

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITE CONSTANTINE 1
FACULTE DES SCIENCES DE LA TECHNOLOGIE
DEPARTEMENT D'ELECTRONIQUE

N° d'ordre :

Série :

MEMOIRE

*Présenté pour obtenir le Diplôme de Magister en
ELECTRONIQUE*

Spécialité : Contrôle des Systèmes

THEME :

***Commande Prédicative Floue Sous Contraintes : Application à des
Systèmes non Linéaires***

PAR

ZAHAF ABDELMALEK

Soutenu le 27 / 11 / 2014

Devant le Jury :

Président : **A. Bennia** Prof. Université Constantine 1

Rapporteur : **S. Filali** Prof. Université Constantine 1

Rapporteur : **B. Boutamina** M.C. Université Constantine 1

Examineurs : **K. Belarbi** Prof. Ecole Nationale Polytechnique Constantine

S. Bououden M.C. Université de Khenchela

À mes très chers parents

À mes sœurs

À mes frères et leurs petites familles

À mes nièces

À tous ceux que j'aime

Abdelmalek

A la Mémoire

Du Professeur Salim Filali, Que Dieu ait pitié de son âme,

Sans lui, notre travail n'aurait pas abouti à ses fins.

*Je tiens à lui rendre hommage et le remercier de tout cœur d'avoir dirigé
mon travail et de n'avais ménagé aucun effort pour mener à bien ce
travail.*

*J'ai l'honneur que vous m'encadrer, me dirigée et me supervisé pendant
la préparation et la progression de ce mémoire.*

Avec votre permission, ce travail est dédié à vous.

Remerciements

Tout d'abord, j'adresse mes sincères remerciements à l'égard de mon encadreur **Feu, Mr Professeur Salim Filali**. En exprimant mes profonds sentiments de gratitude pour ses précieux conseils.

Mr. Boutamina Brahim, Maitre de conférences à l'université de Constantine 1, ma grande gratitude d'avoir accepté de poursuivre mon encadrement jusqu'à l'achèvement de ce mémoire et je le remercie également vivement pour l'intérêt qu'il a porté à ce travail en acceptant de l'examiner.

Mr. Bououden Sofiane, Maitre de conférences à l'université de Khenchela, En tant que Co-Encadreur. Mes profonds sentiments pour sa grande disponibilité pendant la progression de ce travail ; et d'avoir accepté de m'a fait l'honneur d'examiner ce mémoire.

Mr. Belarbi Khaled, professeur de l'Ecole Nationale polytechnique Constantine, ma profonde reconnaissance pour son aide et ses conseils quand j'ai aurai besoin de lui ; et d'avoir accepté de m'a fait l'honneur d'examiner mon travail.

Mr. Bennia Abdelhak, professeur à l'université de Constantine 1, mes sincères remerciements, d'avoir accepté de m'a fait l'honneur d'examiner ce travail et de faire partie du jury et d'avoir assuré sa présidence.

Mr. Chemachema Mohamed, Maitre de conférences à l'université de Constantine 1, mes profonds remerciements pour ses conseils.

A toute l'équipe du Laboratoire du Thermodynamique et Traitements de Surfaces des Matériaux (LTTSM), Mr. Karaali Ahcene, Mr. Hamamda Smail, Mr. Zerouk Imededdine, mes vifs sentiments pour leur aide et leur soutien pendant toute la formation du magister.

Enfin, je tiens à témoigner toute ma gratitude à mes parents, mes sœurs et mes frères pour leurs encouragements, et qui m'ont toujours soutenu inconditionnellement. Et en toutes circonstances. « Maman, Papa, Noura, Leila, Mohamed, Lazher, Said, Hamza, Souraya » c'est à vous que je dédie ce travail.

Sommaire

<i>Introduction Générale</i>	01
------------------------------	----

Chapitre I : Introduction à la modélisation et la commande des modèles T-S

I.1 Introduction	06
I.2 Généralités sur la logique floue	06
I.2.1 Principe de l'inférence floue	06
I.2.2 Fuzzification	08
I.2.3 Base de connaissances	10
I.2.4 Défuzzification	11
I.3 Les différents modèles de la logique floue	13
I.3.1 Modèle floue de Mamdani	13
I.3.2 Modèle floue de Takagi – Sugeno (T-S)	14
I.4 Modèle Flou de Type Takagi-Sugeno	15
I.4.1 Obtention d'un Modèle Flou Takagi-Sugeno	17
I.4.1.1 Modèle Flou basé sur l'approche du secteur non-linéaire	17
I.5 Les Différents Types des Contrôleurs Flous	21
I.5.1 Loi de Commande par Retour d'Etat	21
I.5.2 Stabilité Floue Simultanément (FSS)	21
I.5.3 Compensation et Division pour les Modèles Flous CDF	22
I.5.4 L'approche de la Compensation Parallèle Distribuée PDC	22
I.5.5 L'approche de la Non-Compensation Parallèle Distribuée Non-PDC	23
I.6 Conclusion	24

Chapitre II : Commande Prédictive à Modèle «MPC» et Inégalités Matricielles Linéaires «LMI»: Outils d'Observation, d'Analyse et de Synthèse

II.1 Introduction	26
-------------------------	----

II.2 La Commande prédictive	27
II.2.1 Modèle de prédiction	28
II.2.2 Fonction Coût (critère de performance)	28
II.2.3 Loi de Commande	29
II.2.3.1 le calcul des prédictions à base de modèle d'état	29
II.2.4 La stabilité de la commande prédictive sous contraintes à horizon infini	30
II.2.5 La commande prédictive sous contraintes sous forme des LMIs	31
II.3 Les Inégalités Matricielles Linéaires	35
II.3.1 Quelques Problèmes LMI	35
II.3.1.1 Problème de Faisabilité	35
II.3.1.2 problème de Valeur Propre (EVP : Eigen-Value Problem)	36
II.3.1.3 problème de Valeur Propre Généralisées (General EVP)	36
II.3.2 Les Techniques d'Analyse et Transformation Matricielles	36
II.3.2.1 Complément de Schur	36
II.3.2.2 Congruence	37
II.3.2.3 Lemme d'Elimination	39
II.3.2.4 Le Carré Matriciel et ses Dérivées	39
II.4 Conclusion	41

Chapitre III : Commande prédictive Robuste des Modèles T-S

III.1 Introduction	43
III.2 Modèle T-S et la Fonction Coût	43
III.3 Stabilité et stabilisation du Système Augmenté	44
III.4 Synthèse de la Commande avec l'Approche Non-PDC	45
III.4.1 Problématique : Construction du Système Augmenté	45
III.5 Synthèse de la Commande avec l'Approche PDC	52
III.5.1 Problématique : Construction du Système Augmenté	52
III.6 Conclusion	58

Chapitre IV : Simulations et Discussions

IV.1 Introduction	60
IV.2 Application de la commande MPC-FTS-LMI	60
IV.3 Application de la commande MPC-FTS-LMI aux Système Perturbés	72
IV.4 Conclusion	79
 <i>Conclusion Générale et Perspectives</i>	 80
 <i>Références Bibliographique</i>	 82

Introduction Générale

La diversité des problèmes rencontrés en automatique, notamment ceux portant sur la théorie de la commande et de la conception, ont connu une évaluation considérable ces dernières années. Parmi ces théories, la commande des systèmes linéaires et non-linéaires qui ne cesse de se perfectionner.

La commande prédictive à base de modèle, Model based Predictive Control, MPC, est une des stratégies les plus populaires dans le domaine du contrôle et l'automatique. Elle est basée sur la prédiction du comportement futur du processus à partir d'un modèle explicite d'un système et le choix d'une séquence de commandes optimales basée sur l'utilisation de la fonction coût [1]. De plus, cette technique est très efficace pour calculer la commande optimale des systèmes pour la régulation ou la poursuite sous contraintes, qui nous donne des problèmes d'optimisation dynamique, ce problème est résolu par le calcul en ligne en temps réel à chaque période d'échantillonnage [2]. Récemment, une large classe de commandes prédictives, MPC, ont été proposées par les praticiens de l'automatique. Mais la robustesse du MPC reste toujours l'objectif pour une meilleure conception du système [3]. L'utilisation de la MPC sous contraintes sous formes des LMI (Linear Matrix Inequalities) a permis de bien manipuler et contrôler les variables du système d'une façon systématique durant la conception et l'implémentation du contrôleur [4]. Pour minimiser le temps et l'efficacité de la recherche d'une solution faisable des problèmes, une approche sous-optimale MPC a été étudiée par [5], on trouve également une étude de la robustesse des modèles des systèmes linéaires incertains [6]. Ainsi qu'une étude de la robustesse de la sortie du MPC [7]. Par l'utilisation de la fonction de Lyapunov, une autre loi de commande du MPC pour les systèmes à temps variant a été proposée par [8]. D'autres travaux ont été faits pour trouver des solutions du MPC pour les systèmes non-linéaires [9]. Cependant le problème de robustesse est persiste

toujours vu la présence des contraintes sur l'entrée et la sortie pour les systèmes en boucle fermé [10].

Dans le même sens, la théorie de la logique floue est une des plus fameuses techniques pour la modélisation des systèmes non-linéaires. Dans ce cas, il est possible de mettre une grande partie des modèles non linéaires sous forme de modèles T-S représentant exactement le modèle non linéaire dans un compact de l'espace d'état.

Les lois de commande couramment utilisées sur ce type de modèles sont de type retour d'état non linéaire PDC (Parallel Distributed Compensation) [11]. L'idée principale de la technique de PDC est basée sur la conception de la loi de commande, chaque loi de commande linéaire doit utiliser la même règle que celle du modèle T-S floue du système qui permettra de le compenser. La formulation de celle-ci aboutit à un ensemble d'inégalité écrite, dans la mesure du possible, sous la forme inégalités matricielles linéaires. Les LMI, s'il s'avère qu'elles admettent une solution, peuvent être résolues à l'aide d'outils issus du domaine de l'optimisation convexe (El Ghaoui, 1997) [12], [13], [14] et [15]. Cependant, et afin de réduire le conservatisme des conditions LMI proposées, nous considérons le contexte non quadratique. Dans ce dessein, le choix s'est porté sur l'utilisation d'une fonction candidate de Lyapunov non-quadratique [16], [17], [18] et [19]. Celles-ci sont basées sur une propriété de redondance analytique des descripteurs [20]. Dans ce cas, le principe consiste à réécrire la dynamique de la boucle fermée d'un système dynamique explicite sous forme de descripteur. Ceci conduit à des conditions LMI moins conservatives et réduit leur coût de calcul.

L'objectif de ce travail est d'étudier une procédure systématique qui permet de concevoir un contrôleur stable par retour d'état basé sur la stratégie MPC qui doit répondre aux deux exigences: garantir la stabilité des systèmes en boucle fermée et satisfaire certaines exigences exprimées sous forme de contraintes et de sous optimalité vis à vis d'un critère donné. La synthèse des lois de commande

sera basée sur des techniques de LMIs et afin de réduire les problèmes de conservatisme, une approche basée sur une fonction candidate non quadratique de Lyapunov et une loi de commande non PDC (Compensation Parallèle Distribuée) sont étudiées.

Notre travail sera articulé en quatre chapitres organisés comme suit :

Le premier chapitre sera consacré aux principaux outils ainsi que les résultats les plus couramment utilisés pour l'analyse des modèles de Takagi-Sugeno. Leurs structures ainsi que la façon de les obtenir par l'approche de type secteurs non linéaires sont également présentées. Nous traiterons ensuite la stratégie de commande la plus répandue pour la stabilisation des systèmes non linéaires sous forme de modèles T-S est la commande PDC (Compensation Parallèle Distribuée -Parallel Distributed Compensation).

Le deuxième chapitre servira d'introduction à la problématique de la commande prédictive sous contraintes. On y retrouve les principes généraux des stratégies de commande basées sur le principe de l'horizon infini, introduisant les problèmes d'optimisation associés. Ensuite nous adopterons le formalisme LMI (Linear Matrix Inequality) comme outil d'optimisation.

Le troisième chapitre concernera la commande prédictive sous contraintes basée sur les modèles flous de type Takagi-Sugeno, en utilisant la philosophie de commande du type compensation parallèle distribuée (PDC) et non PDC. Nous commencerons par un aperçu général sur la synthèse (stabilisation) des contrôleurs flous basés sur un modèle flou homogène T-S du système. Ensuite nous décrirons spécifiquement une commande optimale linéaire quadratique par retour d'état et non quadratique basée sur un modèle affine T-S en temps discret [21]. Les gains de la loi de commande sont obtenus en résolvant un problème d'optimisation sous contraintes d'inégalités matricielles linéaires (LMIs).

Dans le quatrième chapitre, nous présenterons des exemples de simulations proposés pour la validation des algorithmes mis aux points, plusieurs tests ont été effectués pour mettre en évidence les performances des approches proposées.

Enfin, nous concluons notre travail par une conclusion générale qui englobera tout ce qui a été développé, ainsi que par quelques-unes des perspectives qui nous semble les plus intéressantes à étudier.

Chapitre I

Introduction à la modélisation et la commande des modèles T-S

Dans ce chapitre, nous allons essayer de présenter quelques notions et propriétés concernant la logique floue de type Takagi-Sugeno, pour l'adopter ensuite à notre travail.

I.1 Introduction

Le concept de la logique floue a été introduit par Lotfi Zadeh [22]. Cette approche a connu un grand développement surtout au Japon à partir des années Quatre-vingts, en particulier avec les idées brillantes de Takagi-Sugeno et la proposition d'un modèle flou en 1985 [23], qui devient après, la base de la plupart des travaux en contrôle et en automatique dans le domaine de la logique floue. En plus de l'existence de modèle T-S, il existe un modèle de Mamdani qui a été le premier modèle floue introduite dans le domaine flou en 1974 [24].

Actuellement, la logique floue est appliquée dans divers domaines, tels que l'intelligence artificielle, l'automatique, la biologie, l'industrie, la médecine, l'économie. Et bien d'autres domaines. Dans le cas de l'automatique et contrôle, la plupart des sujets et thèmes de recherche ont été explorés par ce concept soit pour : La Commande, La Modélisation, l'Approximation, L'Estimation d'état... etc.

Dans ce chapitre, nous nous intéressons à la représentation par un modèle flou de type T-S des systèmes non linéaires ainsi qu'à la transformation par secteurs non linéaires qui permet d'obtenir de manière systématique un modèle T-S à partir d'un système non linéaire. Ensuite, nous rappelons les principaux résultats portant sur la stabilité, la synthèse de la loi de commande PDC (Parallel Distributed Compensation) et non-PDC par retour d'état statique.

I.2 Généralités sur la logique floue

I.2.1 Principe de l'inférence floue

L'idée de base de la logique floue est de modéliser les phénomènes ou des procédés comme le ferait l'être humain, c'est-à-dire de mettre en algorithmes, composés essentiellement par des règles du type «*Si... Alors ...*» (*If... Then ...*), Le raisonnement humain. En effet, l'homme a tendance à tirer des conclusions sur certaines situations à partir de données imprécises ou incomplètes et à agir en fonction d'un ensemble de règles implicites au raisonnement. A titre d'exemple, une personne qui est prête de prendre une douche peut savoir si l'eau a une

température appropriée au moment d'ouvrir le robinet. Si cette personne trouve que l'eau est très froide, sa réaction naturelle sera de bouger le robinet pour réguler la température. Si l'eau devient très chaude, alors elle devra bouger le robinet dans le sens inverse et ainsi de suite jusqu'à trouver la température de l'eau qui lui convient le mieux. La personne en question ne connaît pas forcément la température de l'eau, mais ses capteurs corporels lui donnent une information que son cerveau transforme en une sensation de confort et c'est à partir de laquelle elle prend des décisions [25].

Ainsi, un algorithme qui peut traduire le raisonnement de la personne de l'exemple cité ci-dessus pourrait être le suivant :

Si l'eau est froide alors ouvrir le robinet vers le chaud. Si l'eau est très chaude alors ouvrir le robinet vers le froid... Si l'eau a une température appropriée alors laisser le robinet fixe et prendre la douche.

Si l'idée de mettre en algorithme le raisonnement humain est considérée dans le cas de phénomènes ou procédés plus complexes qui font intervenir plusieurs machines et opérateurs, toute la connaissance empirique des experts peut être exploitée en vue d'une meilleure maîtrise de ces procédés complexes. Pour cela il faut suivre la méthodologie de la logique floue qui divise un système en trois éléments principaux, à savoir, un module de fuzzification, un module de déduction des règles, nommé aussi base de connaissance ou des règles et un module de Défuzzification.

La figure ci-dessous illustre et présente mieux la composition d'un système flou.

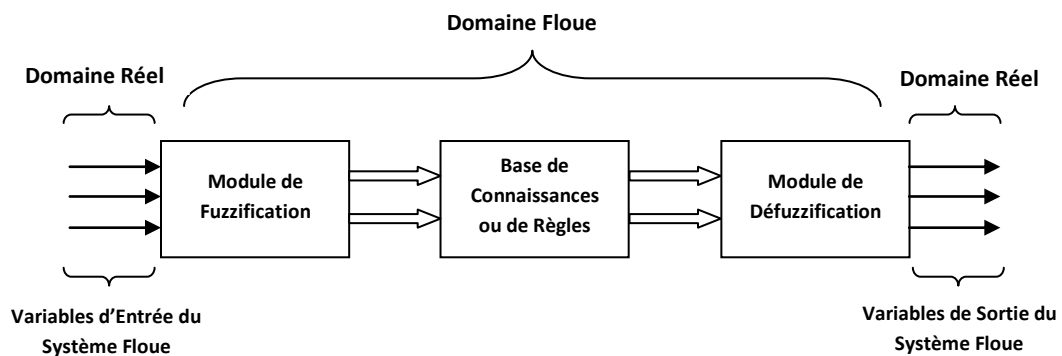


Figure I.1 : Structure d'un Système Floue

Les trois modules ou les éléments essentiels sont détaillés dans les paragraphes suivants :

I.2.2 Fuzzification

Le principe de la Fuzzification est de remplacer l'appartenance 0 et 1 d'un ensemble de la logique classique par un degré d'appartenance pouvant prendre toutes les valeurs comprises entre 0 et 1 dans un ensemble appelé *ensemble flou*. La figure I.2 explique bien ce principe en comparant la température de l'eau classée par la logique classique et par la logique floue.

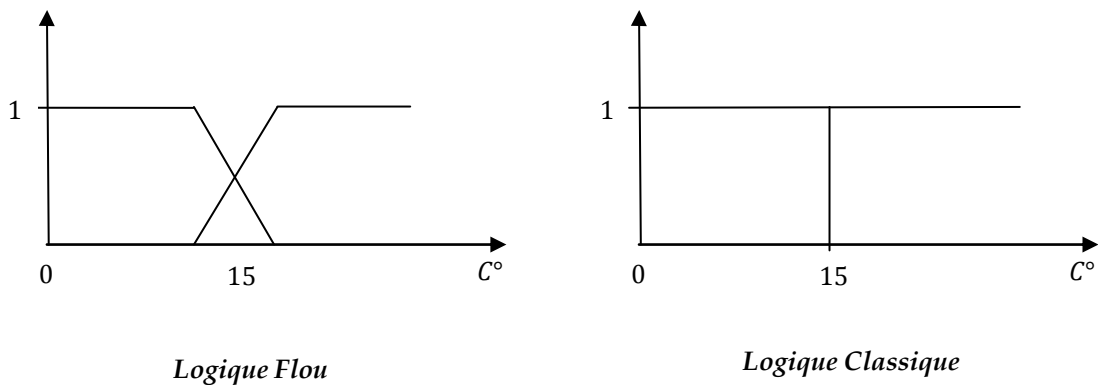


Figure I.2 : Comparaison entre la Logique Classique et la Logique Floue

Dans cet exemple, en logique classique jusqu'à 15°C, l'eau appartient à la classe de l'eau froide, alors qu'en logique floue, à partir de 10°C, l'eau n'appartient plus que partiellement à l'eau froide.

Cette idée est représentée mathématiquement par l'expression (1.1) pour la logique classique et par l'expression (1.2) pour la logique floue.

$$h(z) = \begin{cases} 1, & \text{si } z \in A \\ 0, & \text{sinon} \end{cases} \quad (1.1)$$

$$h(z) = [0, 1] \quad (1.2)$$

Dans la littérature, le degré d'appartenance dans un ensemble flou est désigné par la lettre grecque μ .

Ainsi, le module de Fuzzification a pour but de réaliser le passage des variables du domaine réel vers le domaine flou. Ce passage consiste en la description des variables réelles (variables d'entrées du système flou) à partir des ensembles flous donnant les degrés d'appartenance à différents états identifiés pour ces variables. Les ensembles flous sont désignés par des expressions du langage courant. Les ensembles flous peuvent prendre les formes géométriques les plus variées, mais les plus utilisées sont les formes : Triangulaire, Sigmoidale, gaussienne et Trapézoïdale. Le choix de cette forme est particulier pour chaque application. Les différentes formes sont montrées sur la figure I.3.

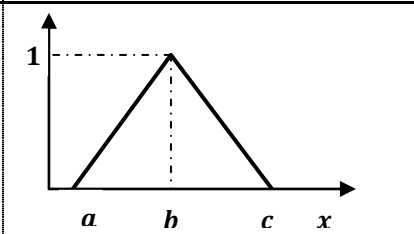
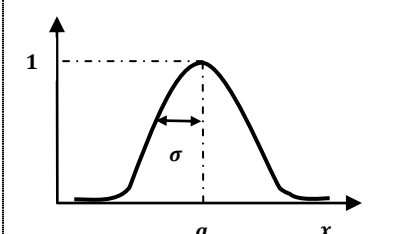
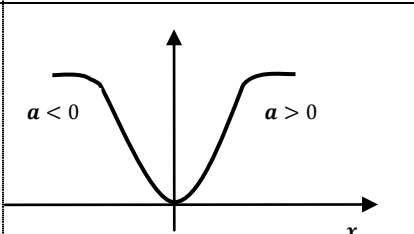
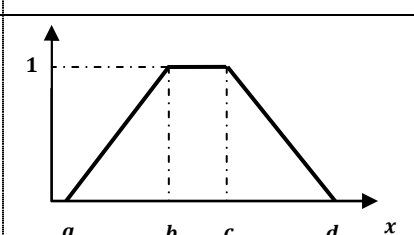
Fonction	Forme	Modèle Mathématique
Triangulaire		$F(x; a, b, c) = \max(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0)$
Gaussienne		$F(x; \sigma, a) = \exp\left(-\left(\frac{x-a}{\sigma}\right)^2\right)$
Sigmoïde		$F(x; a, c) = \frac{1}{1 + \exp(-a(x-c))}$
Trapézoïdale		$F(x; a, b, c) = \max(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-b}\right), 0)$

Figure I.3: Les Différents Types de la Fonction d'Appartenance

Les variables d'entrée ainsi que les variables de sortie du système ou du phénomène à manipuler par la logique floue sont traitées dans le module de Fuzzification. Dans les autres modules, des relations entre ces variables seront établies, comme il est explicité dans les paragraphes suivants.

1.2.3 Base de connaissances

Une fois que les variables d'entrée et de sortie sont exprimées à partir d'ensembles flous, il faut établir des règles d'inférences les liant les unes aux autres. Autrement dit, un traitement des données dans le domaine flou est nécessaire. C'est le rôle de ce module et c'est dans ce module que la connaissance empirique des experts du procédé en question peut être exploitée pour l'établissement des règles.

Ainsi, r règles peuvent être établies (combinaison des ensembles flous des variables d'entrée), dont une règle prise au hasard a la forme suivante :

Si x_1 est A_1^h **et** x_2 est A_2^k ... **et** x_n est A_n^l **alors** y_1 est B_1^p **et** y_2 est B_2^q ... **et** y_m est B_m^r

Avec, x_1, \dots, x_n , les n variables d'entrée. y_1, \dots, y_m , les m variables de sortie. A_1, \dots, A_n , B_1, \dots, B_m , des ensembles flous et les super-indices des fonctions d'appartenance de chaque ensemble flou. Plusieurs combinaisons des opérations logiques (ou, et) sont possibles. La condition de la règle est connue comme *prémisse* et la conclusion aussi nommée *conséquent*.

Ainsi, dans l'exemple de la voiture qui s'approche d'un feu tricolore, une règle d'inférence est :

Si le feu tricolore est proche Et la vitesse est élevée Et la lumière est verte Alors le conducteur maintient sa vitesse.

Pour que le conducteur maintienne sa vitesse, il faut que trois conditions soient remplies. Pour certaines valeurs des variables d'entrée chaque condition va être remplie avec un certain degré d'appartenance. Mais la condition globale ne nécessite qu'un seul. Le degré d'appartenance globale sera alors composé d'une combinaison des degrés individuels. Pour connaître l'apport de ces derniers au degré global, il est nécessaire d'analyser les opérateurs logiques (*NON, OU, ET*) utilisés dans la règle. Il y a plusieurs manières de considérer ces opérateurs, mais

la plus utilisée est la logique de Zadeh [22]. Et c'est celle qui sera employée par la suite.

La logique de Zadeh définit alors l'opération **NON** comme le complément d'une variable dans un ensemble flou. Il est exprimé comme suit :

$$\bar{A} = \{x/x \notin A\} \quad (1.3)$$

Et il est calculé avec :

$$\overline{h(x)} = h_{\bar{A}}(x) = 1 - h_A(x) \quad (1.4)$$

Où le terme $h_A(x)$ exprime le degré d'appartenance de la variable x dans l'ensemble flou A .

L'opérateur logique **ET** représente l'intersection des deux ensembles :

$$A \cap B = \{x/x \in A \cap x \in B\} \quad (1.5)$$

Et le degré retenu est celui de la valeur minimale :

$$h_{A \cap B}(x) = h_A(x) \cap h_B(x) = \min(h_A(x), h_B(x)) \quad (1.6)$$

Le dernier opérateur utilisé pour la composition du degré d'appartenance global est **OU**. Cet opérateur correspond à l'union de deux ensembles :

$$A \cup B = \{x/x \in A \cup x \in B\} \quad (1.7)$$

Et le degré retenu est celui qui a la valeur maximale :

$$h_{A \cup B}(x) = h_A(x) \cup h_B(x) = \max(h_A(x), h_B(x)) \quad (1.8)$$

En résumé, quand des conditions sont liées par un opérateur **OU**, on considère le degré d'appartenance maximum parmi les conditions d'entrée. Quand des conditions sont liées par une logique **ET**, on considère le degré d'appartenance minimum parmi les conditions qui sont en jeu.

I.2.4 Défuzzification

Dans les deux premiers modules, la transformation des variables réelles en variables floues ainsi que le traitement des données dans le domaine flou ont été

réalisés. La dernière étape consiste à passer du domaine flou au domaine réel car les données réelles sont celles qui peuvent être expliquées par les systèmes physiques. Donc le module de Défuzzification a pour but de transformer les valeurs des variables des sorties floues en valeurs numériques. Il existe plusieurs méthodes pour réaliser cette opération, telles que la méthode du maximum, la méthode de la moyenne pondérée, la méthode de la hauteur, la méthode de la hauteur modifiée et la méthode du centre de gravité. Une brève explication de quelques méthodes est présentée ci-dessous.

La méthode du maximum considère seulement, pour chaque sortie, la règle présentant le degré d'appartenance le plus grand.

$$u = \max(h_A(x), h_B(x)) \quad (1.9)$$

Avec h la sortie du système flou, c'est-à-dire, la valeur numérique de la sortie floue.

C'est une méthode qui introduit des ambiguïtés, car il y a plusieurs valeurs de x qui correspondent au maximum. De plus, les règles secondaires (celles qui n'ont pas la valeur maximale) ne sont pas prises en considération même si elles peuvent être importantes au niveau du fonctionnement du système. Cette méthode est peu utilisée.

La méthode de la moyenne pondérée considère comme valeur de sortie la moyenne des valeurs préconisées par chaque règle, pondérées par leurs degrés d'appartenance respectifs. L'équation suivante donne un calcul de la sortie du système flou.

$$u = \frac{\sum_{i=1}^n h_{A_i}(x_i) * x_i}{\sum_{i=1}^n h_{A_i}(x_i)} \quad (1.10)$$

Il s'agit d'une méthode plus évoluée et pourtant plus fiable. En effet, elle prend en compte toutes les règles en diminuant ainsi les ambiguïtés. C'est une méthode plus utilisée que la méthode du maximum.

Enfin, la méthode du centre de gravité consiste à tracer, sur un même diagramme, les différentes zones trapézoïdales correspondant à chacune des règles, et à calculer le centre de gravité de la zone consolidée ; le degré d'appartenance est donc calculé comme suit :

$$u = \frac{\sum_{i=1}^n x_i^* v_i}{\sum_{i=1}^n v_i} \quad (1.11)$$

Où v_i représente la surface active de l'ensemble i , elle est une fonction de la variable d'entrée et du degré d'appartenance. x_i^* Représente le centre de gravité de la surface v_i .

Cette méthode est très utilisée car quoiqu'elle est coûteuse en termes de calcul mais elle est plus performante que les autres méthodes.

Néanmoins, il n'existe pas de règle générale pour le choix d'une méthode de Défuzzification. Le choix de la méthode dépend des exigences de l'application. Il y a souvent un compromis à respecter entre facilité et performance. Dans notre travail, nous allons utiliser les méthodes du centre de gravité et de la moyenne pondérée.

I.3 Les différents modèles de la logique floue

Ils existent deux types de modèles flous :

I.3.1 Modèle floue de Mamdani

Dans ce type de modèle flou de Mamdani [24], les règles floues *If-Then (Si-Alors)* sont écrites sous la forme suivante :

$$\text{Règle } i : \text{Si } z_1(t) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(t) \text{ est } M_{ip} \text{ Alors } y \text{ est } B_i \quad (1.12)$$

Les principaux avantages de ce type sont :

- La simplicité de la représentation des règles floues.
- La flexibilité d'implémentation et utilisation; du point de vue du grand nombre de choix des opérations appropriées en modèle flou.

L'inconvénient de ce modèle est le grand nombre des règles floues utilisées pour représenter un système non-linéaire complexe.

I.3.2 Modèle flou de Takagi - Sugeno (T-S)

Ce modèle de T-S [23]. Devient la technique la plus utilisée car sa simplification de représenter n'importe quel système non-linéaire complexe avec des règles floues réduite par rapport au type de Mamdani, et aussi la conséquence qu'été fonction numérique. Un modèle Takagi-Sugeno flou est défini par la notion des règles floues sous la forme suivante :

$$\begin{aligned} \text{R\`egle } i : \mathbf{Si} \ z_1(t) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(t) \text{ est } M_{ip} \\ \mathbf{Alors} \ y_i = a_0^i + a_1^i z_1(t) + a_2^i z_2(t) + \dots + a_p^i z_p(t) \end{aligned} \quad (1.13)$$

Alors, que M_{i1}, \dots, M_{ip} sont les ensembles flous, a_0^i, \dots, a_p^i sont les coefficients des i^{eme} conséquence linéaire et aussi c' est la sortie de i^{eme} valeur numérique de la sortie pour le modèle flou de T-S est calculée comme suit :

$$y = \frac{\sum_{i=1}^r w_i(t) y_i}{\sum_{i=1}^r w_i(t)} \quad (1.14)$$

Avec, $0 < w_i(t) < 1$, $\sum_{i=1}^r w_i(t) > 0$ et $w_i(t) = \prod_{j=1}^p M_{ij}(z_j(t))$, $M_{ij}(z_j(t))$ est le degré de la fonction d'appartenance de $z_j(t)$ en M_{ij} . (1.13) donne un modèle T-S flou affine, pour $a_0^i = 0$, on obtient un modèle T-S linéaire. Le modèle T-S représente un système dynamique, avec des règles floues *Si-Alors (If-Then)* qui représente des relations localement linéaire entre l'entrée et la sortie du système dynamique non-linéaire. L'aspect principal du modèle T-S flou est de représenter chaque règle floue locale dynamique par un sous-modèle linéaire, après le système flou global est obtenu par le calcul de la somme floue des sous-modèles linéaires [26].

Donc l'avantage de modèle flou T-S est la flexibilité, la simplicité et l'efficacité de représenter les processus non-linéaires complexes par un nombre très réduit des règles floues, par rapport au modèle flou de Mamdani. Avec le principe du centre de gravité de Défuzzification on peut constater des cas

particuliers des modèles flous T-S. En plus, dans le modèle flou T-S, le passage entre les règles floues est fait par une relation linéaire, par exemple, l'interpolation. Donc (1.14) donne une interpolation entre les différentes fonctions linéaires, qui sont des modèles locaux comme indiqué dans la figure suivante.

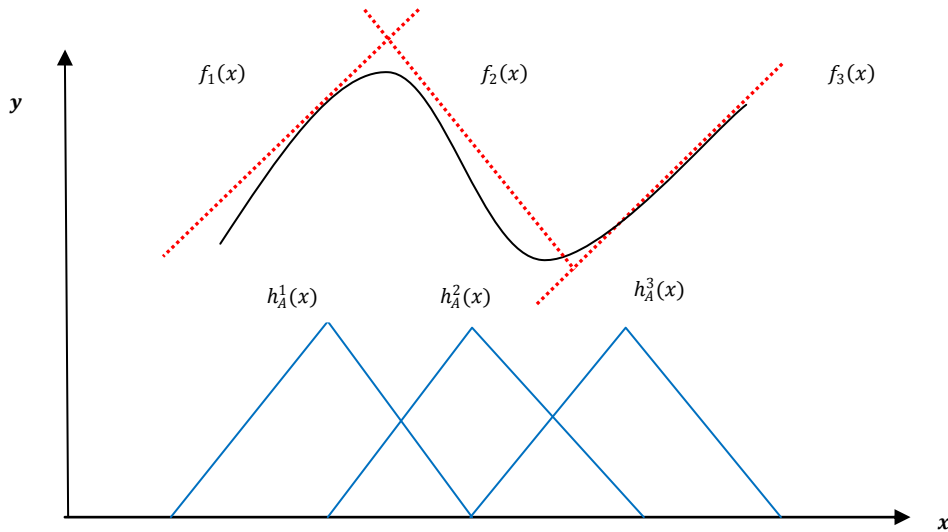


Figure I.4: Une fonction f d'un système non-linéaire représenté par le modèle T-S

I.4 Modèle Flou de Type Takagi-Sugeno

Le modèle flou Takagi-Sugeno (T-S) [23] est décrit par des règles *Si-Alors* (*If-Then*) qui représentent des modèles Linéaires Invariants dans le Temps (LTI). Ainsi, tout modèle flou T-S d'un processus non-linéaire est structuré comme une interpolation des systèmes linéaires. Il a été prouvé que les modèles flous T-S sont des approximateurs universels [27] et [28].

Un modèle T-S est représenté dans le cas continu et le cas discret. La forme du modèle de chaque cas est comme suit :

Systeme Flou Continu : SFC

$$\begin{aligned}
 &\text{R\`egle } i : \text{ Si } z_1(t) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(t) \text{ est } M_{ip} \\
 &\text{Alors } \begin{cases} \dot{x}(t) = A_i x(t) + B_i u(t) \\ y(t) = C_i x(t) + D_i u(t) \end{cases} \quad i = 1 \dots r \quad (1.15)
 \end{aligned}$$

Le modèle flou global ou les sorties finales sont écrites comme suit :

$$\begin{cases} \dot{x}(t) = \sum_{i=1}^r h_i(z(t)) (A_i x(t) + B_i u(t)) \\ y(t) = \sum_{i=1}^r h_i(z(t)) (C_i x(t) + D_i u(t)) \end{cases} \quad i = 1 \dots r \quad (1.16)$$

Systeme Flou Discret : SFD

$$\begin{aligned} & \text{Règle } i: \mathbf{Si} \quad z_1(t) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(t) \text{ est } M_{ip} \\ & \mathbf{Alors} \quad \begin{cases} x(t+1) = A_i x(t) + B_i u(t) \\ y(t) = C_i x(t) + D_i u(t) \end{cases} \quad i = 1 \dots r \end{aligned} \quad (1.17)$$

Le modèle flou global ou les sorties finales sont écrites comme suit :

$$\begin{cases} x(t+1) = \sum_{i=1}^r h_i(z(t)) (A_i x(t) + B_i u(t)) \\ y(t) = \sum_{i=1}^r h_i(z(t)) (C_i x(t) + D_i u(t)) \end{cases} \quad i = 1 \dots r \quad (1.18)$$

Ou $x(t) \in R^n$ est le vecteur d'état, r est le nombre des règles, M_{ij} , sont les fonctions d'appartenances à des ensembles flous, $j = \{1 \dots p\}$, $u(t) \in R^m$ est le vecteur d'entrée ou la commande, $y(t) \in R^q$ est le vecteur de sortie, $A_i(t) \in R^{n \times n}$, $B_i(t) \in R^{n \times m}$ et $C_i(t) \in R^{q \times n}$. $z_1(t), \dots, z_p(t)$ sont les variables des prémisses qui peuvent être des fonctions des variables d'état, des entrées ou une combinaison des deux. A chaque règle est attribué un poids $w_i(z(t))$ qui dépend du vecteur $z(t) = [z_1(t), \dots, z_p(t)]$ et du choix de l'opérateur logique.

Par exemple, l'opérateur *ET* est souvent choisi comme étant le produit, d'où :

$$w_i(z(t)) = \prod_{j=1}^p M_{ij}(z_j(t)), \quad i = \{1 \dots r\} \text{ avec } w_i(z(t)) \geq 0, \text{ pour } t \geq 0,$$

$$h_i(z_i(t)) = \frac{w_i(z(t))}{\sum_{i=1}^r w_i(z(t))} \quad (1.19)$$

La fonction $h_i(z_i(t))$ dite d'activation détermine le degré d'activation du i^{eme} modèle local associé. Selon la zone où évolue le système, cette fonction indique la contribution plus ou moins importante du modèle local correspondant dans le modèle global. Elle assure un passage progressif de ce modèle aux modèles locaux voisins. Ces fonctions dépendent généralement du vecteur d'état $x(t)$. Elles

peuvent être de forme Triangulaire, Sigmoidale, ou gaussienne, et satisferont les propriétés suivantes (convexité) :

$$\begin{cases} \sum_{i=1}^r h_i(z_i(t)) = 1 \\ 0 < h_i(z_i(t)) < 1 \end{cases} \quad (1.20)$$

La particularité d'un modèle T-S est que les ensembles flous continus ou des modèles flous discrets [29] et [30]. Dans ce mémoire, nous allons nous intéresser uniquement au modèle discret dont l'utilisation est plus fréquente.

I.4.1 Obtention d'un Modèle Flou Takagi-Sugeno

I.4.1.3 Modèle Flou basé sur l'approche du secteur non-linéaire

La première apparition de l'approche du secteur non-linéaire pour la conception d'un modèle flou été introduite en 1992 par Kawamoto [31]. L'idée est basée sur la considération d'un système non-linéaire simple $\dot{x}(t) = f(x(t))$ avec $f(0) = 0$. L'objectif est de trouver un secteur global qui satisfait aux conditions du système non-linéaire $\dot{x}(t) = f(x(t)) \in [a_1 \ a_2]$, comme indiqué dans la figure (Fig. 1.5). La conception d'un modèle flou exacte est assurée par cette méthode. Mais, ce n'est pas évident toujours de trouver un secteur global pour n'importe quel système non-linéaire. Pour ces raisons, la proposition d'un secteur local été considérée comme une meilleure solution pour l'obtention d'un modèle flou. L'astuce est de présenter le SNL par plusieurs secteurs non-linéaires, donc chaque région « secteur local » est représenté par un modèle flou exacte avec $x(t) \in [-d_i \ d_i]$ la figure (Fig. 1.6).

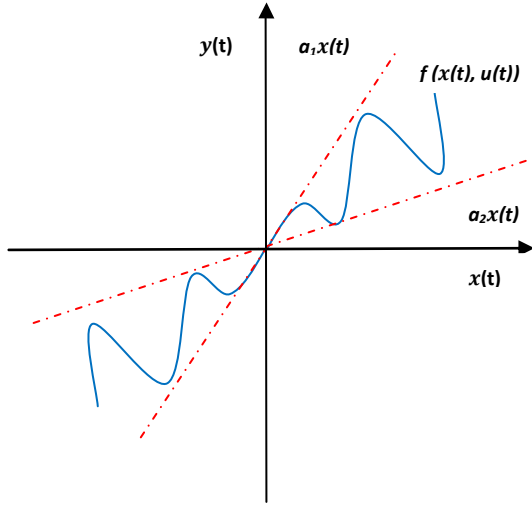


Figure I.5. Secteur Non-Linéaire Global.

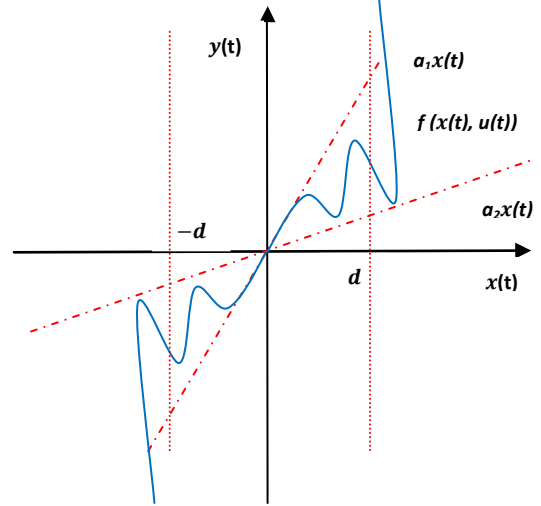


Figure I.6. Secteur Non-Linéaire Local.

Mais l'inconvénient du secteur local est le grand nombre de régions proposées, avec le nombre des règles de chaque modèle flou, qui impose après des difficultés de bien contrôler et commander le système. Ces techniques sont bien traitées, et une méthode de transformation mathématique été proposée par [32] et [33]. Elle est basée sur une transformation des fonctions scalaires origines de la non-linéarité. L'avantage de cette méthode est de ne pas engendrer d'erreur d'approximation et de minimiser le nombre de modèles locaux. Cette méthode est basée sur la bornitude des fonctions continues.

Pour plus de détails, considérons le cas général d'un système continu linéaire :

$$\dot{x}(t) = f(x(t)) + Bu(t) \quad (1.21)$$

Avec: $x(.) \in R^n$, $u(.) \in R^m$, $f(x(.)) \in R^p$, $B \in R^{p*m}$.

Lemme 1 : soit $f(x(t))$ une fonction bornée de $[a, b] \rightarrow R$ pour tout $x \in [a, b]$ avec $[a, b] \in R^{+2}$. Alors il existe deux fonctions

$$\begin{cases} F^i(.) : [a, b] \rightarrow [0, 1], i \in I_2 \\ x(t) \rightarrow F^i(x(t)) \end{cases} \quad (1.22)$$

Avec $F^1(x(t)) + F^2(x(t)) = 1$ et deux scalaires α et β tels que

$$h(x(t)) = \alpha * F^1(x(t)) + \beta * F^2(x(t)) \quad (1.23)$$

Une décomposition de $h(x(t))$ est considérée sur $[a, b]$ comme suivant :

$$\begin{cases} \beta = \min_{x \in [a, b]} h(x(t)), \\ \alpha = \max_{x \in [a, b]} h(x(t)) \end{cases} \quad (1.28)$$

$$\begin{cases} F^1(x(t)) = \frac{h(x(t)) - \beta}{\alpha - \beta}, \\ F^2(x(t)) = \frac{\alpha - h(x(t))}{\alpha - \beta} \end{cases} \quad (1.29)$$

Sous l'hypothèse de la continuité et la bornitude des fonctions $f(x(t))$ et $g(x(t))$ données en (1.21) avec $f(0) = 0$ et $g(0) = 0$, ces fonctions peuvent être réécrites sous la forme suivante :

$$\begin{cases} f(x(t)) = \sum_{i=1}^r F^i(x(t)) A_i x(t), \\ g(x(t)) = \sum_{i=1}^r F^i(x(t)) C_i x(t) \end{cases} \quad (1.30)$$

Le modèle (1.21) devient :

$$\begin{cases} \dot{x}(t) = \sum_{i=1}^r F^i(x(t)) (A_i x(t) + B_i u(t)), \\ y(t) = \sum_{i=1}^r F^i(x(t)) (C_i x(t) + D_i u(t)) \end{cases} \quad (1.31)$$

Dans le cas, le modèle ou le multi-modèle (1.31) obtenu représente de façon exacte le modèle non-linéaire (1.21) sur l'intervalle compact considéré.

Pour cela on va présenter un exemple d'illustration pour bien expliquer.

Exemple d'illustration

A fin d'illustrer la méthode, nous proposons le modèle non-linéaire autonome de dimension deux [26] :

$$\dot{x}(t) = f(x(t)) = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -x_1(t) + x_1(t)x_2^3(t) \\ -x_2(t) + (3 + x_2(t))x_2^3(t) \end{pmatrix} \quad (1.32)$$

Nous supposons que les variables d'état sont bornées :

$$x_1(t) \in [-1, 1] \quad \text{et} \quad x_2(t) \in [-1, 1]$$

Le modèle (1.32) peut être réécrit comme suit :

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -1 & x_1(t)x_2^2(t) \\ (3 + x_2(t))x_2^2(t) & -1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \quad (1.33)$$

L'équation (1.33) présente deux non-linéarités qu'on notera $z_1(x(t))$ et $z_2(x(t))$ respectivement tels que :

$$\begin{cases} z_1(x(t)) = x_1(t)x_2^2(t) \\ z_2(x(t)) = (3 + x_2(t))x_2^2(t) \end{cases} \quad (1.34)$$

Nous évaluons ensuite les valeurs minimales de $z_1(t)$ et $z_2(t)$:

$$\begin{cases} \max_{x_1(t), x_2(t)} z_1(t) = 1 \\ \min_{x_1(t), x_2(t)} z_1(t) = -1 \end{cases} , \quad \begin{cases} \max_{x_1(t), x_2(t)} z_2(t) = 4 \\ \min_{x_1(t), x_2(t)} z_2(t) = 0 \end{cases} \quad (1.35)$$

Ainsi, on peut réécrire les termes non-linéaires $z_1(x(t))$ et $z_2(x(t))$ comme suit :

$$\begin{cases} z_1(x(t)) = F_1^1(x(t)) * 1 + F_1^2(x(t)) * (-1) \\ z_2(x(t)) = F_2^1(x(t)) * 4 + F_2^2(x(t)) * 0 \end{cases} \quad (1.36)$$

Avec

$$\begin{cases} F_1^1(x(t)) + F_1^2(x(t)) = 1 \\ F_2^1(x(t)) + F_2^2(x(t)) = 1 \end{cases} \quad (1.37)$$

Et

$$\begin{cases} F_1^1(x(t)) = \frac{z_1(t) + 1}{2} \\ F_1^2(x(t)) = \frac{1 - z_1(t)}{2} \end{cases} , \quad \begin{cases} F_2^1(x(t)) = \frac{z_2(t)}{4} \\ F_2^2(x(t)) = \frac{4 - z_2(t)}{4} \end{cases} \quad (1.38)$$

On aboutit alors à quatre modèles locaux par combinaison des quatre fonctions F_1^1 , F_1^2 , F_2^1 , F_2^2 .

Le multi-modèle équivalent au modèle non-linéaire (1.32) est :

$$\dot{x}(t) = \sum_{i=1}^4 \mu_i(x(t)) A_i x(t) \quad (1.39)$$

Donc, avec :

$$A_1 = \begin{bmatrix} -1 & 1 \\ 4 & -1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -1 & 1 \\ 0 & -1 \end{bmatrix}, \quad A_3 = \begin{bmatrix} -1 & -1 \\ 4 & -1 \end{bmatrix}, \quad A_4 = \begin{bmatrix} -1 & -1 \\ 0 & -1 \end{bmatrix}$$

Et

$$\begin{cases} \mu_1(t) = F_1^1(x(t)) + F_1^2(x(t)), & \mu_3(t) = F_1^2(x(t)) + F_2^1(x(t)) \\ \mu_2(t) = F_1^1(x(t)) + F_2^2(x(t)), & \mu_4(t) = F_1^2(x(t)) + F_2^2(x(t)) \end{cases} \quad (1.40)$$

Après ce bref rappel sur les méthodes d'obtention d'un modèle flou (multi-modèles flous), nous abordons par la suite les différents types de contrôleurs flous.

I.5 Les Différents Types des Contrôleurs Flous

Plusieurs types des contrôleurs flous été proposés pour stabiliser les modèles flous. Ils sont basés sur la stabilité sous contraintes qui été transformer en forme des LMI. Parmi ces lois de commande on peut citer :

I.5.1 Loi de Commande par Retour d'Etat

Le contrôleur flou est donné par la relation :

$$u(t) = -K_e x(t), \quad K_e \in R^{m*n} \quad (1.41)$$

I.5.2 Stabilité Floue Simultanément (FSS)

Ce type est une loi de commande non-linéaire par retour d'état, été développé par [34] qui été basée sur le travail de [35]. La loi de commande est écrite comme suit :

$$u(x) = g_1(x) + g_2(x) \quad (1.42)$$

Cette approche consiste en d'autres techniques pour la synthèse des autres processus. Du point de vue de stabilité, le FSS est plus conservatisme que d'autres lois de commande, comme la loi de commande PDC.

I.5.3 Compensation et Division pour les Modèles Flous (CDF)

Ce type de loi de commande a évité d'utiliser l'intersection entre les modèles flous. Mais la corrélation linéaire entre les matrices d'entrées est recommandée. La forme générale de la loi de commande est :

$$u(t) = -\frac{\sum_i h_i(z(t)) k_i F_i x(t)}{\sum_{i=1} h_i(z(t)) k_i}, \quad k_i > 0 \quad (1.43)$$

Avec F_i sont les gains du contrôleur. Et le système flou de type T-S en boucle fermé est :

$$\dot{x}(t) = \frac{\sum_i \sum_j h_i(z(t)) h_j(z(t)) k_i [A_i - B_i F_j]}{\sum_{i=1} h_i(z(t)) k_i} x(t) \quad (1.44)$$

I.5.4 L'approche de la Compensation Parallèle Distribuée (PDC)

L'idée principale de la conception du contrôleur PDC (Parallel Distributed Compensation) est basée essentiellement sur l'étude de la stabilité à partir de la fonction Quadratique de Lyapunov [26]. L'idée est d'attribuer une règle pour chaque contrôleur depuis la règle du modèle flou de T-S correspondant, pour compenser le modèle. Donc on obtient un contrôleur flou, en plus il partage le même ensemble flou que le modèle flou. La figure (Fig. 1.7) illustre le principe de fonctionnement de la commande PDC.

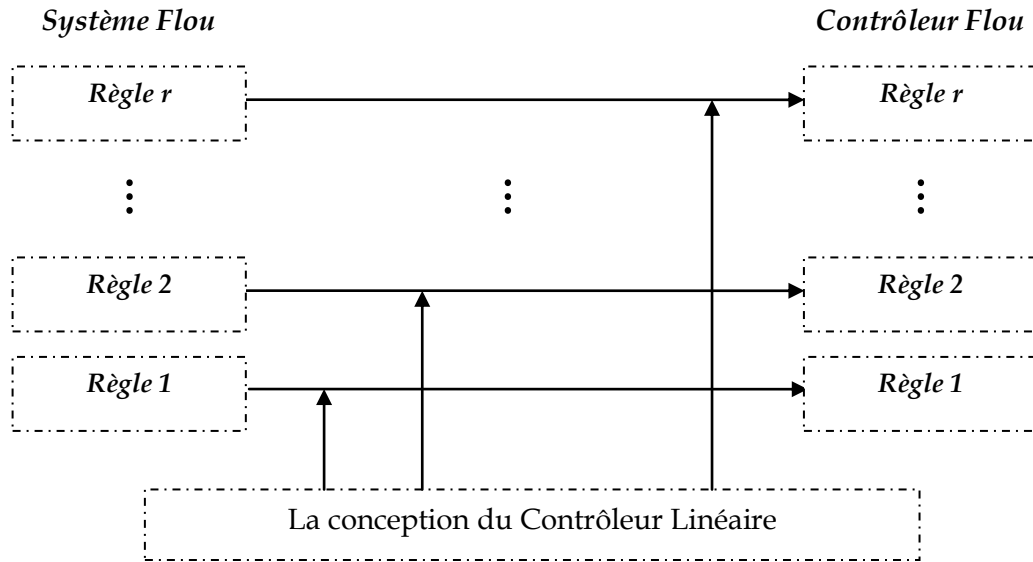


Figure I.7. Principe de la Commande PDC

Pour le système flou (1.15) « (1.17) », le contrôleur PDC flou correspondant est présenté par [26]:

$$\begin{aligned} \text{R\`egle } i : & \text{ Si } z_1(t) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(t) \text{ est } M_{ip} \quad i = 1 \dots r \\ & \text{ Alors } u(t) = -F_i x(t) \end{aligned} \quad (1.45)$$

Qui est un contrôleur par retour d'état à la partie conséquence. Le contrôleur flou est représenté par :

$$u(t) = - \sum_{i=1}^r h_i(z(t)) F_i x(t) \quad (1.46)$$

Ce type de contrôleur est devenu parmi les notions les plus adoptées pour stabiliser les systèmes flous de type T-S sans et sous contraintes, avec la recherche des meilleurs gains F_i qui assure la stabilité en boucle fermé, a fin de résoudre les différentes conditions et contraintes du système par l'outil des LMI.

I.5.5 L'approche de la Non-Compensation Parallèle Distribuée (Non-PDC)

De plus, afin de réduire le conservatisme des conditions LMI , ces résultats ont été étendus dans le contexte non quadratique. Dans ce dessein, le

choix s'est porté sur l'utilisation d'une fonction candidate de Lyapunov non quadratique. [12], [13], [14] et [15]. Ou dans le cas discret [16], [17], [36], [37] et [38] et. La loi de commande non-PDC est donnée par:

$$\begin{aligned} & \text{R\`egle } i: \mathbf{Si } z_1(t) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(t) \text{ est } M_{ip} \\ & \text{Alors } u(t) = - \left(\sum_{i=1}^r Y_i \right) \left(\sum_{i=1}^r G_i \right)^{-1} x(t) \quad i = 1 \dots r \end{aligned} \quad (1.47)$$

Ainsi, la loi de commande Non-PDC peut être réécrite sous la forme :

$$u(t) = - \left(\sum_{i=1}^r h_i(z(t)) Y_i \right) \left(\sum_{i=1}^r h_i(z(t)) G_i \right)^{-1} x(t) \quad (1.48)$$

Des nouveaux résultats sont disponibles aujourd'hui pour ce type de contrôleur [19].

Des recherches sous forme d'une comparaison entre l'approche du PDC et Non-PDC ont été faites pour montrer l'efficacité de chaque approche soit dans le cas continu par [39] ou dans le cas discret par [21].

I.6 Conclusion

Dans ce chapitre, et après la présentation de quelques définitions et concepts de la logique floue, basée essentiellement sur le modèle Takagi-Sugeno, ainsi que les techniques pour obtenir un modèle T-S flou linéaire depuis un système non-linéaire, nous avons présenté par la suite les différents types des contrôleurs flous et nous focalisons particulièrement sur les lois de commande flous PDC et Non-PDC. C'est ce dernier point qui fera l'objet des travaux menés dans les chapitres qui suivent.

Chapitre II

Commande Prédicative à base de Modèle «MPC» et Inégalités Matricielles Linéaires «LMI»: Outils d'Observation, d'Analyse et de Synthèse

Dans ce chapitre, on continuera de présenter les principales notions et concepts qu'on utilisera plus tard dans ce travail. Nous débutons dans ce chapitre par la plus célèbre technique dans le domaine d'automatique (MPC), ensuite, nous passerons à l'outil de LMI qui devient depuis des années parmi les techniques les plus réputées et les plus indispensables.

II.1 Introduction

Durant ces dernières années, la commande prédictive à base de modèle (MPC : Model Predictive Control) est devenue parmi les plus fameuses techniques de commandes avancées utilisées dans le domaine de l'industriel et l'automatique, grâce à sa tolérance pour différents types de systèmes et la satisfaction des contraintes imposées. Le concept MPC n'est pas une stratégie de commande spécifique mais plutôt un ensemble d'algorithmes qui utilisent explicitement le modèle du système dans un problème d'optimisation. à résoudre, afin de déterminer une séquence optimale de commande satisfaisant les contraintes du système et les performances formulées à travers un certain critère d'optimisation sur une dimension finie ou infini à chaque période d'échantillonnage. Historiquement, il est généralement admis que les notions de base de la MPC ont été introduites par Richalet en 1976 [40]. Suivi par d'autres travaux sur le concept de la commande prédictive, mais la popularité de la MPC dans sa forme actuelle, est due à [41] avec la GPC. D'autres idées ont été perfectionnées le MPC par [42].

Avec une étude intéressante sur la robustesse et la formulation du MPC sous forme LMI a été fait par [4], qui permettra de reformuler n'importe quel problème d'optimisation sous forme d'inégalités matricielles linéaires (LMI). Cet outil de LMI est la version des algorithmes semi définie positive (SDP), qui sont des problèmes convexes [43]. Cette technique permet de résoudre et de faire la synthèse de la commande des nombreux problèmes et systèmes dynamiques. Une approche efficace pour la solution des LMI été faite par le développement d'une méthode du point intérieur pour la programmation linéaire, été introduit par [44]. Une amélioration sur cette technique été faite par l'intégration directe de la méthode du point intérieur aux inégalités linéaires matricielles par [45], après, un outil de programmation commerciale implanté au Matlab été écrire par [46], leur avantage est de résoudre les problèmes d'optimisation convexe par un ordinateur dans un temps raisonnable et acceptable du point de vue du temps réel, cet outil est appelé « LMI Toolbox for Matlab ». Mais, la généralisation des LMI pour n'importe quel système ou contrainte a été proposé par [47], cette méthode est

l'une des plus appliquées dans le domaine de la commande grâce à l'existence des méthodes de résolution numérique efficaces.

Dans ce chapitre, nous présenterons quelques concepts de la commande prédictive tels que : son principe de fonctionnement et ses éléments. Ensuite, nous introduisons l'outil de LMI avec quelques notions de bases et les différents lemmes qui permettent de manipuler les inégalités matricielles afin de les rendre linéaires en les inconnues quand cela est possible.

II.2 La Commande Prédictive

L'idée principale de la commande prédictive se résume dans l'utilisation d'un modèle pour prédire le comportement ou l'état du système, au moins sur un certain horizon, et choisir la meilleure décision au sens d'un certain coût tout en respectant les contraintes. A chaque période d'échantillonnage l'utilisation d'un explicite processus modèle pour calculer le processus d'entrées qui optimise le plan futur souhaité sur un intervalle appelé l'horizon de prédiction. Généralement, un ensemble de commande été calculé mais nous ne prendrons en compte que le premier. Dans la prochaine période d'échantillonnage le problème d'optimisation se reformulera et se résoudra avec les nouveaux paramètres obtenus par le système. Cette procédure est ensuite répétée : c'est le principe de l'horizon glissant ou fuyant. Donc le problème d'optimisation en général peut se mettre soit sous forme d'un programme quadratique pour le cas linéaire ou d'un programme non linéaire pour le cas non linéaire.

Généralement, la commande prédictive consiste en les éléments de base suivants :

- Un modèle de prédiction.
- Une fonction coût à minimiser plus les contraintes.
- Un algorithme d'optimisation, pour calculer la commande future.

Chaque élément a eut plusieurs options considérables. Ce qui donne une variété d'algorithmes de commande prédictive.

II.2.1 Modèle de prédiction

Le modèle de prédiction se compose de deux parties : l'une est décrite par la relation entrées sorties et l'autre par les perturbations et les erreurs de modélisation. Le modèle doit être discret, car la commande prédictive est une commande numérique.

Donc, selon le modèle, il existe plusieurs formes de commande prédictive :

- Commande prédictive linéaire à base de modèle d'état, fonction de transfert ... etc.
- Commande prédictive non linéaire à base de modèle d'état non linéaire ... etc.

II.2.2 Fonction Coût (critère de performance)

La fonction coût pénalise les écarts entre les sorties prédites commandées $y(k+t|k)$ et la trajectoire de référence $y_{ref}(k+t|k)$ en plus des variations du vecteur de commande $\Delta u(k) = u(k) - u(k-1)$. La trajectoire de référence peut dépendre de mesures effectuées à l'instant k , en particulier, son point de départ peut être la mesure de la sortie $y(k)$; ainsi elle peut être un ensemble de point fixe ou trajectoire prédéterminée. La fonction coût est souvent donnée par la forme suivante :

$$J = \sum_{i=1}^{\infty} \left[\left(y_{ref}(k+i) - \hat{y}(k+i|k) \right)^T Q \left(y_{ref}(k+i) - \hat{y}(k+i|k) \right) + u(k+i-1)^T R u(k+i-1) + \Delta u(k+i-1)^T S \Delta u(k+i-1) \right] \quad (2.1. a)$$

Où Q, R et S sont les matrices de pondération, avec Q est définie positive et R et S sont semi défini positive. On considère que juste la première commande est prise en compte parmi l'ensemble de la loi de commande calculée à chaque période d'échantillonnage, donc :

$$\Delta u(k+i-1) = 0, \quad i > 1 \quad (2.1. b)$$

II.2.3 Loi de Commande

Pour obtenir la loi de commande, on remplace le prédicteur dans la fonction coût, d'où l'obtention de la synthèse.

II.2.3.1 Le calcul des prédictions à base de modèle d'état

Dans ce chapitre, nous allons introduire un petit rappel de ce que nous avons cité précédemment en ce qui concerne la commande prédictive, ensuite nous essayerons de faire une synthèse sur cette approche afin d'obtenir la loi de commande. Pour cela, nous prendrons le modèle de base suivant qui est un modèle d'état discret afin de calculer les prédictions et ce comme suit :

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) \end{cases} \quad (2.2)$$

Avec $x(k) \in R^n$ est le vecteur d'états appartenant à un espace d'état $M \subset R^n$, $u(k) \in R^m$ le vecteur d'entrées et $y(k) \in R^l$ le vecteur de sorties du système, qui est soumis aux contraintes au niveau de la commande $u(k)$ et des sorties $y(k)$.

$$\begin{aligned} u_{min} &\leq u(k) \leq u_{max} \\ y_{min} &\leq y(k) \leq y_{max} \end{aligned} \quad (2.3)$$

Tout d'abord, on calcule les n prédictions en développant les équations d'état :

$$x(k+1) = Ax(k) + Bu(k) = Ax(k) + Bu(k-1) + B\Delta u(k)$$

$$x(k+2) = Ax(k+1) + Bu(k+1)$$

$$= A^2x(k) + (AB + B)u(k-1) + (AB + B)\Delta u(k) + B\Delta u(k+1)$$

⋮

$$\begin{aligned} x(k+n) &= A^n x(k) + (A^{n-1}B + \dots + B)u(k-1) + (A^{n-1}B + \dots + B)\Delta u(k) + \dots \\ &\quad + B\Delta u(k+n-1) \end{aligned}$$

D'où le prédicteur optimal sous forme matricielle :

$$\hat{X} = \begin{bmatrix} \hat{x}(k) \\ \hat{x}(k+1) \\ \vdots \\ \hat{x}(k+n-1) \end{bmatrix} = \Psi_0 x(k) + \Gamma_0 u(k-1) + \Lambda \Delta u \quad (2.4)$$

$$\text{Où } \Delta U = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+n-1) \end{bmatrix}, \quad \Psi_0 = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^n \end{bmatrix}, \quad \Gamma_0 = \begin{bmatrix} B \\ AB + B \\ \vdots \\ A^{n-1}B + \dots + B \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} B & 0 & \dots & 0 \\ AB + B & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{n-1}B + \dots + B & A^{n-2}B + \dots + B & \dots & B \end{bmatrix}$$

Aussi on a :

$$\hat{y} = C\hat{x} = \Psi_y x(k) + \Gamma_y u(k-1) + \Lambda_y \Delta U \quad (2.5)$$

Avec $\Psi_y = C\Psi_0$, $\Gamma_y = C\Gamma$, $\Lambda_y = C\Lambda$

Par l'utilisation du (2.1.b) le prédicteur de sortie (2.5) peut l'écrire aussi comme suit :

$$\hat{y}(k+n|k) = CA^n x(k) + \sum_{i=0}^{n-1} CA^{n-i-1} Bu(k-1) \quad (2.6)$$

II.2.4 La stabilité de la commande prédictive sous contraintes à horizon infini

La figure ci-dessus montre la différence entre l'horizon fini et infini dans la commande prédictive.

Dans le cas de l'horizon fini à l'instant k une seule trajectoire optimale à suivre pendant tout l'horizon de prédiction H_p . En l'absence des perturbations et avec un modèle parfait (idéal), la prédiction du système à l'instant $k+1$ est exactement à l'état ou elle a été prédite depuis l'instant précédent. Ceci concerne la partie initiale de la trajectoire optimale pendant l'horizon de prédiction depuis l'instant $k+1$ à l'instant $k+1+H_p$ qui convient à la trajectoire optimale calculé

précédemment. Mais avec une nouvelle information qui été introduite maintenant, et qui n'été pas considérée par la conception de l'optimisation du problème, à l'intervalle entre l'instant $k + Hp$ et l'instant $k + 1 + Hp$, la présence de cette intervalle peut mener vers une trajectoire optimale complètement différente par rapport à celle qui été calculée à l'instant précédent [48].

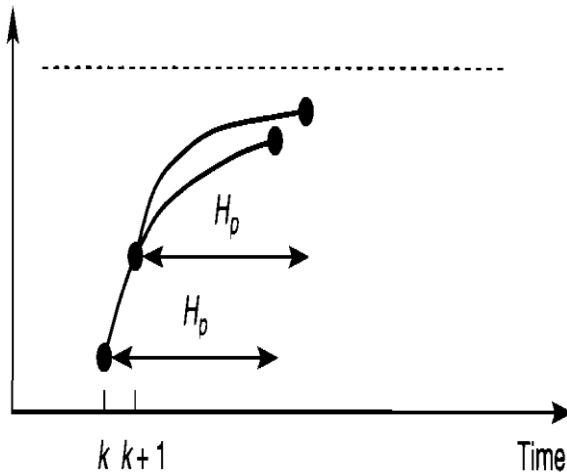


Fig. II.1 L'horizon de prédiction fini

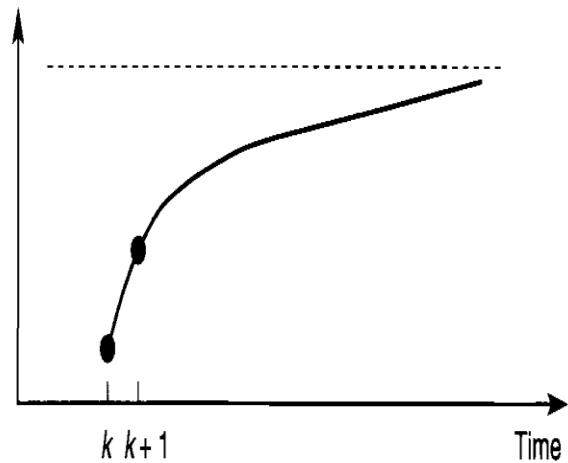


Fig. II.2 L'horizon de prédiction infini

Pour le cas de l'horizon infini, à l'instant $k + 1$ pas de nouvelles informations introduites au problème de l'optimisation, donc la trajectoire optimale est la même qui été calculée précédemment. C'est le célèbre principe de Bellman d'optimalité, avec le même enchaînement d'états pour n'importe quelle trajectoire optimale depuis le point de début. Ce qui n'est pas appliqué dans le cas de l'horizon fini à cause de l'optimisation du problème qui diffère à chaque instant et qui nous conduit au fait que la fonction coût diminue quand k augmente dans l'horizon infini. Ce qui nous permettra d'utiliser la fonction de Lyapunov pour étudier la stabilité du système [48].

II.2.5 La commande prédictive sous contraintes sous forme des LMIs

Comme nous l'avons mentionné, la formulation du MPC sous forme LMI qui a été introduite par [4], permettra de reformuler n'importe quel problème d'optimisation sous forme d'inégalités matricielles linéaires (LMI) se basant sur la fonction de Lyapunov

Quadratique et la loi de commande (1.41). On utilise pour cela la démonstration que nous avons avancée [4].

On commence d'abord par mettre le problème d'optimisation sous forme LMI :

Donc, le problème de minimisation d'après (3.7) est équivalent au (3.18)

$$V(x(k/k)) = x^T(k/k) P x(k/k) \quad P > 0 \quad (2.7)$$

$$\min_{\gamma, P} \gamma \quad (2.8)$$

$$x^T(k/k) P x(k/k) \leq \gamma \Leftrightarrow \gamma - x^T(k/k) P x(k/k) \geq 0 \quad (2.9)$$

Avec $Q = \gamma P^{-1} > 0$, avec l'utilisant du complément de Schur l'inégalité (2.9) est exprimée par :

$$\begin{bmatrix} 1 & x^T(k/k) \\ x(k/k) & Q \end{bmatrix} > 0 \quad (2.10)$$

A chaque instant d'échantillonnage k , la fonction de Lyapunov précédente doit satisfaire l'inégalité suivante pour chaque $i > 0$:

$$V(x(k+i+1/k)) - V(x(k+i/k)) \leq -[X(k+i) + U(k+i)] \quad (2.11)$$

$$V(x(k+i+1/k)) - V(x(k+i/k)) \leq -J_\infty(k) \quad (2.12)$$

En substituant (2.1) dans l'inégalité (2.12) on obtient:

$$\begin{aligned} & V(x(k+i+1/k)) - V(x(k+i/k)) \\ & \leq -[(x^T(k+i/k) Q_0 x(k+i/k)) + (u^T(k+i/k) R_0 u(k+i/k))] \end{aligned}$$

En substituant (2.7) dans l'inégalité précédente on obtient:

$$\begin{aligned} & [x^T(k+i+1/k) P x(k+i+1/k)] - [x^T(k+i/k) P x(k+i/k)] \\ & < -[(x^T(k+i/k) Q_0 x(k+i/k)) + (u^T(k+i/k) R_0 u(k+i/k))] \end{aligned}$$

$$x^T(k + i/k) [(A_i - B_i F_j)^T P (A_i - B_i F_j) - P + Q_0 + F_j^T R_0 F_j] x(k + i/k) \leq 0$$

Cette inégalité est équivalente à l'inégalité suivante :

$$(A_i - B_i F_j)^T P (A_i - B_i F_j) - P + Q_0 + F_j^T R_0 F_j \leq 0$$

En substituant $Q = \gamma P^{-1}$, $Q > 0$ and $Y = FQ$ dans l'inégalité précédente on obtient l'inégalité suivante:

$$\begin{bmatrix} Q & (A_i Q - B_i Y)^T & Q^T Q_0^{1/2} & Y^T R_0^{1/2} \\ (A_i Q - B_i Y) & Q & 0 & 0 \\ Q_0^{1/2} Q & 0 & \gamma I & 0 \\ R_0^{1/2} Y & 0 & 0 & \gamma I \end{bmatrix} \geq 0 \quad (2.13)$$

Après, On passer de mettre les contraintes sous forme LMI :

- Les contraintes sur la commande :

Les contraintes sur la sortie représentent des spécifications sur le comportement. On considérera également une contrainte de type norme euclidienne donnée par :

$$\text{On a: } |u_h(k + i/k)| \leq u_{h,max}, \quad i \geq 0, \quad h = 1, 2, \dots, q$$

$$\|u(k + i/k)\|_{max} \triangleq \max_i u_i(k + i/k)$$

$$\max_{i>0} \|u(k)\|_{max} = \max_{i>0} \|F_i x(k)\|_{max}$$

Avec $Q = \gamma P^{-1} > 0$. On a $x(k) = a = Cst$. Pour chaque $i \geq 0$ danc:

$$\max_{i>0} \|u(k)\|_{max} \geq \max_{a \in \mathbb{R}} |F_i a|^2 \Leftrightarrow \max_{i>0} \|u(k)\|_{max} \geq \|F_i P_i^{-1/2}\|_2^2$$

$$u_{max} \geq \bar{\sigma}[F_i Q^{-1/2}] \Leftrightarrow u_{max}^2 I \geq Q^{1/2} F_i^T F_i Q^{1/2}$$

En utilisant le complément de Schur, l'inégalité précédente est équivalente à la LMI suivante :

$$\begin{bmatrix} u_{max}^2 I & F_j \\ F_j^T & Q \end{bmatrix} \geq 0 \quad (2.14)$$

- Les contraintes sur la sortie :

Les contraintes sur la sortie représentent des spécifications sur le comportement. On considérera également une contrainte de type norme euclidienne donnée par :

$$\text{On a: } |y_h(k + i/k)| \leq y_{h,max} , \quad i \geq 0, \quad h = 1, 2, \dots, q$$

$$\|y(k + i/k)\|_{max} \triangleq \max_i y_i(k + i/k)$$

$$\max_{i>0} \|y(k)\|_{max} = \max_{i>0} \|C_i(A_i - B_i F_i)x(k)\|_{max}$$

Avec $Q = \gamma P^{-1} > 0$. On a $x(k) = a = Cst$. Pour chaque $i \geq 0$ on considère la norme euclidienne donc:

$$\max_{i>0} \|y(k)\|_{max} \geq \max_{a \in \mathfrak{R}} \|C_i(A_i - B_i F_i)a\|_2^2 \Leftrightarrow y_{max} \geq \max_{a \in \mathfrak{R}} \|C_i(A_i - B_i F_i)a\|_2^2$$

$$y_{max} \geq \bar{\sigma}[(C_i(A_i - B_i F_i))Q^{-1/2}] \Leftrightarrow y_{max}^2 I \geq Q^{1/2}[(A_i - B_i F_i)^T C_i^T C_i(A_i - B_i F_i)]Q^{1/2}$$

En utilisant le complément de Schur, l'inégalité précédente est équivalente à la LMI suivante :

$$\begin{bmatrix} y_{max}^2 I & C_i(A_i - B_i F_j) \\ (A_i - B_i F_j)^T C_i^T & P \end{bmatrix} \geq 0 \quad (2.15)$$

II.3 Les Inégalités Matricielles Linéaires

Nous allons essayer de détailler les manipulations nécessaires pour transformer le problème d'optimisation en LMIs dans le cas d'un système flou.

Définition 1 : [47] une inégalité matricielle linéaire est une inégalité matricielle sous la forme suivante :

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i > 0 \quad (2.16)$$

Avec $x(t) = [x_1(t), \dots, x_m(t)]^T$ est le vecteur des variables à trouver, et les matrices symétriques $F_i = F_i^T \in \mathbb{R}^{n \times n}$, $i = 0, \dots, m$, sont des matrices données. Pour qu'on ait une solution d'inégalité (Equa LMI1) il faut que $F(x)$ soit définie positive, i.e. toutes les valeurs propres sont positives. Donc, LMI (Equa LMI1) est une contrainte convexe pour x , et l'ensemble $\{x | F(x) > 0\}$ est convexe. On peut dire aussi que (Equa LMI1) est une matrice diagonale si $F_1(x) > 0, F_2(x) > 0, \dots, F_m(x) > 0$, et ces contraintes sont convexes. Donc, les deux expressions sont équivalentes.

$$\begin{bmatrix} F_1(x) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & F_m(x) \end{bmatrix} > 0 \quad (2.17)$$

Par la suite, nous allons présenter les principes de bases de cette approche.

II.3.1 Quelques Problèmes LMI

On distingue trois types de problèmes d'optimisation convexe rencontrés sous forme des LMI :

II.3.1.1 Problème de Faisabilité

Trouver un vecteur $x(t) = [x_1(t), \dots, x_m(t)]^T$ tel que $F(x) > 0$. Ce problème est résolu généralement en cherchant le vecteur $x(t) = [x_1(t), \dots, x_m(t)]^T$ minimisant le scalaire τ tel que :

$$-F(x) < \tau * I \quad (2.18)$$

Si la valeur minimale de τ est négative alors le problème sera faisable.

II.3.1.2 problème de Valeur Propre (EVP : Eigen-Value Problem)

L'idée est de minimiser la plus grande valeur propre d'une matrice symétrique sous une contrainte de type LMI :

$$\begin{array}{l} \text{Minimiser } \lambda \\ \text{sous les contraintes } \end{array} \begin{cases} \lambda I - A(x) > 0 \\ B(x) > 0 \end{cases} \quad (2.19)$$

II.3.1.3 problème de Valeur Propre Généralisées (General EVP)

L'idée aussi est de minimiser la plus grande valeur propre généralisée d'une paire de matrices par rapport à une contrainte de type LMI :

$$\begin{array}{l} \text{Minimiser } \lambda \\ \text{sous les contraintes } \end{array} \begin{cases} \lambda B(x) - A(x) > 0 \\ B(x) > 0 \\ C(x) > 0 \end{cases} \quad (2.20)$$

II.3.2 Les Techniques d'Analyse et Transformations Matricielles

Dans le cas général, les modèles étudiés, les contraintes et les conditions de stabilité ne sont pas donnés sous forme des LMI directement. Pour cela, ils nécessitent quelques transformations matricielles. Donc nous allons présenter quelques techniques de transformations matricielles utiles pour les résultats obtenus dans ce mémoire.

Généralement, les problèmes lesquels nous avons étudié ne sont pas forcément convexes, ils nécessitent quelques transformations en un problème convexe. Des inégalités non-linéaires convexes peuvent être transformées en LMI.

II.3.2.1 Complément de Schur

Lemme 1 : [43] soient $R(x) \in \mathfrak{R}^{m \times m}$ une matrice définie positive et $R(x) = R(x)^T$, $Q(x) \in \mathfrak{R}^{m \times n}$ une matrice de rang plein en ligne et $Q(x) = Q(x)^T$ et $S(x) \in \mathfrak{R}^{n \times n}$ une matrice quelconque. Les trois matrices sont dépendantes de façon affinée du paramètre x . Les deux inégalités suivantes sont équivalentes :

$$1. Q(x) - S(x)R(x)^{-1}S(x)^T > 0, \quad R(x) > 0 \quad (2.21)$$

$$2. \begin{bmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{bmatrix} > 0 \quad (2.22)$$

- Donc, pour le cas général, par l'utilisation de la propriété dans [25], le complément de Schur généralisé devient :

Soit les deux matrices $U(x)$ et $M(x)$ avec les matrices précédentes, les deux inégalités sont équivalentes :

Avec $R(x) > 0$ et $M(x) > 0$

$$1. Q(x) - S(x)R(x)^{-1}S(x)^T - U(x)M(x)^{-1}U(x)^T > 0, \quad (2.23)$$

$$2. \begin{bmatrix} Q(x) & S(x) & U(x) \\ S(x)^T & R(x) & 0 \\ U(x)^T & 0 & M(x) \end{bmatrix} > 0 \quad (2.24)$$

II.3.2.2 Congruence

Dans certains cas particuliers, des inégalités non-linéaires convexes certaines propriétés permettent d'introduire des variables supplémentaires à rechercher.

Ces propriétés sont présentées par les deux inégalités :

- $A^T P A - Q < 0$,
- $A^T P + P A + Q < 0$,

Pour la première inégalité nous avons les résultats suivants :

Lemme2 : Soient A, G, L, P et Q des matrices de taille appropriée. Les inégalités suivantes sont équivalentes :

$$1. A^T P A - Q < 0, \quad P > 0 \quad (2.25)$$

$$2. \begin{bmatrix} -Q & A^T P \\ P A & -P \end{bmatrix} < 0 \quad (2.26)$$

$$3. \exists G \begin{bmatrix} -Q & -A^T G^T \\ -G A & -G - G^T + P \end{bmatrix} < 0, \quad P > 0 \quad (2.27)$$

$$4. \exists G, L \begin{bmatrix} -Q + A^T L^T + LA & -L + A^T G^T \\ -L^T + GA & -G - G^T + P \end{bmatrix} < 0, \quad P > 0 \quad (2.28)$$

Remarque :

- l'inégalité (2.27) se trouve dans [49].

- l'inégalité (2.28) se trouve dans [50].

Pour la transaction entre les inégalités nous avons :

(2.25) \Leftrightarrow (2.26): Complément de Shur.

(2.26) \Rightarrow (2.27) et(2.28): Il suffit de choisir $G = P$ et $L = 0$

(2.27)et(2.28) \Rightarrow (2.25): Par congruence avec $[I \ A^T]$

Pour la seconde inégalité, nous avons le résultat suivant :

Lemme 3 : [50]. Soient A, G, L, P et Q des matrices de taille appropriée. Les inégalités suivantes sont équivalentes :

$$1. A^T P + PA + Q < 0, \quad (2.29)$$

$$2. \exists G, L \begin{bmatrix} A^T L^T + LA + Q & P - L + A^T G \\ P - L^T + G^T A & -G - G^T \end{bmatrix} < 0, \quad (2.30)$$

Aussi, Pour la transaction entre les inégalités nous avons :

(2.30) \Rightarrow (2.29): Par congruence avec $[I \ A^T]$

(2.29) \Rightarrow (2.30): si (2.29) est vérifiée, $\exists \varepsilon > 0$ tel que $A^T P + PA + Q + \frac{\varepsilon}{2} A^T A < 0$,

en utilisant le complément de Schur il nous aurons :

$$\begin{bmatrix} A^T P^T + PA + Q & \varepsilon A^T \\ \varepsilon A & -2\varepsilon I \end{bmatrix} < 0. \text{ Le résultat s'obtient en posant dans (2.30) } L = P \text{ et } G = \varepsilon I.$$

Note : Les conditions nécessaires pour

- Lemme 2 : (2.26) \Rightarrow (2.27) et(2.28),

- Lemme 3 : (2.29) \Rightarrow (2.30)

Sont garanties si les matrices L, G ne sont pas contraintes.

II.3.2.3 Lemme d'Elimination

Ce lemme est aussi connu sous le nom de projection, il est très utilisé dans les formulations LMI. Dans de nombreux problèmes de commande nous pourrions obtenir des contraintes LMI par élimination de variables. Dans ce cas on obtient un problème de faisabilité de taille réduite. Pour ce faire, on utilise les deux versions du lemme de Finsler.

Lemme 5 (Finsler) : [51]

Soit un vecteur $x(t) = [x_1(t), \dots, x_m(t)]^T \in \mathfrak{R}^n$, et des matrices $Q = Q^T \in \mathfrak{R}^{n \times n}$, $M \in \mathfrak{R}^{m \times n}$ et $N \in \mathfrak{R}^{m \times n}$ telles que $\text{rang}(M) < n$ et $\text{rang}(N) < n$. Les quatre expressions suivantes sont équivalentes :

$$1. x^T Q x < 0 \quad \forall x \neq 0 \text{ tels que } Nx = 0 \quad (2.31)$$

$$2. \text{ Les compléments orthogonaux } M_{\perp} \text{ et } N_{\perp} \text{ de } M \text{ et } N, \text{ respectivement, vérifient :} \\ M_{\perp}^T Q M_{\perp} < 0 \quad \text{et} \quad N_{\perp}^T Q N_{\perp} < 0 \quad (2.32)$$

3. Il existe un scalaire réel $\sigma \in \mathfrak{R}$ tel que :

$$Q - \sigma M^T M < 0 \quad \text{et} \quad Q - \sigma N^T N < 0 \quad (2.33)$$

4. Il existe une matrice réelle $X \in \mathfrak{R}$ tel que :

$$Q + N^T M + M^T X^T N < 0 \quad (2.34)$$

II.3.2.4 Le Carré Matriciel et ses Dérivées

Lemme 6 : [52] Soit X et Y deux matrices aux tailles appropriées.

- Soit γ une constante positive on a :

$$X^T Y + Y^T X \leq \gamma X^T X + \gamma^{-1} Y^T Y \quad (2.35)$$

Pour réduire le conservatisme, cette propriété peut également être écrite en fonction d'une matrice définie positive $Q > 0$, à partir de lemme suivant :

Lemme 7: [53] Soit X et Y deux matrices aux tailles appropriées. L'inégalité suivante est toujours vraie pour toute matrice $Q = Q^T > 0$:

$$XY^T + YX^T \leq XQX^T + YQ^{-1}Y^T \quad (2.36)$$

Preuve [52] :

$$Q > 0 \Rightarrow (QX - Y)^T Q^{-1} (QX - Y) \geq 0 \Leftrightarrow XY^T + YX^T \leq XQX^T + YQ^{-1}Y^T \quad (2.37)$$

Les propriétés de Lemme 6 et 7 ont l'avantage de pouvoir éliminer les termes anti-diagonaux et les ramener dans les blocs diagonaux, comme c'est traité dans l'exemple suivant :

Exemple :

On souhaite éliminer les termes anti-diagonaux X et Y de la matrice suivant :

$$\begin{bmatrix} A & W^T + Y^T X^T \\ W + XY & B \end{bmatrix} < 0 \quad (2.38)$$

L'équation (2.38) peut s'écrire sous la forme :

$$\begin{bmatrix} A & W^T \\ W & B \end{bmatrix} + \begin{bmatrix} 0 & Y^T X^T \\ XY & 0 \end{bmatrix} < 0 \quad (2.39)$$

Ou

$$\begin{bmatrix} 0 & Y^T X^T \\ XY & 0 \end{bmatrix} = \begin{bmatrix} 0 \\ X \end{bmatrix} \begin{bmatrix} Y & 0 \end{bmatrix} + \begin{bmatrix} Y^T \\ 0 \end{bmatrix} \begin{bmatrix} 0 & X^T \end{bmatrix} \quad (2.40)$$

En utilisant la propriété (2.36), on obtient l'inégalité :

$$\begin{bmatrix} 0 \\ X \end{bmatrix} \begin{bmatrix} Y & 0 \end{bmatrix} + \begin{bmatrix} Y^T \\ 0 \end{bmatrix} \begin{bmatrix} 0 & X^T \end{bmatrix} < \begin{bmatrix} 0 \\ X \end{bmatrix} Q \begin{bmatrix} 0 & X^T \end{bmatrix} + \begin{bmatrix} Y^T \\ 0 \end{bmatrix} Q^{-1} \begin{bmatrix} Y & 0 \end{bmatrix} \quad (2.41)$$

L'inégalité (2.38) est équivalente à l'inégalité suivante :

$$\begin{bmatrix} A + Y^T Q^{-1} Y & W^T \\ W & B + XQX^T \end{bmatrix} < 0 \quad (2.42)$$

Dans la partie de simulation, nous utiliserons *YALMIP Toolbox*, la version des solveurs développée par [54] pour résoudre les problèmes sous contraintes LMI.

II.4 Conclusion

Ce deuxième chapitre nous a permis dans un premier temps, de présenter la commande prédictive avec les différentes phases de calcul de la sortie prédite et la fonction de coût. Dans la deuxième partie de ce chapitre, nous avons proposé des solutions en utilisant des contraintes LMI dans des problèmes d'optimisation. Par ailleurs, nous avons présenté également quelques résultats concernant la relaxation des LMI. Des propriétés sur LMI ont été exposées pour faciliter la tâche de traitement du problème d'optimisation et sous mettre se forme d'inégalités matricielles linéaires.

Chapitre III

Commande Prédicative Robuste des Modèles T-S

Dans ce chapitre, on essayera de combiner entre les deux techniques : la commande prédictive à modèle et la logique floue de type T-S Discret. Et de mettre les résultats dans une formulation mathématique pour la recherche de meilleures solutions des systèmes proposés, cette formulation consiste en les inégalités Matricielles Linéaires (LMI).

III.1 Introduction

Dans ce chapitre, nous étudions une approche de commande floue avec retour d'état PDC et non PDC en utilisant une technique de commande prédictive. La conception est effectuée à chaque instant d'échantillonnage par la minimisation d'une borne supérieure d'une fonction objective sur un horizon infini, en suivant la méthodologie MPC. La stratégie permet d'inclure des contraintes sur les variables du processus (limites sur l'entrée et la sortie) pendant l'étape de calcul du contrôleur.

La synthèse de l'approche PDC pour les deux cas ; quadratique (PDC) et Non-Quadratique (Non-PDC) avec le MPC, est basée sur une approche de Lyapunov qui consiste à vérifier un ensemble d'inégalités matricielles linéaires LMI [21].

Ce chapitre s'articule sur deux parties. Dans la première partie, nous présentons l'objectif de la loi de commande du contrôleur Non-PDC et les étapes de synthèse du système de commande prédictive sous contraintes. Cette partie facilite la recherche d'une solution aux problèmes LMI et conduit à une réduction du conservatisme. Dans la deuxième partie, nous nous intéressons uniquement à la synthèse d'une loi de commande de type PDC ainsi qu'aux résultats de stabilisation utilisant ce type de loi de commande.

III.2 Modèle T-S et la Fonction Coût

Soit le système flou T-S discret donnée par:

$$\begin{aligned} \text{R\`egle } i : \mathbf{Si} \quad & z_1(k) \text{ est } M_{i1} \text{ et } \dots \text{ et } z_p(k) \text{ est } M_{ip} \\ \text{Alors} \quad & \begin{cases} x(k+1) = A_i x(k) + B_i u(k) \\ y(k) = C_i x(k) \end{cases} \quad i = 1 \dots r \end{aligned} \quad (3.1)$$

La classe des modèles TS considérée est donnée par :

$$\begin{cases} x(k+1) = \sum_{i=1}^r h_i(z(k)) (A_i x(k) + B_i u(k)) \\ y(k) = \sum_{i=1}^r h_i(z(k)) C_i x(k) \end{cases} \quad i = 1 \dots r \quad (3.2)$$

Avec A_i, B_i et C_i les matrices d'états du système.

Nous considérons le problème d'optimisation min-max avec une fonction de coût prédictive sur un horizon infini :

$$\min_{u(k+i/k)=F(x)x(k+i/k)} \max_{i>0} J_\infty(k) \quad (3.3.a)$$

$$J_\infty(k) = \sum_{i=0}^{\infty} [X(k+i) + U(k+i)] \quad (3.3.b)$$

Avec $\begin{cases} X(k+i) = x^T(k+i/k)Q_0x(k+i/k) \\ U(k+i) = u^T(k+i/k)R_0u(k+i/k)' \end{cases}$

$Q_0 > 0$ et $R_0 > 0$ sont deux matrices de pondération bien définis.

Les restrictions sur le signal de contrôle et sur l'entrée sont données par :

$$\begin{aligned} \|u(k+i/k)\|_2 &\leq u_{max}, & i \geq 0, \\ \|y(k+i/k)\|_2 &\leq y_{max}, & i \geq 1 \end{aligned}$$

III.3 Stabilité et stabilisation du Système Augmenté

L'étude de la stabilité et la stabilisation "la synthèse des contrôleurs flous" pour les modèles T-S sont généralement basées sur la théorie de Lyapunov. Le principe de cette dernière est inspiré d'une réalité physique. De tel sorte que, si l'énergie totale d'un système « qui est une quantité scalaire » se dissipe au cours du temps, ce système tend vers un point d'équilibre. Ainsi, l'idée de Lyapunov est de considérer une fonction scalaire puis d'examiner sa variation afin d'étudier la stabilité d'un système donné.

Théorème 1: [11].

L'équilibre d'un modèle flou discret, décrit par (3.2), est asymptotiquement stable s'il existe une matrice P commune définie positive telle que :

$$A_i^T P A_i - P < 0, \quad i = 1, 2, \dots, r \quad (3.4)$$

Afin d'assurer la stabilisation d'un modèle TS, plusieurs lois de commande ont été proposées. Les plus répandues se basent sur le principe de compensation parallèle distribuée (PDC, Parallel Distributed Compensation) et afin de réduire

le conservatisme des conditions LMI proposées, nous avons jugé utile d'étendre ces résultats dans le contexte non quadratique basée sur la loi de commande Non-PDC.

Nous nous intéressons d'abord, à la stabilité et à la stabilisation Non-Quadratique « Non-PDC » des modèles flous T-S.

III.4 Synthèse de la Commande avec l'Approche Non-PDC

III.4.1 Problématique : Construction du Système Augmenté

Poursuivant l'objectif de fournir des conditions LMI de moindre conservatisme, nous allons présenter une extension de l'approche quadratique qui se base sur une fonction de Lyapunov non quadratique. Une loi de commande Non-PDC par retour d'état est donnée sous sa forme compacte par :

$$u(k) = - \left(\sum_{j=1}^r h_j(z(t)) Y_j \right) \left(\sum_{j=1}^r h_j(z(t)) G_j \right)^{-1} x(k) \quad (3.5)$$

Notation :

Soit Y_j, G_j des matrices de taille appropriée et $h_j(z(t))$ des fonctions scalaires ayant la propriété de somme convexe, on pose

$$Y_z = \sum_{j=1}^r h_j(z(k)) Y_j, \quad G_z = \sum_{j=1}^r h_j(z(k)) G_j$$

De manière classique, l'expression du modèle TS en boucle fermée s'obtient en substituant la loi de commande (3.5) dans (3.2), c'est-à-dire:

$$\begin{cases} x(k+1) = (A_z - B_z Y_z G_z^{-1}) x(k) \\ y(k) = C_z x(k) \end{cases} \quad (3.6)$$

Avec

$$A_z = \sum_{i=1}^r h_i(z(k)) A_i, \quad B_z = \sum_{i=1}^r h_i(z(k)) B_i, \quad C_z = \sum_{i=1}^r h_i(z(k)) C_i,$$

On définit une fonction de Lyapunov non-quadratique comme suit :

$$V(x(k/k)) = x^T(k/k) S_{zz}^{-1} x(k/k) \quad (3.7)$$

$$V(x(k/k)) = x^T(k/k) \left(\sum_{i=1}^m \sum_{j=1}^m h_i h_j S_{ij} \right)^{-1} x(k/k) \quad (3.8)$$

Avec $S > 0$, $V(x)$ majore de $J(k)$ et γ sa borne supérieure :

$$\min_{u(k+i/k), i>0} \gamma \quad (3.9)$$

$$-V(x(k/k)) \leq -J_\infty(k) \quad (3.10)$$

$$\max_{A(k+i), B(k+i) \in \Omega, i>0} J_\infty(k) \leq V(x(k/k)) \leq \gamma \quad (3.11)$$

Le problème de contrôle prédictif peut être résolu par le problème LMI suivant :

Théorème 2 : [21]

Le système TS (3.6) est globalement asymptotiquement stable via la loi de commande non-PDC (3.5) et minimise la fonction de coût (3.3), s'il existe des matrices $S_{zz} > 0$, G_z , Y_z et H telles que les LMI suivantes soient vérifiées.

$$\min_{S_{zz}, G_z, Y_z} \gamma \quad (3.12)$$

$$\begin{bmatrix} \gamma & x^T(k/k) \\ x(k/k) & S_{zz} \end{bmatrix} > 0 \quad (3.13)$$

$$\begin{bmatrix} G_z^T + G_z - S_{zz} & (A_z G_z - B_z Y_z)^T & Y_z^T R_0 & G_z^T Q_0 \\ (A_z G_z - B_z Y_z) & S_{zz} & 0 & 0 \\ R_0 Y_z & 0 & R_0 & 0 \\ Q_0 G_z & 0 & 0 & Q_0 \end{bmatrix} > 0 \quad (3.14)$$

Une fois le problème LMI précédent est résolu, le contrôleur de retour d'état est déterminé par

$$F = -Y_z G_z^{-1}.$$

Preuve :

A chaque instant d'échantillonnage k , la fonction de Lyapunov précédente doit satisfaire l'inégalité suivante pour chaque $i > 0$:

$$V(x(k+i+1/k)) - V(x(k+i/k)) \leq -[X(k+i) + U(k+i)] \quad (3.15)$$

$$V(x(k+i+1/k)) - V(x(k+i/k)) \leq -J_\infty(k) \quad (3.16)$$

Si la fonction objectif (3.6) est définie correctement, c'est-à-dire, a $i \rightarrow \infty, x(i/k) \rightarrow 0 \Rightarrow x(\infty/k) = 0$, et $V(x(\infty/k)) = 0$; en sommant (3.8) depuis $i = 0$ jusqu'à $i = \infty$, on obtient (3.10).

La fonction non quadratique (3.7) représente donc une borne supérieure de la fonction objective (3.3). Par la suite, l'approche proposée consistera à synthétiser à chaque instant d'échantillonnage k , un contrôleur par retour d'état (3.5) qui minimise $V(x(k/k))$, tout en satisfaisant la condition (3.9). Qui nous donne :

$$\max_{A_i, B_i, i > 0} J_\infty(k) \leq V(x(k/k)) \leq \gamma \quad (3.17)$$

Donc le problème de minimisation d'après (3.3), (3.7), et (3.17) devient :

$$\min_{S_{zz}, G_z, Y_z} \gamma \quad (3.18)$$

Avec :

$$x^T(k/k) S_{zz}^{-1} x(k/k) \leq \gamma \Leftrightarrow \gamma - x^T(k/k) S_{zz}^{-1} x(k/k) \geq 0 \quad (3.19)$$

En utilisant le complément de Schur, l'inégalité (3.19) est exprimée par :

$$\begin{bmatrix} \gamma & x^T(k/k) \\ x(k/k) & S_{zz} \end{bmatrix} > 0 \quad (3.20)$$

D'après (3.11) on a :

$$\begin{aligned} & V(x(k+i+1/k)) - V(x(k+i/k)) \\ & \leq -[(x^T(k+i/k)Q_0x(k+i/k)) + (u^T(k+i/k)R_0u(k+i/k))] \end{aligned}$$

En substituant (3.7) dans l'inégalité précédente on obtient:

$$\begin{aligned} & [x^T(k+i+1/k)S_{zz}^{-1}x(k+i+1/k)] - [x^T(k+i/k)S_{zz}^{-1}x(k+i/k)] \\ & < -[(x^T(k+i/k)Q_0x(k+i/k)) + (u^T(k+i/k)R_0u(k+i/k))] \end{aligned}$$

En remplaçant $u(k+i/k)$ par (3.5) on a:

$$\begin{aligned} & x^T(k+i/k)[x^T(k+1/k)S_{zz}^{-1}x(k+1/k) - S_{zz}^{-1}x(k+i/k)] \\ & < -x^T(k+i/k)[Q_0 + G_z^{-T}Y_z^T R_0 Y_z G_z^{-1}]x(k+i/k) \end{aligned}$$

Par le remplacement de $x(k+1/k)$ par (3.6) on obtient:

$$\begin{aligned} & x^T(k+i/k)[(A_z - B_z Y_z G_z^{-1})^T S_{zz}^{-1}(A_z - B_z Y_z G_z^{-1}) - S_{zz}^{-1}]x(k+i/k) \\ & < -x^T(k+i/k)[Q_0 + G_z^{-T}Y_z^T R_0 Y_z G_z^{-1}]x(k+i/k) \end{aligned}$$

Cette inégalité est équivalente à l'inégalité suivante :

$$(A_z - B_z Y_z G_z^{-1})^T S_{zz}^{-1}(A_z - B_z Y_z G_z^{-1}) - S_{zz}^{-1} < -Q_0 - G_z^{-T}Y_z^T R_0 Y_z G_z^{-1} \Leftrightarrow$$

En utilisant une congruence avec la matrice de rang plein G_z il vient :

$$(A_z G_z - B_z Y_z)^T S_{zz}^{-1}(A_z G_z - B_z Y_z) - G_z^T S_{zz}^{-1} G_z < -G_z^T Q_0 G_z - Y_z^T R_0 Y_z \Leftrightarrow$$

En multipliant l'inégalité précédente à gauche par G_z et à droite par G_z on obtient :

$$G_z^T S_{zz}^{-1} G_z - (A_z G_z - B_z Y_z)^T S_{zz}^{-1}(A_z G_z - B_z Y_z) - G_z^T Q_0 G_z - Y_z^T R_0 Y_z > 0 \quad (3.21)$$

Parfois les conditions obtenues ne sont pas linéaires et il n'est pas possible d'utiliser directement des programmes de solution de problèmes LMIs, toutefois,

en faisant usage d'algorithmes de relaxation, dans (3.21) nous allons utiliser la propriété du *Carré Matriciel et ses Dérivées* pour simplifier le terme $G_z^T S_{zz}^{-1} G_z$, comme suit :

$$\begin{aligned} (G_z^T - S_{zz}) S_{zz}^{-1} (G_z - S_{zz}) \geq 0 &\Rightarrow G_z^T S_{zz}^{-1} G_z - G_z^T S_{zz}^{-1} S_{zz} - S_{zz} S_{zz}^{-1} G_z + S_{zz} S_{zz}^{-1} S_{zz} \geq 0 \Leftrightarrow \\ G_z^T S_{zz}^{-1} G_z - G_z^T - G_z + S_{zz} \geq 0 &\Leftrightarrow G_z^T + G_z - S_{zz} \leq G_z^T S_{zz}^{-1} G_z \Leftrightarrow \\ G_z^T + G_z - S_{zz} \leq G_z^T S_{zz}^{-1} G_z &\quad (3.22) \end{aligned}$$

En substituant (3.22) dans (3.21), l'inégalité (3.21) devient :

$$\begin{aligned} G_z^T + G_z - S_{zz} - (A_z G_z - B_z Y_z)^T S_{zz}^{-1} (A_z G_z - B_z Y_z) - G_z^T Q_0 G_z - Y_z^T R_0 Y_z &> 0 \\ G_z^T + G_z - S_{zz} - (A_z G_z - B_z Y_z)^T S_{zz}^{-1} (A_z G_z - B_z Y_z) - G_z^T Q_0 Q_0^{-1} Q_0 G_z - Y_z^T R_0 R_0^{-1} R_0 Y_z &> 0 \end{aligned}$$

En utilisant le complément de Schur et la propriété à [26] on obtient :

$$\begin{bmatrix} G_z^T + G_z - S_{zz} & (A_z G_z - B_z Y_z)^T & Y_z^T R_0 & G_z^T Q_0 \\ (A_z G_z - B_z Y_z) & S_{zz} & 0 & 0 \\ R_0 Y_z & 0 & R_0 & 0 \\ Q_0 G_z & 0 & 0 & Q_0 \end{bmatrix} > 0 \quad (3.23)$$

Définition des contraintes

Les processus réels sont, en général, soumis à des restrictions portant sur les variables du processus. Dans ce travail, nous montrons comment les contraintes peuvent être intégrées dans l'algorithme sous forme de LMIs.

- Les contraintes sur la commande :

Les contraintes sur l'entrée $u(k)$ représentent les limitations physiques des équipements présents dans le processus. On considèrera une contrainte de type norme euclidienne donnée par :

$$|u_h(k + i/k)| \leq u_{h,max} , \quad i \geq 0, \quad h = 1, 2, \dots, p$$

$$\|u(k)\|_{max} \triangleq \max_i u_i(k)$$

$$\max_{i>0} \|u(k)\|_{max} \geq \max_{i>0} \|Y_z G_z^{-1} S_{zz}^{-1} x(k)\|_{max}$$

On a $x(k) = a = Cst.$ Pour chaque $i \geq 0$ danc:

$$\max_{i>0} \|u(k)\|_{max} \geq \max_{a \in \mathbb{R}} |Y_z G_z^{-1} S_{zz}^{-1} a|^2 \Leftrightarrow \max_{i>0} \|u(k)\|_{max} \geq \|(Y_z G_z^{-1} S_{zz}^{-1/2})\|_2^2$$

Par l'utilisation de l'inégalité de Cauchy-Schwarz à [4] et avec $\max_{i>0} \|u(k)\|_{max} = u_{max} = Z.$ L'inégalité précédente devient :

$$Z \geq (Y_z G_z^{-1} S_{zz}^{-1} (Y_z G_z^{-1})^T) \Leftrightarrow Z - (Y_z G_z^{-1} S_{zz}^{-1} (Y_z G_z^{-1})^T) \geq 0$$

En utilisant le complément de Schur on obtient :

$$\begin{bmatrix} Z & Y_z G_z^{-1} \\ (Y_z G_z^{-1})^T & S_{zz} \end{bmatrix} > 0 \quad (3.24)$$

Par l'utilisation du théorème de Congruence avec la matrice en rang plein $\begin{bmatrix} I & 0 \\ 0 & G_z^T \end{bmatrix}$ on obtient :

$$\begin{bmatrix} Z & Y_z \\ Y_z^T & G_z^T + G_z - S_{zz} \end{bmatrix} > 0 \quad (3.25)$$

- Les contraintes sur la sortie :

Les contraintes sur la sortie sont moins strictes que celles qui précèdent puisqu'elles représentent des spécifications sur le comportement. On considérera également une contrainte de type norme euclidienne donnée par :

On a: $|y_h(k + i/k)| \leq y_{h,max} , \quad i \geq 0, \quad h = 1, 2, \dots, q$

$$\|y(k + i/k)\|_{max} \triangleq \max_i y_i(k + i/k)$$

$$\max_{i>0} \|y(k)\|_{max} \geq \max_{i>0} \|C_z(A_z - B_z Y_z G_z^{-1})S_{zz}^{-1}x(k)\|_{max}$$

On a $x(k) = a = Cst$. Pour chaque $i \geq 0$ danc:

$$\begin{aligned} \max_{i>0} \|y(k)\|_{max} &\geq \max_{a \in \mathfrak{R}} |C_z(A_z - B_z Y_z G_z^{-1})S_{zz}^{-1}a|^2 \Leftrightarrow \max_{i>0} \|y(k)\|_{max} \\ &\geq \|(C_z(A_z - B_z Y_z G_z^{-1}))S_{zz}^{-1/2}\|_2^2 \end{aligned}$$

Par l'utilisation de l'inégalité de Cauchy-Schwarz à [4] et avec $\max_{i>0} \|y(k)\|_{max} = Y_{max} = W$. L'inégalité précédente devient :

$$\begin{aligned} W &\geq ((C_z(A_z - B_z Y_z G_z^{-1}))S_{zz}^{-1}(C_z(A_z - B_z Y_z G_z^{-1}))^T) \\ &\Leftrightarrow W - ((C_z(A_z - B_z Y_z G_z^{-1}))S_{zz}^{-1}(C_z(A_z - B_z Y_z G_z^{-1}))^T) \geq 0 \end{aligned}$$

En utilisant le complément de Schur, l'inégalité précédente est équivalente à la LMIs suivante :

$$\begin{bmatrix} -W & -C_z(A_z - B_z Y_z G_z^{-1}) \\ -(A_z - B_z Y_z G_z^{-1})^T C_z^T & S_{zz} \end{bmatrix} > 0 \quad (3.26)$$

Par l'utilisation du théorème de Congruence avec la matrice en rang plein $\begin{bmatrix} I & 0 \\ 0 & G_z^T \end{bmatrix}$ on obtient :

$$\begin{bmatrix} W & C_z(A_z G_z - B_z Y_z) \\ (A_z G_z - B_z Y_z)^T C_z^T & G_z^T + G_z - S_{zz} \end{bmatrix} > 0 \quad (3.27)$$

En reprenant de manière similaire les développements établis aux paragraphes précédents, en l'absence de contraintes sur la commande et sur la sortie, le problème d'optimisation sous contraintes est résumé par le théorème suivant :

Théorème 3 : [21].

Le système en boucle fermée (3.6) est globalement asymptotiquement stable et minimise la fonction de cout (3.3) s'il existe des matrices S_{ij} symétriques définies positive,

Avec les Y_{ij} définie à (3.23), L'équilibre d'un modèle flou discret, décrit par (3.6), est asymptotiquement stable, s'il existe des matrices S_{ij} commune définie positive, G_j , Y_j , et $X_{ii} > 0$ et $X_{ij} = X_{ij}^T$. Telle que les LMI suivantes soient vérifiées et satisférons le problème d'optimisation convexe suivant :

$$\min_{S_{ij}, G_j, Y_j} \gamma \quad (3.28)$$

$$\begin{bmatrix} \gamma & x^T(k/k) \\ x(k/k) & S_{ij} \end{bmatrix} > 0 \quad (3.29)$$

$$Y_{ii} > X_{ii} \quad i \in \{1, \dots, r\} \quad (3.30)$$

$$Y_{ij} + Y_{ij} > X_{ij} + X_{ij}^T \quad i < j \quad i, j \in \{1, \dots, r\}, \quad (3.31)$$

$$\text{Avec: } X_l = \begin{bmatrix} 2X_{11} & (*) & (*) & (*) \\ X_{12} & 2X_{22} & \dots & (*) \\ \vdots & \vdots & \ddots & \vdots \\ X_{1r} & X_{2r} & \dots & 2X_{rr} \end{bmatrix} > 0 \quad (3.32)$$

Avec

$$\begin{bmatrix} Z & Y_j \\ Y_j^T & G_j^T + G_j - S_{ij} \end{bmatrix} > 0 \quad (3.33)$$

$$\begin{bmatrix} W & C_i(A_i G_j - B_i Y_j) \\ (A_i G_j - B_i Y_j)^T C_i^T & G_j^T + G_j - S_{ij} \end{bmatrix} > 0 \quad (3.34)$$

$$Y_{ij} = \begin{bmatrix} G_j^T + G_j - S_{ij} & (A_i G_j - B_i Y_j)^T & Y_j^T R_0 & G_j^T Q_0 \\ (A_i G_j - B_i Y_j) & S_{ij} & 0 & 0 \\ R_0 Y_j & 0 & R_0 & 0 \\ Q_0 G_j & 0 & 0 & Q_0 \end{bmatrix} \quad (3.35)$$

III.5 Synthèse de la Commande avec l'Approche PDC

III.5.1 Problématique : Construction du Système Augmenté

Une loi de commande fréquemment utilisée pour les modèles non linéaires TS est la loi PDC. Elle est donnée par la loi de commande suivante :

$$u(k) = - \sum_{j=1}^r h_j(z(t)) F_j x(t) = - \sum_{j=1}^r h_j(z(t)) Y_j G^{-1} x(k) \quad (3.36)$$

Notation :

Soit Y_j, G des matrices de taille appropriée et $h_j(z(t))$ des fonctions scalaires ayant la propriété de somme convexe, on pose :

$$F_z = \sum_{j=1}^r h_j(z(k)) F_j = \sum_{j=1}^r h_j(z(k)) Y_z G^{-1}$$

Pour obtenir le modèle flou en boucle fermée, on applique la commande PDC au modèle T-S (3.2). Ainsi le modèle flou obtenu est le suivant :

$$\begin{cases} x(k+1) = (A_z - B_z F_z) x(k) \\ y(k) = C_z x(k) \end{cases} \quad (3.37)$$

$$\text{Avec: } A_z = \sum_{i=1}^r h_i(z(k)) A_i, \quad B_z = \sum_{i=1}^r h_i(z(k)) B_i, \quad C_z = \sum_{i=1}^r h_i(z(k)) C_i,$$

Le problème de contrôle prédictif peut être résolu à chaque itération k par le problème LMI suivant :

Théorème 4 : [21]

Le système TS (3.6) est globalement asymptotiquement stable via la loi de commande PDC (3.36) et minimise la fonction de coût (3.3), s'il existe des matrices $Q_z > 0, Y_z$, et G , telles que les LMI suivantes soient vérifiées :

$$\min_{Q_z, G, Y_z} \gamma \quad (3.38)$$

$$\begin{bmatrix} \gamma & x^T(k/k) \\ x(k/k) & Q_z \end{bmatrix} > 0 \quad (3.39)$$

$$\begin{bmatrix} G^T + G - Q_z & (A_z G - B_z Y_z)^T & Y_z^T R_0 & G^T Q_0 \\ (A_z G - B_z Y_z) & Q_z & 0 & 0 \\ R_0 Y_z & 0 & R_0 & 0 \\ Q_0 G & 0 & 0 & Q_0 \end{bmatrix} > 0 \quad (3.40)$$

Preuve :

En considérant la fonction candidate non-quadratique de Lyapunov comme suit :

$$V(x(k/k)) = x^T(k/k) Q_z^{-1} x(k/k) \quad (3.41)$$

En suivant la même démarche que dans le cas non-PDC, on aboutit au résultat suivant :

Les gains de retour de la commande PDC (3.36) sont obtenus en résolvant le problème d'optimisation suivant :

$$\min_{Q_z, Y_z, G} \gamma \quad (3.42)$$

Avec :

$$x^T(k/k) Q_z^{-1} x(k/k) \leq \gamma \Leftrightarrow \gamma - x^T(k/k) Q_z^{-1} x(k/k) \geq 0 \quad (3.43)$$

En appliquant un complément de Schur sur (3.43) on obtient :

$$\begin{bmatrix} \gamma & x^T(k/k) \\ x(k/k) & Q_z \end{bmatrix} > 0 \quad (3.44)$$

On a $V(0) = 0$. A chaque instant d'échantillonnage k , $V(x(k))$ doit satisfaire l'inégalité suivante :

$$V(x(k+i+1/k)) - V(x(k+i/k)) \leq -[X(k+i) + U(k+i)] \quad (3.45)$$

$$V(x(k+i+1/k)) - V(x(k+i/k)) \leq -J_\infty(k) \quad (3.46)$$

Il suffit de suivre les mêmes démarches que celle dans la section précédente avec $F_j = Y_j G^{-1}$, on aboutit au résultat suivant donc après l'analyse sur (3.45) on obtient :

$$\begin{bmatrix} G^T + G - Q_z & (A_z G - B_z Y_z)^T & Y_z^T R_0 & G^T Q_0 \\ (A_z G - B_z Y_z) & Q_z & 0 & 0 \\ R_0 Y_z & 0 & R_0 & 0 \\ Q_0 G & 0 & 0 & Q_0 \end{bmatrix} > 0 \quad (3.47)$$

Contraintes sur l'entrée et sur la sortie

Dans cette section, est montré comment les contraintes sur l'entrée et sur la sortie, peuvent être intégrées comme des LMIs dans le problème d'optimisation.

- Les contraintes sur la commande :

A l'instant k , on considère la contrainte de type norme euclidienne suivante :

$$|u_h(k + i/k)| \leq u_{h,max}, \quad i \geq 0, \quad h = 1, 2, \dots, p$$

$$\|u(k)\|_{max} \triangleq \max_i u_i(k)$$

$$\max_{i>0} \|u(k)\|_{max} \geq \max_{i>0} \|Y_z G^{-1} Q_z^{-1} x(k)\|_{max}$$

On a $x(k) = a = Cst$. Pour chaque $i \geq 0$ danc:

$$\max_{i>0} \|u(k)\|_{max} \geq \max_{a \in \mathbb{R}} |Y_z G^{-1} Q_z^{-1} a|^2 \Leftrightarrow \max_{i>0} \|u(k)\|_{max} \geq \|(Y_z G^{-1} Q_z^{-1/2})\|_2^2$$

Par l'utilisation de l'inégalité de Cauchy-Schwarz à [4] et avec $\max_{i>0} \|u(k)\|_{max} = u_{max} = Z$. L'inégalité précédente devient :

$$Z \geq (Y_z G^{-1} Q_z^{-1} (Y_z G^{-1})^T) \Leftrightarrow Z - (Y_z G^{-1} Q_z^{-1} (Y_z G^{-1})^T) \geq 0$$

En utilisant le complément de Schur on obtient :

$$\begin{bmatrix} Z & Y_z G^{-1} \\ (Y_z G^{-1})^T & Q_z \end{bmatrix} > 0 \quad (3.48)$$

Par l'utilisation du théorème de Congruence avec la matrice en rang plein $\begin{bmatrix} I & 0 \\ 0 & G^T \end{bmatrix}$ on obtient :

$$\begin{bmatrix} Z & Y_z \\ Y_z^T & G^T - G - Q_z \end{bmatrix} > 0 \quad (3.49)$$

- Les contraintes sur la sortie :

A l'instant k , on considère la contrainte de type norme euclidienne suivante :

$$|y_h(k + i/k)| \leq y_{h,max} , \quad i \geq 0, \quad h = 1, 2, \dots, q$$

$$\|y(k + i/k)\|_{max} \triangleq \max_i y_i(k + i/k)$$

$$\max_{i>0} \|y(k)\|_{max} \geq \max_{i>0} \|C_z(A_z - B_z Y_z G^{-1})Q_z^{-1}x(k)\|_{max}$$

On a $x(k) = a = Cst.$ Pour chaque $i \geq 0$ danc:

$$\max_{i>0} \|y(k)\|_{max} \geq \max_{a \in \mathbb{R}} |C_z(A_z - B_z Y_z G^{-1})Q_z^{-1}a|^2 \Leftrightarrow$$

$$\max_{i>0} \|y(k)\|_{max} \geq \|(C_z(A_z - B_z Y_z G^{-1}))Q_z^{-1/2}\|_2^2$$

Par l'utilisation de l'inégalité de Cauchy-Schwarz à [4] et avec $\max_{i>0} \|y(k)\|_{max} = Y_{max} = W$. L'inégalité précédente devient :

$$W \geq ((C_z(A_z - B_z Y_z G^{-1}))Q_z^{-1}(C_z(A_z - B_z Y_z G^{-1}))^T)$$

$$\Leftrightarrow W - ((C_z(A_z - B_z Y_z G^{-1}))Q_z^{-1}(C_z(A_z - B_z Y_z G^{-1}))^T) \geq 0$$

En utilisant le complément de Schur on obtient :

$$\begin{bmatrix} -W & -C_z(A_z - B_z Y_z G^{-1}) \\ -(A_z - B_z Y_z G^{-1})^T C_z^T & Q_z \end{bmatrix} > 0 \quad (3.50)$$

Par l'utilisation du théorème de Congruence avec la matrice en rang plein $\begin{bmatrix} I & 0 \\ 0 & G^T \end{bmatrix}$ on obtient :

$$\begin{bmatrix} W & C_z(A_z G - B_z Y_z) \\ (A_z G - B_z Y_z)^T C_z^T & G^T + G - Q_z \end{bmatrix} > 0 \quad (3.51)$$

Théorème 5 : [21].

L'équilibre d'un modèle flou discret, décrit par (3.6), est asymptotiquement stable, en minimisant une borne supérieure du cout quadratique sous contrainte, si en chaque instant d'échantillonnage, Avec les Y_{ij} définie à (3.47), s'il existe des matrices Q_i commune définie positive, Y_j, G et $X_{ii} > 0$ et $X_{ij} = X_{ij}^T$. Telle que les LMI suivantes soient vérifiées et satisférons le problème d'optimisation convexe suivant :

$$\min_{Q_i, G, Y_j} \gamma \quad (3.52)$$

$$\begin{bmatrix} \gamma & x^T(k/k) \\ x(k/k) & Q_i \end{bmatrix} > 0 \quad (3.53)$$

$$Y_{ii} > X_{ii} \quad i \in \{1, \dots, r\} \quad (3.54)$$

$$Y_{ij} + Y_{ij} > X_{ij} + X_{ij}^T \quad i < j \quad i, j \in \{1, \dots, r\}, \quad (3.55)$$

$$\text{Avec: } X_l = \begin{bmatrix} 2X_{11} & (*) & (*) & (*) \\ X_{12} & 2X_{22} & \dots & (*) \\ \vdots & \vdots & \ddots & \vdots \\ X_{1r} & X_{2r} & \dots & 2X_{rr} \end{bmatrix} > 0 \quad (3.56)$$

Avec

$$\begin{bmatrix} Z & Y_j \\ Y_j^T & G^T + G - Q_i \end{bmatrix} > 0 \quad (3.57)$$

$$\begin{bmatrix} W & (A_i G - B_i Y_j) \\ (A_i G - B_i Y_j)^T & G^T + G - Q_i \end{bmatrix} > 0 \quad (3.58)$$

$$Y_{ij} = \begin{bmatrix} G^T + G - Q_i & (A_i G - B_i Y_j)^T & Y_j^T R_0 & G^T Q_0 \\ (A_i G - B_i Y_j) & Q_i & 0 & 0 \\ R_0 Y_j & 0 & R_0 & 0 \\ Q_0 G & 0 & 0 & Q_0 \end{bmatrix} \quad (3.59)$$

Les résultats présentés précédemment sur la stabilité des systèmes T-S sont conservatifs. En effet, ceux-ci ne constituent que des conditions suffisantes mais non nécessaires. Une alternative, efficace lorsque le modèle T-S est obtenu sur la base d'un modèle de connaissance non linéaire, consiste en l'utilisation des fonctions candidates non quadratique de Lyapunov, ou encore un contrôleur non-PDC.

III.6 Conclusion

Dans ce chapitre, nous avons présenté une synthèse d'un contrôleur par retour d'état pour systèmes nominaux (sans incertitude). La conception est effectuée à chaque instant d'échantillonnage par la minimisation d'une borne supérieure d'une fonction objective sur un horizon infini, en suivant la méthodologie MPC. La stratégie permet d'inclure des contraintes sur les variables du processus (limites sur l'entrée et la sortie) pendant l'étape de calcul du contrôleur.

Chapitre IV

Simulations et Discussions

Dans ce dernier chapitre, des tests proposés sur l'efficacité des méthodes étudiées dans le chapitre précédent, et aussi des discussions autour les résultats obtenues.

IV.1 Introduction

Dans ce chapitre, nous allons présenter des exemples d'application pour mettre en évidence les performances de chacune des méthodes décrites au chapitre précédent (PDC et Non-PDC) et mieux appréhender leurs limitations. Nous prenons trois exemples non-linéaires en [21], [55] et [56] pour tester la fiabilité de la méthode développée. En se basant sur les simulations des problèmes de régulation dans le traitement des exemples étudiés. Nous utiliserons pour la simulation *YALMIP Toolbox*, la version des solveurs pour les LMI améliorée par [54].

IV.2 Application de la commande MPC-FTS-LMI

Exemple 1 : [21]

On considère la classe du système Flou discret défini comme suit :

$$\begin{aligned}
 \text{R\`egle 1: Si } x_1(k) \text{ est } M_1, \text{ Alors } & \begin{cases} x(k+1) = A_1x(k) + B_1u(k) \\ y(k) = C_1x(k) \end{cases} \\
 \text{R\`egle 2: Si } x_1(k) \text{ est } M_2, \text{ Alors } & \begin{cases} x(k+1) = A_2x(k) + B_2u(k) \\ y(k) = C_2x(k) \end{cases}
 \end{aligned} \tag{4.1}$$

Avec les matrices d'états :

$$\begin{aligned}
 A_1 &= \begin{bmatrix} -0.5 & 2 \\ -0.1 & 1.1 \end{bmatrix}, & A_2 &= \begin{bmatrix} -0.19 & 0.5 \\ -0.1 & -1.2 \end{bmatrix} \\
 B_1 &= \begin{bmatrix} 4.1 \\ 4.8 \end{bmatrix}, & B_2 &= \begin{bmatrix} 3 \\ 0.1 \end{bmatrix}; & C_1 &= [1 \quad 0.3], & C_2 &= [0.8 \quad 0.2]
 \end{aligned}$$

Et Avec les matrices de pondération :

$$Q_0 = \begin{bmatrix} 0.8 & 0.1 \\ 0.1 & 0.95 \end{bmatrix}, \quad R_0 = 0.9, \quad \text{Avec } T = 0.08s.$$

Et les fonctions d'appartenance pour la règle 1 et la règle 2 sont :

$$M_1(x_1(k)) = \frac{1}{1 + \exp(-2x_1(k))}, \quad M_2(x_2(k)) = 1 - M_1(x_1(k))$$

Soumis aux contraintes suivantes :

$$\text{Sur la commande} \quad -1 < u(k) < 1$$

$$\text{Sur la sortie} \quad -2.5 < y(k) < 2.5$$

Avec les conditions initiales pour la loi de commande flou PDC et Non-PDC sont :

$$x_1(0) = -0.3, \quad x_2(0) = -1, \quad u(0) = -0.5, \quad y(0) = -0.5.$$

Les figures (Figure. IV.1 - IV.4) illustrent les résultats obtenus par les deux approches PDC et Non- PDC, les performances avec le PDC moins bonnes, en termes de rapidité, temps de réponse et erreur statique, que dans le cas d'un contrôleur Non-PDC, on observe une amélioration de la performance de la réponse du système. Les sorties convergent en un temps approximatif de 0.7s, bien réduit en comparaison avec les 2.0s obtenues dans le cas PDC.

Les figures IV.3 et IV.4 donnent l'évolution de signal d'entrée et le signal de sortie respectivement, on constate que le modèle converge vers l'origine et que la commande avec approche Non-PDC, est beaucoup moins sollicitée que celle de l'approche PDC.

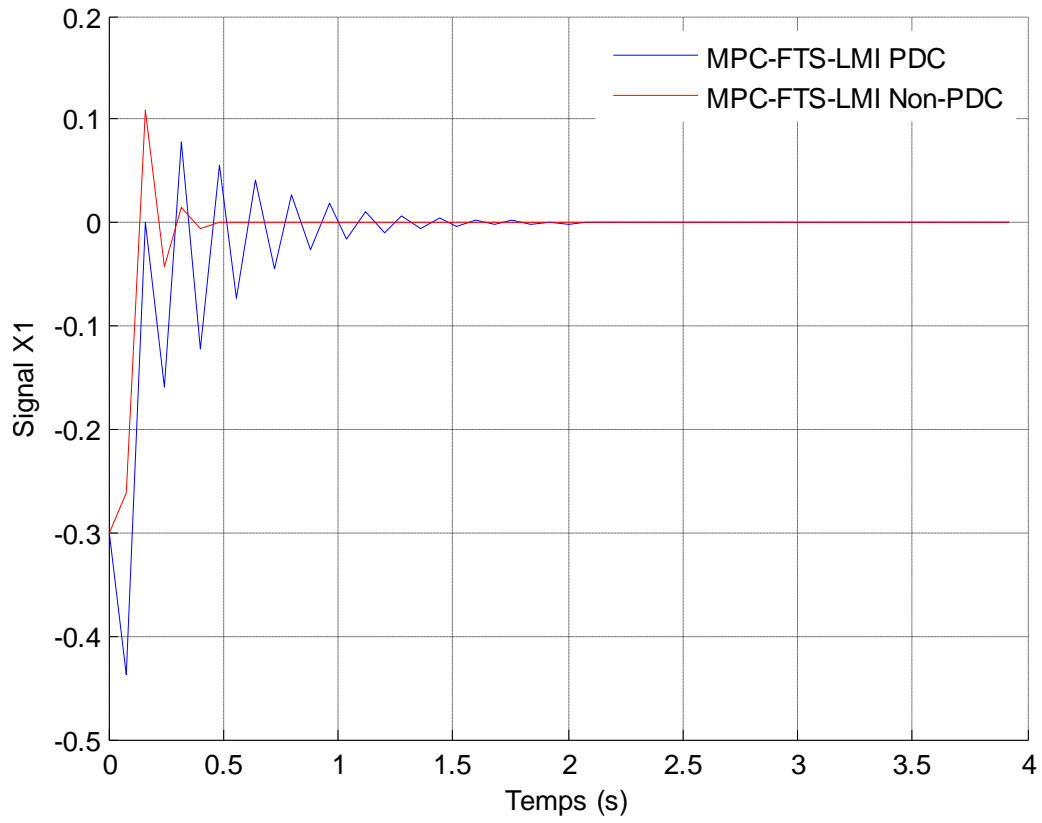


Figure IV.1: La Réponse du Signal d'état X1

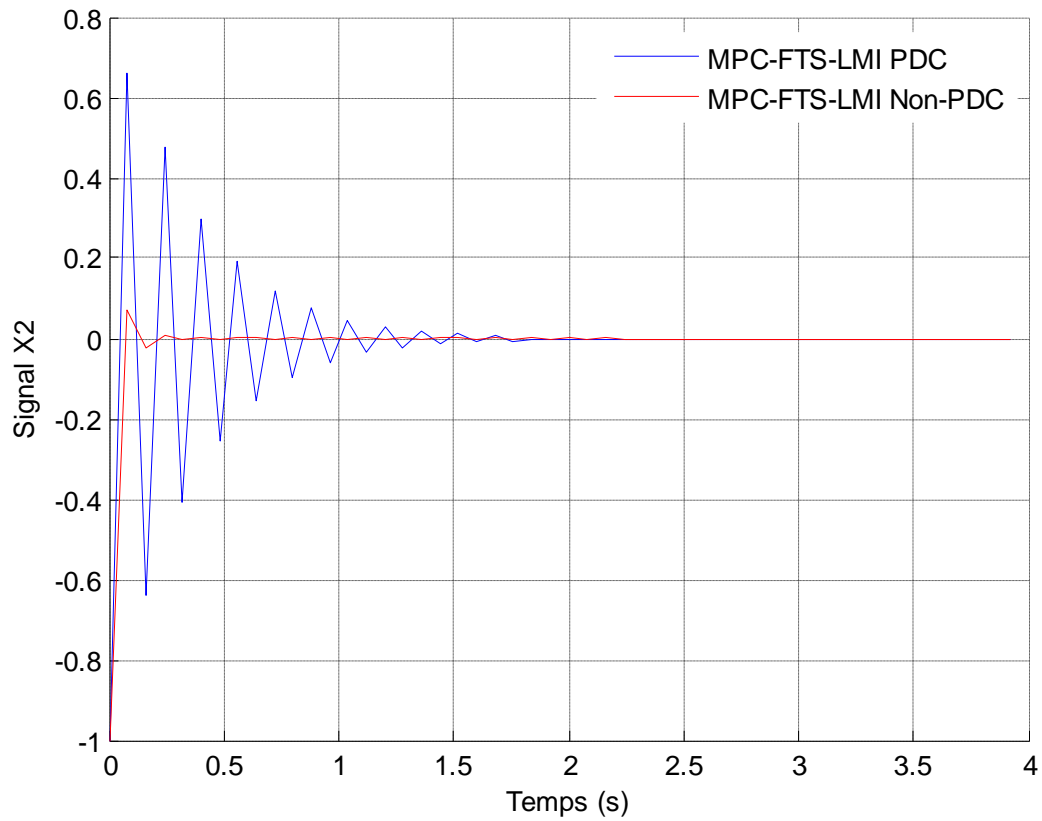


Figure IV.2: La Réponse du Signal d'état X2

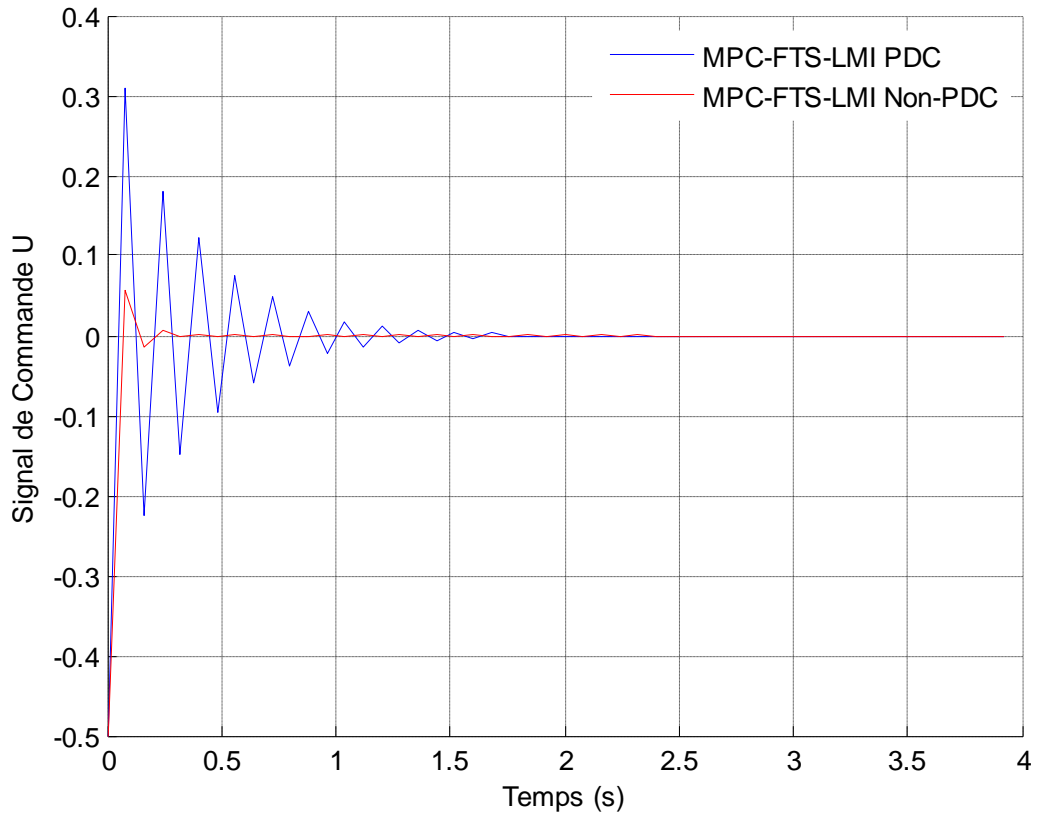


Figure IV.3: La Réponse du Signal de Commande

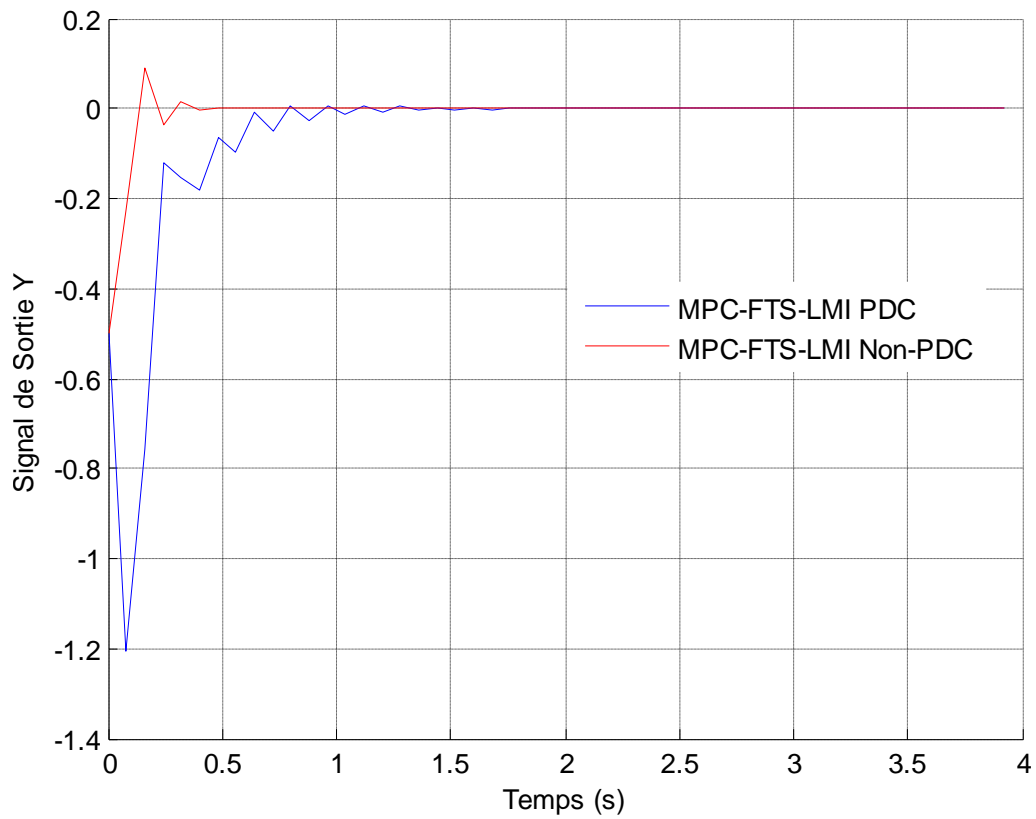


Figure IV.4: La Réponse du Signal de Sortie

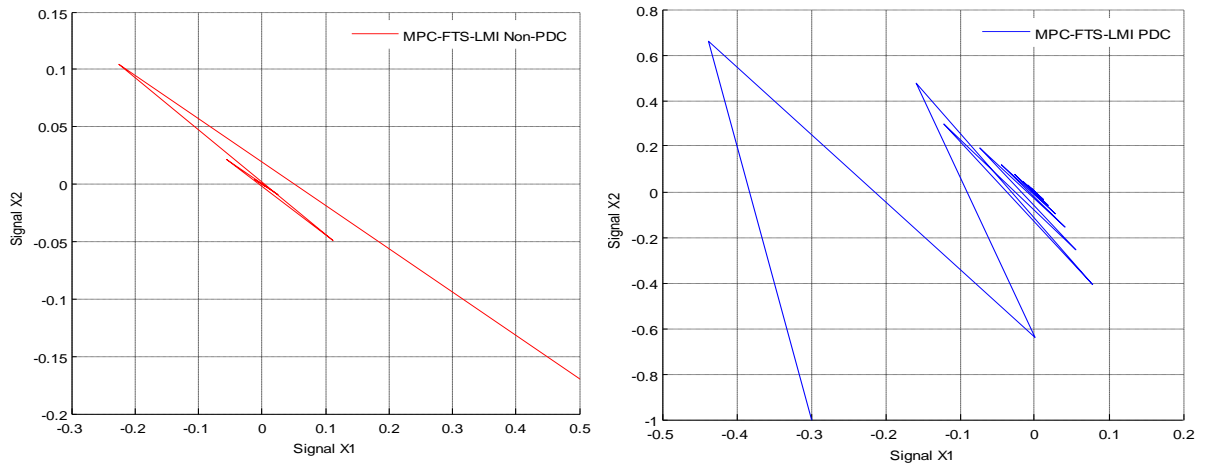


Figure IV.5: Représentation des résultats par MPC-FTS-LMI dans l'Espace s'Etat

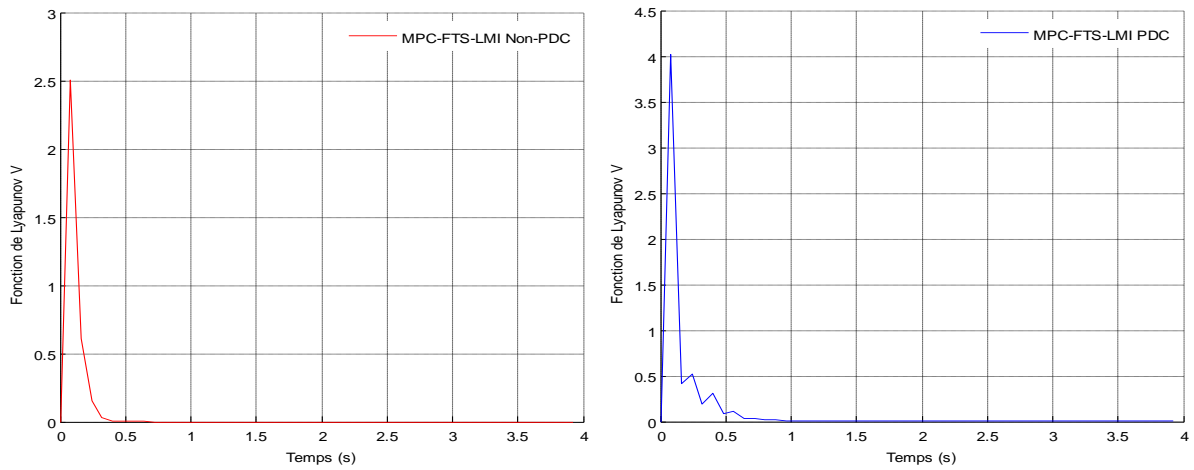


Figure IV.6: Représentation de la Fonction de Lyapunov

Exemple 2 : [55]

On considère la classe du système Flou discret défini comme suit :

$$\left\{ \begin{array}{l} \text{R\`egle 1: Si } x_1(k) \text{ est } 0, \text{ Alors } \begin{cases} \dot{x}(t) = A_1x(t) + B_1u(t) \\ y(t) = C_1x(t) \end{cases} \\ \text{R\`egle 2: Si } x_1(k) \text{ est } \pm \pi/2, \text{ Alors } \begin{cases} \dot{x}(t) = A_2x(t) + B_2u(t) \\ y(t) = C_2x(t) \end{cases} \end{array} \right. \quad (4.2)$$

Avec les matrices d'états :

$$A_1 = \begin{bmatrix} 0.9504 & 0.9834 \\ -0.09834 & 0.9504 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.9635 & 0.6218 \\ -0.06218 & 0.3417 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0.4958 \\ 0.9834 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0.365 \\ 0.6218 \end{bmatrix}; \quad C_1 = [1 \quad 0], \quad C_2 = [1 \quad 0]$$

Et Avec les matrices de pondération :

$$Q_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R_0 = 0.5 \quad \text{Avec } T = 1s.$$

Et les fonctions d'appartenance pour la règle 1 et la règle 2 sont :

$$M_1(x_1(k)) = -x^2 + 1, \quad M_2(x_2(k)) = x^2$$

Soumis aux contraintes suivantes :

$$\text{Sur la commande} \quad -0.5 < u(k) < 0.5$$

$$\text{Sur la sortie} \quad -2 < y(k) < 2$$

Avec les conditions initiales pour la loi de commande flou PDC et Non-PDC sont :

$$x_1(0) = 0.1, \quad x_2(0) = 0, \quad u(0) = 0.1, \quad y(0) = 0.1.$$

Les figures (IV.7-IV.9) illustrent les résultats obtenus par les deux approches PDC et Non- PDC, les performances avec le PDC moins bonnes, en termes de rapidité, temps de réponse et erreur statique, que dans le cas d'un contrôleur Non-PDC, on observe une amélioration de la performance de la réponse du système. Les sorties convergent en un temps approximatif de 15s, bien réduit en comparaison avec les 50s obtenues dans le cas PDC

Les figures IV.10 et IV.11 donnent l'évolution de signal d'entrée et le signal de sortie respectivement, on constate que le modèle converge vers l'origine et que la commande avec approche Non-PDC, est beaucoup moins sollicitée que celle de l'approche PDC.

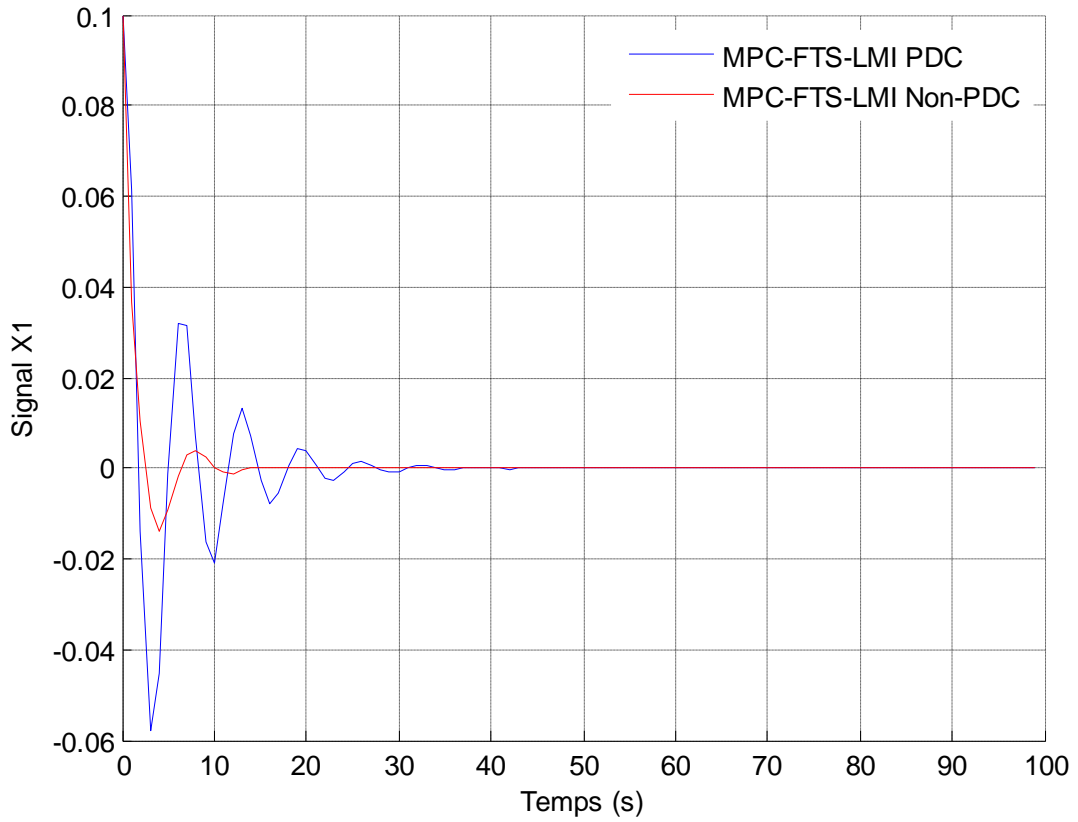


Figure IV.7: La Réponse du Signal d'état X1

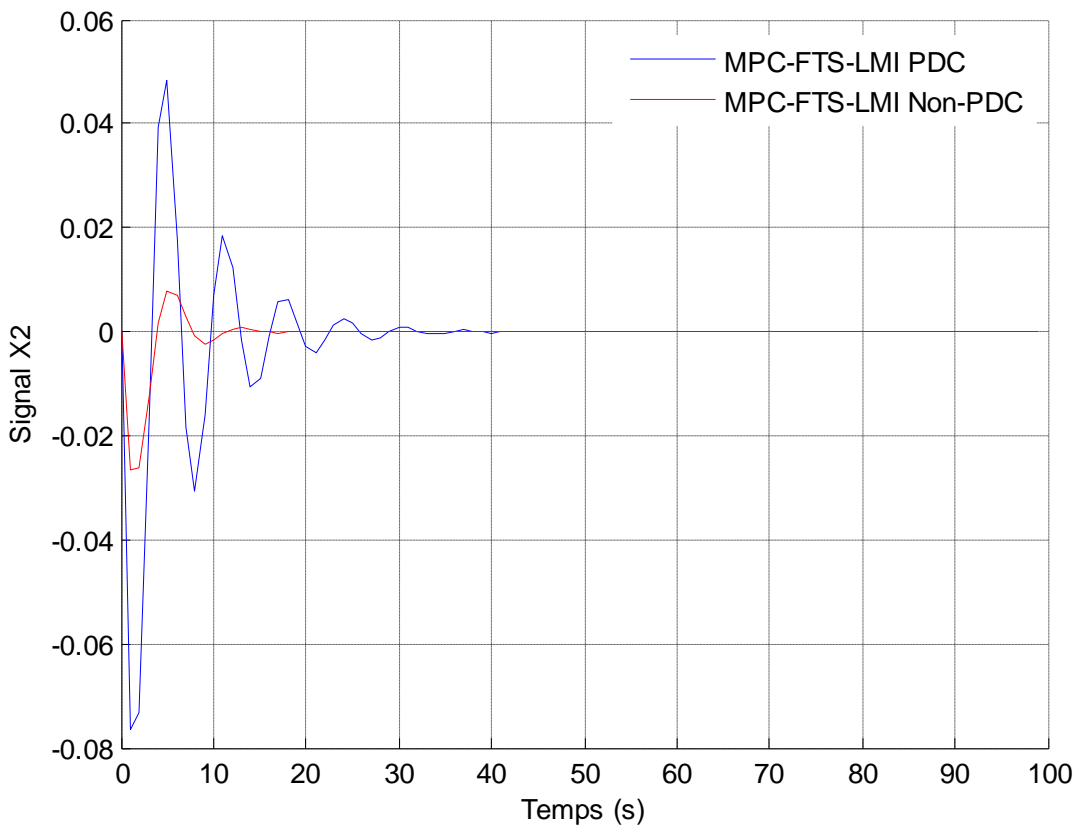


Figure IV.8: La Réponse du Signal d'état X2

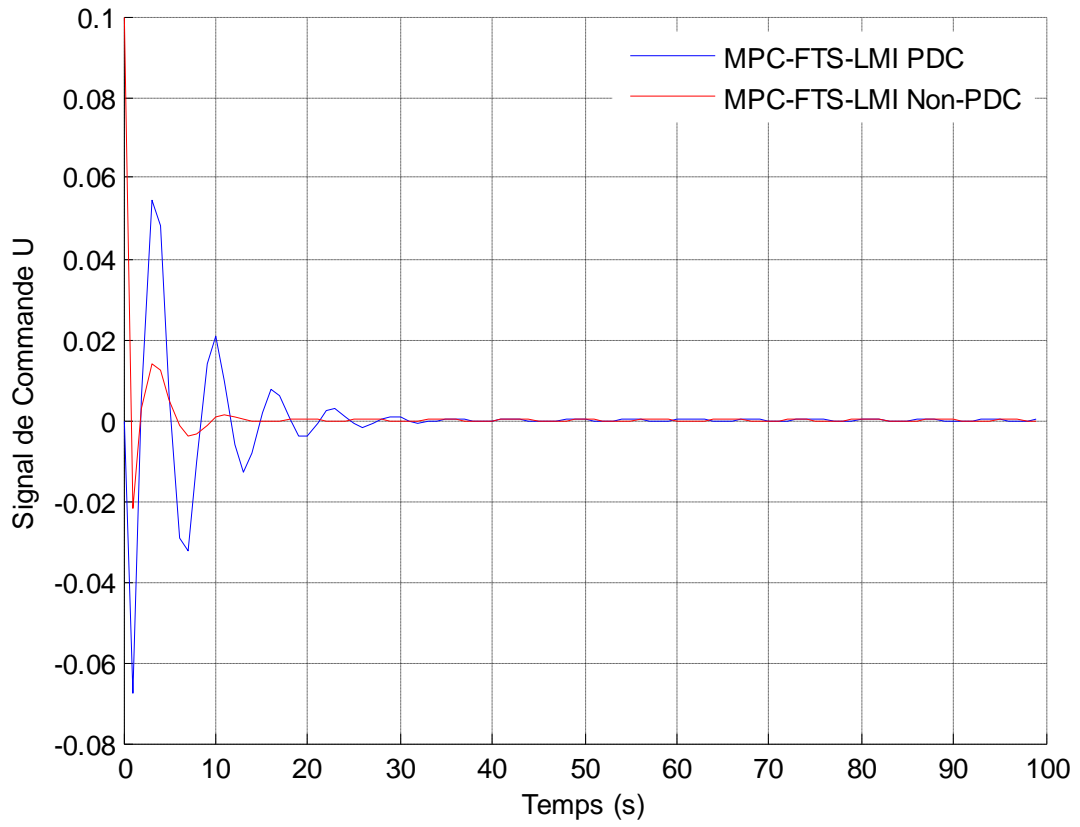


Figure IV.9: La Réponse du Signal de Commande

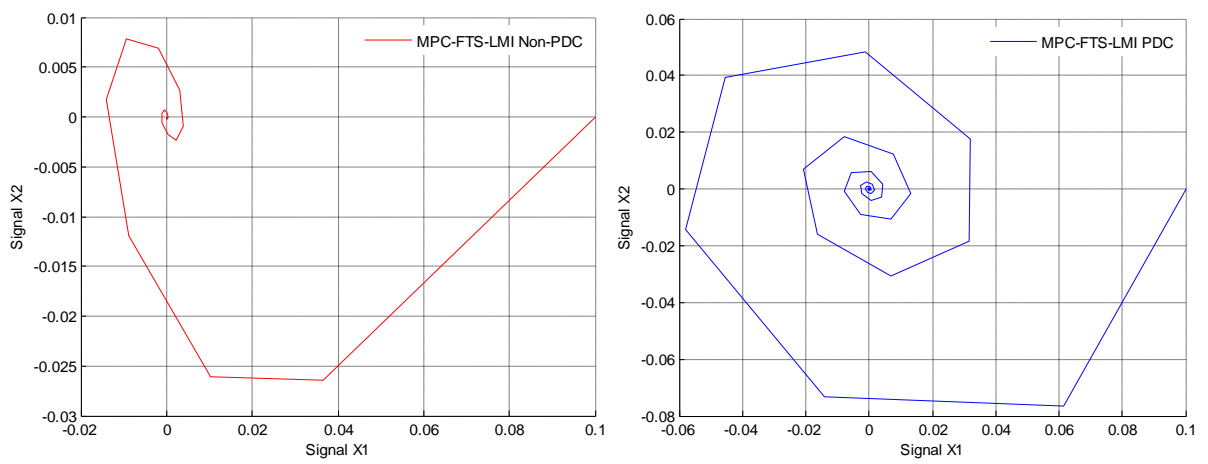


Figure IV.10: Représentation des résultats par MPC-FTS-LMI dans l'Espace s'Etat

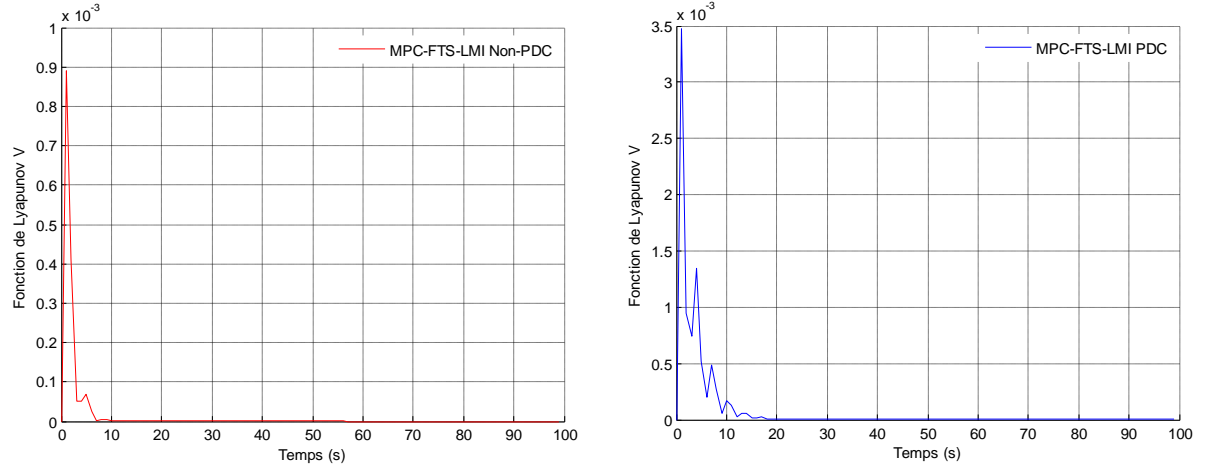


Figure IV.11: Représentation de la Fonction de Lyapunov

Exemple 3 : Pendule Inversé

Considérons le pendule inversé donnée on [56] et [21]. L'objectif de commande est de balancer le pendule vers le point d'origine de $\pi/2$.

$$\left\{ \begin{array}{l} \text{R\`egle 1: Si } x_1(k) \text{ est } 0, \text{ Alors } \begin{cases} \dot{x}(t) = A_1 x(t) + B_1 u(t) \\ y(t) = C_1 x(t) \end{cases} \\ \text{R\`egle 2: Si } x_1(k) \text{ est } \pm \pi/2, \text{ Alors } \begin{cases} \dot{x}(t) = A_2 x(t) + B_2 u(t) \\ y(t) = C_2 x(t) \end{cases} \end{array} \right. \quad (4.3)$$

Avec les matrices d'états :

$$A_1 = \begin{bmatrix} 0 & 1 \\ \frac{g}{4l/3 - aml} & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 1 \\ -\frac{2g}{4l/3 - amlb^2} & 0 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 \\ -\frac{a}{4l/3 - aml} \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ -\frac{ab}{4l/3 - amlb^2} \end{bmatrix}; \quad C_1 = [1 \quad 0], \quad C_2 = [1 \quad 0]$$

Avec les constantes :

$$a = 0.1, \quad b = \cos(88^\circ), \quad g = 9.8 \text{ m/s}^2, \quad m = 2 \text{ kg}, \quad l = 1 \text{ m}$$

Ce système est dans le cas continu, pour obtenir le cas discret il suffit de discrétiser le système T-S flou avec un période d'échantillonnage $T = 0.03s$, on obtient le système T-S flou discret avec les matrices d'états suivantes :

$$A_1 = \begin{bmatrix} 1.0078 & 0.0301 \\ 0.5202 & 1.0078 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1.0073 & 0.0300 \\ 0.2445 & 1.0037 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 4 \\ 3 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 4 \\ 3 \end{bmatrix}; \quad C_1 = [1 \quad 0], \quad C_2 = [1 \quad 0]$$

Et Avec les matrices de pondération :

$$Q_0 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}, \quad R_0 = 1$$

Et les fonctions d'appartenance pour la règle 1 et la règle 2 sont :

$$M_1(x_1(k)) = \frac{x_1 + 1}{2}, \quad M_2(x_2(k)) = \frac{1 - x_1}{2}$$

Soumis aux contraintes suivantes :

$$\text{Sur la commande} \quad -0.5 < u(k) < 0.5$$

$$\text{Sur la sortie} \quad -\pi/2 < y(k) < \pi/2$$

Avec les conditions initiales pour la loi de commande flou PDC et Non-PDC sont :

$$x_1(0) = 0.73, \quad x_2(0) = 0, \quad u(0) = 0.05, \quad y(0) = 0.05.$$

Les figures (Figure. IV.12, IV.13 et IV.14) illustrent les résultats obtenus par les deux approches PDC et Non- PDC, les performances avec le PDC sont moins bonnes, en termes de rapidité, temps de réponse et erreur statique, que dans le cas d'un contrôleur Non-PDC, on observe une certaine amélioration de la performance de la réponse du système. Les sorties convergent en un temps approximatif de 0.4s, bien réduit en comparaison avec les 2s obtenues dans le cas PDC. On constate également que le modèle converge vers l'origine et que la

commande avec approche Non-PDC, est beaucoup moins sollicitée que celle de l'approche PDC.

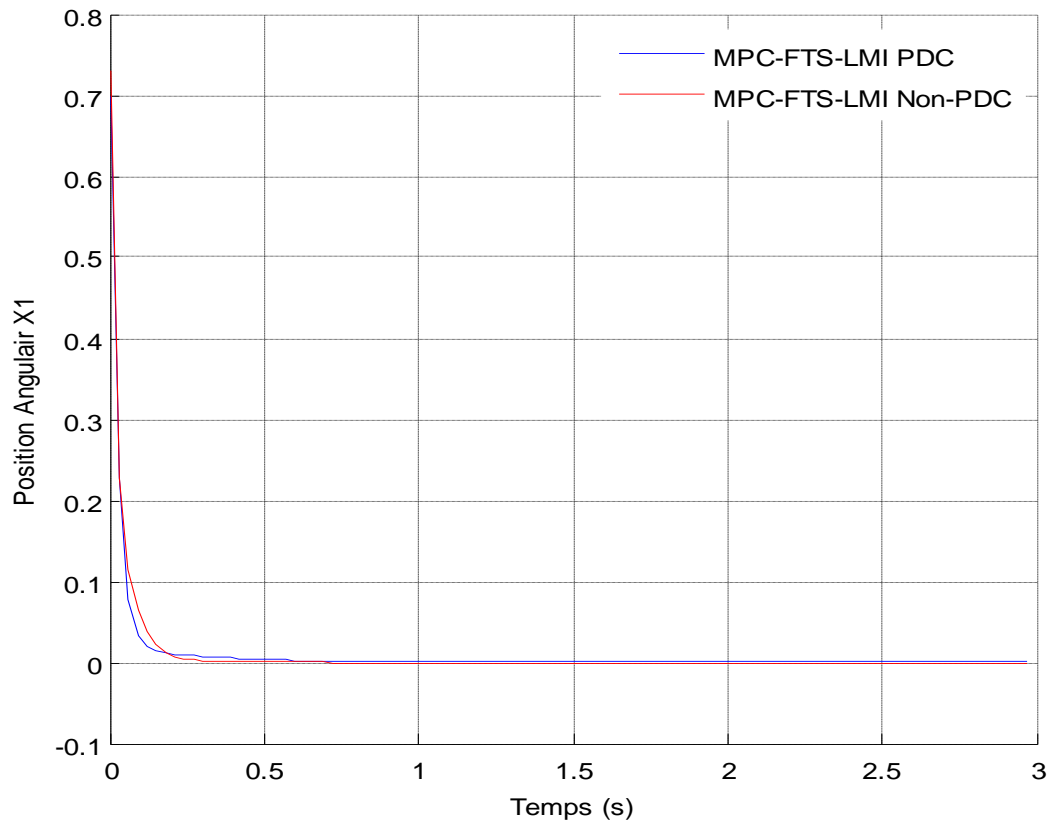


Figure IV.12: La Réponse du Signal Position Angulaire (X_1)

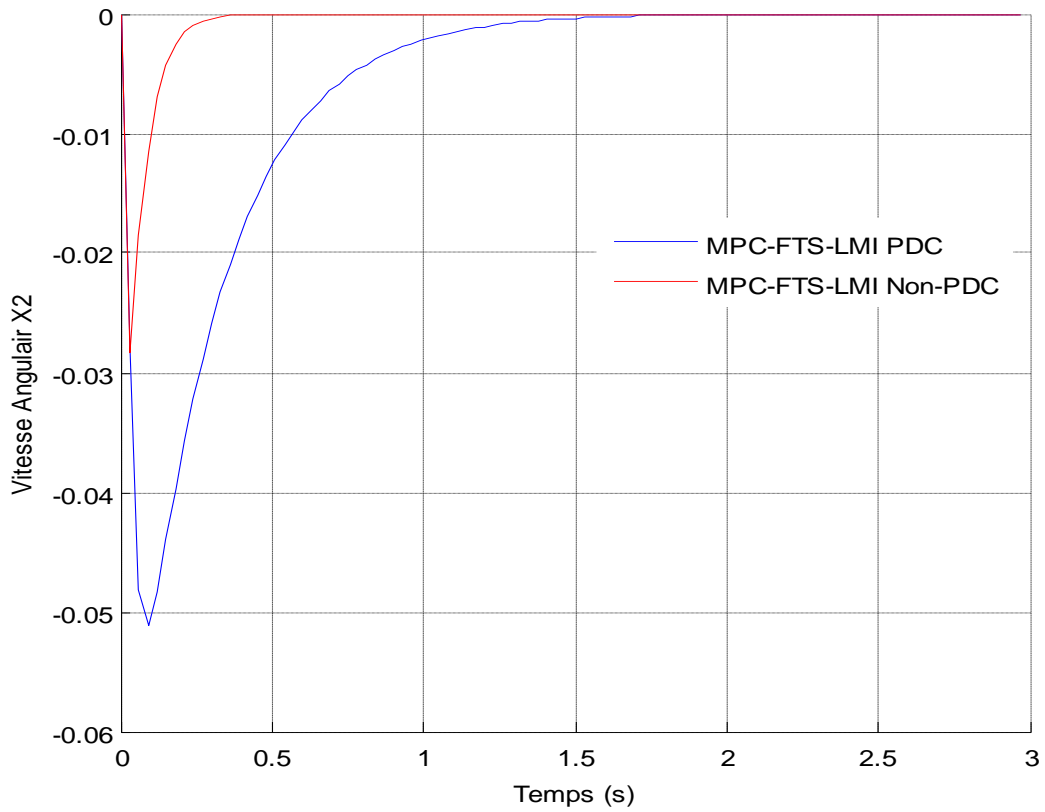


Figure IV.13: La Réponse du Vitesse Angulaire (X_2)

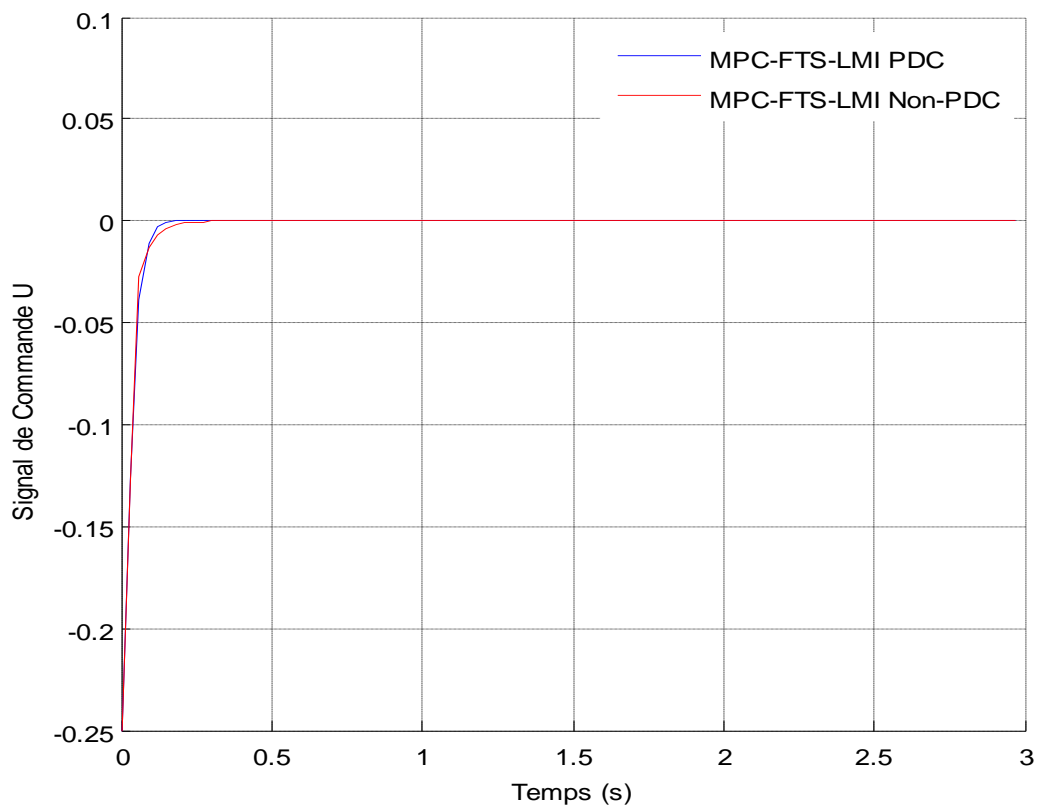


Figure IV.14: La Réponse du Signal de Commande

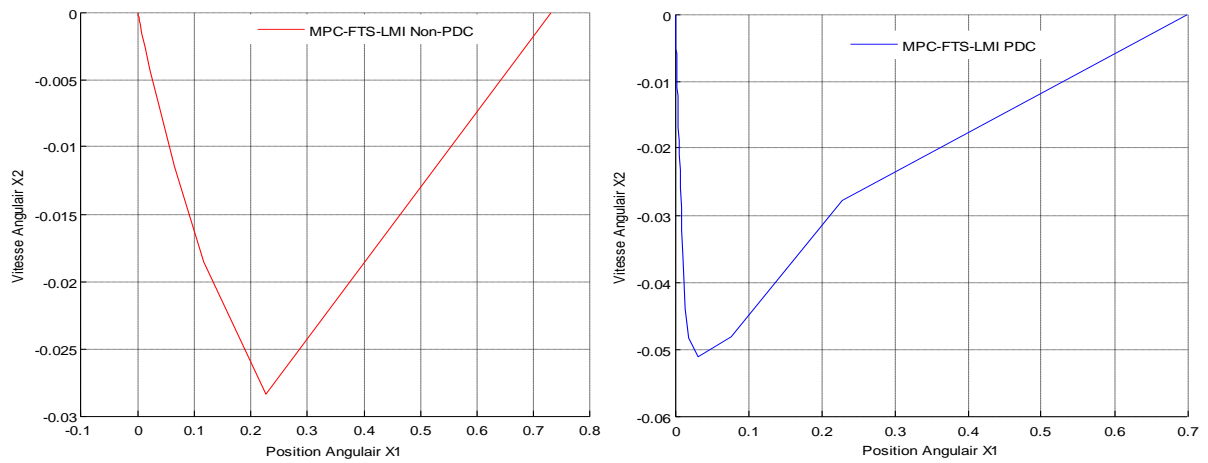


Figure IV.15: Représentation des résultats par MPC-FTS-LMI dans l'Espace s'Etat

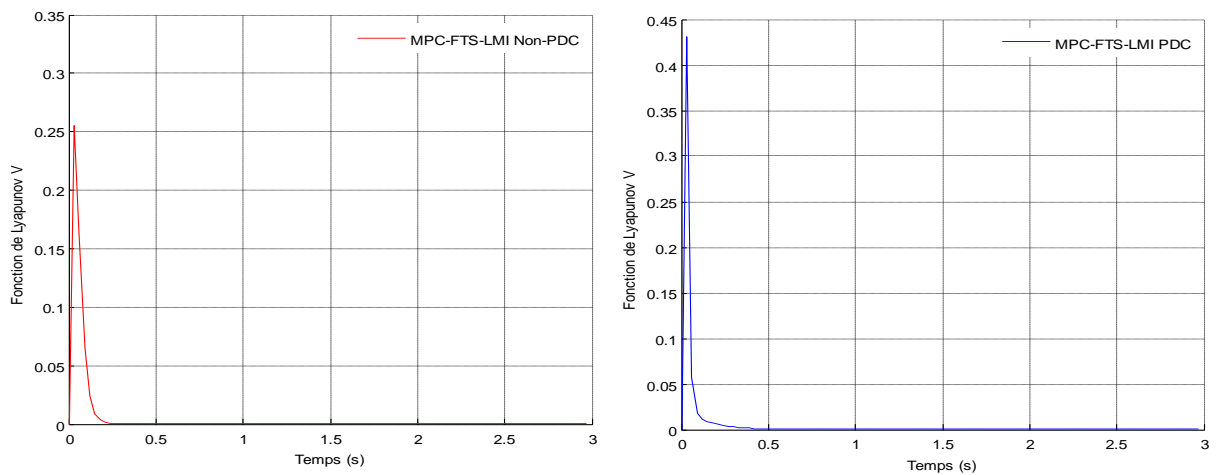


Figure IV.16: Représentation de la Fonction de Lyapunov

IV.3 Application de la commande MPC-FTS-LMI aux Système Perturbés

Pour évaluer la robustesse des lois de commande étudiées dans le chapitre 3, nous allons les examiner en présence des perturbations externes. Pour cela, nous allons prendre les mêmes exemples présentés précédemment :

Exemple 1 : [21] à la présence des perturbations externes.

Dans cet exemple, nous allons appliquer un pic de perturbation externe de valeur $d = 0.3$ sur les états du système à l'instant $t = 1.8s$.

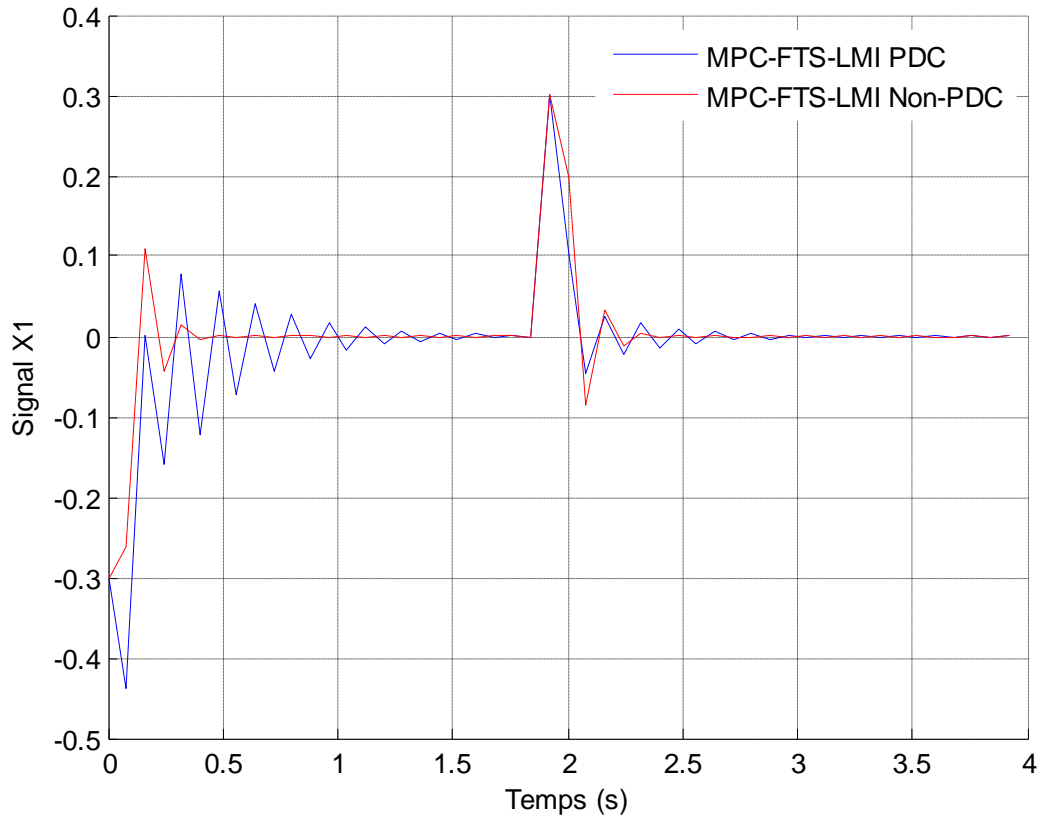


Figure IV.17: La Réponse du Signal d'état X2 Perturbé

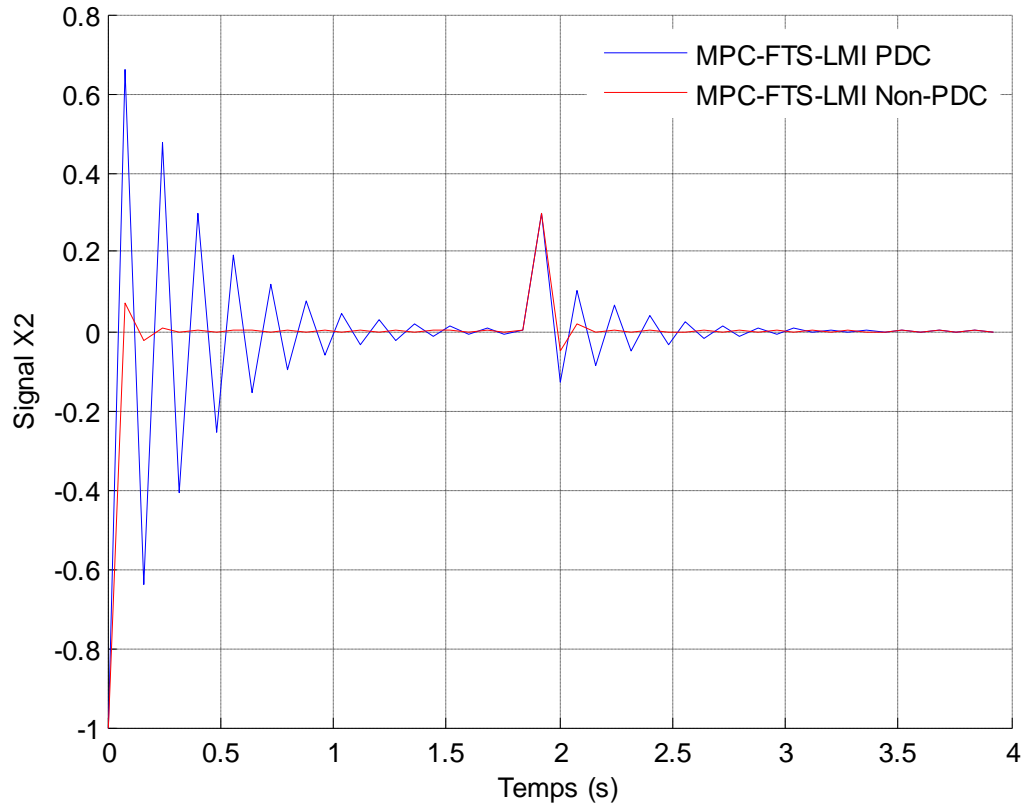


Figure. IV. 18 : La Réponse du Signal d'état X_2 Perturbé.

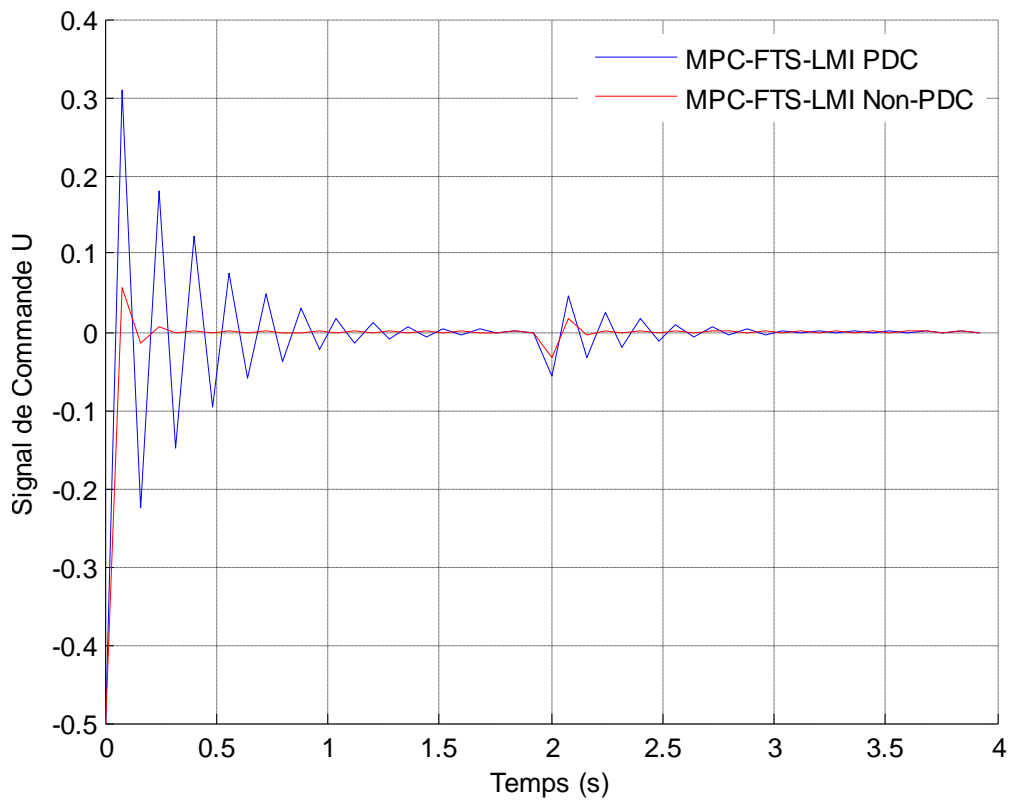


Figure IV.19: La Réponse du Signal de Commande Perturbé.

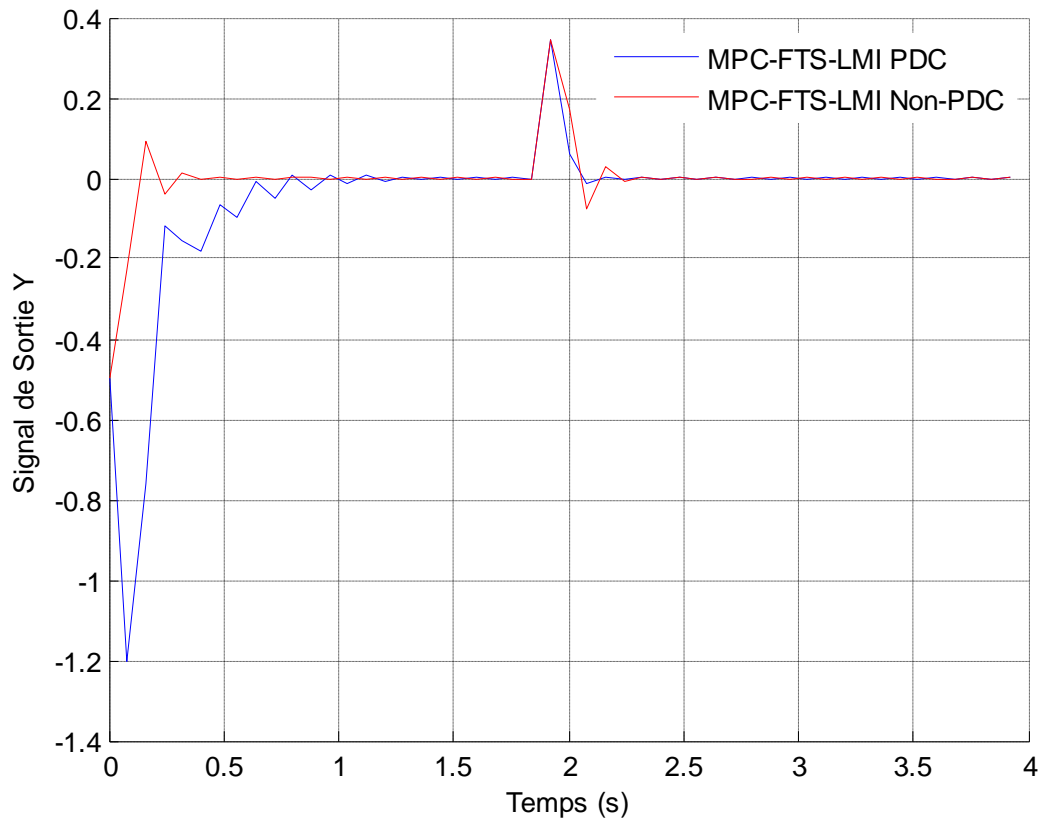


Figure IV.20: La Réponse du Signal de Sortie Perturbé.

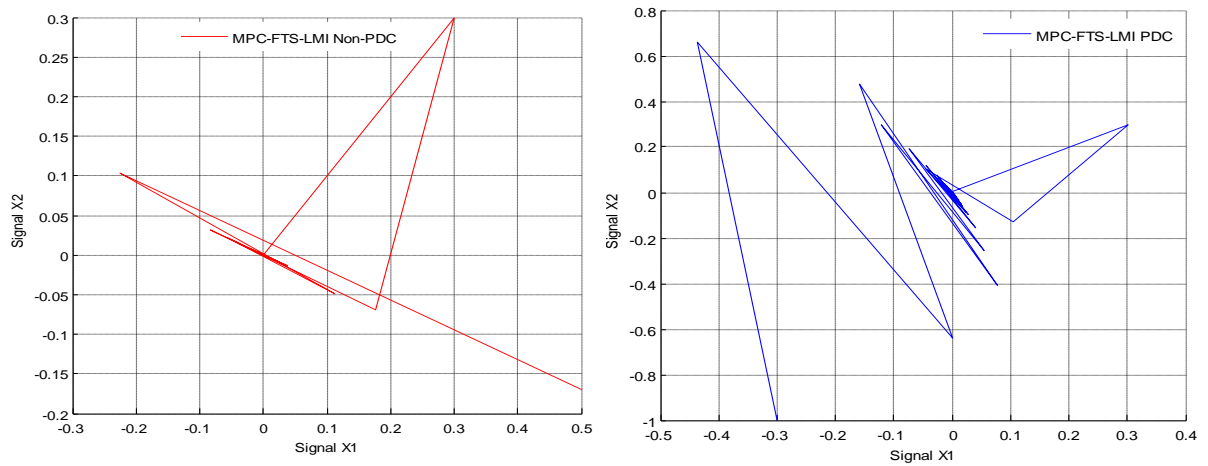


Figure IV.21 : Représentation des résultats par MPC-FTS-LMI dans l'Espace d'Etat

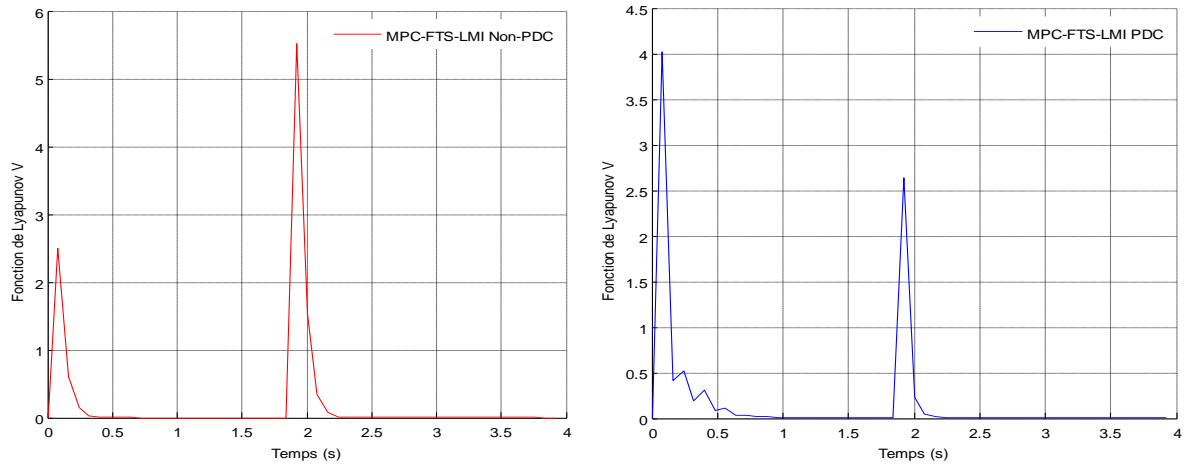


Figure IV.22: Représentation de la Fonction de Lyapunov

Les figures (IV.17-IV.22) Montrent l'apport en termes de robustesse. L'approche non quadratique permet de prendre en compte des incertitudes plus importantes que l'approche quadratique. On constate que les performances sont assurées ainsi que la rapidité de convergence vers l'origine. Les sorties convergent en un temps approximatif de 0.5s, bien réduit en comparaison avec les 1.2s obtenues dans le cas PDC.

Exemple 2 : [55] à la présence des perturbations externes.

Dans cet exemple nous allons appliquer un pic de perturbation externe de valeur $d = 0.1$ sur les états du système à l'instant $t = 49s$.

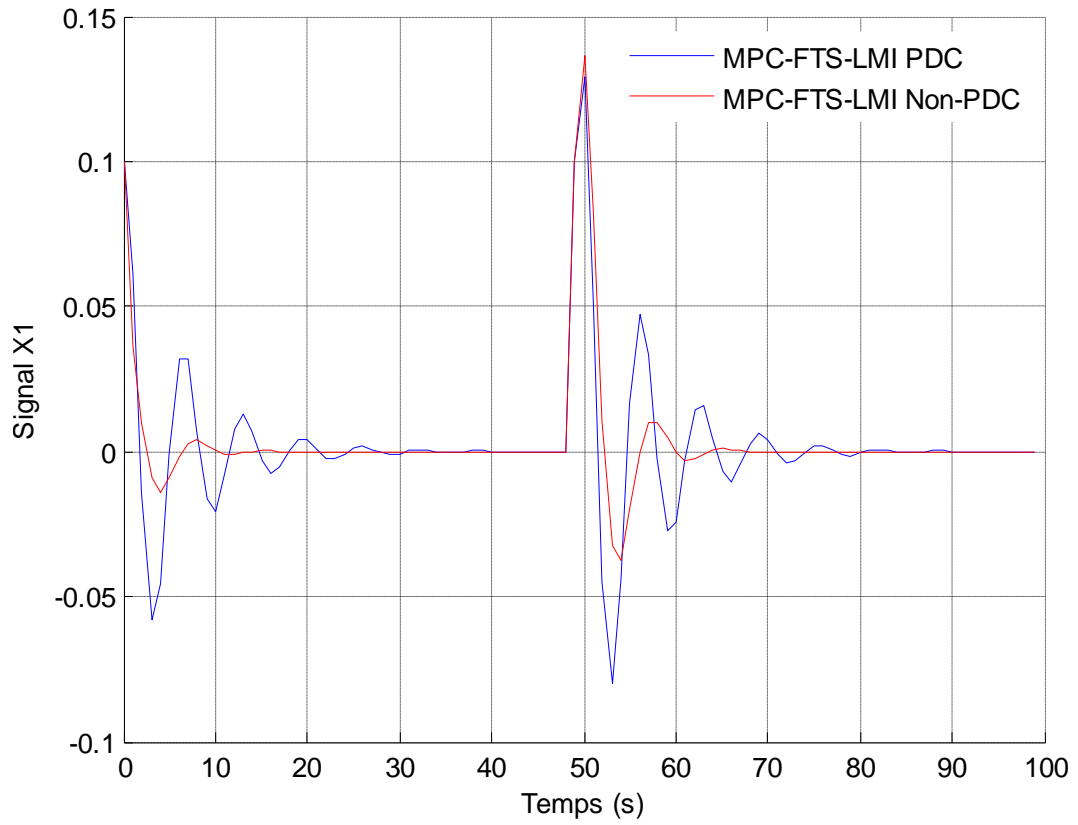


Figure IV.23: La Réponse du Signal d'état X1 Perturbé.

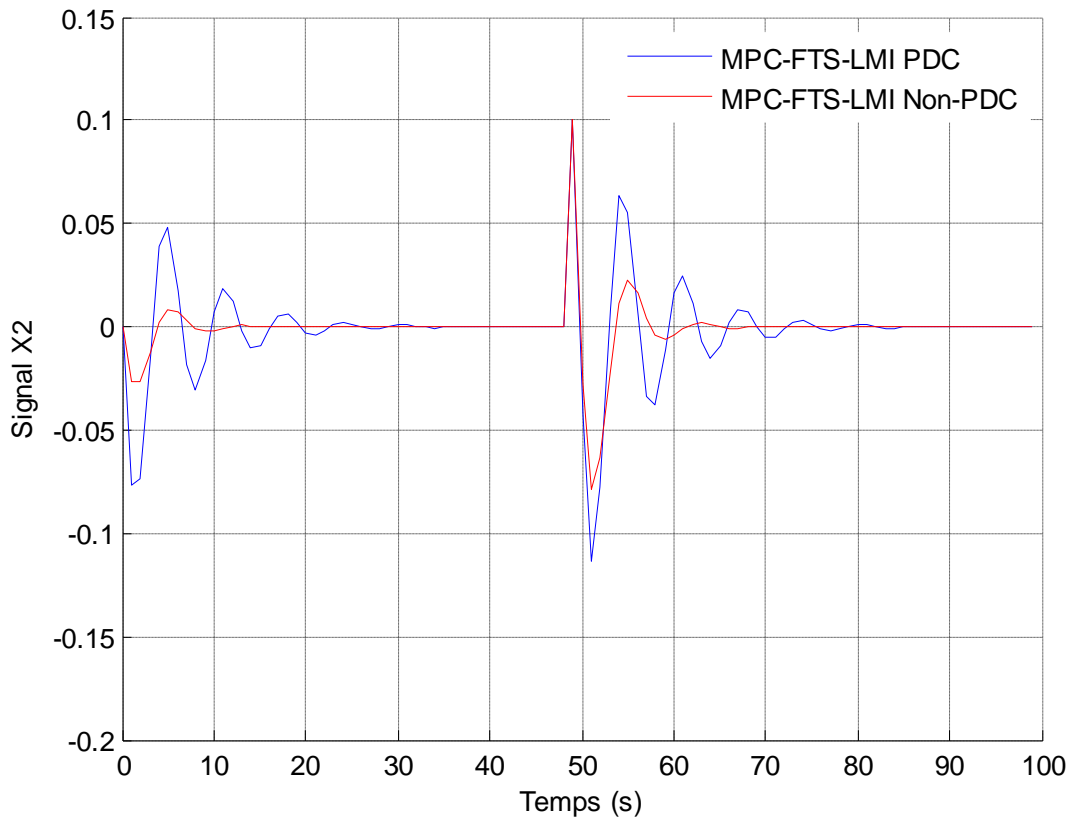


Figure IV.24: La Réponse du Signal d'état X2 Perturbé.

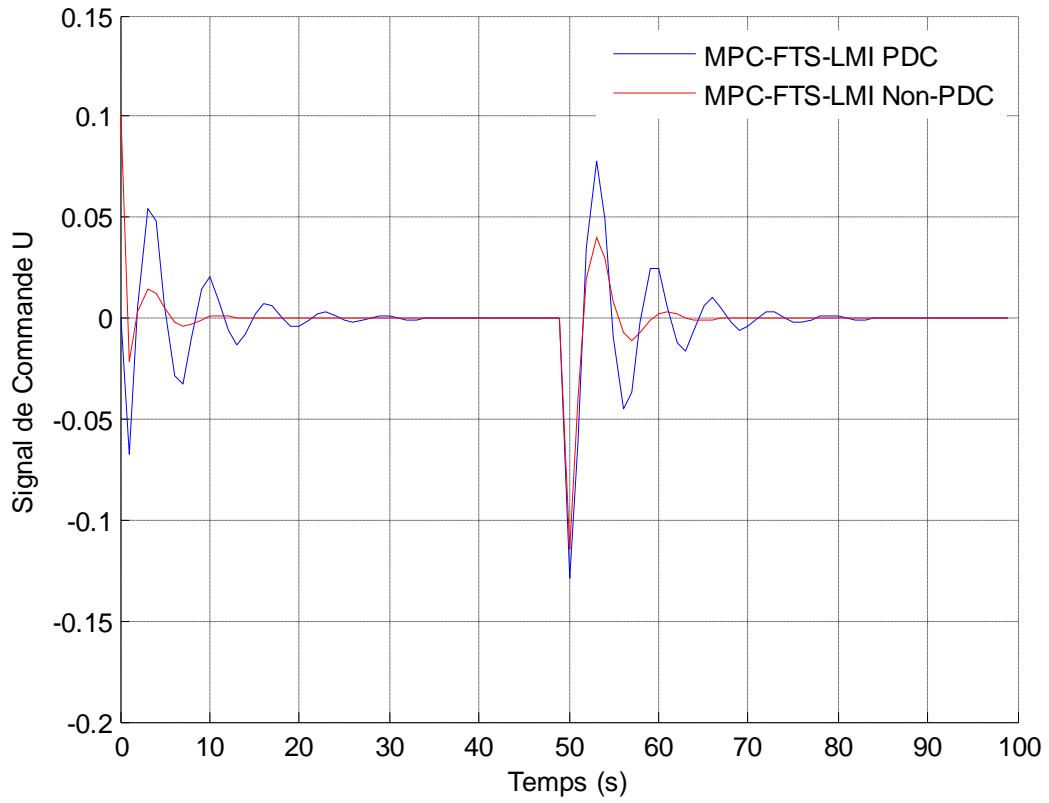


Figure IV.25 : La Réponse du Signal de Commande Perturbé.

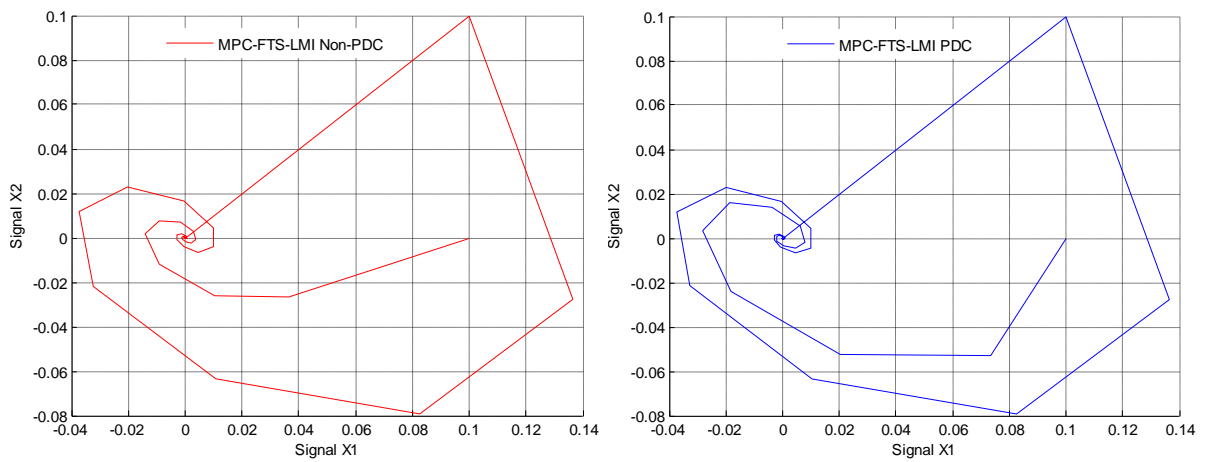


Figure IV.26: Représentation des résultats par MPC-FTS-LMI dans l'Espace s'Etat

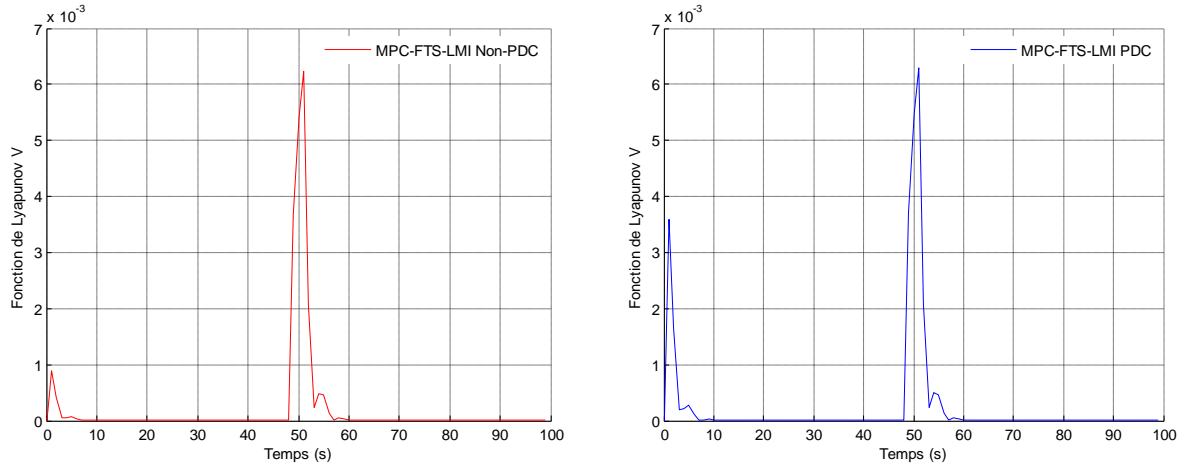


Figure IV.27: Représentation de la Fonction de Lyapunov

Les figures (IV.23-IV.27) Montrent l'apport en terme de robustesse. L'approche non quadratique permet de prendre en compte des incertitudes plus importantes que l'approche quadratique. On constate que les performances sont assurées ainsi par la rapidité de convergence vers l'origine. Les sorties convergent en un temps approximatif de 20s, bien réduit en comparaison avec les 30s obtenues dans le cas PDC.

IV.4 Conclusion

Dans ce chapitre nous avons montré par les simulations les performances de la loi de commande MPC-FTS-LMI PDC ou Non-PDC, la stratégie de commande étudiée présente de meilleures performances en termes de régulation surtout par l'utilisation de l'approche Non-PDC.

Donc, l'utilisation d'un contrôleur flou Non-PDC nous donne une forte amélioration pour la régulation et la stabilité du système.

Conclusion Générale et Perspectives

Durant ces dernières années la méthodologie de commande prédictive a connu un développement important, et comme nous l'avons rappelé dans ce travail, ceci est dû au fait que sa formulation permet d'inclure de manière explicite des contraintes sur les variables du système pendant la conception du contrôleur. En outre, la commande prédictive est facile à implanter et simple à comprendre par des personnes ayant des connaissances minimales en commande.

Dans ce mémoire, la synthèse d'un contrôleur robuste prédictive à partir d'une modélisation entrée-sortie a été reprise. A chaque instant d'échantillonnage, on effectue le calcul de la commande par la minimisation d'une borne supérieure d'une fonction objectif d'horizon infini à l'aide d'inégalités linéaires matricielles (LMI).

Dans le premier chapitre, nous avons mené une étude bibliographique, portant sur la modélisation floue, et dans le deuxième chapitre nous avons présenté les résultats concernant la relaxation des LMI, pour la commande des systèmes T-S dans le cadre quadratique. Ceux-ci font intervenir une fonction candidate de Lyapunov conduisant à la recherche de variables de décisions communes à un ensemble de contraintes LMI. Les techniques d'obtention de ces conditions LMI quadratique sont à ce jour bien maîtrisées. Elles sont basées sur l'application de lemmes de transformations matricielles (Complément de Schur, Lemme d'élimination, lemme de Finsler...). Ajoutons à cela que, même s'il est possible que tous les problèmes de commande ou d'observation des systèmes T-S ne soient pas encore été traités, l'obtention de conditions de stabilité quadratiques ne constituent plus un enjeu majeur, mais tout au plus une $n^{\text{ième}}$ extension des techniques existantes. Cela suppose que, le défi de l'extension dans le contexte non quadratique demeure un problème ouvert. Poursuivant l'objectif de fournir des conditions LMI de moindre conservatisme, des extensions de l'approche non quadratique à certains problèmes de commande ont été présentés.

Des travaux récents ont permis de réduire le conservatisme de l'approche non quadratique en définissant des paramètres dépendants, ou avec relations augmentées, faisant intervenir des variables d'écart.

La synthèse passe par l'utilisation des inégalités linéaires matricielles LMI puisqu'il existe des algorithmes puissants qui permettent d'obtenir la solution du problème en un temps polynomiale. Cela fait que l'optimisation par la solution de LMI peut être faite en ligne ce qui s'avère essentiel pour l'approche MPC. Dans ce contexte, nous avons présenté dans le chapitre III une loi de commande par retour d'état pour une classe de systèmes non-linéaires discrets, nous avons étudié la stratégie de la commande prédictive pour les systèmes flous T-S discret sous contraintes mise sous forme LMI « MPC-FTS-LMI ».

Enfin, dans un objectif de présenter les performances obtenues par chaque approches étudiées dans ce mémoire, des tests de simulation ont été effectués sur trois systèmes non linéaires discrets.

Les résultats de simulations nous montrent la fiabilité et les performances de la loi de commande MPC-FTS-LMI PDC ou Non-PDC, l'approche de commande étudiée dans ce mémoire présente de meilleures performances en termes de régulation surtout par l'utilisation de l'approche Non-PDC.

Donc, l'implémentation d'un contrôleur flou Non-PDC nous donne une forte amélioration pour la régulation et la stabilité du système.

En ce qui nous concerne, nous allons projeter quelques perspectives qui nous semblent les plus intéressantes à étudier :

L'extension de l'analyse de la stabilité pour le système constitué du modèle flou de type affine Takagi-Sugeno Multi-variable (MIMO) en temps discret.

Le développement d'un observateur flou basé sur le modèle affine Takagi-Sugeno, afin d'estimer les variables non mesurables du système.

La synthèse de notre loi de commande non-PDC pour des modèles TS incertains et perturbés.

Références Bibliographiques

- [1] E. F. Camacho and C. Bordons, “Model predictive control”, Ed. Springer-Verlag, London, 2004.
- [2] M. N. Zeilinger, D. M. Raimondo, A. Domahidi, M. Morari and C.N. Jones, “On Real-Time Robust Model Predictive Control”, *Automatica*, vol. 50, pp. 683–694, 2014.
- [3] S. Oliveira and M. Morari, “Robust Model Predictive Control for Nonlinear Systems”, in Proc. 33rd IEEE Conf. Decision Control, Lake Buena Vista, FL, pp. 3561–3566, Dec.1994.
- [4] M. V. Kothare, V. Balakrishnan, and M. Morari, “Robust Constrained Model Predictive Control Using Linear Matrix Inequalities,” *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.
- [5] P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings, “Suboptimal Model Predictive Control (Feasibility Implies Stability),” *IEEE Trans. Automat. Control*, vol. 44, no. 3, pp. 648–654, Mar. 1999.
- [6] L. Hu, B. Huang, and Y. Cao, “Robust Digital Model Predictive Control for Linear Uncertain Systems with Saturations,” *IEEE Trans. Automat. Control*, vol. 49, no. 5, pp. 792–796, May 2004.
- [7] V. Vesely and D. A. Rosinova, “Robust Output Model Predictive Control Design: BMI Approach,” *Int. J. Innovative Comput., Inf. Control*, vol. 5, no. 4, pp. 1115–1124, 2009.
- [8] N. Wada, K. Saito, and M. Saeki, “Model Predictive Control for Linear Parameter Varying Systems Using Parameter Dependent Lyapunov Function,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 12, pp. 1446–1450, Dec. 2006.
- [9] E. Scholte and M. E. Campbell, “Robust Nonlinear Model Predictive Control with Partial State Information,” *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 4, pp. 636–651, Sep. 2008.
- [10] Y. Xia, G. P. Liu, P. Shi, J. Chen, and D. Rees, “Robust Constrained Model Predictive Control Based on Parameter-Dependent Lyapunov Functions”, *Circuits Syst. Signal Process.*, vol. 27, no. 4, pp. 429–446, 2008.
- [11] K. Tanaka and M. Sugeno, “Stability Analysis and Design of Fuzzy Control System,” *Fuzzy Sets and Systems*, vol. 45, no. 2, pp. 135 – 156, 1992.
- [12] Y. Blanco, W. Perruquetti, and P. Borne, “Non-Quadratic Stability of Non Linear Systems in the Takagi-Sugeno Form”. Proceedings of the ECC 2001, Porto, Portugal. 2001.
- [13] K. Tanaka, T. Hori, T. Taniguchi, and H. O. Wang, “Stabilization of nonlinear systems based on fuzzy Lyapunov function”. Workshop IFAC Advances in Fuzzy and Neural Control. Valencia, Spain: Albertos & Sala Eds. 2001a.

- [14] K. Tanaka, T. Hori, and H. O. Wang, "A fuzzy Lyapunov Approach to Fuzzy Control System Design". ACC'2001, Washington, USA. 2001b.
- [15] M. Bernal and T. Guerra, "Generalized nonquadratic stability of continuous-time Takagi-Sugeno models", IEEE Transactions on Fuzzy Systems, vol. 18, no. 4, pp. 815-822, 2010.
- [16] Y. Morère, "Control Laws for Fuzzy Models of Takagi-Sugeno, Thesis", University of Valenciennes & Hainaut CambrXesis, Jan 2001.
- [17] T. M. Guerra and L. Vermeiren, "LMI-Based Relaxed Non-Quadratic Stabilization Conditions for Nonlinear Systems in the Takagi-Sugeno's Form. Automatica, vol. 40, no. 5, pp. 823-829, 2004.
- [18] B. Ding, H. Sun, and P. Yang, "Further Studies on LMI-Based Relaxed Non-Quadratic Stabilization Conditions for Nonlinear Systems in The Takagi- Sugeno's Form," Automatica, vol. 42, no. 3, pp. 503-508, 2006.
- [19] X. Xie, H. Ma, Y. Zhao, D. Ding and Y. Wang, "Control Synthesis Discrete-time T-S Fuzzy Systems Based on a Novel Non-PDC Control Scheme", IEEE Transactions on Fuzzy Systems, vol. 21, no. 1, pp. 147-157, 2013.
- [20] D. Peaucelle, Modèles descripteurs redondants et réduction du pessimisme des conditions LMI, JNMACS, Reims, 2007.
- [21] Y. Xia, H. Yang, P. Shi, and M. Fu, "Constrained Infinite-Horizon model predictive control for Fuzzy-Discrete Time Systems," IEEE Trans. Fuzzy Syst., vol. 19, no. 3, pp. 429-436, Apr. 2010.
- [22] L. A. Zadeh, "Information and Control", *Fuzzy Sets*, no. 8, pp. 338-353, 1965.
- [23] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and its Application to Modeling and Control," IEEE Trans. Syst., Man, Cybern., vol. SMC-15, no. 1, pp. 116-132, Feb. 1985.
- [24] E. H. Mamdani, "Application of Fuzzy Algorithm for Control of Simple Dynamic Plant". IEE, vol. 121, pp. 1585-1588, 1974.
- [25] C. H. Salvador, "Stratégie de Commande Intégrée Intelligente de Procèdes de Traitement des Eaux Usées par la Digestion Anaérobie". Thèse de doctorat, Institut National Polytechnique de Grenoble, 11 Jul. 2005.
- [26] K. Tanaka, and H. O. Wang, "Fuzzy Control Systems Design and Analysis. A Linear Matrix Inequalities Approach. " John Wiley and Sons Inc, 2001.
- [27] J. J. Buckley, "Universal Fuzzy Controllers". Automatica, vol. 28, 1992.
- [28] J. Castro, "Fuzzy Logic Controllers are Universal Approximator". IEEE Transactions on Systems Man and Cybernetics-part C, vol. 25, pp. 629-635, 1995.
- [29] K. Tanaka, and M. Sano. "Trajectory Stabilization of Model Car Via Fuzzy Control". Fuzzy Sets and Systems, vol. 70, pp. 155-170, 1995.

- [30] T. A. Johansen and A. B. Foss. "Constructing Narmax Using Armax". *International Journal of Control*, vol. 58, 1993.
- [31] S. Kawamoto and al. "an Approach to Stability Analysis of Second Order Fuzzy Systems", *Proceedings of First IEEE International Conference on Fuzzy Systems*, vol. 1, pp. 1427-1434, 1992.
- [32] K. Tanaka, T. Ikeda, and H. O. Wang. "Robust Stabilization of a Class of Uncertain Nonlinear Systems Via Fuzzy Control : Quadratic Stabilizability, H_∞ Control Theory, and Linear Matrix Inequalities". *IEEE Trans. Fuzzy Systems*, vol. 4, 1996.
- [32] H. O. Wang, K. Tanaka and M.F. Griffin, "An Approach to Fuzzy Control of Nonlinear Systems: Stability and Design Issues". *IEEE Transactions on Fuzzy Systems*, vol.4, no.1, pp. 14-23, 1996.
- [34] L. Vermeirin, "Proposition de Lois de Commandes pour la Stabilisation de Modèles Flous", Ph.D. thesis, Valenciennes University, France, 1998.
- [35] I. R. Petersen, "A Procedure for Simultaneously Stabilizing a Collection of Single Input Linear Systems Using Non-Linear State Feedback Control". *Automatica*, vol. 23, no. 1, pp. 33-40, 1987.
- [36] T. M. Guerra and W. Perruquetti, "Non-Quadratic Stabilization of Discrete Takagi Sugeno Fuzzy Models". *FUZZ-IEEE'2001*, Melbourne, Australia, 2001.
- [37] T. M. Guerra and L. Vermeiren, "Conditions for Non-Quadratic Stabilization of Discrete Fuzzy Models". *Workshop IFAC Advances in Fuzzy & Neural Control*. Valencia, Spain: Albertos & Sala Eds, 2001.
- [38] T. M. Guerra, L. Vermeiren and H. Tirmant, "Some Conditions for Non-Quadratic Stability of a Class of Non Linear Systems". *IPMU 2002*, Annecy, France, 2002.
- [39] W. J. Chang, C. C. Ku, and C. H. Chang, "PDC and Non-PDC Fuzzy Control with Relaxed Stability Conditions for Continuous-Time Multiplicative Noised Fuzzy Systems", *Journal of the Franklin Institute*, vol. 349, no. , pp. 2664-2686, 2012.
- [40] J. Richalet, A. Rault, J.L.Testud, J. Papon, "Model predictive heuristic control: application to industrial processes", *Automatica*, vol. 14, no. 5, pp. 413-428, 1978.
- [41] D.W. Clark, C.Mohtadi, P.S. Tuffs, "Generalized Predictive Control: Part I: The Basic Algorithm, Part II: Extensions and Interpretation", *Automatica*, vol. 23, no. 2, pp.137-160, 1987.
- [42] M. Morari, "Advances in Model Based Predictive Control", Oxford University Press, 1994.
- [43] L. El Ghaoui, F. Oustry and M.A. Ait Rami, "Cone complementary linearization algorithm for static output-feedback and related problems". *IEEE Transaction and Automatic Control*, Vol. 42, No. 8, pp. 1171-1176, 1997.
- [44] N. Karmakar, "A new polynomial time algorithm for linear programming". *Combinatorica*. vol. 4, no. 4, pp. 373-395, 1984.

- [45] Y. Nesterov and A. Nemirovski, "Interior point polynomial methods in convex programming: theory and applications". SIAM, Philadelphia, PA, 1994.
- [46] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, "LMI Control Toolbox". Natick, MA: MathWorks, 1995.
- [47] S. Boyd, L. El Ghaoul, E. Feron, and V. Balakrishnan, "Linear Matrix Inequalities in System and Control Theory", vol. 15. Philadelphia: Society for Industrial Mathematics, 1994.
- [48] J. M. Maciejowski, "Predictive Control with Constraints", Prentice-Hall, Harlow, UK 2002.
- [49] M.C. De Oliveira, J. Bernussou, and J.C. Geromel, "A New Discrete-Time Robust Stability Condition", Systems & Control Letters, vol. 37, pp. 261-265, 1999.
- [50] D. Peaucelle, D. Arzelier, O. Bachelier, and J. Bernussou, "A New Robust D-Stability Condition for Real Convex Polytopic Uncertainty", Systems & Control Letters, vol.40, pp. 21-30, 2000.
- [51] M. C. De Oliveira and R. E. Skelton, "Stability Tests for Constrained Linear Systems, Perspectives in Robust Control", Lecture Notes in Control and Information Sciences 268 edited by S. O. Reza Moheimani, Springer. 2001.
- [52] K. Zhou and P. Khargonedkar, "Robust Stabilization of Linear Systems with Norm-Bounded Time-Varying Uncertainty". Sys. Control Letters, vol. 10, pp. 17-20, 1988.
- [53] L.X. Wang and J.M. Mendel, "Fuzzy Basis Functions, Universal Approximation and Irthogonal Least-Squares", IEEE T. on Neural Networks, vol. 3, no. 5, pp. 807-814, 1992.
- [54] J. Löfberg, "YALMIP: a Toolbox for Modeling and Optimization in MATLAB". In Proc. of the CACSD Conference, Taipei, Taiwan, 2004.
- [55] M.Khairy, A. L. Elshafei and H. M. Emara, "LMI Based Design of Constrained Fuzzy Predictive Control", Fuzzy Sets and Systems, vol. 161, no. , pp. 893-918, 2010.
- [56] R. Rajesh and M. R. Kaimal, "T-S Fuzzy Model with Nonlinear Consequence and PDC Controller for a Class of Nonlinear Control Systems," Appl. Soft Comput., vol. 7, no. 3, pp. 772-782, 2007.

ملخص

التحكم التنبؤي الغامض (الضبابي) في وجود القيود المفروضة عليها يعتبر من أكثر التقنيات فعالية في الميدان الصناعي وكذا مجال الأتوماتيك، وذلك لاحترامه للقيود المفروضة على الأنظمة. ومع ذلك يبقى تصميم قانون ديناميكي فعال للتحكم بمختلف الأنظمة محل نقاش.

في هذه الأطروحة حاولنا مناقشة و تصميم مقارنة التحكم التنبؤي النموذجي للأنظمة الرقمية ذات النماذج الغامضة T-S مع القيود المفروضة عليها، وقد فضلنا لذلك استعمال قانوني التحكم الضبابي PDC و Non-PDC لبناء الشروط اللازمة والكافية، و من ثم وضعها على شكل أنماط رياضية (مصفوفة عدم المساواة الخطية) LMI ، لتمكيننا من إيجاد أفضل حل ممكن لكل مقارنة قانون تحكم (نظام مراقبة). هذه الأخيرة يجب أن تضمن استقرار الأنظمة المدروسة حتى في وجود مؤثرات خارجية. و النتائج المتحصل عليها بعد إجراء محاكاة على أنظمة مختلفة، والتي بينت أفضلية استعمال مقارنة Non-PDC على PDC وذلك لأخذها بعين الاعتبار مشكلة شدة الشروط المفروضة لإيجاد أفضل حل.

كلمات مفتاحية : النموذج التحكم التنبؤي (MPC) ، الأنظمة الغامضة (الضبابية) T-S ، مصفوفة عدم المساواة الخطية LMI ، قانوني التحكم الضبابي PDC و Non-PDC.

Abstract

Constrained Fuzzy predictive control become among efficient techniques in control and industrial field for its tolerance, and respect of imposed constraints.

In the framework of predictive control, the synthesis follows the minimization at each sampling time, of an upper bound for a quadratic cost associated with an infinite time horizon. The optimization problem which takes into account state and control constraints is described in terms of a semi definite programming one including linear matrix inequalities. The obtained results, were presenting in tow approaches, PDC and Non-PDC. This last, make the searching of solution for the problems of LMI easier than using the PDC controller and lead to reduce the problem of conservatism.

Key words: Model predictive control (MPC), Takagi-Sugeno (T-S) fuzzy systems, Linear matrix inequality (LMI), Parallel distributed compensation PDC, Non-PDC.

Résumé

La commande prédictive floue sous contraintes est devenue parmi les techniques les plus utilisées dans le domaine d'automatique et le milieu industriel pour sa tolérance, sa fiabilité et son respect des contraintes imposées.

Dans le cadre de commande prédictive, la synthèse découle de la minimisation à chaque instant d'échantillonnage d'une borne supérieure d'un coût quadratique évalué sur un horizon temporel infini. Le problème d'optimisation qui inclut des contraintes sur l'état et la commande est formulé comme un problème de programmation semi définie à base d'inégalités matricielles linéaires. Les résultats obtenus, sont présentés en deux approches, PDC puis Non-PDC. Cette dernière approche facilite la recherche d'une solution aux problèmes LMI et conduit à une réduction du conservatisme.

Mots Clés : Commande prédictive à modèle (MPC), Systèmes flous Takagi-Sugeno, Inégalités matricielle linéaire (LMI), PDC, Non-PDC