suivant l'axe Oy ?). Si ce n'est pas le cas, le rejet de perturbation peut être ajouté dans les objectifs de commande.

Dans le reste de notre étude, cet aspect du contrôle n'est pas pris en compte. Seul le contrôle du véhicule dans le plan vertical est considéré. Celui-ci nécessite cependant une vitesse d'avance constante (pour ne pas influencer les résultat) et donc le modèle d'avance est utilisé pour synthétisé un contrôleur contrôlant u. Le plan horizontal est laissé de coté à partir de maintenant.

Remarque : Rappelons que l'objectif de l'étude n'est pas le contrôle du sous-marin en lui même mais la commande sous contrainte de ressources informatiques d'un système complexe comme peut l'être le sous-marin.

3.3 Plan vertical : structure cascade z/θ

Le mouvement du véhicule dans le plan vertical résulte de l'action des ailerons horizontaux à l'avant du véhicule, et de ceux inclinés à l'arrière, commandés par des angles β_1 et β_2 respectivement. Grâce à ces 2 paires d'actionneurs, on peut théoriquement créer de façon indépendante un couple autour de l'axe de tangage

et une poussée suivant l'axe z du véhicule, et ainsi découpler l'altitude et l'angle d'incidence (par exemple changement d'altitude en gardant le véhicule horizontal).

Cependant ce mode de déplacement génère une force de trainée importante, due à l'angle d'incidence entre le véhicule et le fluide en mouvement. Comme la gestion de l'énergie à bord d'un AUV est un enjeu crucial, l'angle entre le sous-marin et sa trajectoire doit être limité le plus possible (de façon à réduire les pertes dues aux forces de trainée). De ce fait les forces exercées par les ailerons avant et arrière doivent être coordonnées pour maintenir le véhicule tangent à sa trajectoire.

Comme l'altitude est directement liée à l'angle d'incidence, la meilleure façon de gravir une pente est de maintenir le véhicule parallèle à la pente en contrôlant l'angle d'incidence (Varrier (2010)).

C'est pourquoi une structure de contrôle en cascade est mise en place, voir figure 3.8.

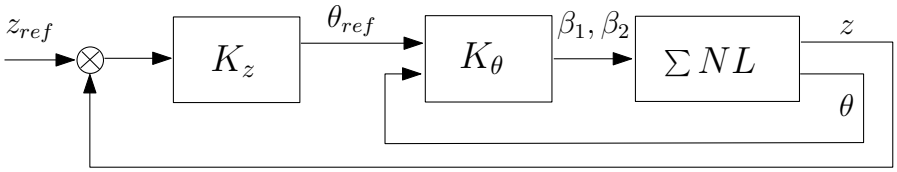

FIGURE 3.8 – Structure cascade pour le contrôle de l'altitude

Le contrôleur d'altitude donne une référence d'angle d'incidence ; cette référence est ensuite utilisée par le contrôleur d'angle d'incidence pour calculer l'action à appliquer sur l'AUV. Cette structure est destinée à remplacer le modèle réduit au plan vertical de la sous-section 3.2.3.

Notons qu'une structure cascade similaire a déjà été décrite dans Leveille (2007) où une structure PD/PI en cascade permet le rejet de perturbation.

Comme cela sera décrit dans le chapitre 4 ; le correcteur de l'angle de tangage est synthètisé à partir du model réduit au plan vertical de la section 3.2.3 pour le cas LTI, où à partir du modèle LPV (polytopique ou LFR) qui sera construit dnas la section 3.4. Pour le contrôleur d'altitude, ayant comme sortie la référence de l'angle de tangage, on exploite la relation géométrique entre tangage et altitude, présenté sur la figure :

D'après la figure 3.9, la relation entre l'angle de tangage et l'altitude est :

$$z = l \sin(\theta) \tag{3.21}$$

En dérivant cette relation pour un angle θ constant (relation géométrique) on obtient : $\dot{z} = \dot{l} \sin \theta$.

De plus la vitesse d'avance est la dérivée de la longueur l : $u = \dot{l}$, d'où :

$$\dot{z} = u \sin \theta \tag{3.22}$$

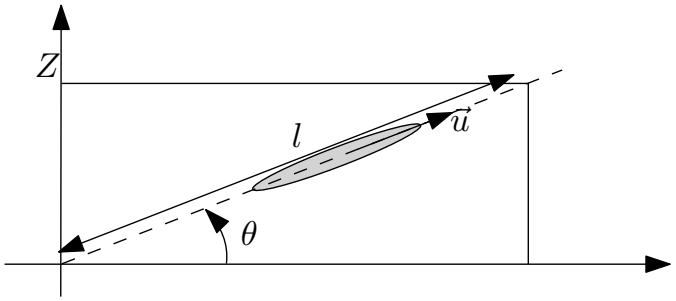

FIGURE 3.9 – Relation entre z θ

Par un développement limité à l'ordre 1 de la fonction *sinus*, on obtient la relation suivante :

$$\dot{z} \simeq u\theta \tag{3.23}$$

d'où on notera :

$$G_\theta(p) = \frac{z}{\theta} = \frac{u}{p} \tag{3.24}$$

Par la suite, la vitesse d'avance u est choisie égale à 1 mètre/seconde.

La boucle interne de la structure cascade est donc approximée par un intégrateur. Un modèle LPV est construit à partir de ce modèle simplifié suivant la démarche suivante : d'abord on réalise la discrétisation du modèle : $x_{k+1} = x_k + h.u_k$ En tenant compte de la variation δ de période dans un intervalle $[h_{min}, h_{max}]$ avec $h_{min} > 0$ et en considérant une variation autour de la valeur nominale h_0, on obtient le modèle d'altitude LPV suivant :

$$x_{k+1} = x_k + (h_0 + \delta).u_k \tag{3.25}$$

Ce modèle est affine en la variation de période δ, qui est bornée, donc le modèle est déjà polytopique. Il peut également se mettre de façon évidente sous une forme d'une LFR.

3.4 Modèle LPV du sous-marin

Le sous-marin autonome considéré comme application est décrit par des équations non-linéaires présentées dans le section 3.1. Nous avons choisi une approche de commande robuste utilisant des outils de commande linéaire, ce qui a nécessité la linéarisation du modèle. Au cours de cette étape de linéarisation, des informations importantes du modèle de départ ont été perdues, ce qui peut entrainer, lors des simulations sur le système non-linéaire, des performances dégradées, voire même l'instabilité du système.

Plusieurs travaux ont montré l'intérêt des systèmes linéaires à paramètres variants (LPV) pour la modélisation sous forme linéaire des systèmes non-linéaire, en gardant certaines non linéarités comme paramètres du système linéaire (Biannic, 1996; Bruzelius, 2004; Gauthier, 2007).

Pour l'exemple du contrôle d'altitude du sous-marin autonome, en se limitant au plan vertical et en considérant la structure cascade présentée précédemment, une des variables importantes du modèle qu'il peut être intéressant de conserver sous forme de paramètre est l'angle de tangage θ, qui est la variable à contrôler par la boucle interne de la figure 3.8.

L'objectif est donc d'obtenir un modèle LPV pour la synthèse du contrôleur de tangage, dont le paramètre variant est l'angle de tangage θ.

Le modèle utilisé pour la synthèse du correcteur de l'angle de tangage K_θ, n'est composé que de 2 variables d'état : l'angle de tangage θ et la vitesse correspondante q. Les équations d'état non-linéaires décrivant le comportement dynamique de ces états (obtenues en ajoutant les différentes forces décrites dans la section 3.1) sont les suivantes :

$$\begin{aligned}\dot{\theta} =& cos(\phi)q - sin(\phi)r \\ M\dot{q} =& -pr(I_x - I_z) - m[Z_g(qw - rv)] - (Z_q m - Z_f \mu V)gsin(\theta) \\ & - (X_q m - X_f \mu V)gcos(\theta)cos(\phi) + M_{wq}w|q| + M_{qq}q|q| + F_{fins}\end{aligned} \tag{3.26}$$

où M est la matrice d'inertie, m la masse du véhicule, V le volume et μ la masse volumique. Les autres paramètres : $I_x, I_z, Z_g, Z_q, Z_f, X_q, X_f, M_{wq}$ sont présents dans les équations non linéaires présentées dans la section 3.1, dans les fonctions dynamiques et hydrodynamiques. La variable F_{fins} représente les forces et moments dues aux différents ailerons du véhicule (correspondant aux équations 3.13 et 3.14 après projections sur les axes appropriés). Elle dépend de la vitesse d'avance au carré u^2 ainsi que de l'angle entre l'aileron et le fluide. Dans cette première approche, la vitesse d'avance est considérée comme constante, mais il pourrait être intéressant de l'ajouter aux paramètres variants. Afin de ne conserver que l'angle de tangage comme paramètre, on effectue une linéarisation tangentielle autour du point d'équilibre suivant : $X_{eq} = [0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ \theta_{eq}\ 0\ 0\ 0]$, toutes les vitesse et positions d'équilibre sont égales à zero, sauf la vitesse d'avance $u_{eq} = 1m/s$ et l'angle de tangage θ_{eq} (gardé comme paramètre). Aucune valeur numérique n'est assignée à cette valeur à l'équilibre, puisqu'elle sera par la suite choisie comme paramètre variant du modèle (sa valeur sera donc variable, et dépendante de l'état du système).

La linéarisation tangentielle autour de X_{eq} permet d'obtenir les équations suivantes :

$$\dot{\tilde{\theta}} = \tilde{q}$$
$$M\dot{\tilde{q}} = [-(Z_g m - Z_f \mu V) g cos(\theta_{eq}) + (X_g m - X_f \mu V) g sin(\theta_{eq})]\tilde{\theta} + F_{fins_{eq}} \quad (3.27)$$

où $\tilde{\theta}$ et $\tilde{q}$ représente la variation de θ et q par rapport au point d'équilibre, ce qui représente le nouvel état du système.

3.4.1 Modèle LPV polytopique

A partir des équations linéarisées (3.27), on peut construire un modèle polytopique, en considérant 2 paramètres variants : $\rho_1 = cos(\theta_{eq})$ et $\rho_2 = sin(\theta_{eq})$. Pour borner ces 2 paramètres, on fixe un intervalle de variation pour l'angle θ entre -30° et 30°, ce qui correspond au limites physiques de l'AUV dû à l'équilibre hydrostatique, ce qui mène aux bornes suivantes pour les paramètres : $\rho_1 \in [cos(+30^\circ); cos(0^\circ)] = [0.86, 1]$ et $\rho_2 \in [sin(-30^\circ); sin(+30^\circ)] = [-0.5, 0.5]$.

Les équations 3.27 sont affines par rapport aux 2 paramètres considérés, et ces 2 paramètres évoluent à l'intérieur d'un polytope défini par 4 sommets : $\underline{\rho_1}\underline{\rho_2}$, $\underline{\rho_1}\overline{\rho_2}$, $\overline{\rho_1}\underline{\rho_2}$, $\overline{\rho_1}\overline{\rho_2}$.

En calculant les équations (3.27) à chacun de ces sommets, on obtient 4 systèmes sommets, définissant la représentation polytopique du modèle pour l'angle de tangage.

Il faut noter qu'en réalité les deux paramètres ne dépendent que d'une seule variable θ_{eq}. Cette approche peut donc introduire un certain conservatisme.

En appliquant la méthode présentée dans la section 2.4, il est possible de construire un modèle LPV polytopique continu, dépendant à la fois des paramètres ρ_1 et ρ_2 du système, et également du paramètre $\tilde{h}$ représentant la période d'échantillonnage. Ceci aboutit à un modèle polytopique défini par 8 sommets :

$$\underline{\rho_1}\underline{\rho_2}\underline{\tilde{h}},\ \underline{\rho_1}\underline{\rho_2}\overline{\tilde{h}},\ \underline{\rho_1}\overline{\rho_2}\underline{\tilde{h}},\ \underline{\rho_1}\overline{\rho_2}\overline{\tilde{h}},\ \overline{\rho_1}\underline{\rho_2}\underline{\tilde{h}},\ \overline{\rho_1}\underline{\rho_2}\overline{\tilde{h}},\ \overline{\rho_1}\overline{\rho_2}\underline{\tilde{h}},\ \overline{\rho_1}\overline{\rho_2}\overline{\tilde{h}} \quad (3.28)$$

Ce modèle sera utilisé dans le chapitre 4 pour la synthèse d'un contrôleur polytopique dans le cas d'un système LPV.

3.4.2 Modèle LPV sous la forme d'une LFR

A partir des équations (3.27), on peut également construire un modèle LPV sour la forme d'une Représentation Linéaire Fractionnelle (LFR), dont la forme générale est rappelée sur la figure 3.10.

Le bloc Δ contient la partie variable, dépendante des paramètres du système, qui dépend du point de linéarisation (variable) choisi : θ_{eq}. La forme LFR est définie de la façon suivante :

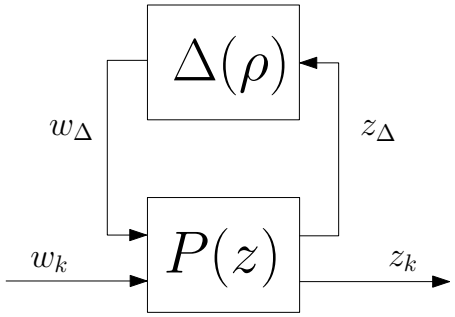

FIGURE 3.10 – Système sous forme LFR

$$z_\Delta = \begin{bmatrix} \tilde{\theta} \\ \tilde{\theta} \end{bmatrix}; \quad w_\Delta = \Delta z_\Delta; \; \Delta = \begin{bmatrix} \rho_1 & 0 \\ 0 & \rho_2 \end{bmatrix} \tag{3.29}$$

avec :

$$\begin{cases} \rho_1 = cos(\theta_{eq}) \\ \rho_2 = sin(\theta_{eq}) \end{cases} \tag{3.30}$$

A partir des equations 3.26, le système P est défini de la façon suivante :

$$\begin{aligned} \begin{pmatrix} \dot{\tilde{\theta}} \\ \dot{\tilde{q}} \end{pmatrix} &= A_{LFT} \begin{pmatrix} \tilde{\theta} \\ \tilde{q} \end{pmatrix} + B_{LFT} \begin{pmatrix} w_\Delta \\ w_k \end{pmatrix} \\ \begin{pmatrix} z_\Delta \\ z_k \end{pmatrix} &= C_{LFT} \begin{pmatrix} \tilde{\theta} \\ \tilde{q} \end{pmatrix} + D_{LFT} \begin{pmatrix} w_\Delta \\ w_k \end{pmatrix} \end{aligned} \tag{3.31}$$

avec w_k et z_k les entrées et sorties telles que définies sur la figure 3.10, et les matrices du système définies comme suit :

$$A_{LFT} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B_{LFT} = \begin{bmatrix} B_\Delta & B \end{bmatrix} \tag{3.32}$$

$$C_{LFT} = \begin{bmatrix} C_\Delta \\ C \end{bmatrix}, \quad D_{LFT} = \begin{pmatrix} D_\Delta & 0 \\ 0 & D \end{pmatrix} \tag{3.33}$$

avec $B_\Delta = \begin{pmatrix} 0 & 0 \\ -(Z_g m - Z_f \mu V)g & X_g m - X_f \mu V)g \end{pmatrix}$, $C_\Delta = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$, $D_\Delta = 0_{2\times 2}$, les matrices B et D liées aux forces et moments des ailerons (F_{fins} de l'équation 3.26) et $C = \mathbb{I}_{2\times 2}$ car les deux états sont mesurés.

En suivant la méthode présentée dans la section 2.6, un modèle LPV discret sous la forme LFR peut être construit, dépendant à la fois des paramètres ρ_1 et

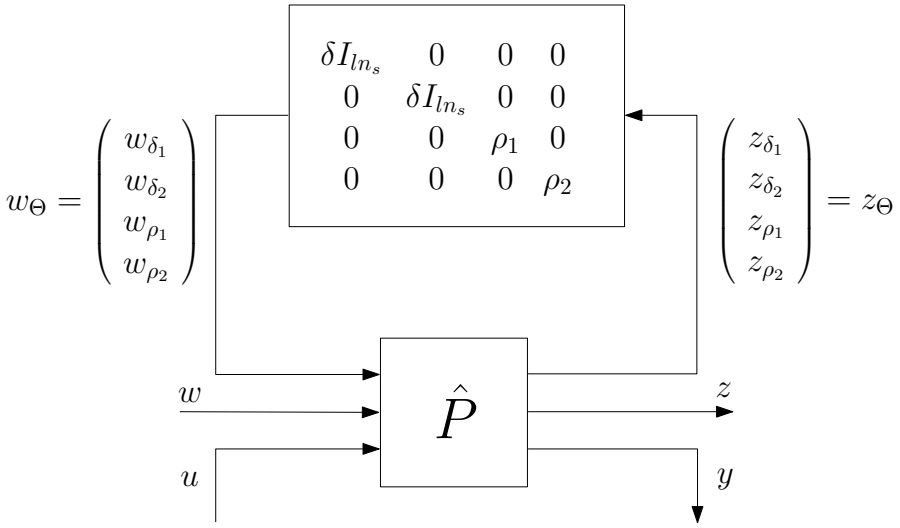

FIGURE 3.11 – Système sous forme LFR

ρ_2 mais aussi de la période d'échantillonnage h comme présenté sur la figure 3.11, avec l l'ordre du développement en série de Taylor (ici $l = 1$) et n_s le nombre d'état (ici $n_s = 2$).

Les matrices du système discret $\hat{P}$ sont définies par les équations 2.49 à 2.52 appliquées au système sous la forme d'une LFR de l'AUV (voir équation 3.31).

3.4.2.1 Conclusion

Dans ce chapitre, différents modèles de sous-marins ont étés présentés. Tout d'abord le modèle non-linéaire, le plus complet et le plus proche du système réel, mais dont la formulation n'est pas adaptée aux lois de commande que l'on souhaite appliquer. C'est ce modèle qui sera utilisé pour les simulations dans le chapitre suivant.

Puis un modèle linéarisé a été obtenu, par une linéarisation tangentielle autour d'un point d'équilibre. Les couplages de ce modèle en font un système difficile à contrôler dans son ensemble. Une stratégie pour séparer les mouvements suivant l'axe Ox et les plans Oxy et Oxz a donc été mise en place, ce qui permet de découpler le mouvement et donc de simplifier les modèles.

Pour le contrôle de l'altitude de l'AUV, qui est l'application choisie comme illustration des approches de contrôle, une structure en cascade permet de tenir compte du lien fort existant entre l'altitude et l'angle de tangage. Cette structure permet encore de simplifier le modèle, mais surtout de tenir compte des contraintes de navigation du sous-marin lors de la modélisation.

Finalement, deux modèles LPV de l'AUV, l'un sous forme polytopique et le

second sous la forme d'une LFR ont été développés, afin de conserver lors de la modélisation un paramètre essentiel pour le contrôleur d'altitude qui est l'angle de tangage, et dont les équations non-linéaires (pour l'altitude et le tangage) dépendent très fortement.

Nous obtenons ainsi des modèles LPV à temps discret dont les paramètres variants représentent soit des non-linéarités du modèle, soit un paramètre structurel lié à la mise en œuvre informatique (temps réel) de la commande.

Chapitre 4

Commandes LPV d'un sous-marin autonome à intervalle d'échantillonnage variable

Ce chapitre présente la mise en application des méthodes de synthèse de contrôleur à période d'échantillonnage variable, pour un système LTI puis pour un système LPV. L'application choisie est un véhicule sous-marin autonome, dont la modélisation a été effectuée dans le chapitre 3. Le but est le contrôle de l'altitude en utilisant la structure cascade de la section 3.3.

Après la description du scénario de simulation mis en œuvre, la synthèse du contrôleur de vitesse d'avance K_u est détaillée. Puis la méthode de synthèse de contrôleur à période d'échantillonnage variable est testée, dans un premier temps pour un modèle LTI. Dans ce cas nous présentons les deux approches permettant la synthèse d'un correcteur à période variable (pour un modèle LTI) : l'approche polytopique et l'approche LFR. Ensuite cette méthode est étendue aux systèmes LPV, dont le modèle en temps continu est supposé dépendre d'un ou plusieurs paramètres variants. Les deux approches proposées, LPV polytopique et LFR, sont comparées afin de dégager quelques conseils pour le concepteur de lois de commandes à période variable.

4.1 Scénario de simulations

La mission considérée dans l'ensemble du manuscrit est le contrôle d'altitude d'un sous-marin, en vue par exemple de la cartographie du fond de l'océan. Comme expliqué dans le chapitre 3 consacré aux modèles, nous avons restreint notre étude au contrôle d'altitude. Cependant, pour le type de mission envisagé, plusieurs contraintes sur les états du véhicule doivent être satisfaites.

Tout d'abord la vitesse d'avance u du véhicule doit rester la plus constante possible pour faciliter l'interprétation des résultats obtenus. Typiquement, la vitesse d'avance d'un AUV comme l'$Asteri^X$ se situe entre 0.5 et 2.5 m/s ; nous avons choisi une vitesse de $1m/s$, pas trop faible pour permettre aux ailerons d'avoir une force de portance suffisante, et pas trop élevée pour ne pas arriver en limite des capacités du véhicule.

La vitesse de roulis p doit être maintenue la plus proche possible de zéro, pour permettre une bonne interprétation des mesures collectées en mission. La solution $\mathcal{H}_\infty$ standard consisterait à placer une fonction de pondération sur le sortie p du modèle de synthèse, de façon à limiter son évolution. C'est une solution impossible dans notre cas, puisque la synthèse de contrôleur utilise un modèle réduit du sous-marin, où la vitesse de roulis ne fait pas partie des variables d'état. Cependant, les ailerons des deux paires contenues dans le modèle réduit (la paire horizontale à l'avant et la paire inclinée à l'arrière) sont actionnées avec la même consigne, ce qui permet d'avoir des forces égales des deux cotés du véhicule, et donc un angle de roulis nul lors du contrôle de l'altitude en simulation.

Remarque : Dans le cas du contrôle complet du sous-marin, il faut notamment ajouter un contrôle de l'angle de lacet ψ, qui nécessite une action différente des ailerons d'une même paire, qui pourrait conduire à une vitesse de roulis trop importante. Pour remédier à ce problème, on pourrait ajouter la vitesse de roulis au modèle de synthèse (qui ne contient que 3 états : v, ψ et r) et la limiter par une pondération.

Les simulations sont toutes effectuées sur le modèle complet et non-linéaire du sous-marin, contenant les 12 variables d'état, puisqu'il est le plus proche du véhicule réel.

Comme présenté dans le chapitre 3 de modélisation du véhicule, l'essentiel de notre travail porte sur le contrôle de l'altitude z, couplé à l'angle de tangage θ. Cependant pour respecter la contrainte sur la vitesse d'avance du sous-marin, qui doit rester la plus constante possible, un contrôleur doit être synthétisé pour cet état. Le schéma de contrôle global mis en place dans la suite de ce mémoire est décrit sur la figure 4.1.

Au total trois contrôleurs seront donc mis en œuvre : le contrôleur de vitesse d'avance K_u, et la structure cascade pour l'altitude comprenant le correcteur d'altitude K_z qui délivre une référence de tangage au régulateur K_θ qui actionne les ailerons du véhicule.

La suite du chapitre décrit la synthèse puis la simulation sur le modèle non-linéaire des différents contrôleurs. Le contrôleur de vitesse d'avance est le même pour l'ensemble des simulations présentées ; sa synthèse est développée dans la section 4.2. Puis le contrôle d'altitude de l'AUV en utilisant la structure cascade est étudiée dans le cas où le modèle continu du système est LTI : seule la pé-

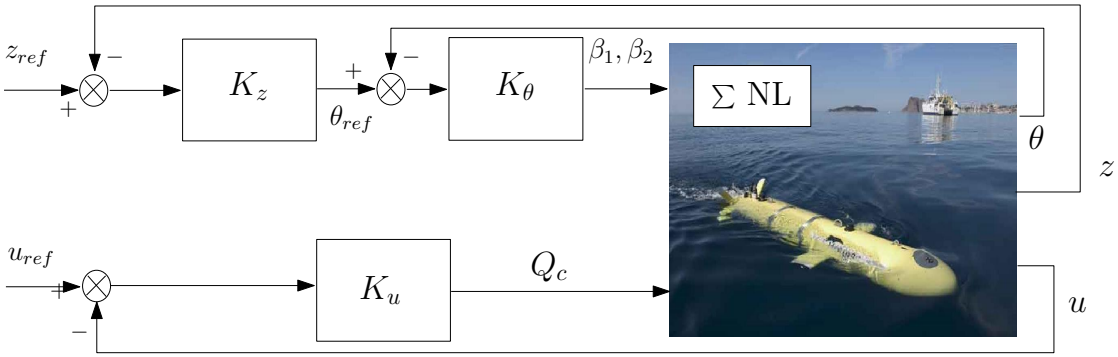

FIGURE 4.1 – Schéma de contrôle global

riode d'échantillonnage est utilisée pour séquencer le contrôleur. Les approches polytopique et LFR sont comparées dans ce cas, dans la section 4.3. Enfin le cas LPV est considéré dans la section 4.4, avec un contrôleur séquencé à la fois par des paramètres conservés lors de la linéarisation du système, et par la période d'échantillonnage (là encore les méthodes polytopique et LFR sont comparées).

4.2 Contrôleur K_u pour la vitesse d'avance

Le contrôleur de vitesse d'avance est synthétisé à partir du modèle linéarisé réduit présenté à la section 3.2.1, qui contient les 12 variables d'état et la poussée Q_c de l'hélice pour seul actionneur. La méthode $\mathcal{H}_\infty$ (décrite à la section 1.1) est employée pour la synthèse, suivant le schéma de contrôle présentée sur la figure 4.2

Remarque : On rappelle que le modèle utilisé est obtenu en considérant u_{eq} constant.

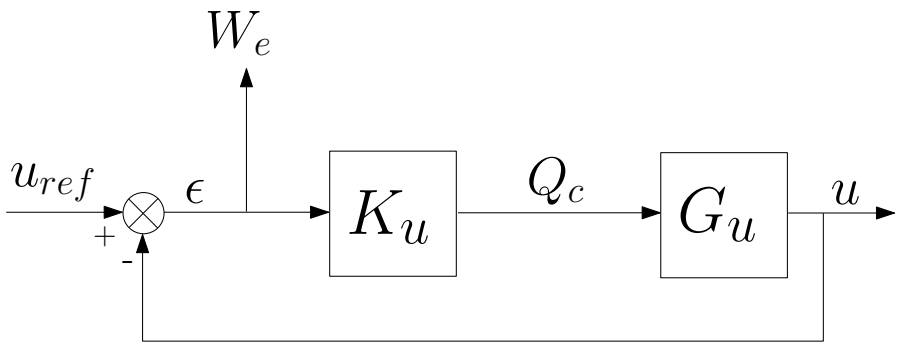

FIGURE 4.2 – Schéma de contrôle pour la vitesse d'avance u

Les objectifs en boucle fermée sont fixés grâce à la fonction de pondération W_e

définie de la façon suivant :

$$W_{u_e} = \left(\frac{\frac{s}{M_{S_{u_e}}} + \omega_{S_{u_e}}}{s + \omega_{S_{u_e}} \epsilon_{u_e}} \right)^2 \tag{4.1}$$

avec $M_{S_{u_e}} = 2$, $\epsilon_{u_e} = 0.01$ et $\omega_{S_{u_e}} = 0.46$.

La synthèse du contrôleur est réalisée en continu, en utilisant la méthode présentée dans la section 1.1.2. A l'issue de la synthèse, on obtient un niveau d'atténuation $\gamma = 102$ et la fonction de sensibilité S donnée sur la figure 4.3.

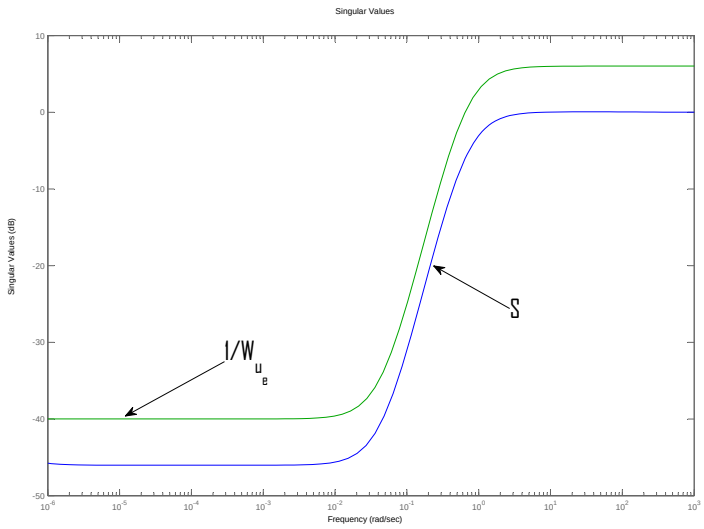

FIGURE 4.3 – Fonction de sensibilité S ainsi que la pondération $1/W_{u_e}$

Bien que γ soit élevé (ce qui correspond à un non respect de la fonction de pondération et donc a priori à de mauvaises performances en boucle fermée), on voit sur la figure 4.3 que la fonction de sensibilité est respectée dans une plage de fréquence $[10^-5, 10^3]$ rad/sec, qui parait suffisante pour le système considéré (ce qui a été vérifié en simulation).

Ce correcteur est finalement discrétisé en utilisant la fonction *c2d* de *Matlab* (avec la méthode *tustin*), à une période d'échantillonnage fixe et égale à 0.1 secondes, de façon à s'intégrer au schéma de contrôle discret.

Remarque : La synthèse discrète de ce correcteur a été essayée, mais n'a pas abouti à cause de problèmes numériques.

4.3 Contrôle d'un système LTI à période variable

Dans un premier temps, la méthode de synthèse de contrôleur LPV discret avec période d'échantillonnage variable présentée dans le chapitre 2 est appliquée à la structure cascade pour le contrôle de l'altitude. Le modèle de départ est un système LTI, rappelons que c'est la phase de discrétisation en gardant la période d'échantillonnage comme paramètre variant qui aboutit à un système LPV. La structure de contrôle développée dans la section 3.3 est rappelée ici :

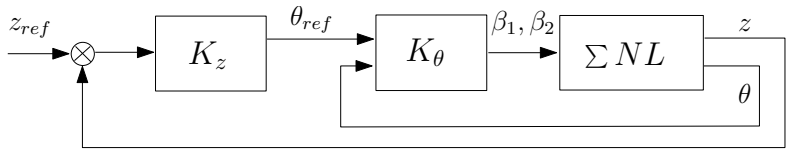

FIGURE 4.4 – Structure cascade pour le contrôle de l'altitude

Le contrôle de l'altitude par la structure cascade nécessite 2 régulateurs :

- le contrôleur d'altitude K_z qui est synthétisé à partir d'un intégrateur, obtenu en étudiant la relation géométrique entre l'altitude et l'angle de tangage (voir section 3.3. Ce contrôleur reçoit en entrée la différence entre la référence d'altitude et sa mesure, effectuée par le capteur à ultrason qui introduit un retard pris en compte dans la synthèse comme un intervalle variable d'application de la commande. Le calcul de K_z utilise donc un modèle LPV du système (avec la période d'échantillonnage comme paramètre variant) puis une synthèse LPV, soit polytopique soit LFR est réalisée. Le modèle de synthèse du contrôleur est au départ un système LTI, c'est la phase de discrétisation, en gardant la période comme paramètre variant dans le modèle, qui va mener à un système LPV. Le modèle LPV discrétisé avec la période d'échantillonnage comme paramètre variant, utilisé pour le synthèse du contrôleur d'altitude, est rappelé ici :

$$\hat{G}_z(h) : \begin{cases} z_{k+1} = z_k + (h_0 + \delta).U_k \\ y_k = z_k \end{cases} \tag{4.2}$$

- le contrôleur de l'angle de tangage K_θ qui est synthétisé à partir du modèle réduit dans le plan vertical, composé des 4 variables d'état suivante : z, w, θ et q (la mesure de θ étant acquise instantanément), et des entrées de contrôle β_1 et β_2. Ce régulateur est synthétisé en temps discret, avec une période d'échantillonnage fixe et égale à $T_e = 0.005s$ (pour être plus rapide que la boucle externe). Le modèle utilisé est le suivant :

$$G_\theta : \begin{cases} \dot{X}_\theta(t) = A_\theta X_\theta(t) + B_\theta U_\theta(t) \\ Y_\theta = \mathbb{I}_4 X_\theta(t) + \mathbb{O}_{4\times 2} U_\theta(t) \end{cases} \tag{4.3}$$

où

$$X_\theta(t) = \begin{bmatrix} z \\ w \\ \theta \\ q \end{bmatrix}, \quad U_\theta(t) = \begin{bmatrix} \beta_1 \\ \beta_1 \end{bmatrix} \tag{4.4}$$

Remarque : dans le cas du contrôle LPV qui sera présenté dans la section 4.4, ce modèle LTI sera remplacé par le modèl poytopique ou LFR obtenus section 3.4.

4.3.1 Contrôleur d'angle d'incidence K_θ discret à période d'échantillonnage constante

Le régulateur pour l'angle d'incidence est de type $\mathcal{H}_\infty$ discret classique ; c'est à dire avec une période d'échantillonnage constante. Le problème de sensibilité mixte est considéré, avec les fonctions de pondération, W_{θ_e} sur l'erreur de poursuite et W_{θ_u} sur la commande, définies comme suit :

- la fonction de pondération W_{θ_e} placée sur l'erreur $\theta_{ref} - \theta$ est définie par le filtre du premier ordre suivant :

$$W_{\theta_e} = \left(\frac{\frac{s}{M_{S_{\theta_e}}} + \omega_{S_{\theta_e}}}{s + \omega_{S_{\theta_e}} \epsilon_{\theta_e}} \right) \tag{4.5}$$

  avec : $M_{S_{\theta_e}} = 1$; $\omega_{S_{\theta_e}} = 2$ et $\epsilon_{\theta_e} = 0.0001$.
- la fonction de pondération sur l'entrée de commande (définie par le vecteur $\begin{bmatrix} \beta_1 & \beta_2 \end{bmatrix}^T$) est définie par la matrice bloc diagonale suivante :

$$W_{\theta_u} = \begin{pmatrix} 0,3 \times W_t & 0 \\ 0 & 0,15 \times W_t \end{pmatrix} \tag{4.6}$$

  avec :

$$W_t = \left(\frac{s + \frac{\omega_{S_{\theta_u}}}{M_{S_{\theta_u}}}}{\epsilon_{\theta_u} s + \omega_{S_{\theta_u}}} \right) \tag{4.7}$$

  et où : $M_{S_{\theta_u}} = 2$; $\omega_{S_{\theta_u}} = 100$ et $\epsilon_{\theta_u} = 0.001$.
  Cette fonction de pondération utilise le même filtre du premier ordre pour les deux commandes, mais les coefficients ($0,3$ pour β_1 et $0,15$ pour β_2) permettent de forcer le contrôleur à plus utiliser les ailerons situés à l'arrière du véhicule, qui ont une force de portance plus faible du fait de leur inclinaison.

Les fonctions de pondération sont définies en temps continu, puis connectées au système continu G_θ de l'equation 4.3. C'est le système augmenté obtenu qui est discrétisé à la période d'échantillonnage constante de $T_e = 0.005s$. La synthèse du régulateur est ensuite effectuée sous *Matlab* en utilisant la fonction *dhinflmi*.

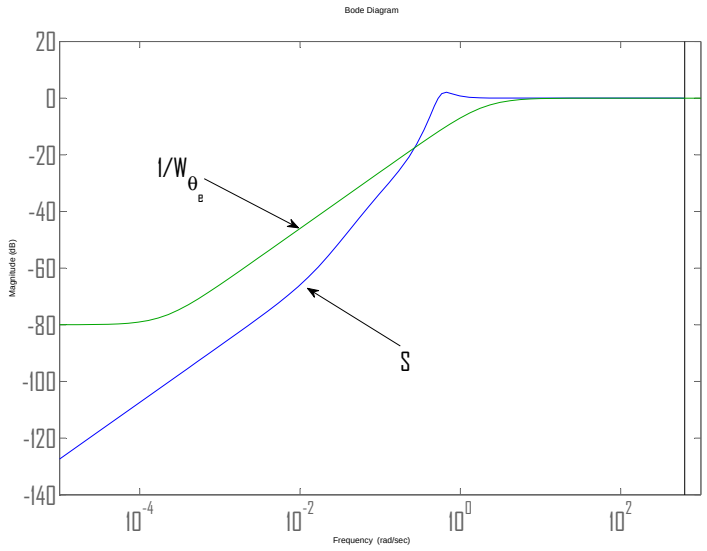

FIGURE 4.5 – Régulateur de l'angle de tangage : fonction de sensibilité S ainsi que la pondération $1/W_{\theta_e}$

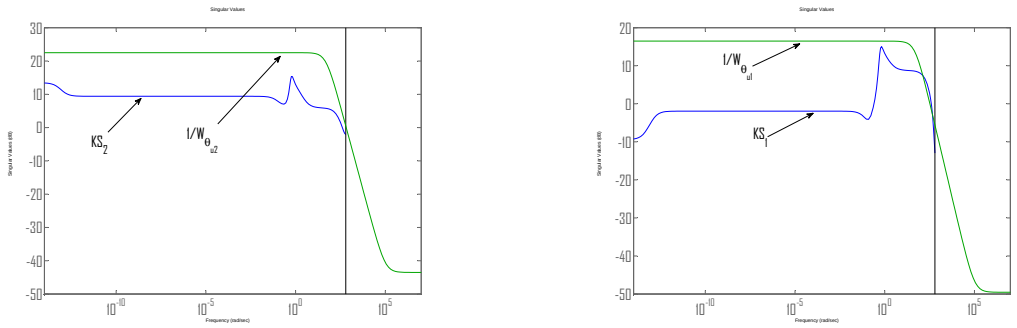

FIGURE 4.6 – Régulateur de l'angle de tangage : fonction de sensibilité KS ainsi que les pondérations, pour chacune des 2 entrées (en temps discret)

On obtient un niveau d'atténuation $\gamma_{\theta_{opt}} = 5.41$ et les fonctions de sensibilité S et KS présentées sur les figures 4.5 et 4.6. Le niveau d'atténuation $\gamma_{\theta_{opt}} = 5.41$ est supérieur à 1 ce qui montre que les gabarits fixés pour la synthèse n'ont pas été respectés. On le vérifie sur la figure 4.5, où la fonction de sensibilité dépasse le gabarit entre 0.3 et 5 rad/s. Cependant le gabarit a été choisi de façon trop restrictive, pour pouvoir atteindre les objectifs de commande malgré une valeur de γ supérieur à 1 (pour un gabarit moins restrictif, $\gamma_{\theta_{opt}} > 1$ mais ainsi les objectifs ne sont pas atteints), donc le résultat de la synthèse est convenable. De plus, pour la fonction de sensibilité KS présentée sur la figure 4.6, on peut voir que les deux gabarits sont respectés par chacune des commandes β_1 et β_2.

4.3.2 Synthèse du correcteur d'altitude $K_z(h)$: approche LPV polytopique discrète

Le correcteur d'altitude est calculé à partir d'un modèle LPV polytopique de l'AUV $Asteri^X$ obtenu en suivant la méthode présentée dans la section 2.3, pour tenir compte dans la conception du régulateur des mesures du capteur à ultrason reçues de façon apériodique.

4.3.2.1 Présentation de la méthodologie de synthèse

Là encore le problème de sensibilité mixte de la figure 1.3 est considéré, avec l'utilisation de deux fonctions de pondération :

- la première sur l'erreur de poursuite, $W_{z_e}(h)$ dépendante de la période d'échantillonnage (ce qui permet l'adaptation des performances, comme expliqué dans Robert et al. (2010)). Elle est définie par un filtre du premier ordre en temps continu, avec une bande passante dépendant de $f = \frac{1}{h}$ suivant :

$$W_{z_e}(h) = \frac{\frac{s}{M_{S_{z_e}}} + w_{S_{z_e}} f}{s + w_{S_{z_e}} f \epsilon_{S_{z_e}}} \tag{4.8}$$

  où $M_{S_{z_e}} = 2$, $w_{S_{z_e}} = 0.0001$ et $\epsilon_{S_{z_e}} = 0.1$. Cette fonction est ensuite réécrite sous forme canonique commandable puis discrétisée (suivant la méthode exposée dans la sous-section 2.3.2) de façon à s'intégrer dans le schéma de synthèse de contrôleur discret : pendant cette phase, une simplification entre la période et la fréquence amène à une pondération exprimée par la fonction LTI de l'équation 4.9 qui peut être connectée au système LPV.

$$\hat{W}_{z_e}(z) : \begin{cases} x_{k+1} = \hat{A}_{W_{z_e}} x_k + \hat{B}_{W_{z_e}} u_k \\ y_k = x_k + u_k \end{cases} \tag{4.9}$$

– la seconde sur la commande W_{z_u} constante, et définie directement en temps discret :

$$W_{z_u} = 0.1 \tag{4.10}$$

Ces deux pondérations discrètes (et LTI) sont ensuite connectées au système $\hat{G}_z(h)$ (de l'équation 4.2) discret et sous forme LPV polytopique par rapport au paramètre δ (on rappelle que δ est la variation de la période d'échantillonnage par rapport à sa valeur nominale h_0 : $h = h_0 + \delta$, avec $h_0 = 0.05s$) pour former le système augmenté. Dès lors la synthèse de contrôleur LPV polytopique en temps discret présentée à la section 1.3 peut être appliquer : on rappelle que l'intervalle de période d'échantillonnage considéré pour la synthèse est $h \in [0.05;\ 0.3]s$.

La synthèse permet d'obtenir un niveau d'atténuation $\gamma_{z_{opt}} = 1.09$ et les fonctions de sensibilité S et KS données sur les figures 4.7, 4.8.

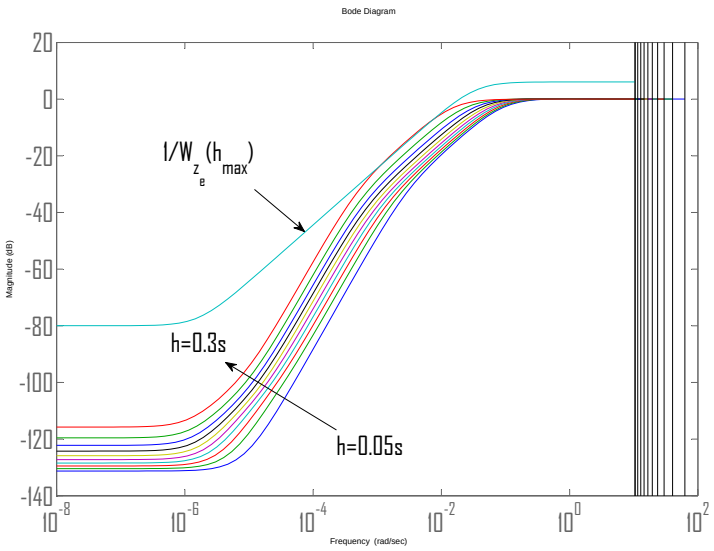

FIGURE 4.7 – Fonction de sensibilité S du correcteur LPV polytopique pour l'altitude, et fonction de pondération correspondante (en temps discret)

4.3.2.2 Analyse des résultats fréquentiels de la synthèse

La valeur du $\gamma_{z_{opt}}$ est très proche de 1, ce qui signifie que les performances spécifiées pour la synthèse sont presque respectées (ce qui est vérifié sur le tracé des fonctions de sensibilité).

Les fonctions de sensibilité sont tracées pour 10 valeurs gelées de la période d'échantillonnage à l'intérieur de l'intervalle considéré pour la synthèse. Le gabarit

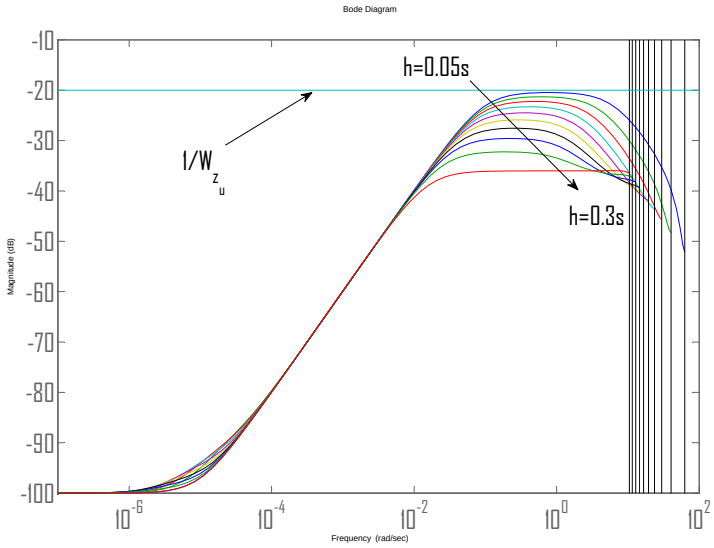

FIGURE 4.8 – Fonction de sensibilité KS du correcteur LPV polytopique pour l'altitude, et fonction de pondération correspondante (en temps discret)

$\frac{1}{\hat{W}_{ze}}$ est également tracé sur la même figure (en temps discret, ce gabarit est une fonction LTI). On peut voir que la fonction S respecte quasiment le gabarit (très léger dépassement autour de 3.10^{-3} rad/s. De plus on vérifie l'adaptation des performances en boucle fermée réalisée grâce à la fonction de pondération variable avec la fréquence, puisque la bande passante de la fonction de sensibilité S, qui est liée au temps de réponse en boucle fermée varie avec la période. Plus la période d'échantillonnage augmente, plus la bande passante diminue et plus le temps de réponse en boucle fermée augmente.

La fonction de sensibilité KS, présentée sur la figure 4.8 permet de vérifier le respect de la pondération sur les actionneurs. Elle met également en évidence l'adaptation des performances, avec la variation de période d'échantillonnage, car le gain de KS diminue si la période augmente.

4.3.2.3 Résultat de simulation temporelle

Le schéma de simulation global, représenté sur la figure 4.1, et utilisant les régulateurs K_u, K_θ et $K_z(h)$, est mise en œuvre sous *Matlab*, en utilisant le modèle non-linéaire complet (12 variables d'état) du sous-marin.

La référence de vitesse est constante et égale à $1m/s$ tout au long de la simulation. La référence d'altitude est un échelon filtré par une fonction du deuxième

ordre (pour éviter un changement trop brutal) et d'amplitude $5m$. Aucune référence pour l'angle de tangage n'est spécifiée, puisque c'est le contrôleur d'altitude qui la calcule à partir de la référence et de la mesure d'altitude. Pendant la simulation, une variation de période de type sinusoïdale est effectuée, entre 0.05 et 0.3 s.

Les résultats de simulations sont présentés sur les figures 4.9, 4.10, 4.11, 4.13 et 4.12.

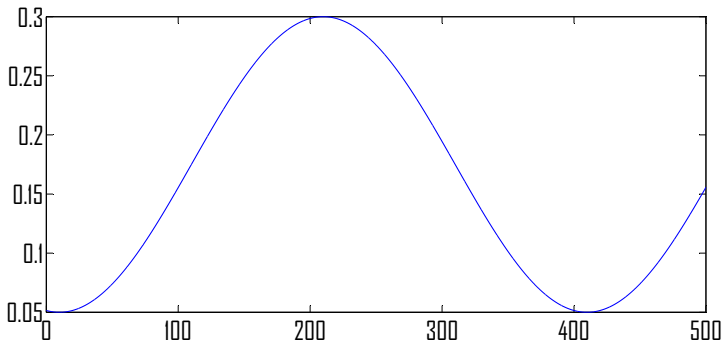

FIGURE 4.9 – Référence de période d'échantillonnage $h = h_0 + \delta$

Les résultats obtenus sont satisfaisants, puisque le système reste stable, avec globalement de bonnes performances. En particulier le suivi d'altitude est réalisé avec un temps de réponse satisfaisant, malgré un comportement légèrement oscillatoire au début de la simulation. La figure 4.10 montre également l'adaptation des performances par rapport à la valeur courante du pas d'échantillonnage : pour le premier échelon (positif) d'altitude, vers $t = 10s$, la période d'échantillonnage h est à son minimum, et la réponse pour le suivi d'altitude est rapide, plus que pour le second échelon (négatif) vers $t = 200s$ (où la période est plus grande).

Le suivi de référence pour l'angle de tangage est également réalisé, avec de bonnes performances (temps de réponse rapide et pas d'erreur statique).

Sur la figure 4.12, excepté au début de la simulation, où la vitesse d'avance présente une légère oscillation, elle reste parfaitement constante et quasiment égale à $1m/s$ (erreur statique de 0.5%) malgré les variations d'altitude et d'angle de tangage.

Bien que l'ensemble des correcteurs ait été synthétisé sur des modèles linéarisés et simulés sur le modèle non-linéaire, la robustesse apportée par le choix de la méthode de synthèse, basée sur la méthodologie $\mathcal{H}_\infty$permet d'obtenir la stabilité du système, avec des performances satisfaisantes.

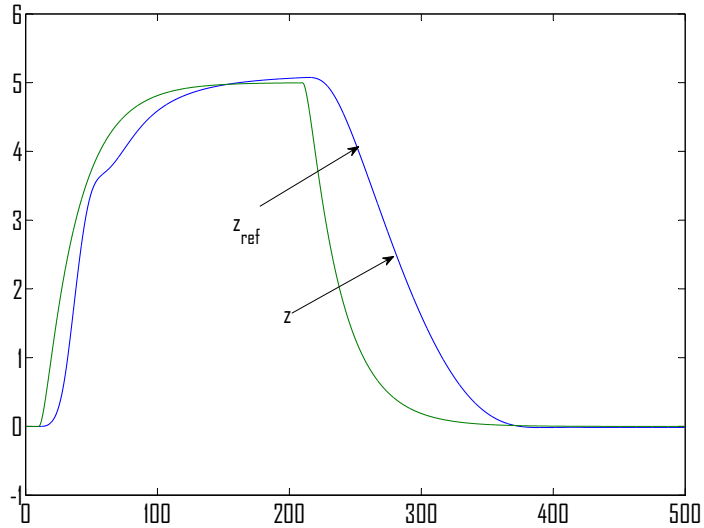

FIGURE 4.10 – Référence et mesure de l'altitude z

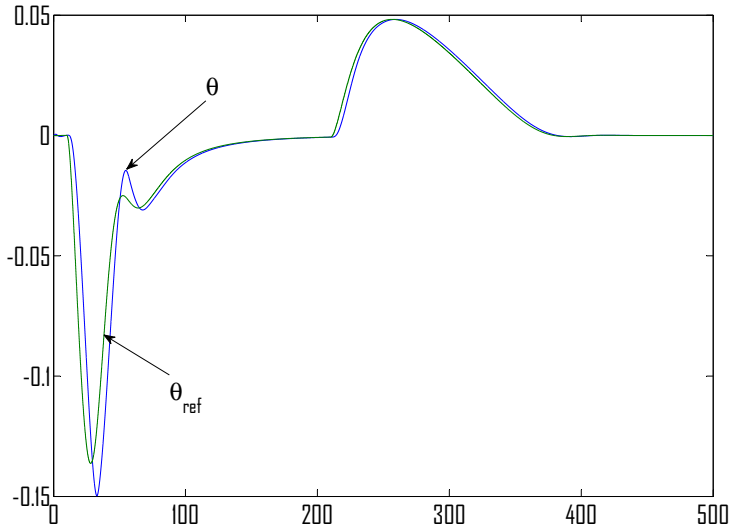

FIGURE 4.11 – Référence et mesure de l'angle de tangage θ

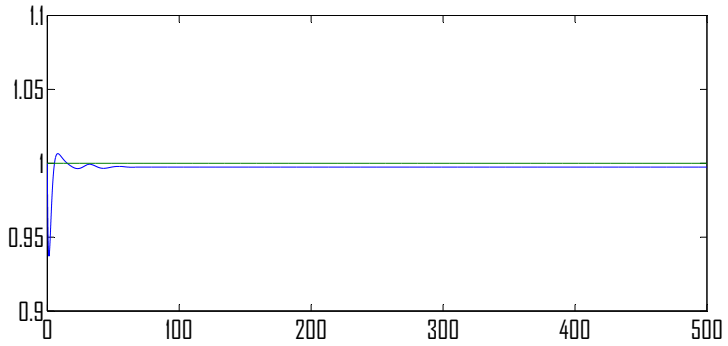

FIGURE 4.12 – Référence et mesure de la vitesse d'avance u

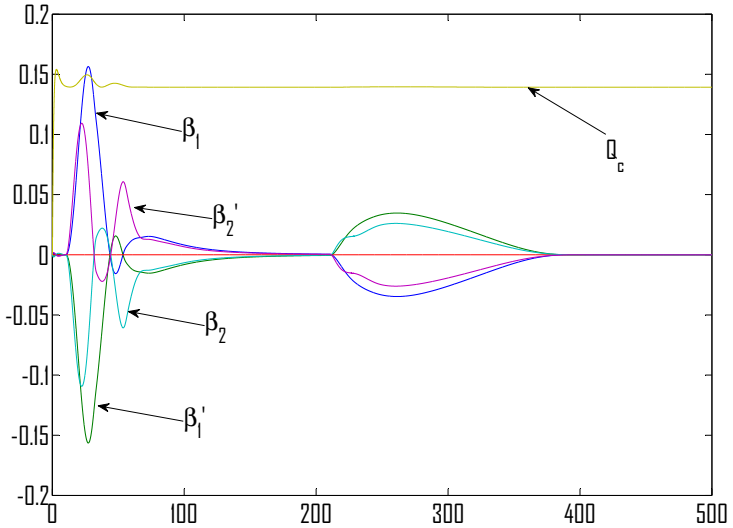

FIGURE 4.13 – Commandes appliquées sur le véhicule

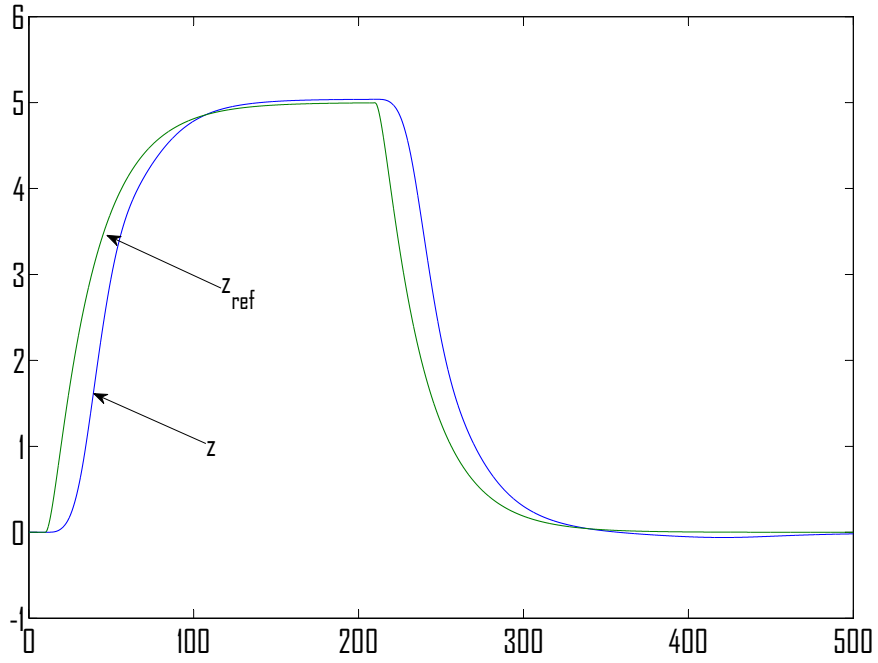

FIGURE 4.14 – Référence et mesure de l'altitude z

4.3.2.4 Remarque : en utilisant une fonction de pondération W_e constante

La synthèse du contrôleur d'altitude LPV dépendant de la période d'échantillonnage a également été testée avec une fonction de pondération sur l'erreur de poursuite constante, c'est à dire sans adaptation des performances de la boucle fermée à la valeur courante de la période.

Pour cela, la fonction de pondération est définie en temps continu, puis connectée au système continu avant la phase de discrétisation, où la période d'échantillonnage est conservée comme paramètre. La même méthode de synthèse est appliquée que pour le cas précédent.

Les résultats sont présentés sur les figure 4.14, 4.15, 4.16.

Même sans adaptation des performances, le suivi de l'altitude et de l'angle de tangage est réalisé avec de bonnes performances. Contrairement aux travaux présentés dans Robert et al. (2010), où le même type de lois de commande est appliqué à un pendule inversé et où l'absence d'adaptation des performances mène à l'instabilité du système, ce problème n'est pas rencontré pour le contrôle d'altitude de l'AUV. Le sous-marin autonome semble être un véhicule assez lent pour

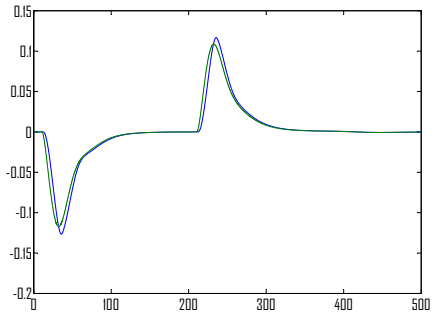

FIGURE 4.15 – Référence et mesure de l'angle de tangage θ

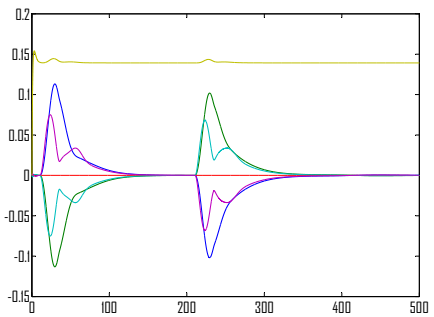

FIGURE 4.16 – Commande appliquée sue le véhicule

que les problèmes de stabilité n'apparaissent pas. Toutefois, par la suite, on ne considérera que des fonctions de pondérations dépendant de la période d'échantillonnage, puisque la simplification période-fréquence permet des simplifications intéressantes.

4.3.3 Synthèse LFR discrète pour le correcteur d'altitude $K_z(h)$

La seconde approche étudiée dans ce manuscrit pour la synthèse de correcteur LPV est la forme LFR, qui est appliquée ici à la synthèse du contrôleur d'altitude à période d'échantillonnage variable.

Le même modèle de départ est considéré, c'est à dire l'intégrateur représentant la relation géométrique entre l'altitude et l'angle de tangage. Le correcteur d'altitude est calculé à partir d'un modèle LPV mis sous la forme d'une LFR de l'AUV $Asteri^X$ obtenu en suivant la méthode présentée dans la section 2.5. La version approchée de la méthode (voir 2.5.2) est utilisée, puisqu'elle permet de réduire le conservatisme et la complexité de l'approche, avec un dévéloppement en série de Taylor (voir équations 2.37 et 2.38) à l'ordre $l = 1$.

Le problème de sensibilité mixte est une nouvelle fois considéré, avec les deux pondérations :

- la première sur l'erreur de poursuite, $W_{z_e}(h)$. Comme pour le cas polytopique, elle est définie en temps continu et dépendante de la période d'échantillonnage par un filtre du premier ordre suivant :

$$W_{z_e}(h) = \frac{\frac{s}{M_{S_{ze}}} + w_{S_{ze}} h}{s + w_{S_{ze}} h \epsilon_{S_{ze}}} \tag{4.11}$$

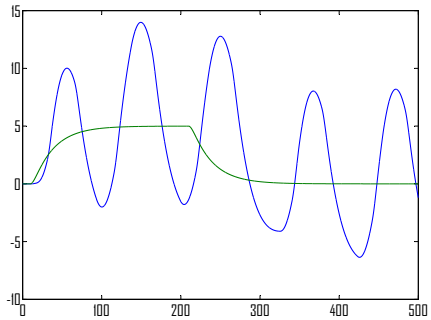

FIGURE 4.17 – Altitude

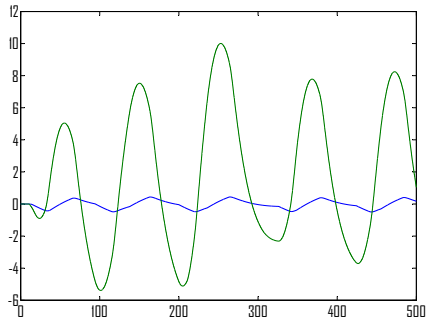

FIGURE 4.18 – Tangage

où $M_{S_{ze}} = 2$, $w_{S_{ze}} = 5$ et $\epsilon_{S_{ze}} = 0.001$. Puis cette fonction est discrétisée pour être connectée au système.

– la seconde sur la commande W_{z_u} constante, et définie directement en temps discret :

$$W_{z_u} = 5 \tag{4.12}$$

La simulation est effectuée dans les mêmes conditions que pour le cas polytopique, avec le schéma de simulation global, représenté sur la figure 4.1, et en remplaçant le régulateur d'altitude polytopique par le régulateur LFR (les contrôleurs pour la vitesse d'avance et l'angle de tangage sont les mêmes que dans le cas précédent). La simulation du correcteur d'altitude sous la forme d'un LFR mène à un résultat instable présenté sur le figures 4.17 et 4.18. Le seul élément modifié par rapport aux résultats précédents est le correcteur d'altitude, qui est forcement en cause. Le résultat reste instable en refaisant la synthèse du contrôleur d'altitude avec une valeur de γ non optimale, par exemple pour $\gamma = 7$ (ce qui permet de réduire les problèmes numériques, et d'améliorer le conditionnement des matrices servant au calcul du contrôleur : M et N de l'équation (1.27)). La même simulation avec un scénario de simulation plus favorable, en imposant une période d'échantillonnage constante pendant la simulation, mène également à l'instabilité du système.

La synthèse de ce correcteur d'altitude sous forme LFR a été également essayée en utilisant la forme exacte présentée à la sous-section 2.5.1. La encore la synthèse aboutit, avec un niveau d'atténuation optimal $\gamma_{opt} = 5.38$, mais la simulation mène à l'instabilité du système.

4.3.3.1 Conclusion sur l'approche LTI

Dans cette section, la structure cascade présentée dans la section 3.3 est appliquée au contrôle de l'altitude. Le modèle LTI réduit au plan vertical du sous-marin

est utilisé pour la synthèse du régulateur de tangage (synthèse $\mathcal{H}_\infty$discrète classique) et un modèle géométrique liant l'altitude z à l'angle de tangage θ pour la synthèse du régulateur d'altitude. Ce second contrôleur est de type LPV, dépendant de la période d'échantillonnage, soit en utilisant l'approche polytopique, soit l'approche LFR.

Les résultats obtenus dans le cas polytopique sont satisfaisants, malgré quelques oscillations, mais globalement les performances obtenues sont convenables, et l'adaptation des performances, voulues par la fonction de pondération dépendante de la période d'échantillonnage, est réalisée.

Dans le cas LFR, bien que la synthèse du contrôleur d'altitude $K_z(h)$ soit réalisée avec un niveau d'atténuation γ_{opt} convenable, la simulation (sur le modèle complet) mène à l'instabilité du système. Puisque la boucle interne de contrôle de l'angle de tangage (LTI) est identique au cas précédent, le problème vient bien du correcteur d'altitude. Nous pensons que ce problème vient du fait que le comportement de la boucle interne n'est pas assez proche de celui d'un intégrateur (peut être à cause d'un mauvais asservissement de l'angle de tangage par le contrôleur K_θ). L'approche polytopique doit apporter une robustesse supplémentaire permettant la stabilisation du système.

Nous allons résoudre ce problème en utilisant une deuxième approche, qui permet de garder certaines non-linéarités dans le modèle continu et donc de considérer un modèle de synthèse des contrôleurs plus proche du système non-linéaire. La section suivante décrit donc les résultats obtenus pour la conception d'une loi de commande à échantillonnage variable, pour les systèmes LPV.

4.4 Contrôle d'un système LPV à période variable

L'extension de la méthode de synthèse de contrôleur avec période d'échantillonnage variable aux systèmes LPV présentée dans les sections 2.4 pour l'approche polytopique et 2.6 pour la LFR est maintenant testée dans le cadre du contrôle d'altitude du sous-marin $Asteri^X$. La même structure globale de contrôle est utilisée (voir figure 4.1), avec le contrôleur de vitesse d'avance K_u, et la structure cascade pour la contrôle d'altitude.

Dans cette structure, le correcteur d'altitude LPV $K_z(h)$ (soit polytopique soit LFR) dépendant de la période d'échantillonnage est conservé. La nouvelle méthode est appliquée pour la synthèse du régulateur de l'angle de tangage. Le modèle LPV dérivé des équations 3.27, avec comme paramètres variants $\cos(\theta_{eq})$ et $\sin(\theta_{eq})$ est utilisé.

Remarque : Les deux sections suivantes présentent le contrôle d'altitude de l'AUV dans un premier temps avec une structure entièrement polytopique (contrôleur de l'angle de tangage et d'altitude LPV polytopique) puis avec une structure

entièrement LFR.

4.4.1 Structure cascade polytopique

Dans cette section, le contrôle d'altitude K_z polytopique obtenu précédemment (voir 4.3.2) est utilisé pour la boucle externe, et la méthode de synthèse d'un contrôleur à période d'échantillonnage variable pour un système LPV est mise en place pour la boucle interne de la structure cascade (le contrôle de l'angle de tangage K_θ).

4.4.1.1 Contrôleur de l'angle de tangage $K_\theta(h, \rho)$: synthèse Polytopique continue

La méthode proposée dans la section 2.3, est mise en application sur le modèle LPV réduit du sous-marin, pour le contrôle de l'angle de tangage θ. Comme expliqué auparavant, la synthèse du contrôleur est faite en temps continu, à partir du modèle LPV polytopique du système dépendant des paramètres $cos(\theta_{eq})$ et $sin(\theta_{eq})$ augmenté de fonctions de pondération dépendant du paramètre $\tilde{h}$ (qui est lié à la période d'échantillonnage).

Le problème de sensibilité mixte est considéré, avec les fonctions de pondération suivantes :

- W_{θ_u} sur le vecteur des commandes $[\beta_1,\ \beta_2]$ qui est définie par une constante (différente pour les 2 entrées, ce qui permet de rééquilibrer l'utilisation des 2 paires d'ailerons) :

$$W_{\theta_u} = \begin{pmatrix} 0.75 & 0 \\ 0 & 1.5 \end{pmatrix} \tag{4.13}$$

- W_{θ_e} sur l'erreur de poursuite $\theta - \theta_{ref}$, qui est définie par une fonction du premier ordre, donc la bande passante dépend du paramètre $\tilde{h}$, suivant :

$$W_{\theta_e}(\tilde{h}) = \frac{\frac{s}{M_{S_{\theta_e}}} + w_{S_{\theta_e}}\tilde{h}}{s + w_{S_{\theta_e}}\tilde{h}\epsilon_{S_{\theta_e}}} \tag{4.14}$$

  où $M_{S_{\theta_e}} = 2$, $w_{S_{\theta_e}} = 10$, $\epsilon_{S_{\theta_e}} = 0.0001$ et $\tilde{h}$ est un paramètre variant dans l'intervalle $[\underline{\tilde{h}},\ \overline{\tilde{h}}] = [0.005,\ 0.03]$. La boucle interne de la structure cascade (contrôle de l'angle de tangage) aura donc une période d'échantillonnage 10 fois plus rapide que la boucle externe (contrôle d'altitude, dont la période d'échantillonnage varie est 0.05 et 0.3 secondes).

La connexion du système LPV polytopique dépendant des paramètres $\cos(\theta_{eq})$ et $\sin(\theta_{eq})$ avec les fonctions de pondérations $W_{\theta_e}(\tilde{h})$ et W_{θ_u} permet d'obtenir un

modèle polytopique continu de la forme suivante :

$$P \ : \ \begin{cases} \dot{x}(t) = A(cos(\theta_{eq}), sin(\theta_{eq}), \tilde{h})x(t) + B(cos(\theta_{eq}), sin(\theta_{eq}), \tilde{h})w(t) \\ z(t) = C(cos(\theta_{eq}), sin(\theta_{eq}), \tilde{h})x(t) + D(cos(\theta_{eq}), sin(\theta_{eq}), \tilde{h})w(t) \end{cases} \tag{4.15}$$

La synthèse du contrôleur est réalisée en continu à partir de ce modèle polytopique, en utilisant la méthode présentée dans la section 1.3, basée sur la synthèse $\mathcal{H}_\infty$. La résolution du problème d'optimisation permet d'obtenir un niveau d'atténuation $\gamma_{opt} = 1.39$.

4.4.1.2 Analyse fréquentielle du correcteur de tangage K_θ LPV polytopique

Les fonctions de sensibilité S et KS sont tracées pour 8 valeurs figées de paramètres, voir figure 4.19 et 4.20.

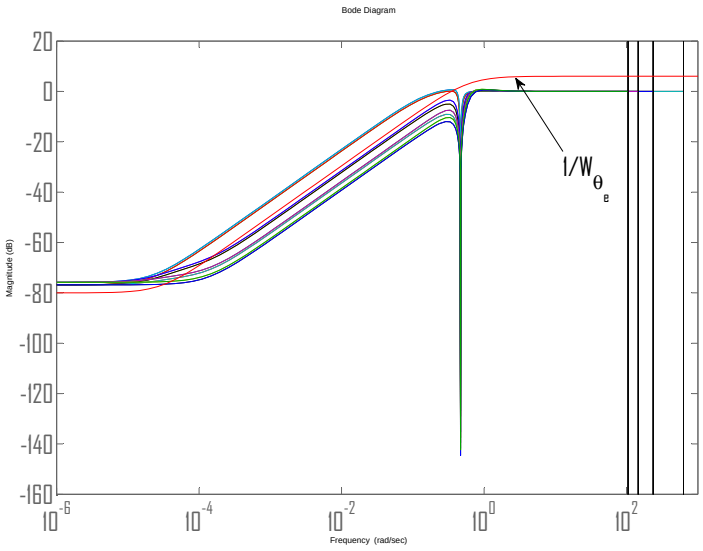

FIGURE 4.19 – Fonction de sensibilité S du correcteur LPV polytopique pour l'angle de tangage, et fonction de pondération correspondante (en temps discret)

La fonction de sensibilité S montre un léger dépassement du gabarit (celui tracé correspond aux valeurs maximales pour chacun des paramètres), mais globalement les performances souhaitées sont respectées. Les deux fonctions de sensibilité KS ont également le profil souhaité, avec un filtrage du bruit de mesure en haute fréquence.

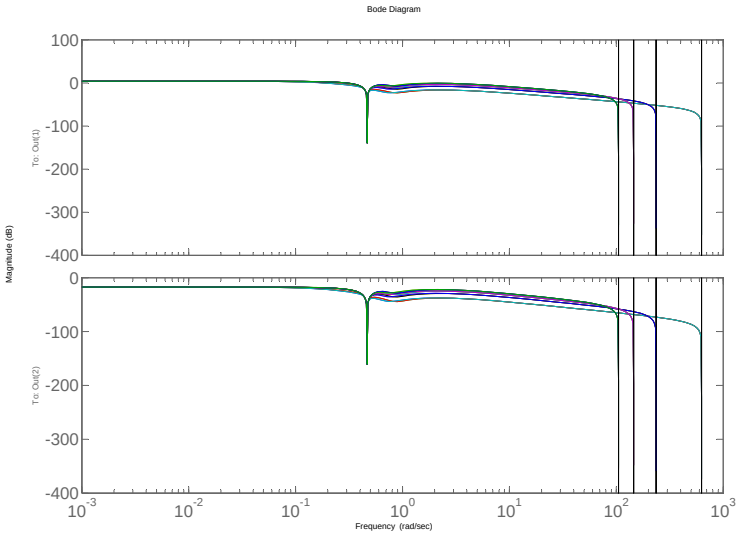

FIGURE 4.20 – Fonction de sensibilité KS du correcteur LPV polytopique pour l'angle de tangage(en temps discret)

A l'issue de la synthèse, on obtient un ensemble de 8 correcteurs correspondant à chacun des sommets du polytope. La combinaison convexe en continu de ces correcteurs sera effectuée lors de la simulation pour s'adapter au point de fonctionnement, puis le correcteur obtenu sera discrétisé (avec la valeur courante de la période d'échantillonnage).

Remarque : Le paramètre $\tilde{h}$ symbolise la période d'échantillonnage pour la pondération $W_{\theta_e}(\tilde{h})$ (et donc pour le système polytopique utilisé pour la synthèse). Ce qui signifie que pour le calcul du correcteur à l'instant k par la combinaison convexe des correcteurs sommets, ce paramètre $\tilde{h}$ est pris égal à la période d'échantillonnage courante h_k.

4.4.1.3 Résultat de simulation temporelle : angle de tangage seul (Cas 1)

Dans un premier temps, le contrôleur de l'angle de tangage est testé seul en simulation, sur le modèle non-linéaire complet de l'AUV. Pour cela, une variation de période d'échantillonnage sinusoïdale est imposée, entre les bornes minimale et maximale choisies pour le paramètre $\tilde{h}$ pour la synthèse $[\underline{\tilde{h}},\ \overline{\tilde{h}}] = [0.005,\ 0.03]$. La mesure de l'angle d'incidence θ est utilisée pour le calcul des paramètres $\cos(\theta_{eq})$ et $\sin(\theta_{eq})$ dont dépend également le correcteur K_θ.

La référence à suivre pour l'angle de tangage est un échelon filtré par un second ordre, avec une amplitude de $0.5rad$, comme présenté sur la figure 4.22.

Les résultats de l'asservissement de l'angle θ (correspondant à la boucle interne de la structure cascade) sont présentés sur les figures 4.21, 4.22 et 4.23.

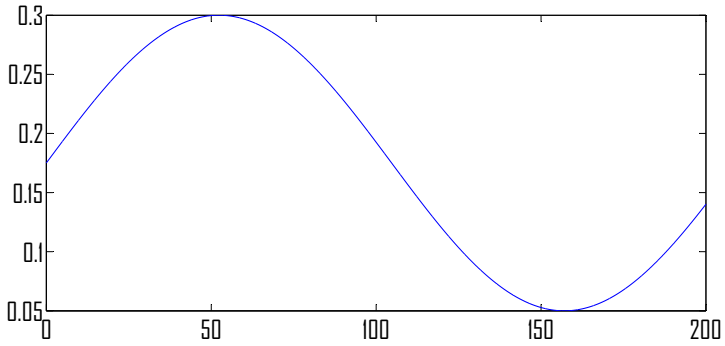

FIGURE 4.21 – Cas 1 : Référence de période d'échantillonnage h

Les résultats obtenus ne sont pas de très grande qualité. Bien que le stabilité du système soit préservée, des oscillations importantes, en particulier sur la fin de la simulation, où la période d'échantillonnage est faible, mènent à de mauvais résultats pour la suivi d'angle de tangage. Toutefois il est à noté que l'angle de tangage donné en référence ($0.5rad$) est à la limite de ce qui peut être atteint par le véhicule. On remarque d'ailleurs que les commandes entrent en saturation à plusieurs reprises au cours de la simulation (voir sur la figure 4.23).

4.4.1.4 Résultat de simulation temporelle : structure cascade complète pour le contrôle d'altitude (Cas 2)

Le même correcteur est ensuite inséré dans le schéma global de simulations, qui reprend le correcteur de vitesse d'avance constante de la section 4.2 et le contrôleur d'altitude LPV polytopique paramétré par la période d'échantillonnage de la sous-section 4.3.2. La référence d'altitude est la même que précédemment, c'est à dire un échelon de $5m$ d'amplitude filtré par un second ordre. Les résultats sont présentés sur les figure 4.24 4.25 4.26 4.29 4.27.

Le suivi d'altitude est réalisé avec de bonnes performances : on constate une amélioration par rapport au cas LTI de la section précédente, en terme de temps de réponse, et plus aucunes oscillations ne sont visibles. Le suivi de référence de l'angle de tangage est réalisé, avec des performances satisfaisantes.

On remarque également que les commandes sont assez peu utilisées, avec une amplitude assez faible ($0.05\ rad$) soit une utilisation de 20% de leur plage de

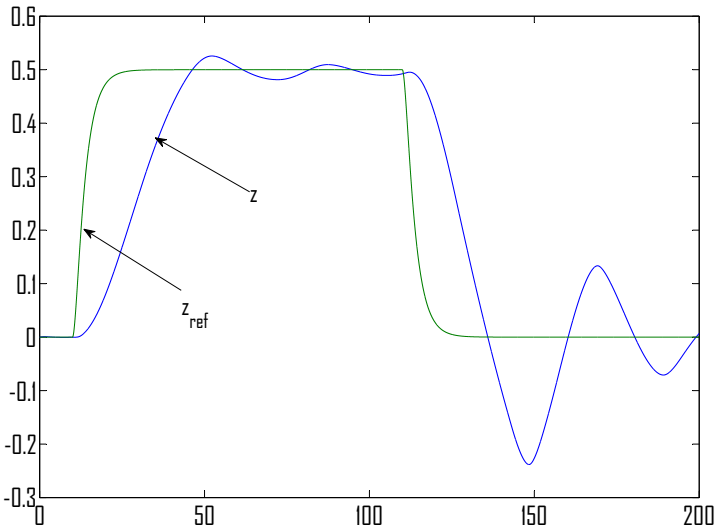

FIGURE 4.22 – Cas 1 : Référence et mesure de l'angle de tangage θ

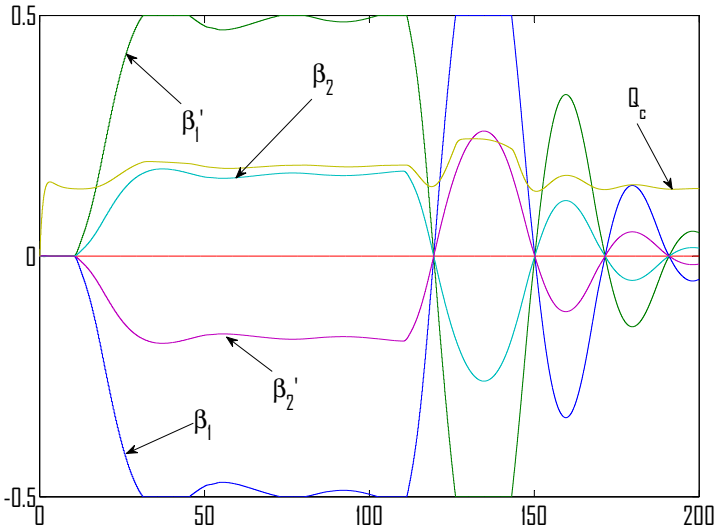

FIGURE 4.23 – Cas 1 : Commande calculées par le régulateur et appliquée au véhicule

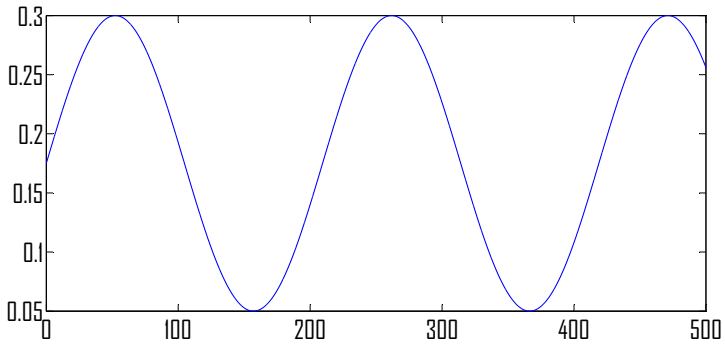

FIGURE 4.24 – Cas 2 : Référence de période d'échantillonnage h

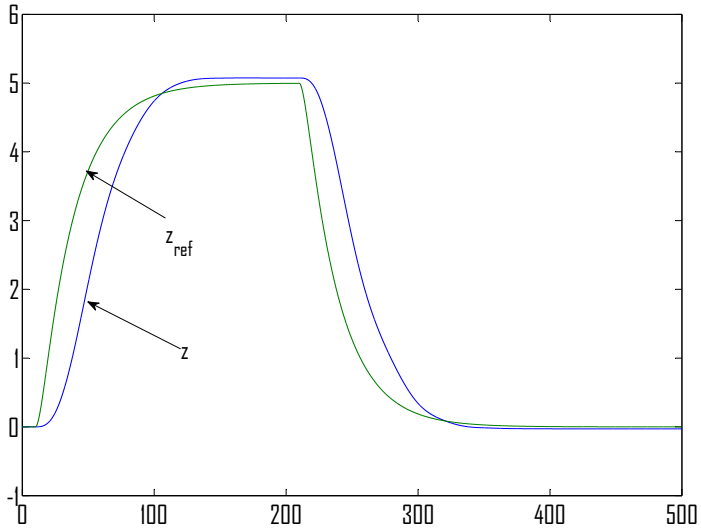

FIGURE 4.25 – Cas 2 : Référence et mesure de l'altitude z

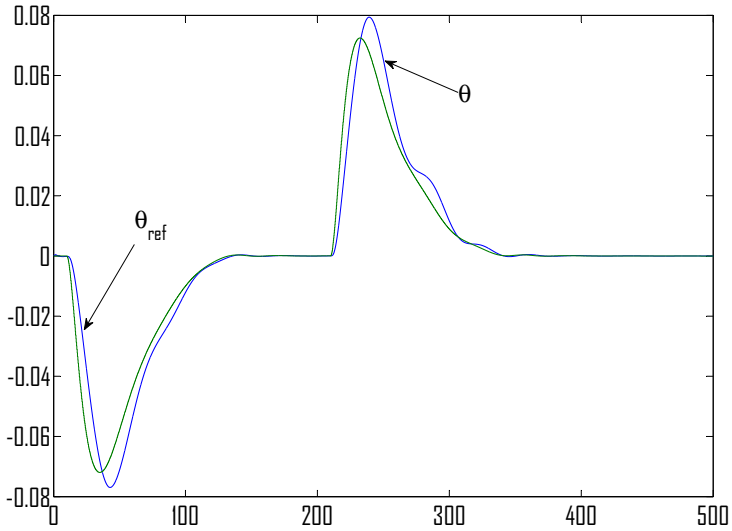

FIGURE 4.26 – Cas 2 : Référence et mesure de l'angle de tangage θ

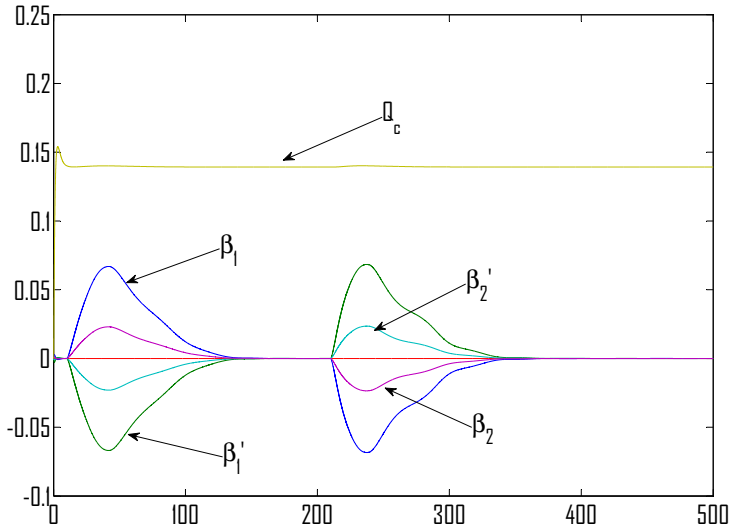

FIGURE 4.27 – Cas 2 : Commandes appliquées sur le véhicule

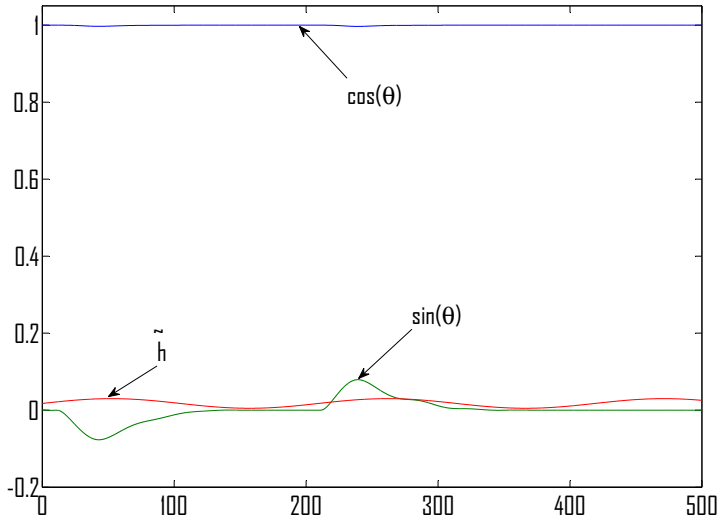

FIGURE 4.28 – Cas 2 : Evolution des paramètres au cours de la simulation

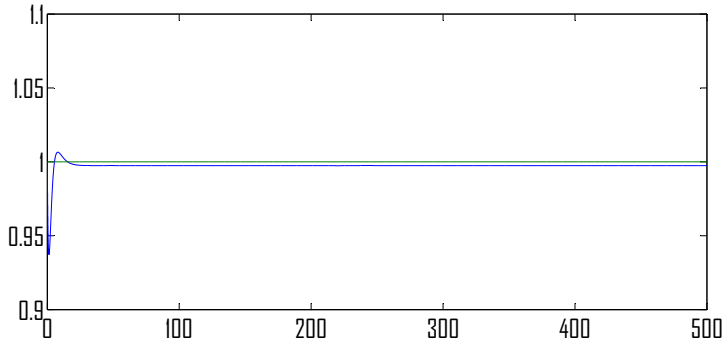

FIGURE 4.29 – Cas 2 : Référence et mesure de la vitesse d'avance u

fonctionnement (ce qui est une bonne nouvelle pour le contrôle global du véhicule, puisque ces ailerons seraient également utilisés pour la commande dans le plan horizontal).

Il est à noter que les résultats présentés dans cette thèse sont les meilleurs obtenus suite à de nombreux essais. L'utilisation relativement faible des actionneurs montre qu'il doit être possible d'améliorer ces performances, cependant des problèmes numériques apparaissent lorsque l'on modifie les pondérations pour aller dans ce sens.

Cela montre bien l'intérêt de l'approche polytopique dans ce cadre par rapport à la synthèse LTI.

4.4.2 Structure cascade LFR

Dans cette section, nous avons choisi de conserver le contrôle d'altitude LFR $K_z(h)$ synthétisé dans la section 4.3.3, utilisé pour la boucle externe du contrôle cascade de l'altitude (malgré des résultats instables lors des simulations temporelles). Le contrôleur de l'angle de tangage de la boucle interne est ici calculée en utilisant la méthode LFR pour les systèmes LPV développée dans cette thèse.

4.4.2.1 Contrôle de l'angle de tangage $K_\theta(h,\rho)$: synthèse LFR discrète

Le correcteur de l'angle de tangage LPV est maintenant synthétisé en utilisant la deuxième méthode présentée : la Representation Linéaire Fractionnelle (LFR). A partir du modèle LPV continu réduit du sous-marin, dépendant des paramètres $\cos(\theta_{eq})$ et $\sin(\theta_{eq})$, le modèle discret et dépendant en plus de la période d'échantillonnage est construit sous la forme d'une LFR, suivant les équations 2.49 à 2.52. La synthèse du régulateur de l'angle de tangage est réalisée suivant l'approche LFR pour les systèmes LPV présentée dans la section 1.4. Le problème de sensibilité mixte est considéré, avec les pondérations suivantes :

- W_{θ_u} sur le vecteur des commandes $[\beta_1,\ \beta_2]$ qui est définie par une constante (contrairement au cas précédent, les meilleurs résultats sont obtenus en pondérant les deux actionneurs avec la même constante), suivant :

$$W_{\theta_u} = \begin{pmatrix} 0.8 & 0 \\ 0 & 0.8 \end{pmatrix} \tag{4.16}$$

- W_{θ_e} sur l'erreur de poursuite $\theta - \theta_{ref}$, qui est définie par une fonction du premier ordre, donc la bande passante dépend du paramètre $\tilde{h}$, suivant :

$$W_{\theta_e}(h) = \frac{\frac{s}{M_{S_{\theta_e}}} + w_{S_{\theta_e}} h}{s + w_{S_{\theta_e}} h \epsilon_{S_{\theta_e}}} \tag{4.17}$$

où $M_{S_{\theta_e}} = 2$, $w_{S_{\theta_e}} = 10$, $\epsilon_{S_{\theta_e}} = 0.01$ et h la période d'échantillonnage variant dans l'intervalle $[\underline{h},\ \overline{h}] = [0.005,\ 0.03]$ (là encore dix fois plus rapide que la boucle externe de la structure cascade)

On obtient un niveau d'atténuation $\gamma_{opt} = 37.03$. La première LMI dérivant du lemme réel borné appliqué aux systèmes LPV sous la forme d'une LFR permet d'aboutir à ce résultat. Cependant la seconde LMI qui calcul le correcteur n'est pas résolue pour cette valeur optimale de γ. Pour éviter les problèmes numériques, la synthèse est ensuite réalisée pour une valeur sous-optimale du niveau d'atténuation : $\gamma = 39$. Cette fois le correcteur est bien calculé et peut être testé en simulation.

4.4.2.2 Résultat de simulation temporelle : angle de tangage seul (cas 3)

Le correcteur LFT obtenu est testé en simulations pour le contrôle de l'angle de tangage (boucle interne de la structure cascade). Une variation de période d'échantillonnage sinusoïdale est imposée, entre les bornes minimale et maximale $[\underline{h},\ \overline{h}] = [0.005,\ 0.03]$. La mesure de l'angle d'incidence θ est utilisée pour le calcul des paramètres $\cos(\theta_{eq})$ et $\sin(\theta_{eq})$ dont dépend également le correcteur K_θ.

La référence à suivre pour l'angle de tangage est un échelon filtré par un second ordre, avec une amplitude de $0.5rad$.

Les résultats de l'asservissement de l'angle θ sont présentés sur les figures 4.30, 4.31, 4.32 et 4.33.

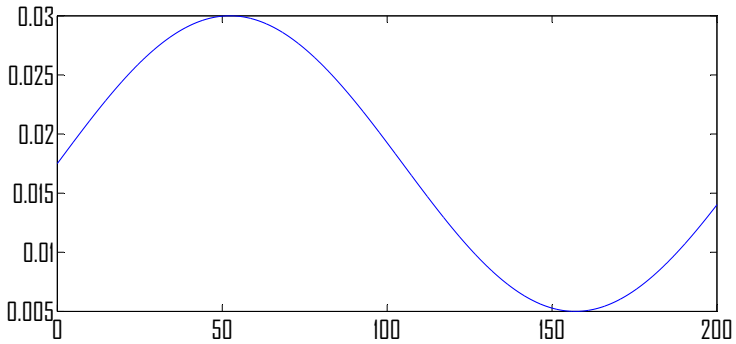

FIGURE 4.30 – Cas 3 : Référence de période d'échantillonnage h

Le suivi de la référence de l'angle de tangage est réalisé, malgré une erreur statique non nulle, de l'ordre de 3%. Le temps de réponse est assez rapide, surtout pour le deuxième changement de référence, où la période d'échantilonnage est alors minimale (ce qui montre l'adaptation des performances). Comme pour le cas

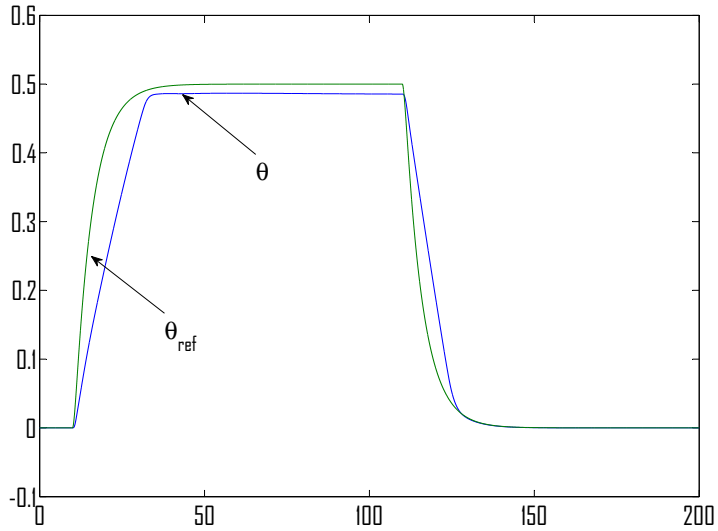

FIGURE 4.31 – Cas 3 : Référence et mesure de l'angle de tangage θ

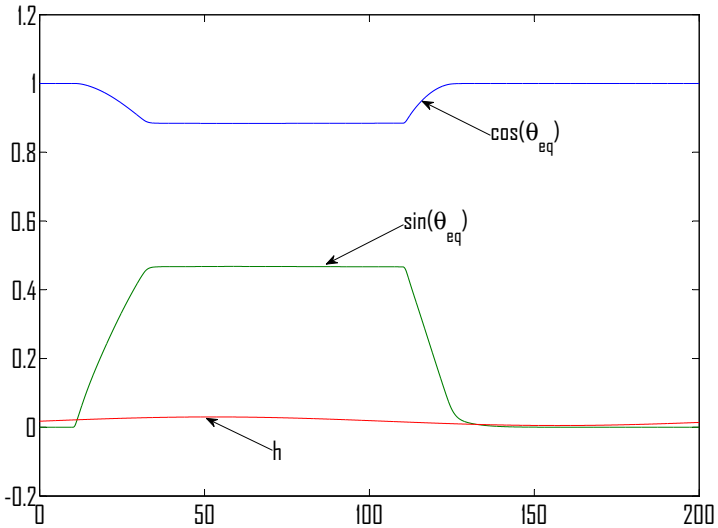

FIGURE 4.32 – Cas 3 : Evolution des paramètres du correcteur LFT

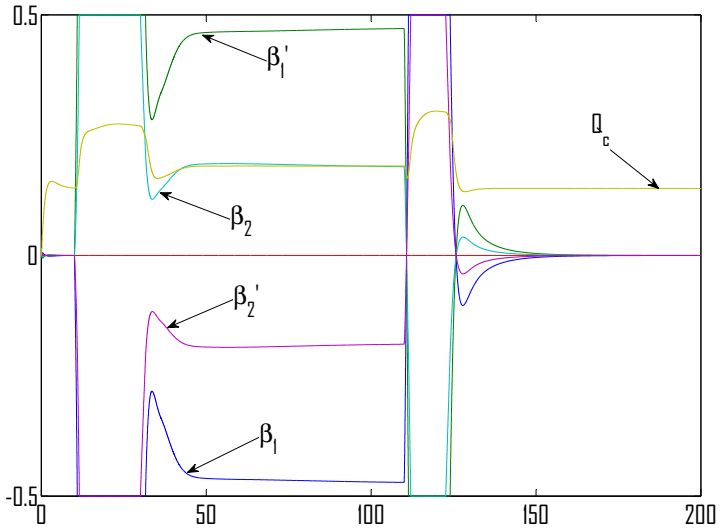

FIGURE 4.33 – Cas 3 : Commande calculées par le régulateur et appliquée au véhicule

polytopique, les commandes sont saturées sur une partie de la simulation, à cause de la valeur importante demandée pour l'angle de tangage ($0.5rad$).

4.4.2.3 Résultat de simulation temporelle :structure cascade LFR complète pour le contrôle d'altitude (cas 4)

Le contrôleur sous forme LFR dépendant des 2 paramètres liés à la linéarisation ($\cos(\theta_{eq})$ et $\sin(\theta_{eq})$) et de la variation de période d'échantillonnage est maintenant testé dans le schéma de simulation global (voir figure 4.1). Le correcteur d'altitude LFR dépendant de la période d'échantillonnage obtenu dans la section 4.3.3 est utilisé pour le contrôle de l'altitude ainsi que le correcteur $\mathcal{H}_\infty$ discret pour la vitesse d'avance.

Un référence d'altitude consistant en un échelon de $5m$ filtré par un deuxième ordre est appliquée. Une variation de période d'échantillonnage sinusoïdale utilisée à la fois pour la correcteur d'altitude (entre 0.05 et $0.3s$) et pour le correcteur d'angle de tangage (entre 0.005 et $0.03s$). Les résultats obtenus sont présentés sur les figures 4.34, 4.35, 4.36, 4.37, 4.38 et 4.39.

Le suivi de référence pour l'altitude ainsi que pour l'angle de tangage est parfaitement réalisé, avec un temps de réponse très rapide et aucune erreur statique.

Après une légère oscillation lors du changement de référence en altitude (à $210s$)

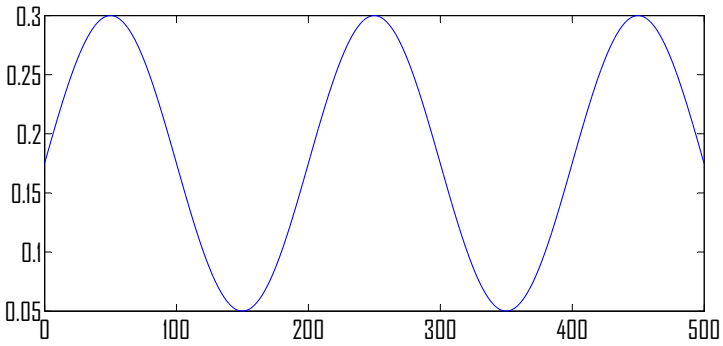

FIGURE 4.34 – Cas 4 : Référence de période d'échantillonnage h

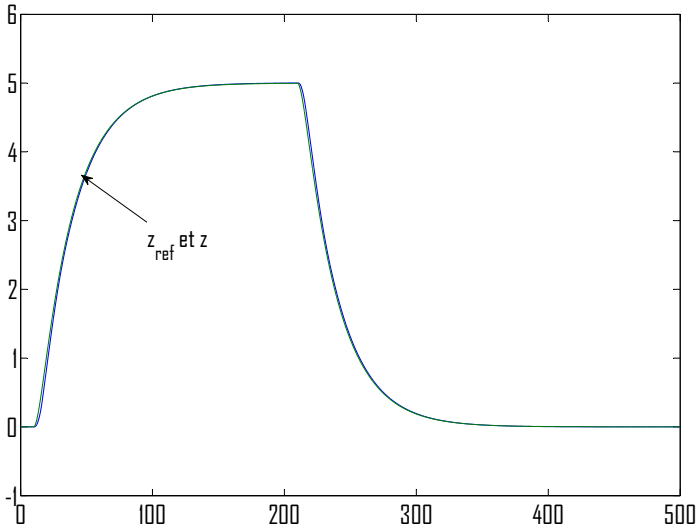

FIGURE 4.35 – Cas 4 : Référence et mesure de l'altitude z

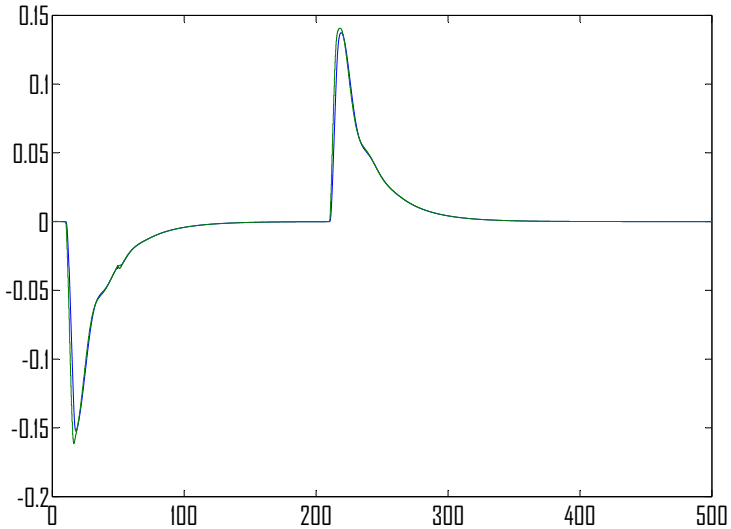

FIGURE 4.36 – Cas 4 : Référence et mesure de l'angle de tangage θ

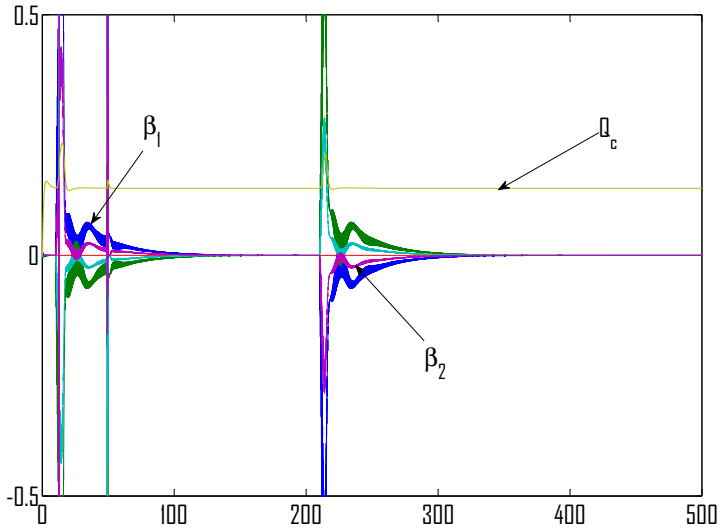

FIGURE 4.37 – Cas 4 : Commandes appliquées au véhicule

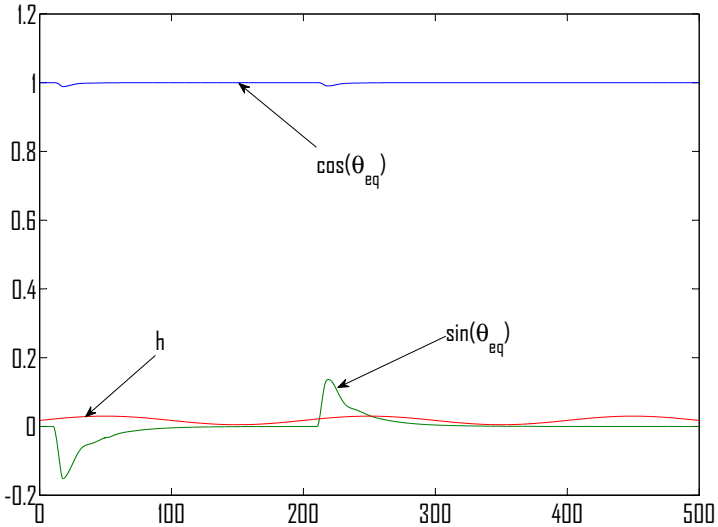

FIGURE 4.38 – Cas 4 : Evolution des paramètres au cours de la simulation

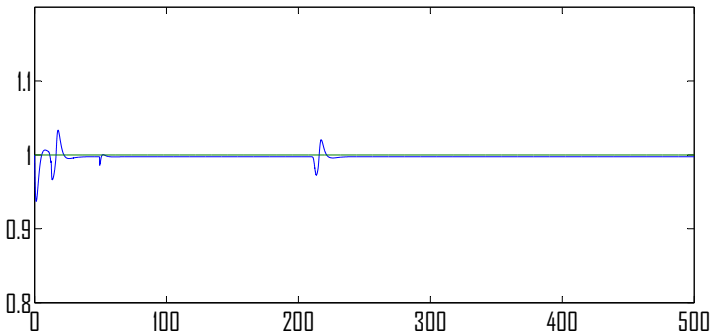

FIGURE 4.39 – Cas 4 : Référence et mesure de la vitesse d'avance u

la vitesse d'avance reste constante et égale à $1m/s$. Les perturbation engendrée par les variation d'altitude et d'angle de tangage sont bien rejetée par ce contrôleur.

Les commandes envoyées aux actionneurs présentées sur la figure 4.37 montrent une saturation au moment du changement de référence (à 10 et 210 secondes) mais elles reviennent à des valeurs plus faibles très rapidement. En revanche un comportement légèrement oscillatoire (trait épais qui correspond à des oscillations très rapide mais de faible amplitude) nuit à l'efficacité énergétique du contrôle.

4.5 Conclusion : comparaison des deux approches

Au vu des résultats obtenus dans les sections 4.3 et 4.4, il est difficile de conclure sur la meilleure technique LPV pour le contrôle d'un véhicule sous marin avec une période d'échantillonnage variable. En effet dans le cas du modèle LTI du sous-marin, la synthèse polytopique donne de bons résultats en simulation, alors que l'approche LFR mène à l'instabilité du système en boucle fermée. Il semblerait donc que dans ce cas, l'approche polytopique apporte un peu plus de robustesse, ce qui permet le contrôle du système malgré les nombreuses approximations et simplifications réalisées sur les modèles de synthèses (que ce soit pour le contrôle de l'altitude ou de l'angle de tangage).

En revanche pour l'extension de cette technique au systèmes LPV (dépendant déjà de paramètres avant la phase de discrétisation, et qui permet d'ajouter la période d'échantillonnage à l'ensemble des paramètres existants) l'approche LFR est plus adaptée du point de vue théorique. En effet la formulation LFR, qui extrait la partie variable du système permet de ramener la discrétisation d'un système LPV sous la forme d'une LFR, à la discrétisation de la partie LTI de ce système. Pour l'approche polytopique, une méthode détournée (en temps continu) a été mise en place, pour permettre la synthèse d'un contrôleur dépendant à la fois des paramètres physiques du système et de la période d'échantillonnage. Les simulations ont permis de confirmer que l'approche LFR est la mieux adaptée, puisqu'elle a permis d'obtenir de très bons résultats. Bien sûr cette méthodologie nécessite une modélisation adéquate du système ; prenant en compte des incertitudes ou des paramètres variants, sous peine d'être moins efficace (comme on l'a vu dans la section 4.3.3).

D'un point de vue des calculs à effectuer pour la synthèse, les deux approches sont également différentes : pour l'approche polytopique, le nombre de LMIs à résoudre augme avec le nombre de paramètres variants (en 2^N, N étant le nombre de paramètres), alors que pour l'approche LFR, le nombre de LMI reste constant, c'est leur taille qui augmente (plus il y a de paramètres, plus la matrice Δ augmente, plus il y a d'entrées et sorties au système). Lors de la synthèse pour le système continu LTI, un seul paramètre variant est considéré, et les deux mé-

thodes permettent d'aboutir sans trop de problème numériques. Dans les deux cas (polytopique et LFR), quand le nombre de paramètres devient important (au delà de trois ou quatre), des difficultés numériques apparaissent lors de la résolution, et notamment pour le calcul du contrôleur à partir de la solution de la première LMI. Relâcher un peu les contraintes en s'éloignant du γ optimal peut permettre dans certain cas de résoudre ce problème (cas de la synthèse LFR pour le système LPV de la sous-section 4.4.2.1).

Les méthodes de synthèse de contrôleur employées dans cette thèse donnent de bons résultats, cependant plusieurs travaux ont été menés récemment pour améliorer ces techniques de synthèse. Citons par exemple l'utilisation de fonctions de Lyapunov dépendant du paramètre variant (Oliveira and Peres, 2006; Wu and Dong, 2006), ou encore l'utilisation de multiplieurs (Scherer, 2001) permettant de réduire le conservatisme. Ces méthodes pourraient montrer leur intérêt dans des cas plus complexes où d'autres paramètres variants sont ajoutés au modèle du système considéré.

Conclusion et perspectives

Le travail présenté dans ce manuscrit concerne le développement d'un contrôleur à période d'échantillonnage variable pour un système continu sous forme LPV. L'application considérée est un véhicule sous-marin autonome.

Le chapitre 1 rappelle les différents outils théoriques utilisés tout au long de cette thèse. Après diverses définitions qui sont utiles pour la compréhension des méthodes employées, la commande robuste $\mathcal{H}_\infty$est présentée, de la définition des pondérations fréquentielles permettant de fixer des performances pour la boucle fermée, jusqu'au calcul du contrôleur, par la résolution de LMI. Puis l'extension de cette méthode pour des systèmes Linéaires à Paramètres Variants est exposée pour deux approches : l'approche polytopique, où le vecteur de paramètres variants évolue dans un polytope, ce qui apporte une propriété de convexité qui permet de réduire l'étude de stabilité (où de synthèse de contrôleur) aux sommets du polytope, puis l'approche LFR, où la dépendance envers le vecteur de paramètres est extraite du reste du système, dans une matrice de forme bloc diagonale, qui permet de séquencer le système (et le contrôleur).

Le chapitre 2 présente la méthode pour obtenir, à partir d'un système continu, un système discret paramétré par la période d'échantillonnage (variable). Dans un premier temps, la méthode est expliquée pour les systèmes continus LTI, pour obtenir un système LPV soit sous forme polytopique, soit sous la forme d'une LFR. Puis cette méthode est étendue pour des systèmes dont le modèle à temps continu dépend déjà d'un ou plusieurs paramètres. Il s'agit alors d'ajouter la période d'échantillonnage au vecteur de paramètres existant. Là encore cette technique est développée pour le cas polytopique et pour le cas LFR.

Le chapitre 3 définit les différents modèles de sous-marins qui ont été considérés tout au long de la thèse. Tout d'abord le modèle non-linéaire complet du véhicule (avec ces douze variables d'état) est présenté ; il est utilisé lors du chapitre 4 pour les simulations. Puis une linéarisation tangentielle autour d'un point de fonctionnement fixe permet d'obtenir un modèle linéaire complet. Ce modèle est ensuite séparé en trois sous modèles, pour la vitesse d'avance, le plan vertical et le plan horizontal. Une plus grande attention est ensuite portée au plan vertical, et en particulier au contrôle de l'altitude via le contrôle de l'angle de tangage grâce

à une structure cascade. Finalement un modèle LPV du sous-marin est obtenu en effectuant une linéarisation tangentielle autour d'un point de fonctionnement variable ; le modèle LPV est alors paramétré par l'angle de tangage à l'équilibre θ_{eq}.

Finalement le chapitre 4 présente la mise en application (en simulation) des différentes méthodes présentées dans le chapitre 2 sur l'AUV. Après la présentation du scénario de simulation, un correcteur $\mathcal{H}_\infty$ discret (à période d'échantillonnage constante) est synthétisé pour la vitesse d'avance : le but est de maintenir la vitesse égale à $1m/s$ malgré les perturbations engendrées par les variation d'altitude et de tangage. Les approches polytopique et LFR sont ensuite comparées sur les deux cas étudiés tout au long de cette thèse (système de départ LTI puis LPV) dans le cas du contrôle d'altitude de l'AUV. Dans le cas LTI, il est difficile de conclure sur la méthode la plus adaptée, les deux ayant données des résultats de simulations corrects, une complexité de calcul du contrôleur équivalente (pour un seul paramètre variant). Dans le cas où le système continu de départ est déjà LPV, l'approche LFR semble la plus adaptée, à cause notamment de la complexité pour obtenir un modèle polytopique après l'étape de discrétisation. Cette intuition est confirmée en simulation, puisque la méthode LFR permet d'obtenir de très bons résultats (même si l'approche polytopique avec synthèse en continu permet de maintenir la stabilité du système avec des performances satisfaisantes). Dans le second cas, la synthèse d'un correcteur à période variable pour un système LPV, l'approche LFR est alors la mieux adaptée au problème.

Nous avons donc montré dans ce travail que l'approche LFR, malgré un conservatisme certain et malgré le peu d'études qui y sont consacrées, est un outil très intéressant pour la synthèse de contrôleurs robustes LPV, prenant en compte le contexte de la mise en œuvre temps-réel des lois de commande.

Parmi les différentes perspectives envisageables pour ce travail on peut distinguer les suivantes :

D'abord il est nécessaire d'envisager des missions plus complètes et compliquées pour l'AUV. Pour cela il peut être intéressant d'ajouter comme paramètre variant la vitesse d'avance du véhicule, qui est une variable importante de la force exercée par les ailerons et qui entre en compte dans plusieurs non-linéarités du modèle du sous-marin. Ce paramètre viendra augmenter la taille du vecteur de paramètres, qui est l'un des paramètres essentiels lors du calcul du correcteur. Il serait donc intéressant de voir comment se comportent les deux méthodes face à cette augmentation, notamment répondre à la question : qu'en est-il du gain en performance par rapport à l'augmentation de la compléxité de la synthèse ?

Une autre façon de considérer des missions plus complexes est évidement de rajouter le contrôle du sous-marin dans le plan horizontal. Dans cette nouvelle configuration, la question est : qu'en est-il de l'utilisation partagée des gouvernes ? Lors

du contrôle de l'altitude, l'utilisation des gouvernes est assez loin de la saturation (exceptée aux instant de changement de consigne), ce qui est plutôt encourageant pour le contrôle global du véhicule ; ceci reste toutefois à vérifier en simulation.

Une autre perspective possible consiste à améliorer les outils utilisés dans les deux approches étudiées : pour l'approche polytopique, il serait intéressant de voir comment évolue les résultats en utilisant une fonction de Lyapunov dépendant du vecteur de paramètres (Oliveira and Peres, 2006). Pour l'approche LFR, plusieurs travaux ont été menés sur l'utilisation de multiplieurs permettant de réduire le conservatisme (Scherer, 2001), ou encore l'utilisation de fonction de Lyapunov dépendant du paramètre variable (Wu and Dong, 2006; Wu, 2001). Il serait donc intéressant de voir comment évoluent les résultats présentés dans cette thèse en utilisant ces nouvelles techniques de synthèse.

Enfin une dernière perspective à plus long terme consiste à tester les lois de commandes synthétisées tout au long de cette thèse sur un vrai AUV, afin de voir si les bons résultats obtenus en simulation peuvent être transposable au cas réel.

Bibliographie

Almeida, J., Silvestre, C., and Pascoal, A. (2011). Self-triggered output feedback control of linear plants. In *Proceedings of the European Control Conference*, San Fransisco, CA, USA.

Apkarian, P. (1997). On the discretization of LMI-synthesized linear parameter-varying controllers. *Automatica*, 33(4) :655 – 661.

Apkarian, P. and Adams, R. (2008). Advanced gain-scheduling techniques for uncertain systems. In *IEEE Transaction on Automatic Control*, volume 6, pages 21–32.

Apkarian, P. and Gahinet, P. (1994). A linear matrix inequality approach to $\mathcal{H}_\infty$ control. *International journal of robust and nonlinear control*, 4 :421–448.

Apkarian, P. and Gahinet, P. (1995). A convex characterisation of gain-scheduled $\mathcal{H}_\infty$ controllers. *IEEE Transaction on Automatic Control*, 40 :853 – 864.

Apkarian, P., Gahinet, P., and Becker, G. (1995). Self-scheduled $\mathcal{H}_\infty$ control of linear parameter-varying systems : a design example. *Automatica*, 31(9) :1251–1261.

Åström, K. J. and Wittenmark, B. (1997). *Computer-Controlled Systems*. Information and systems sciences series. Prentice Hall, New Jersey, 3rd edition.

Biannic, J. (1996). *Commande robuste des systèmes à paramètres variables - Application en aéronautique*. PhD thesis, SUPAERO Toulouse.

Boyd, S., El-Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in applied mathematics.

Bruzelius, F. (2004). *Linear Parameter-Varying Systems : an approach to gain scheduling*. PhD thesis, Department of Signals and Systems – CHALMERS UNIVERSITY OF TECHNOLOGY.

Cervin, A. (2003). *Integrated Control and Real-Time Scheduling.* PhD thesis, Department of Automatic Control, Lund Institute of Technology, Sweden.

Cuenca, A., Garcia Gil, P. J., Årzén, K.-E., and Albertos, P. (2009). A predictor-observer for a networked control system with time-varying delays and non-uniform sampling. In *Proceedings of the European Control Conference, Budapest, Aug 2009.*

Feng, Z. and Allen, R. (2004). Reduced order $\mathcal{H}_\infty$control of an autonomous underwater vehicle. *Control Engineering Practice*, 12 :1511 –1520.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles.* John Wiley & Sons.

Gauthier, C. (2007). *Commande multivariable de la pression d'injection dans un moteur diesel common rail.* PhD thesis, Institut National Polytechnique de Grenoble.

Gauthier, C., Sename, O., Dugard, L., and Meissonnier, G. (2007). An $\mathcal{H}_\infty$ linear parameter-varying (LPV) controller for a diesel engine common rail injection system. In *Proceedings of the European Control Conference*, Budapest, Hungary.

Healey, A. and Lienard, D. (1993). Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles. *Oceanic Engineering*, 18(3) :327 – 339.

Jalving, B. and Storkersen, N. (1994). The control system of an autonomous underwater vehicle. In *Proceedings of the Third IEEE Conference*, volume 2, pages 851 – 856.

Leveille, E. (2007). Analysis, redesign and verification of the iver2 autonomous underwater vehicle motion controller. Master's thesis, University of Massachusetts Dartmouth.

Lofberg, J. (2004). YALMIP : A toolbox for modeling and optimization in matlab. In *Proceedings of the CACSD Conference.*

Miyamaoto, S., Aoki, T., Maeda, T., Hirokawa, K., Ichikawa, T., Saitou, T., Kobayashi, H., Kobayashi, E., and Iwasaki, S. (2001). Maneuvering control system design for autonomous underwater vehicle. *MTS/IEEE Conference and Exhibition*, 1 :482 – 489.

Oliveira, R. C. and Peres, P. L. (2006). LMI conditions for robust stability analysis based on polynomially parameter-dependent lyapunov functions. *Systems and Control Letters*, 55(1) :52 – 61.

Poussot-Vassal, C. (2008). *Commande Robuste LPV Multivariable de Châssis Automobile*. PhD thesis, Grenoble INP.

Robert, D. (2007). *Contribution à l'interconnection commande / ordonnancement*. PhD thesis, Institut National Polytechnique de Grenoble.

Robert, D., Sename, O., and Simon, D. (2006). Synthesis of a sampling period dependent controller using LPV approach. In *5th IFAC Symposium on Robust Control Design ROCOND'06*, Toulouse, France.

Robert, D., Sename, O., and Simon, D. (2007). A reduced polytopic LPV synthesis for a sampling varying controller : experimentation with a T inverted pendulum. In *European Control Conference, ECC'07*, Kos, Greece.

Robert, D., Sename, O., and Simon, D. (2010). An h_∞ LPV design for sampling varying controllers : experimentation with a t inverted pendulum. *IEEE Transactions on Control Systems Technology*, 18(3) :741–749.

Roche, E., Sename, O., and Simon, D. (2010). LPV / $\mathcal{H}_\infty$ varying sampling control for autonomous underwater vehicles. In *Proceedings of the IFAC SSSC*, Ancona, Italie.

Roche, E., Sename, O., and Simon, D. (2011). A hierarchical varying sampling $\mathcal{H}_\infty$control of an AUV. In *Proceedings of the IFAC World Congress*, Milan, Italie.

Roche, E., Sename, O., and Simon, D. (2012). Varying sampling control for LPV systems : comparison of polytopic and LFR approaches for an AUV application. In *Submitted to ACC*, Montréal, Canada.

Sala, A. (2005). Computer control under time-varying sampling period :an LMI gridding approach. *Automatica*, 41(12) :2077–2082.

Salgado-Jimenez, T. and Jouvencel, B. (2003). Using a high order sliding modes for diving control a torpedo autonomous underwater vehicle. In *OCEANS*, volume 2, pages 934 – 939.

Santos, A. S. (1995). *Contribution à la conception des sous-marins autonomes : architecture des capteurs d'altitude, et commande référencées capteurs*. PhD thesis (in french), Ecole nationale supérieure des Mines de Paris.

Scherer, C., Gahinet, P., and Chilali, M. (1997). Multiobjective output-feedback control via lmi optimization. *Automatic Control, IEEE Transactions on*, 42(7) :896 –911.

Scherer, C. and Wieland, S. (2004). *Linear Matrix inequalities in Control.* lecture support, DELFT University.

Scherer, C. W. (2001). LPV control and full block multipliers. *Automatica*, 37(3) :361 – 375.

Sename, O., Simon, D., and Robert, D. (2003). Feedback scheduling for real-time control of systems with communication delays. In *Proceedings of the 2003 IEEE Conference on Emerging Technologies and Factory Automation*, pages 454–61, Lisbon, Portugal.

Silvestre, C. and Pascoal, A. (2004). Control of the INFANTE AUV using gain scheduled static output feedback. *Control Engineering Practice*, 12(12) :1501 – 1509.

Silvestre, C. and Pascoal, A. (2007). Depth control of the INFANTE AUV using gain-scheduled reduced order output feedback. *Control Engineering Practice*, 15(7) :883 – 895.

Simon, D., Seuret, A., Hokayem, P., Lygeros, J., and Camacho, E. (2009). FeedNetBack-D04.01- State of the art in control/computing co-design. Technical report.

Skogestad, S. and Postlethwaite, I. (1996). *Multivariable Feedback Control : analysis and design.* John Wiley and Sons.

Stankovic, J. A., Lu, C., Son, S. H., and Tao, G. (1999). The case for feedback control real-time scheduling. In *Proceedings of the 11th Euromicro Conference on RealTime Systems*, pages 11–20, York, England.

Tóth, R. (2010). *Modeling and Identification of Linear Parameter-Varying Systems.* Springer Berlin / Heidelberg.

Tóth, R., Heuberger, P., and den Hof, P. V. (2010). Discretisation of linear parameter-varying state-space representations. *IET Control Theory and Applications*, 4(10) :2082–2096.

Tóth, R., Lovera, M., Heuberger, P., and den Hof, P. V. (2009). Discretisation of linear fractional representations of LPV systems. In *Proceedings of the IEEE CDC*, Shangai, China.

Varrier, S. (2010). Robust control of autonomous underwater vehicles. Master's thesis, ENSE3 Master in Automatic Control, Systems & Information Technology, Grenoble INP, France.

Wu, F. (2001). A generalized LPV system analysis and control synthesis framework. *International Journal of Control*, 74(7) :745–759.

Wu, F. and Dong, K. (2006). Gain-scheduling control of LFT systems using parameter-dependent lyapunov functions. *Automatica*, 42(1) :39 – 50.

Zanoli, S. M. and Conte, G. (2003). Remotely operated vehicle depth control. *Control Engineering Practice*, 11(4) :453 – 459.

Zhou, K., Doyle, J., and Glover, K. (1996). *Robust and Optimal Control*. Prentice hall.