

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITÉ MENTOURI DE CONSTANTINE
FACULTÉ DES SCIENCES DE L'INGÉNIEUR
DÉPARTEMENT D'ÉLECTRONIQUE

THÈSE

Présentée pour obtenir le diplôme de
Doctorat en Sciences

Spécialité : Electronique
Option : Contrôle

Par :

OUASSILA BOUREBIA

THÈME

Commande Prédictive Floue des Systèmes Non Linéaires

Soutenue le 26 /05 /2011 devant le jury composé de:

Président	Salim FILALI Professeur, Université de Constantine
Rapporteur	Khaled BELARBI Professeur, Université de Constantine
Examineurs	Brahim BOUTAMINA Maître de conférences, Université de Constantine
	Noureddine SLIMANE Maître de conférences, Université de Batna
	Samir LADACI Maître de conférences, Université de Skikda

Remerciements

J'exprime toute ma reconnaissance à monsieur Khaled BELARBI, Professeur à l'université de Constantine, pour avoir assuré l'encadrement de ce travail. Je le remercie pour son soutien, son orientation et ses consignes. Son expérience et sa connaissance ont contribué à ma formation scientifique.

Je tiens à exprimer mes vifs remerciements à monsieur Salim Filali, Professeur à l'université de Constantine, pour avoir présidé le jury de ma soutenance de thèse.

Mes sincères remerciements vont également à Messieurs Brahim BOUTAMINA, Maître de Conférences à l'université Mentouri Constantine, monsieur Samir LADACI Maître de Conférences à l'université de Skikda, ainsi qu'à Monsieur SIIMANE Nouredine Maître de Conférences à l'université de Batna, pour l'intérêt qu'ils ont porté à ce travail et qui ont accepté d'en être les rapporteurs.

J'adresse un grand merci à mon amie et collègue Mme KHNIENNE Nour el Houda, pour son considérable aide à la mise en forme de ce manuscrit.

Je n'oublierai pas de remercier mon mari Mahfoud FERKOUS pour sa patience et son aide dans les moments difficiles. Enfin un grand merci à mes enfants M^{ed} Haithem, Ghada-Chaima et Maroua, source de mon plus grand bonheur, ainsi que mon père et mes frères et sœurs pour leur amour et leur soutien, également à ma belle famille.

Ouassifa Bourebia

Sommaire

INTRODUCTION GENERALE	7
------------------------------------	----------

Chapitre I: La Commande Prédictive

I.1 INTRODUCTION.....	11
I. 2 Principes de la commande prédictives.	11
I.2.1 Concepts de base	11
I.2.2 Paramètres de réglage	12
I.3.Commande Prédictive Généralisée.....	13
I.3.1 Le cas sans contraintes	13
I.3.1.1 Le modèle de prédiction et la fonction coût	13
I.3.1.2 La loi de commande prédictive	14
I.3.2 La Commande prédictive généralisée avec contraintes	15
I.4. La Commande prédictive à base d'un modèle	16
I.4.1. Cas sans contraintes.....	16
I.4.1.1 Calcul des prédictions.....;	16
I.4.1.2 Calcul de la commande.....	17
I.4.2 Cas avec contraintes.....	18
I.4.2.1 Position du problème.....	18
I.4.2.1 Méthodologies de Solutions.....	18
I.5 Commande prédictive non linéaire.....	19
I.5.1Formulation Générale.....	19
I.5.2 Solution du problème de la commande prédictive non linéaire.....	20
I.6 Stabilité de la commande Prédictive.....	20
I.7 La commande Prédictive fonctionnelle.....	21
I.7.2. Le modèle interne	21
I.7.2 La trajectoire de référence.....	21

I.7.3 Structuration de la variable manipulée	23
I.7.4.Principe de l'horizon fuyant.....	24
I.8 Conclusion	24

Chapitre II : La Commande prédictive généralisée floue hiérarchisée

II.1 INTRODUCTION.....	27
II.2 Commande Prédictive Généralisée Floue	28
II.3 La commande hiérarchisée	30
II.3.1 Couche inférieure.....	30
II.3.2 Couche supérieure	30
II.3.3 Calcul des contrôleurs locaux	32
II.4.Résultats de la simulation.....	34
II.4.1.Control de concentration dans un CSTR.....	34
II.4.1.1 Modélisation floue	35
II.4.1.2 Commande Floue Hiérarchisée.....	36
II.4.2 Système mécanique Masse-Ressort-Amortisseur.....	39
II.4.2.1 Modélisation floue	40
II.4.2.2 Commande Floue Hiérarchisée.....	41
II.4 Conclusion	43

Chapitre III : LA Commande Prédictive Fonctionnelle Floue Hiérarchisée

III.1 INTRODUCTION.....	45
III.2 La PFC Floue hiérarchisée.....	45
III.2.1 Le modèle flou.....	45
III.2.2 La commande fonctionnelle hiérarchisée	46
III.2.1 La Couche inférieure.....	46
III.2.2 La couche supérieure.....	48

III.3. Exemple de simulation.....	49
III.4 Conclusion.....	51

Chapitre IV : Une solution explicite stable et admissible à la commande prédictive floue à base de modèle d'état

IV.1 INTRODUCTION.....	53
IV.2 La commande prédictive floue.....	53
IV.2.1 Prise de décision dans un environnement flou.....	53
IV.2.2 Commande prédictive floue à base de modèle d'état.....	55
IV.3 La solution explicite admissible.....	56
IV.3.1 Position du problème.....	56
IV.3.2 Principal résultat: conditions d'admissibilité.....	57
IV.4. Résultats de simulation.....	61
IV.5 Conclusion	65
CONCLUSION GENERALE	67
ANNEXE A.....	69
ANNEXE B	77
BIBLIOGRAPHIE.....	81

Introduction générale

Introduction Générale

La commande prédictive est une méthode qui repose sur l'utilisation d'un modèle, afin de prédire le comportement du système à commander sous un certain nombre de contraintes. Cette méthode n'a connu un réel essor qu'au début des années 80, grâce aux travaux de D.W Clarke et al [14].

Toutefois cette technique de commande, a suscité un intérêt dans le domaine industriel dès la fin des années 70. En effet en 1978, J.Richalet *et al* publient leurs premiers résultats obtenus dans le cas des applications industrielles en réalisant un logiciel appelé PFC (Predictive Functional Control) [29]. Ce dernier permet une mise en œuvre aisée à partir d'information disponibles sur le système et des spécifications adoptées [28]. Il a fallu attendre 1987, pour voir publier les premiers résultats obtenus par J. Richalet *et al*, sur des systèmes électro-mécaniques rapides, tel que des commandes d'axes d'un robot [11].

La commande prédictive a connu très tôt un vif engouement dans le secteur de l'industrie chimique et pétrolière [6][15]. Seulement, elle reste non adéquate pour les systèmes fortement non linéaires, d'où l'intérêt de développer une commande prédictive qui ne se limite pas uniquement à des systèmes linéaires mais s'étend à des systèmes non linéaire par l'élaboration d'un système de type MPC non linéaire (NMPC) [27] [22].

L'idée du sujet proposé dans le cadre de cette thèse est d'étudier une technique [7] [8] [9] de commande de système fortement non linéaire qui fait appel à la commande prédictive associée à la logique floue FMPC (Fuzzy Model Predictive Control) [12] [33]. Cette approche exploite les capacités de modélisation de la logique floue à la commande prédictive à base de modèle, en particulier en utilisant les propriétés d'approximation universelle des modèles flous de Takagi Sugeno [32]. Ces modèles flous consistent en une subdivision du système global non linéaires en sous systèmes locaux linéaires [24].

L'objectif de la première partie de notre étude est alors d'étendre les méthodes de commande prédictive linéaire aux systèmes non linéaires par l'utilisation de ces modèles.

Dans ce contexte, deux approches peuvent être adoptées. Dans la première approche, le système d'inférence flou T-S sera utilisé pour fusionner les modèles linéaires locaux. Un modèle de prédiction linéaire global variant dans le temps sera alors élaboré, puis il sera exploité dans le calcul de la loi de commande linéaire globale adaptée au système MBPC [20] [26] [30]. Dans la seconde approche, une commande linéaire sera calculée pour chaque modèle linéaire local de chaque règle. Ces contrôleurs locaux seront fusionnés afin d'élaborer le contrôleur global. Dans ce cas, la méthode adaptée (aux contrôleurs locaux) est généralement basée sur les poids associé à chaque modèle local [24] [25]. Sauf que l'optimalité de la loi de commande globale n'est pas garantie.

Afin de pallier à ce problème, plusieurs auteurs ont suggérés l'élaboration d'un algorithme de contrôle hiérarchique garantissant l'optimalité de la commande prédictive floue [20]. La méthodologie a été appliquée à la commande prédictive basée sur un modèle floue à réponse impulsionnelle où chaque règle du modèle flou a été générée en fonction de la réponse du système à une impulsion. Cependant, cela conduit généralement à des modèles d'ordre élevé [20].

Dans le cadre de notre étude, nous proposons une extension de l'approche hiérarchique de [20] à la conception d'une loi de commande prédictive généralisée GPC [7] et d'une loi de commande prédictive fonctionnelle PFC [8] [9].

Nous consacrerons un effort particulier à l'élaboration d'un modèle qui se base sur la modélisation floue de Takagi-Sugeno. Notons que les commandes GPC et PFC sont des techniques adaptées aux modèles d'entrée-sortie, qui sont plus compacts et d'une moindre dimension que ceux d'impulsion.

La stabilité de la commande prédictive floue sera aussi étudiée dans le cadre de cette thèse, où nous proposons une nouvelle solution dite explicite [2] [4] basée sur les ensembles invariants polyédraux [12], en tenant compte des contraintes sur la commande.

Cette thèse est organisée comme suit :

Le premier chapitre a pour but de rappeler les principes généraux de la commande prédictive, puis d'analyser de façon plus détaillée les systèmes discrets invariants dans le temps. Les structures, qui serviront de support aux développements futurs de la commande prédictive généralisée (GPC), celles à base de modèle d'état (MPC) ainsi que la Commande Prédictive Fonctionnelle PFC seront étudiées.

Dans le deuxième chapitre, nous présenterons la commande prédictive généralisée floue hiérarchisée. Un ensemble d'exemples de simulation mettra en évidence la validité de cette approche.

Le troisième chapitre sera consacré à l'élaboration de la loi de commande prédictive fonctionnelle floue hiérarchisée.

Le quatrième chapitre traitera de la solution explicite admissible introduite par la nouvelle méthodologie de résolution du problème d'optimisation de la commande prédictive floue à base de modèle d'état. La méthode mise au point utilisera les ensembles invariants tout en considérant des différentes contraintes.

Chapitre I

La Commande Prédictive

Chapitre I

La Commande Prédicative

I.1. Introduction:

Ce chapitre a pour but d'introduire les principes généraux de la commande prédictive et d'analyser les structures les plus utilisées en l'occurrence la commande prédictive généralisée (GPC), la commande prédictive à base de modèle d'état (MPC) et la commande prédictive fonctionnelle. Pour chacune de ces formulations, la représentation des solutions sous la forme d'une loi de commande est représenté dans les deux cas avec et sans contraintes.

I.2 Principes de la commande prédictive:

Fondamentalement, l'algorithme de commande prédictive prend en compte le comportement futur du système afin d'élaborer une commande permettant le meilleur suivi d'une trajectoire connue [10] [11] [15].

I.2.1 Concepts de base:

Toutes les techniques proposées dans la littérature ont commun les concepts suivants :

- Utilisation d'un modèle pour construire la prédiction des signaux intéressants : états ou sorties.
- Connaissance de la trajectoire à suivre sur un horizon au moins aussi long que l'horizon de prédiction.
- Minimisation d'un critère le plus souvent quadratique incluant l'erreur de poursuite et l'effort de commande
- Utilisation d'un algorithme de résolution produisant en temps réel la solution optimale /ou sous optimale/admissible
- Application du premier élément de la séquence de commande calculée.
- Répétition de la procédure à la période d'échantillonnage suivante, selon le principe de l'horizon fuyant.

Le modèle de prédiction est fondamental l'application de la commande prédictive, il constitue la cheville de la loi de commande, il faut donc le construire avec le plus grand soin [11].

Le principe de l'horizon fuyant est tel qu'à chaque période d'échantillonnage, il est nécessaire de calculer une séquence de commandes en boucle ouverte, améliorées par l'arrivée de nouvelles mesures obtenues sur le procédé. La loi de commande prédictive est ainsi implémentée par une suite de problèmes d'optimisation.

La figure I.1 illustre le principe de la commande prédictive.

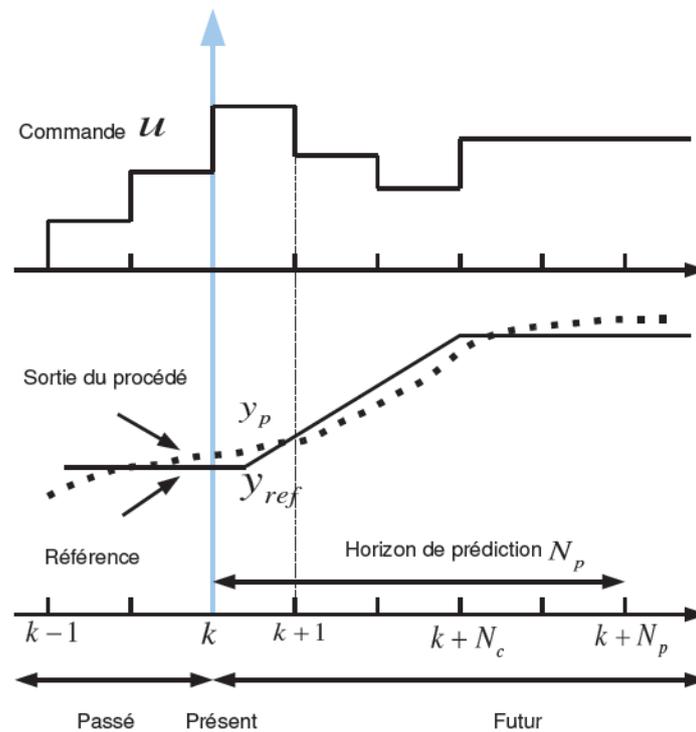


Figure I.1 : Schéma de principe de la commande prédictive

I. 2.2 Paramètres de réglage :

Les paramètres de réglages qu'on retrouve dans la majorité des stratégies de commande prédictives sont:

- Les horizons de prédiction des sorties et/ou des états;
- L'horizon de commande;

- Les pondérations intervenant dans le critère de performance sur les erreurs et sur l'effort de commande;
- Les paramètres terminaux tels que les coûts terminaux.

I.3 Commande Prédictive Généralisée (GPC)

I.3.1 Le cas sans contraintes

C'est en 1985 que D.W. Clarke *et al* [14], présente la première version de la Commande Prédictive Généralisée (Generalised Predictive Control, GPC), par la suite cette méthode a connue le plus grand nombre d'applications [6]. Dans ce paragraphe nous développons les idées de cette stratégie.

I.3.1.1 Le modèle de prédiction et la fonction coût :

La commande GPC utilise pour la prédiction du comportement un modèle entrée/sortie par fonction de transfert de type **CARIMA** (Controlled Auto Regressive Integrated Moving Average) [14]:

Dans le cas du GPC, $x(t)$ le modèle de prédiction est donné par

$$A(z^{-1}) \cdot y(t) = B(z^{-1}) \cdot u(t-1) + C(z^{-1}) \cdot e(t) / \Delta \quad (I.1)$$

où : $C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{nc} z^{-nc}$ (nc est le degré de C).

A et B sont des polynômes en l'opérateur de retard z^{-1} de degrés na , nb respectivement

$\Delta = 1 - z^{-1}$, appelé opérateur de différence, son rôle est d'assurer une action intégrale sur le contrôleur à fin d'annuler l'effet des perturbations.

$e(t)$: est un bruit gaussien blanc à moyenne nulle.

Le modèle décrit par l'équation (I.1) représentatif du système à contrôler est appelé modèle **CARIMA**, ce modèle constitue le modèle de base de la méthode GPC à partir duquel sera dérivée l'expression de la loi de commande.

Pour déterminer la commande à appliquer au système à l'instant t , la méthode GPC minimise le critère suivant :

$$J(t) = \sum_{j=N1}^{N2} [y(t+j) - w(t+j)]^2 + \sum_{j=N1}^{Nu} \lambda * [\Delta u(t+j-1)]^2 \quad (I.2)$$

Avec : $\Delta u(t+j) = 0$ pour $j \geq Nu$

Où :

$N1$: est l'horizon minimal de prédiction.

$N2$: est l'horizon maximal de prédiction.

Nu : est l'horizon de commande.

λ : est un facteur de pondération du signal de commande ($\lambda > 0$).

$w(t+j)$: est la trajectoire future de référence connue à l'avance.

$\Delta u(t+j)$: est l'incrément de contrôle.

Pour simplifier le calcul de la loi de commande *GPC*, on va poser : $C(z^{-1}) = 1$,

On montre que le prédicteur s'écrit : sous forme matricielle :

$$\hat{Y} = G \cdot \tilde{U} + F \quad (\text{I.3})$$

Où : \hat{Y} , U et F sont des vecteurs de dimension $N \times 1$ avec :

$$\hat{Y} = [y(t+1), y(t+2), \dots, y(t+N)]^t$$

$$\tilde{U} = [\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+N-1)]^t$$

$$F = [f(t+1), f(t+2), \dots, f(t+N)]^t$$

$$\text{et } f(t+j) = [g(j,j).z^j + g(j,j+1).z^{-(j+1)} + \dots + g(j, nb+j-1).z^{-(nb+j-1)}].\Delta u(t+j-1) + F_j.y(t).$$

Pour $N_1=1$ et $N_2=N$

La matrice G définie dans l'équation (I.3) serait une matrice de dimension $N \times N$ tel que :

$$G = \begin{bmatrix} g(1,0) & 0 & 0 & \dots & 0 \\ g(2,1) & g(2,0) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ g(N,N-1) & g(N,N-2) & g(N,N-3) & \dots & .g(N,0) \end{bmatrix} \quad (\text{I.4})$$

1.3.1.2 La loi de la commande prédictive :

La loi de commande est calculée de façon à minimiser le critère quadratique suivant :

$$J = \sum_{j=1}^{N_2} [(y(t+j) - w(t+j))]^2 + \sum_{j=1}^{N_u} \lambda .[\Delta u(t+j-1)]^2 \quad (\text{I.5})$$

avec : $\Delta u(t+j-1) = 0$ pour $j \geq N_u$

Le terme $[(y(t+j) - w(t+j))]^2$ représente la somme pondérée des erreurs futures entre les sorties futures et les signaux de consignes $w(t+j)$ ($j=1..N_2$).

Le terme $[\lambda .\Delta u(t+j-1)]^2$ représente le coût de l'effort de commande

Le critère (I.2) peut être écrit sous forme matricielle comme suit :

$$J = (Y - W)^T \cdot (Y - W) + \lambda \cdot U^T \cdot U \quad (\text{I.6})$$

La relation du prédicteur est: $Y = G.U + F$ en remplaçant cette dernière dans l'équation (I.5)

$$\text{ont trouve : } J = (G.U+F-W)^T \cdot (G.U+F-W) + \lambda.U^T \cdot U \quad (\text{I.7})$$

J est minimale si $\delta J/\delta \dot{U} = 0$, qui donne

$$U = [(G^T \cdot G) + \lambda I]^{-1} \cdot G^T \cdot (W - F) \quad (\text{I.8})$$

Où I représente la matrice identité.

Cette dernière équation fournit les incréments futurs de contrôle pour les instants ' t ' à $(t+N_u-1)$, basés sur les informations disponibles à l'instant ' t '. Seulement $\Delta u(t)$ sera appliqué au système et la commande est alors tel que : $u(t) = \Delta u(t) + u(t-1)$

La minimisation du critère sera reprise à chaque échantillon pour le calcul de la nouvelle commande à appliquer au système.

1.3.2 La commande prédictive généralisée avec contraintes :

Dans toutes les techniques de commande prédictive, les variables d'entrées, d'états et de sorties d'un système sont souvent contraintes par leurs domaines de définitions. Ces contraintes sont de natures diverses: limitation physique des actionneurs, spécification de la qualité d'un produit, exigence de sécurité et domaine de tolérance pour une sortie, ...etc. Par exemple, en pratique, le signal de commande doit satisfaire la contrainte du domaine de validité de l'actionneur (son domaine de fonctionnement). Dans ce cas, si on suppose l'intervalle $[u_{min}, u_{max}]$ comme le domaine de validité de l'actionneur, alors la contrainte de commande sur $u(t)$ s'écrit $u_{min} \leq u \leq u_{max}$

Cette contrainte peut aussi s'exprimer par l'incrément de commande $\Delta u(k)$

$$\Delta u_{min} < \Delta u < \Delta u_{max} \quad (\text{I.9})$$

Dans ce cas le problème de la GPC avec contraintes est de minimiser J en tenant compte de (I.9):

$$\text{Donc on peut écrire } J \text{ sous la forme : } J = \frac{1}{2} U^T Q U + d^T U + c \quad (\text{I.10})$$

Avec :

$$Q = G^T G + \lambda I \quad (\text{I.11})$$

$$d = 2(f_{conu} - W)^T G \quad (I.12)$$

$$c = (f_{conu} - W)^T (f_{conu} - W) \quad (I.13)$$

Pour résoudre ce problème on utilise la programmation quadratique, *QP* (Quadratic programming) [22] [23].

1.4 La commande prédictive à base d'un modèle d'état

1.4.1 Cas sans contraintes:

1.4.1.1 Calcul des prédictions [11] :

Le modèle de base pour calculer les prédictions est un modèle d'état discret donné par :

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

$$y(k) = C_y x(k)$$

$$z(k) = C_z x(k) \quad (I.14)$$

$x(k)$ variable d'état, $y(k)$ sorties du système, $z(k)$ sorties à commander, k l'instant d'échantillonnage. Souvent on a $z(k) = y(k)$ Aussi on suppose que nous ne savons rien au sujet des perturbations ou bruit de mesure, alors tout ce que nous pouvons faire est de prédire à partir des équations ci-dessus [11], nous obtenons le prédicteur sous forme matricielle :

$$\begin{bmatrix} \hat{x}(k+1|k) \\ \vdots \\ \hat{x}(k+N_u|k) \\ \hat{x}(k+N_u+1|k) \\ \vdots \\ \hat{x}(k+N_2|k) \end{bmatrix} = \underbrace{\begin{bmatrix} A \\ \vdots \\ A^{N_u} \\ A^{N_u+1} \\ \vdots \\ A^{N_2} \end{bmatrix} x(k)}_{\text{passé}} + \underbrace{\begin{bmatrix} B \\ \sum_{i=0}^{N_u-1} A^i B \\ \sum_{i=0}^N A^i B \\ \vdots \\ \sum_{i=0}^{N_2-1} A^i B \end{bmatrix} u(k-1)}_{\text{futur}} + \underbrace{\begin{bmatrix} B & \dots & 0 \\ Ab+B & \dots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{N_u-1} A^i B & \dots & B \\ \sum_{i=0}^{N_u} A^i B & \dots & AB+B \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{N_2-1} A^i B & \dots & \sum_{i=0}^{N_2-N_u} A^i B \end{bmatrix}}_{\text{futur}} \begin{bmatrix} \Delta \hat{u}(k|k) \\ \vdots \\ \Delta \hat{u}(k+N_u-1|k) \end{bmatrix} \quad (I.15)$$

Les prédictions de z sont maintenant obtenues simplement

$$\hat{z}(k+1|k) = C_z x(k+1|k)$$

$$\hat{z}(k+2|k) = C_z \hat{x}(k+2|k)$$

$$\vdots$$

$$\hat{z}(k+N_2|k) = C_z \hat{x}(k+N_2|k)$$

Où:

$$(I.16)$$

$$\begin{bmatrix} \hat{z}(k+1 | k) \\ \vdots \\ \hat{z}(k+N_2 | k) \end{bmatrix} = \begin{bmatrix} C_z & 0 & \dots & 0 \\ 0 & C_z & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & C_z \end{bmatrix} \begin{bmatrix} \hat{x}(k+1 | k) \\ \vdots \\ \hat{x}(k+N_2 | k) \end{bmatrix}$$

1.4.1.2 Calcul de la commande

La fonction coût s'écrit :

$$V(k) = \sum_{i=N_1}^{N_2} \|\hat{z}(k+i | k) - r(k+i)\|_{Q(i)}^2 + \sum_{i=0}^{N_u-1} \|\Delta\hat{u}(k+i | k)\|_{R(i)}^2 \quad (\text{I.17})$$

Avec : $r(k)$ référence, $Q(i)$ et $R(i)$ sont des matrices des pondérations.

Nous pouvons récrire ceci sous forme condensée :

$$V(k) = \|Z(k) - T(k)\|_Q^2 + \|\Delta U(k)\|_R^2$$

$$\text{Où } Z(k) = \begin{bmatrix} \hat{z}(k+N_1 | k) \\ \vdots \\ \hat{z}(k+N_2 | k) \end{bmatrix} \quad T(k) = \begin{bmatrix} \hat{r}(k+N_1 | k) \\ \vdots \\ \hat{r}(k+N_2 | k) \end{bmatrix} \quad \Delta U(k) = \begin{bmatrix} \Delta\hat{u}(k/k) \\ \vdots \\ \Delta\hat{u}(k+N_u-1/k) \end{bmatrix}$$

A partir des prédictions on peut écrire :

$$Z(k) = \Psi x(k) + Y u(k-1) + \Theta \Delta U(k) \quad (\text{I.18})$$

$$\text{En posant : } \varepsilon(k) = T(k) - \Psi x(k) - Y u(k-1) \quad (\text{I.19})$$

On obtient :

$$V(k) = \text{const} - \Delta U(k)^T G + \Delta U(k)^T H \Delta U(k)$$

$$\text{Où: } G = 2\Theta^T Q \Theta \varepsilon(k) \text{ et } H = \Theta^T Q \Theta + R \quad (\text{I.20})$$

Et ni F ni H dépendent de $\Delta U(k)$. Pour calculer $\Delta U(k)$ optimal, il faut calculer le gradient de $V(k)$ la solution est alors donnée par

$$\Delta U(k)_{opt} = \frac{1}{2} H^{-1} G \quad (\text{I.21})$$

Nous utilisons seulement la partie de cette solution qui correspond au premier élément du vecteur, nous pouvons représenter ceci comme :

$$\Delta u(k)_{opt} = \begin{bmatrix} I_l, \underbrace{0_l, \dots, 0_l}_{(N_U-1) \text{ times}} \end{bmatrix} \Delta U(k)_{opt} \quad (\text{I.22})$$

Où : I_l est la $l \times l$ identité matrice, et 0_l est la $l \times l$ zéro matrice.

1.4.2 Cas avec contraintes

1.4.2.1 Position du problème

Le problème de la commande prédictive avec contraintes est de minimiser

$$\Delta U(k)^T H \Delta U(k) - G^T \Delta U(k) \quad (\text{I.23})$$

Avec les contraintes. Ce problème est formulé par l'équation:

$$\min_{\theta} \frac{1}{2} \theta^T \Phi \theta + \Phi' \theta$$

Qui est un problème standard appelé programme quadratique : QP .

1.4.2.2 Méthodologies de solutions:

1°) Méthode de l'ensemble actif:

Il s'agit ici d'une des méthodes les plus connues pour résoudre les problèmes QP, qui doit son nom au fait que la procédure essaie itérativement de trouver la séparation entre l'ensemble de contraintes actives et inactives pour la solution optimale par la résolution d'une suite de problèmes QP avec contraintes égalité. Les problèmes QP avec contraintes égalité se résument par projection à la résolution d'un problème sans contrainte [13].

2°) Méthode du point intérieur

Ces algorithmes font partie de la classe qui utilise une fonction barrière et des algorithmes de type Newton. Ces algorithmes peuvent être vus comme une généralisation des méthodes d'optimisation non linéaire classiques pour les problèmes d'optimisation avec contraintes convexes. Leurs performances font qu'ils sont particulièrement adaptés aux problèmes d'optimisation convexe de grandes dimensions [18].

3°) Programmation Quadratique multiparamétrique (MPQP)

Il s'agit d'un développement relativement récent qui est toujours un domaine de recherche active. Si les trois méthodes décrites précédemment se résument à l'emploi de

techniques performantes permettant de résoudre en ligne une certaine classe de problèmes d'optimisation, cette méthode conduit à l'implémentation de la même stratégie de commande prédictive par l'intermédiaire de l'évaluation d'une fonction explicite du vecteur d'état [2]. Cette fonction est construite hors ligne. La solution numérique des problèmes d'optimisation est dite explicite lorsqu'un calcul direct des variables dépendantes (arguments optimaux, optimum) peut être réalisé à partir de quantités connues, le calcul est alors dite explicite [2] [5] [17].

1.5 Commande prédictive non linéaire

L'utilisation de la commande prédictive non linéaire est justifiée dans les secteurs où les non-linéarités de processus sont fortes et les demandes du marché exigent des changements fréquents en régimes de fonctionnement. D'un point de vue théorique, employer un modèle non linéaire change le problème d'optimisation d'un QP convexe en programme non linéaire non convexe, dont la solution est beaucoup plus difficile. Il n'y a aucune garantie, par exemple, que l'optimum global peut être trouvé.

1.5.1 Formulation générale

Soit un système non linéaire échantillonné décrit par l'équation aux différences suivante :

$$x(k + 1) = f(x(k), u(k)) \quad (\text{I.24})$$

Où x est l'état du système, u son entrée de commande et f est une application continue de $\mathbb{R}^n \times \mathbb{R}^m$ dans \mathbb{R}^n . L'indice k est associé à une période d'échantillonnage qui est supposée constante.

La commande u est supposée appartenir à un ensemble compact et convexe \mathbb{U} de \mathbb{R}^m :

$$u(k) \in \mathbb{U} \subset \mathbb{R}^m \quad (\text{I.25})$$

Il est supposé aussi que l'ensemble admissible \mathbb{U} contient l'origine $\mathbf{0}$ dans son intérieur,

L'objectif de la loi de commande est de stabiliser le système en $\mathbf{0}$ tout en garantissant que les trajectoires de l'état du système restent dans un ensemble convexe et fermé \mathbb{X} :

$$x(k) \in \mathbb{X} \quad (\text{I.26})$$

Une fonction de coût est associée à chaque profil de commande \mathcal{U} de la façon suivante :

$$J_N(x, k, \mathcal{U}) = F(x(k + N)) + \sum_{i=k}^{k+N-1} L(x(i), u(i)) \quad (\text{I.27})$$

Notons que la fonction de coût contient une pénalisation sur l'état final $x(k + N)$.

En plus du terme de pondération sur l'état final, une contrainte finale explicite sur l'état peut aussi être utilisée. Elle peut s'écrire d'une façon générale comme suit :

$$x(k + N) \in X_f \subset \mathbb{X} \quad (\text{I.28})$$

Où X_f est un sous-ensemble fermé et convexe de \mathbb{X} .

Le problème de commande optimale suivant peut alors être défini pour le système à l'état x à l'instant k :

$$\min_{\mathcal{U}} J_N(x, k, \mathcal{U})$$

Sous les contraintes ci-dessus.

Il s'agit donc d'un problème d'optimisation, *généralement non convexe*, dans lequel la variable de décision est le profil de commande $\mathcal{U} \in \mathbb{U}^N \subset \mathbb{R}^{Nm}$.

1.5.2 Solution du problème de la commande prédictive non linéaire

Comme on l'a vu dans la section précédente, l'utilisation d'un modèle non linéaire change le problème de commande d'un programme quadratique convexe en un problème non linéaire non convexe, qui est beaucoup plus difficile à résoudre et ne fournit aucune garantie que l'optimum global peut être trouvé. Comme dans la commande en temps réel l'optimum doit être obtenu en un intervalle de temps limité, le temps nécessaire pour trouver l'optimum (ou une approximation acceptable) est une question importante.

Le problème est souvent résolu en utilisant des techniques de Programmation quadratique séquentielle (SQP). Ce sont des extensions des méthodes de Newton pour converger à la solution des conditions de Karush-Kuhn-Tucker (KKT) du problème d'optimisation avec contraintes. Une autre méthode utilisée pour résoudre le problème de la commande prédictive non linéaire est la programmation dynamique (DP) [13].

I.6 Stabilité de la commande prédictive

Même si le problème de la commande prédictive est résolu parfaitement à chaque instant de décision, son utilisation ne conduit pas nécessairement à un système stable en boucle fermée. Le problème de la stabilité qui pourrait surgir lorsque le principe de l'horizon fuyant est utilisé sans précaution particulière a été soulevé très tôt par Kalman en 1960. En effet, Kalman a observé même pour les systèmes linéaires : « L'optimalité n'implique pas la stabilité ».

Accompagnant le succès de la MBPC dans le contexte industriel, des travaux académiques commençaient à voir le jour visant à caractériser les conditions de stabilité. Le passage aux représentations d'état des systèmes a certainement amorcé le virage dans la bonne direction. Le pas décisif a été franchi par S. S. Keerthi, E. G. Gilbert [21] où la première preuve formelle de stabilité asymptotique de la boucle fermée pour un système non linéaire sous la MBPC a été démontrée en utilisant une méthode de Lyapunov. Depuis une floraison de formulations a vu le jour. Aujourd'hui, une vision épurée et extrêmement synthétique des conditions de stabilité des schémas de la MBPC est ainsi disponible [23] [26].

I.7 La Commande Prédicative Fonctionnelle

Cette méthode introduite par *J. Richalet et al* [29] applique la même philosophie que la commande prédictive généralisée de *D.W. Clarke et al* [14].

Néanmoins, la mise en œuvre de cette philosophie commune s'avère relativement différente, par les notions utilisées et la mise en équations.

Cette méthode repose sur quatre principes essentiels:

- Modèle interne.
- Trajectoire de référence.
- Structuration de la variable manipulée.
- Principe de l'horizon fuyant.

1.7.1 Modèle interne :

Le modèle interne de prédiction de la sortie du système est choisi linéaire et fait intervenir:

1/ Soit les coefficients de la réponse impulsionnelle :

$$S_m(t+j) = \sum_{i=1}^N a_i u(t+j-i) \quad (\text{I.29})$$

2/ Soit une mise sous forme de variable d'état:

$$\begin{aligned} x_m(t) &= F_m x_m(t-1) + G_m u(t-1) \\ s_m(t) &= C_m x_m(t) \end{aligned} \quad (\text{I.30})$$

1.7.2. Trajectoire de référence

Le ralliement de la consigne se fait par l'intermédiaire d'une trajectoire de référence réinitialisé à chaque instant sur la sortie réelle du processus. Il est usuel de choisir comme trajectoire de référence un premier ordre afin de rallier la consigne en douceur, selon la relation

$$C(t+j) - y_r(t+j) = \lambda^j [C(t) - y_p(t)] \quad (\text{I.31})$$

Avec C , y_r , y_p sont la consigne, la trajectoire de référence, la sortie réelle du processus, respectivement. Ainsi que la variable λ est définie par $\lambda = \exp(-3T_e / T_r)$, où T_e est la période d'échantillonnage et T_r représente le temps de réponse de la boucle fermée.

On préfère parler de points de coïncidence, plutôt que d'horizons de prédiction. En ces points de coïncidence, la sortie prédite du système est supposée coïncider avec la trajectoire de référence. Il n'existe aucune indication précise pour fixer ces points particuliers. Pour cela, on introduit la notion d'horizon de coïncidence (H_1 , H_2) il ne s'agit pas en effet de rechercher une coïncidence à tous les instants futurs, mais seulement sur un certain nombre de point entre H_1 et H_2 . (Figure I.2)

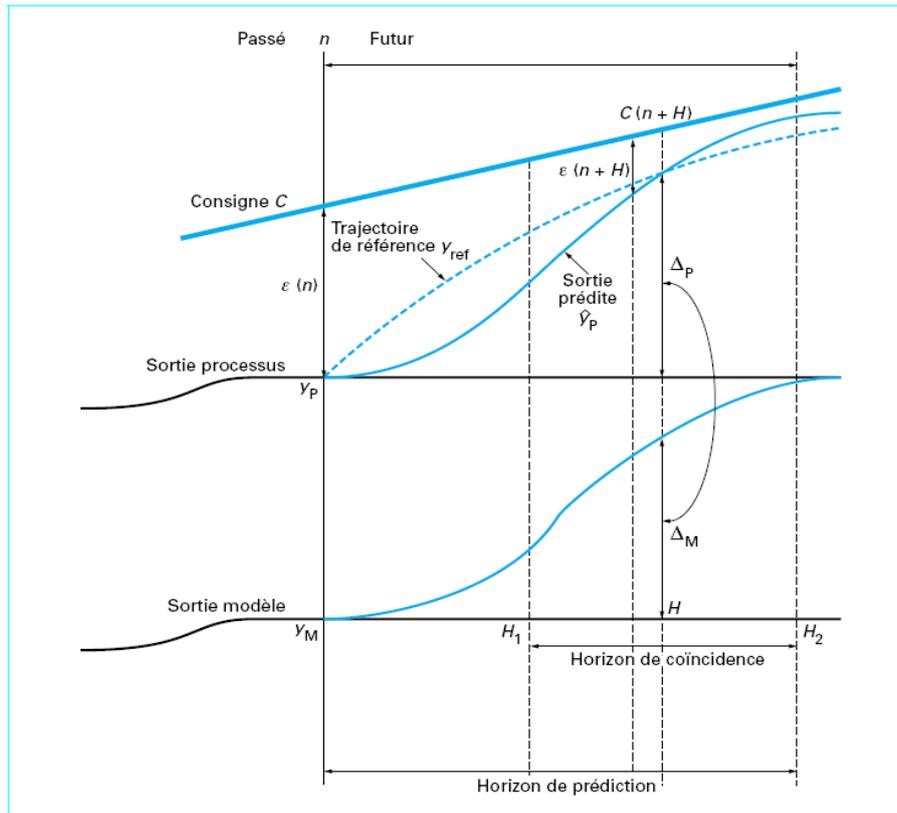


Figure I.2: Trajectoire de référence et horizon de prédiction

La dynamique de la trajectoire de référence peut éventuellement être variable dans le temps ou suivant l'état du processus. Elle est le facteur qui va régler la dynamique en boucle fermée avec une grande simplicité, directement interprétable par n'importe quel régleur.

La méthode de commande consiste alors à transférer l'incrément de sortie désiré du processus, Δ_p spécifiée par la trajectoire de référence, vers un modèle mathématique qui va permettre de calculer une commande qui réalise le même incrément de sortie modèle

$$\Delta_M = \Delta_P . \quad (\text{I.32})$$

Ce transfert de spécification est un élément clé de la méthode.

La trajectoire de référence est prise exponentielle de décrement λ , elle relie le point courant de la sortie à la consigne $C(k)$. A un point de coïncidence $H \in [H_1, H_2]$ on a :

$$y_r(k+H) = y_p(k+H) \quad (\text{I.33})$$

Prenant par exemple le cas d'une consigne constante C à d $C(k)=C_0$, et remplaçant dans l'équation (I.34) on abouti a la formulation:

$$\Delta_P = (C_0 - y_p(k))(1 - \lambda^H) = \Delta_M = y_M(k+H) - y_M(k) \quad (\text{I.34})$$

1.7.3. Structuration de la variable manipulée :

La commande futures est définie à partir d'une combinaison linéaire d'un ensemble de n_B fonctions UB_k , dénommées fonctions de bases:

$$u(n+i) = \sum_1^{n_B} u_k(n) UB_k(i)$$

La détermination des coefficients inconnues $\mu_{n,k}$ s'effectue en minimisant un critère quadratique lié à l'écart entre la sortie prédite \hat{y} du processus et la trajectoire de référence en un ensemble n_B de points ($n_h \geq n_B$) de coïncidences:

$$J_k = \sum_{j=1}^{n_h} [y^r(k+h_j) - \hat{y}(k+h_j)]^2 \quad (\text{I.35})$$

Chaque entrée de base UB_k induit une sortie de base SB_k connue à priori pour un modèle donné, un exemple est donnée par la figure I.3.

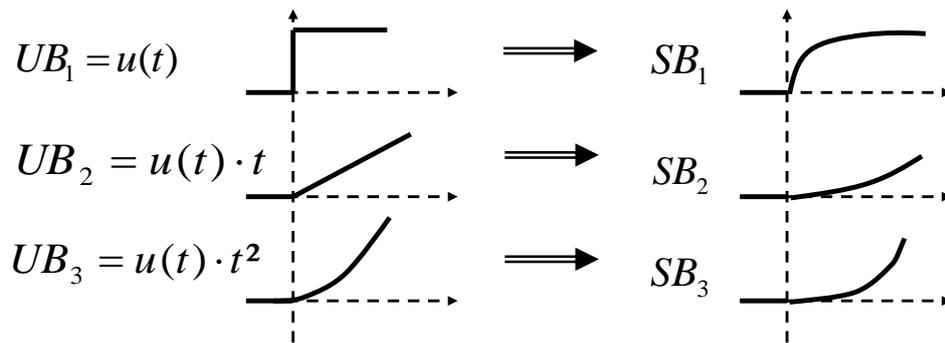


Figure I.3: Entrées et sorties de base

1.7.4 Principe de l'horizon fuyant.

Seul le 1er élément de la séquence optimale précédente est appliqué sur le système. Tous les autres éléments peuvent être omis car à la période d'échantillonnage suivante, les séquences sont décalées, une nouvelle sortie est mesurée et la procédure complète est répétée. Ce procédé repose sur le principe de l'horizon fuyant.

$$u_k = \sum_{j=1}^{n_B} \mu_{j,k} UB_j(0)$$

- le 1er élément de la séquence « optimale » précédente est appliqué sur le système
- les séquences sont décalées
- une nouvelle sortie est mesurée
- la procédure complète est répétée.

I.8 Conclusion :

Ce premier chapitre a présenté les principes généraux de la commande prédictive, détaillant plus spécifiquement les formulations GPC, MPC et PFC, dans le cas à temps discret invariant, faisant intervenir des fonctions de coût quadratiques et des contraintes. Ensuite nous avons donné un aperçu sur la commande prédictive non linéaire et ses conditions de stabilité.

Chapitre II

La Commande Prédicative Généralisée Floue hiérarchisée

Chapitre II

La Commande Prédictive Généralisée Floue hiérarchisée

II.1 Introduction:

L'objectif du travail présenté dans ce chapitre est d'étendre les méthodes de commande prédictive linéaire aux systèmes non linéaires en utilisant les systèmes d'inférences floues de type Takagi Sugeno, TS [25] [32]. Deux approches peuvent être adoptées. Dans la première approche, le système d'inférence flou de T-S est utilisé pour fusionner les modèles linéaires locaux. Le but est alors de produire un modèle de prédiction linéaire global variant dans le temps [30]. Ce dernier sera exploité dans le calcul de la loi de commande linéaire globale [16] [20]. La seconde approche utilise le modèle linéaire local basé sur les contrôleurs prédictifs qui peuvent être calculé pour chaque règle. Ces contrôleurs locaux sont fusionnés afin d'élaborer le contrôleur global. Dans ce cas, la méthode utilisée pour fusionner les contrôleurs locaux est généralement basée sur les poids associé à chaque modèle local [25] [33], ce qui ne garanti pas l'optimalité de la loi de commande globale. Pour pallier ce problème, il a été suggéré dans [20] l'élaboration d'un algorithme de commande hiérarchique garantissant un quasi optimalité de la loi de commande prédictive obtenue. La méthodologie a été appliquée à la commande prédictive basée sur un modèle floue à réponse impulsionnelle où chaque règle du modèle flou a été générée en fonction de la réponse du système à une impulsion. Cependant, cela conduit généralement à des modèles d'ordre élevé. Dans ce travail, nous proposons l'approche hiérarchique pour la conception d'une loi de commande prédictive généralisée GPC [7].

Le modèle élaboré se base sur la modélisation floue de Takagi Sugeno. Rappelons que la GPC est une technique adaptée aux modèles d'entrée-sortie, qui sont plus compacts et d'une moindre dimension que ceux d'impulsion [20].

II.2. Commande Prédicative Généralisée Floue (FGPC: Fuzzy Generalized Predictive Control) :

Dans le cadre de notre étude, nous considérons un système non linéaire monovarié, (SISO Single Input Single Output). Pour obtenir la prédiction de ce système dans le cadre de la GPC, ce système sera représenté par un modèle flou de type TS [32]. La conséquence de chaque règle sera exprimée par un prédicteur GPC tel que décrit dans le chapitre précédent, soit donc :

$$\begin{aligned} \mathbf{R}^i: & \text{IF } y(t) \text{ IS } A_{i0}^i, y(t-1) \text{ IS } A_{i1}^i, \dots, y(t-m+1) \text{ IS } A_{i,m-1}^i, \text{ AND } u(t) \text{ IS } B_{i0}^i, u(t-1) \text{ IS } B_{i1}^i, \dots, u(t-l+1) \\ & \text{IS } B_{i,l-1}^i \text{ THEN } y^i(t+j/t) = \sum_{i=1}^j g_i \Delta u(t+j-i) + f_j \end{aligned} \quad (\text{II.1})$$

A_j^i : Ensemble flou correspond à la sortie $y(t-j)$ dans la $i^{\text{ème}}$ règle floue;

B_j^i : Ensemble flou correspond à l'entrée $u(t-j)$ dans la $i^{\text{ème}}$ règle floue;

La sortie globale est obtenue par fusion des règles donnée par :

$$y(t+1) = \frac{\sum_{j=1}^P w^j y^i(t+1)}{\sum_{j=1}^P w^j} \quad (\text{II.2})$$

Où w^i représente le poids associé à la règle R^i de manière à :

$$w^j = \bigwedge_i A_i^j \bigwedge_k B_k^j \quad (\text{II.3})$$

Avec \bigwedge est l'opérateur minimum

Rappelons que la fonction coût de la GPC donnée par :

$$J(t) = \sum_{i=1}^{Ny} \mu_i (\hat{y}(t+i) - y^r(t+i))^2 + \sum_{i=1}^{Nu} \nu_i \Delta u(t+i)^2 \quad (\text{II.4})$$

Or, μ_i et ν_i sont les facteurs de pondérations sur l'erreur de prédiction et l'incrément de commande, respectivement. Aussi la fonction coût dépend de la décomposition en p sous

systèmes. Dans l'approche analytique de commande prédictive floue à base de modèle, l'incrément de commande optimale globale, $\Delta u(t+i)$, $i=1,2,\dots,N_u$ est calculé par pondération des incréments de commande locaux $\Delta u^j(t+i)$, $i=1,2,\dots,N_u$, $j=1,\dots,p$, par la règle du centre de gravité :

$$\Delta u(t+i) = \sum_{j=1}^p w^j \Delta u^j(t+i) \quad (\text{II.5})$$

Deux principales méthodes de résolutions sont proposées. La première calcule les commandes locales par résolution de manière indépendante les contrôleurs GPC locaux, la seconde calcule la commande global à partir de l'équation (II.5)

Dans la deuxième méthode de résolution, nous substituons l'équation (II.5) dans (II.4) afin d'obtenir une formulation du coût global exprimé par:

$$J(t) = \sum_{i=1}^{N_y} \mu_i \left(\sum_{j=1}^p w^j (\hat{y}^j(t+i) - y^r(t+i)) \right)^2 + \sum_{i=1}^{N_u} v_i \left(\sum_{j=1}^p w^j \Delta u^j(t+i) \right)^2 \quad (\text{II.6})$$

Étant donné que la minimisation de cette fonction coût exige un calcul important, il a été suggéré dans [57] de transformer l'expression (II.6) par une approximation basée sur l'inégalité de Cauchy. La fonction coût équivalente sera évaluée par l'expression :

$$\tilde{J}(t) = \sum_{j=1}^p \left((w^j)^2 \left(\sum_{i=1}^{N_y} \mu_i (\tilde{y}^j(t+i) - y^r(t+i))^2 + \sum_{i=1}^{N_u} v_i (\Delta u^j(t+i))^2 \right) \right) \quad (\text{II.7})$$

Le problème d'optimisation moyennant cette approximation est reformulé de telle sorte que :

$$\begin{aligned} & \min_{\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+N_u)} \tilde{J}(t) \\ & = \min_{\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+N_u)} \sum_{j=1}^p (w^j)^2 \tilde{J}^j(t) \end{aligned}$$

$$\text{Où } \tilde{J}^j(t) = \sum_{i=1}^{N_y} \mu_i (\hat{y}^j(t+i) - y^r(t+i))^2 + \sum_{i=1}^{N_u} v_i (\Delta u^j(t+i))^2 \quad (\text{II.8})$$

En se basant sur la décomposition de la fonction coût (II.8), la solution du problème d'optimisation se résume par l'utilisation du concept de la décomposition et de la coordination de [20] que nous développons dans le paragraphe suivant.

II.3 La commande hiérarchisée

Il est important de rappeler que le système global se décompose en P sous systèmes dans le but de construire des contrôleurs prédictifs locaux qui seront coordonnés pour former le contrôleur optimal global. Deux couches sont proposées pour l'élaboration du système hiérarchisé: couche inférieure et la couche supérieure.

II.3.1 Couche inférieure

La couche inférieure englobe les P sous systèmes où pour le $j^{\text{ème}}$ sous système le problème d'optimisation sera redéfini comme suit:

$$\min_{\Delta u^j(t), \Delta u^j(t+1), \dots, \Delta u^j(t+Nu)} \tilde{J}^j(t) \quad (\text{II.9})$$

La méthode est basé sur le terme d'erreur $\varepsilon^j(t+k-1) = e^j(t+k-1) - y^j(t+k-1)$ qui permet d'assurer la coordination entre les deux couches. Après avoir introduit ce terme dans l'équation (II.1) qui désigne le terme de coordination pour chaque sous système local, nous aboutissons à une nouvelle expression :

R^j : IF $y(t+k-1)$ IS A^j_0 , $y(t+k-2)$ IS $A^j_1, \dots, y(t+k-m)$ IS A^j_{m-1} , AND $u(t+k-1)$ IS B^j_0 ,
 $u(t+k-2)$ IS $B^j_1, \dots, u(t+k-l)$ IS B^j_{l-1} THEN

$$y^j(t+k) = \sum_{i=1}^k g_i \Delta u(t+j-i) + fk + \varepsilon^j(t+k-1) \quad (\text{II.10})$$

Notons que ce terme d'erreur sera déterminé au niveau de la couche supérieure. Les résultats des calculs obtenus sur cette couche permettront la détermination de l'ensemble des informations qui sera transmis à la couche supérieure. Cet ensemble est défini par:

$$S_{L \rightarrow U}^j = \left\{ \begin{array}{l} \hat{y}^j(t+i), \Delta u^j(t+l) / i = 1, 2, \dots, Ny; \\ l = 1, 2, \dots, Nu \end{array} \right\} \quad (\text{II.11})$$

II.3.2 Couche supérieure

La couche supérieure de coordination a pour objectif la détermination des séquences de commande optimales grâce au terme d'erreur $\varepsilon^j(t+k)$ introduit dans le modèle. Ainsi, la fonction coût associée à cette couche sera reformulée de manière à résoudre:

$$\min_{\varepsilon(t), \varepsilon(t+1), \dots, \varepsilon(t+Nu)} \tilde{J}(t) \quad (\text{II.12})$$

Où la minimisation de cette fonction coût est accomplie par l'identification de la variable erreur $\varepsilon(t+k-1)$ qui définit l'ensemble d'information donné par :

$$S_{U \rightarrow L}^j = \{ \varepsilon^j(t+k-1) / k = 1, 2, \dots, Ny \} \quad (\text{II.13})$$

La communication entre les deux couches est réalisée comme suit :

-La couche inférieure rassemble les informations locales de la sortie et de la commande dans l'ensemble $S_{L \rightarrow U}^j$ de chaque p sous système puis le transmet à la couche supérieure.

-La couche supérieure calcule les variables d'erreurs dont l'expression $\varepsilon^j(t+k-1) = e^j(t+k-1) - y^j(t+k-1)$. Ces valeurs seront comparées avec celles obtenues par le calcul itératif de la séquence précédente $\varepsilon^j(t+k-1)$ puis évaluées par rapport au paramètre de tolérance ζ . Les itérations entre les couches inférieure et supérieure sont répétées tant que

$$\sum_{j=1}^p \sum_{k=1}^{Ny} |e^j(t+k-1) - \varepsilon^j(t+k-1)| > \zeta \quad (\text{II.14})$$

Lorsque la somme des erreurs ci-dessus est inférieure à la tolérance, la commande obtenue est appliquée au système.

La commande prédictive hiérarchique est résumée par la représentation donnée par la Figure II.1.

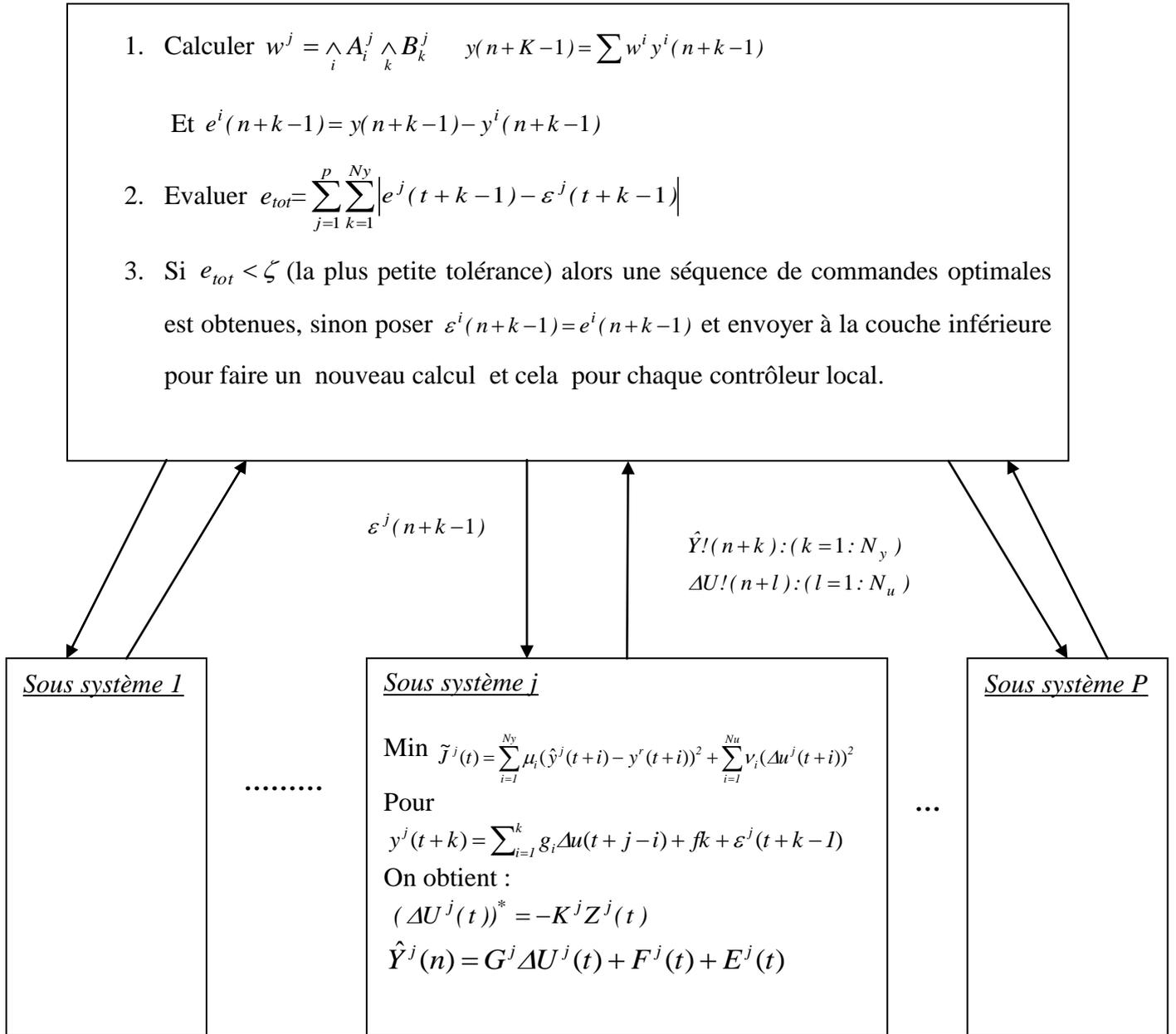


Figure II.1: Structure hiérarchisée de la conception du contrôleur prédictif flou

II.3.3 Calcul des contrôleurs locaux:

Pour le calcul des contrôleurs prédictifs généralisés locaux, nous considérons le modèle de prédiction du j^{ème} sous système flou. La sortie prédite à N_y pas du j^{ème} sous système sera exprimé par:

$$\hat{Y}^j(n) = G^j \Delta U^j(t) + F^j(t) + E^j(t) \tag{II.15}$$

$$\text{Où } G^j = \begin{bmatrix} g_0^j & 0 & \dots & 0 & 0 \\ g_1^j & g_0^j & 0 & \dots & 0 \\ g_2^j & g_1^j & g_0^j & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{N_y-1}^j & \dots & g_2^j & g_1^j & g_0^j \end{bmatrix}$$

$$F^j(t) = [f(t+1) \quad \dots \quad f(t+N_y)]$$

$$E^j(n) = \left[0 \quad \sum_{k=1}^2 \varepsilon^j(t+k) \quad \dots \quad \sum_{k=1}^{N_y} \varepsilon^j(t+k) \right]^T$$

Il faut remarquer que ce dernier terme définit les termes d'erreur de coordination. Ainsi

l'équation (14) sera reformulé sous forme matricielle telle que:

$$\tilde{J}^j(t) = (\hat{Y}^j(t) - Y^r(t))^T W_1^j (\hat{Y}^j(t) - Y^r(t)) + (\Delta U^j(t))^T W_2^j (\Delta U^j(t)) \quad (\text{II.16})$$

$$\hat{Y}^j(t) = (\hat{y}^j(t+1) \hat{y}^j(t+2) \dots \hat{y}^j(t+N_y))^T \quad (\text{II.17})$$

$$\hat{Y}^r(t) = (y^r(t+1) y^r(t+2) \dots y^r(t+N_y))^T \quad (\text{II.18})$$

$$\Delta U^j(t) = (\Delta u^j(t+1) \Delta u^j(t+2) \dots \Delta u^j(t+N_u))^T \quad (\text{II.19})$$

$$W_1^j = \text{diag} \{ \mu_1^j, \mu_2^j, \dots, \mu_{N_y}^j \} \quad (\text{II.20})$$

$$W_2^j = \text{diag} \{ \nu_1^j, \nu_2^j, \dots, \nu_{N_u}^j \} \quad (\text{II.21})$$

Ce qui donne une nouvelle expression du coût :

$$\begin{aligned} \tilde{J}^j(t) = & (\Delta U^j(t))^T (G^{jT} W_1^j G^j + W_2^j) \Delta U^j(t) + (\Delta U^j(t))^T G^{jT} W_1^j Z^j(t) \\ & + (Z^j(t))^T W_1^j G^j \Delta U^j(t) + (Z^j(t))^T W_1^j Z^j(t) \end{aligned} \quad (\text{II.22})$$

Où

$$Z^j(t) = F^j(t) - Y^r(t) + E^j(t)$$

Dérivant l'équation (II. 23) on obtient :

$$\frac{\partial \tilde{J}^j(t)}{\partial \Delta U^j(t)} = 2(G^{jT} W_1^j G^j + W_2^j) \Delta U^j(t) + 2G^{jT} W_1^j Z^j(t) = 0 \quad (\text{II.23})$$

Ainsi, la loi de commande du $j^{\text{ème}}$ sous système flou sera défini par :

$$(\Delta U^j(t))^* = -K^j Z^j(t) \quad (\text{II.24})$$

Où K^j est la matrice Gain du $j^{\text{ème}}$ sous système, formulé par l'expression

$$K^j = (G^{jT} W_1^j G^j + W_2^j)^{-1} G^{jT} W_1^j \quad (\text{II.25})$$

Au niveau de la couche supérieure la loi de commande optimale globale est :

$$\Delta U(t) = (\Delta u(t+1) \quad \Delta u(t+2) \quad \dots \quad \Delta u(t+Nu))^T$$

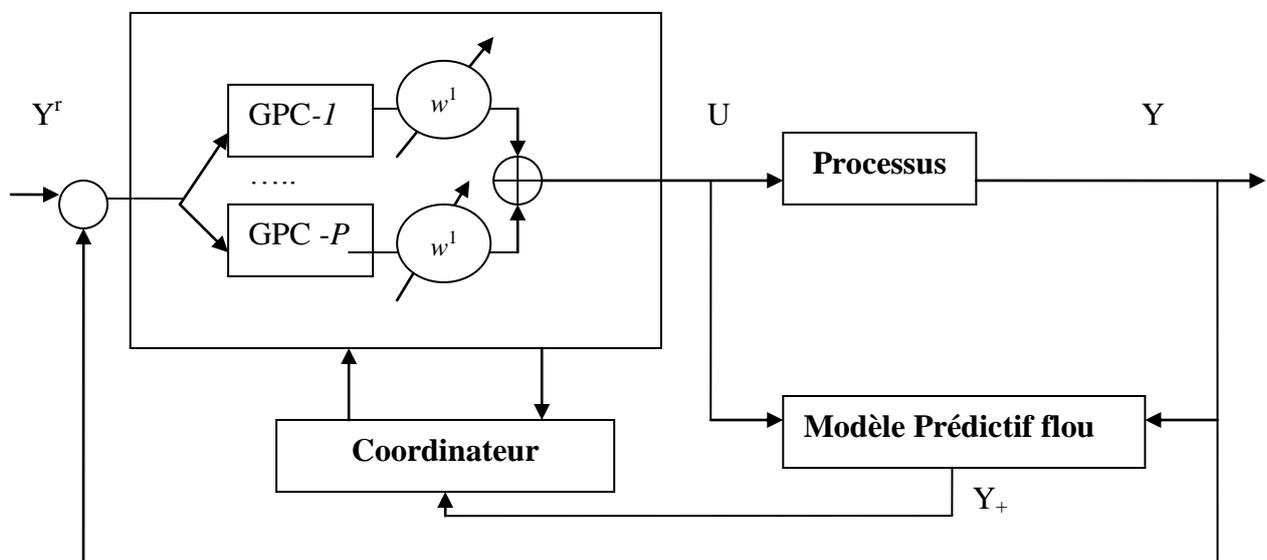


Figure II. 2 Structure du contrôleur FGPC

II.4. Résultats de la simulation:

II.4.1 Contrôle de concentration dans un CSTR

Dans le cadre de cette étude nous considérons un exemple de contrôle de la concentration dans un réacteur chimique CSTR (Continuous Stirred Tank Reactor) [7]. Dans ce processus, la réaction exothermique $A \rightarrow B$ a lieu dans un réacteur à volume constant, refroidi par écoulement du liquide de refroidissement. Le processus est modélisé par les équations mathématiques suivantes:

$$\frac{dC_A(t+d)}{dt} = \frac{q(t)}{V} (C_{A0}(t) - C_A(t+d) - k_0 C_A(t+d) \times \exp\left(\frac{-E}{RT(t)}\right)) \quad (\text{II.26})$$

$$\begin{aligned} \frac{dT(t)}{dt} = & \frac{q(t)}{V} (T_0(t) - T(t)) - \frac{(-\Delta H) k_0 C_A(t+d)}{\rho C_p} \times \exp\left(\frac{-E}{RT(t)}\right) + \\ & \frac{\rho_c C_{pc}}{\rho C_p V} q_c(t) \left\{ 1 - \exp\left(\frac{-hA}{q_c(t) \rho C_p}\right) \right\} \times (T_{c0}(t) - T(t)). \end{aligned} \quad (\text{II.27})$$

L'objectif est de commander la concentration notée $C_A(t)$, en manipulant le taux d'écoulement du liquide de refroidissement $q_c(t)$ dans le réacteur.

Les paramètres nominaux du modèle sont donnés par le tableau II.1

<i>Variable</i>	<i>Valeur nominale</i>
Concentration en alimentation (C_A)	0.1 mol/l
Température du Réacteur(T)	438.54K
Débit du liquide de refroidissement (q_c)	103.41l/min
Taux du débit (q)	100.0 l/min
Concentration Nominale(C_{A0})	1 mol/l
Temperature Nominale (T0)	350.0K
volume du réacteur (V)	100l
Terme de transfert (hA)	7.2×10^5 cal/(min.K)
Constante du taux de réaction (k0)	7.2×10^{10} min ⁻¹
Terme d'activation d'énergie (E/R)	1×10^4 K
La chaleur de la réaction ($-\Delta H$)	-2×10^5 cal/mol
Densité du liquide ((ρ, ρ_c))	1×10^3 g/l
Capacité de chaleur (Cp,Cpc)	1 cal/(g.K)

Tableau II.1: paramètres nominaux du CSTR

II.4.2 Modélisation floue

Pour la modélisation floue, le modèle non linéaire ci-dessus est utilisé pour constituer des données d'entrées-sorties. Avec un temps d'échantillonnage qui peut être fixé à 5 secondes.

On considère deux règles floues selon les formules :

$$R^1: \text{IF } q_c \text{ is } Q^1$$

$$\text{THEN } C_A^1(n) = a_{11} C_A(n-1) + \dots + a_{1na} C_A(n-na) + b_{11} q_c(n-1) + \dots + b_{1nb} q_c(n-nb) \quad (\text{II.28})$$

R^1 : IF q_c is Q^2

$$\text{THEN } C^2_{A(n)} = a_{21}C_A(n-1) + \dots + a_{2na}C_A(n-na) + b_{21}q_c(n-1) + \dots + b_{2nb}q_c(n-nb) \quad (\text{II.29})$$

Le modèle flou est structurellement très simple et ne nécessite que deux règles floues données par R^1 et R^2 . Les ensembles flous Q_1 et Q_2 sont présentés par la figure II.3.

La figure II.4 montre la réponse en boucle ouverte avec des changements pour différentes valeurs du débit du liquide de refroidissement. Le modèle non linéaire est représenté par un trait continu, alors que le modèle flou par un trait discontinus.

Le vecteur des paramètres de la $i^{\text{ème}}$ règle floue est obtenu en utilisant la méthode des moindres carrés récursifs.

$$\begin{array}{cccc} a_{11} = -1.7016 & b_{11} = -0.2006 & a_{21} = -1.8480 & b_{21} = -0.1397 \\ a_{12} = 0.7715 & b_{12} = 0.2375 & a_{22} = 0.8984 & b_{22} = 0.1973 \end{array}$$

II.4.3 Commande floue hiérarchisée:

Pour les deux contrôleurs locaux, l'horizon de prédiction et de l'horizon de contrôle sont fixées à 10 et 2 respectivement. Les facteurs de pondération sont W^j_1 et W^j_2 tous égaux à 20000 et à 6 sur la diagonale, respectivement. Nous avons étudié la qualité du contrôleur FGPC par une série de simulations. La réponse dynamique du système est représentée dans la même figure II.5 où la consigne est représentée en pointillés. La courbe de la concentration suit la consigne sans dépassement et que le signal de commande est lisse et sans actions brusques.

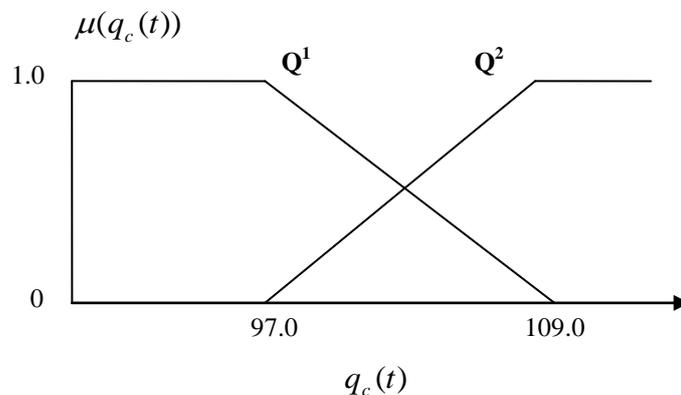


Figure II.3: Les ensembles flous Q^1 et Q^2 pour les règles R^1 et R^2 , respectivement.

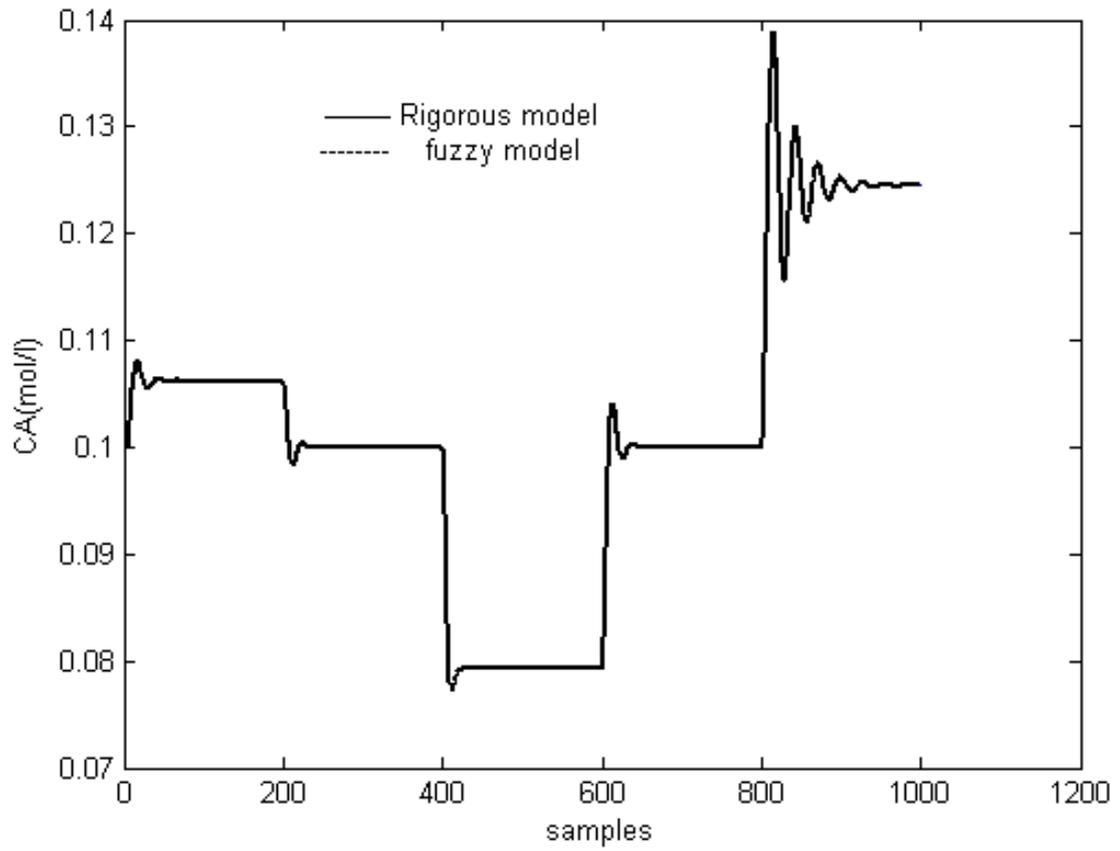


Figure II.4: Réponse en boucle ouverte pour la modélisation floue

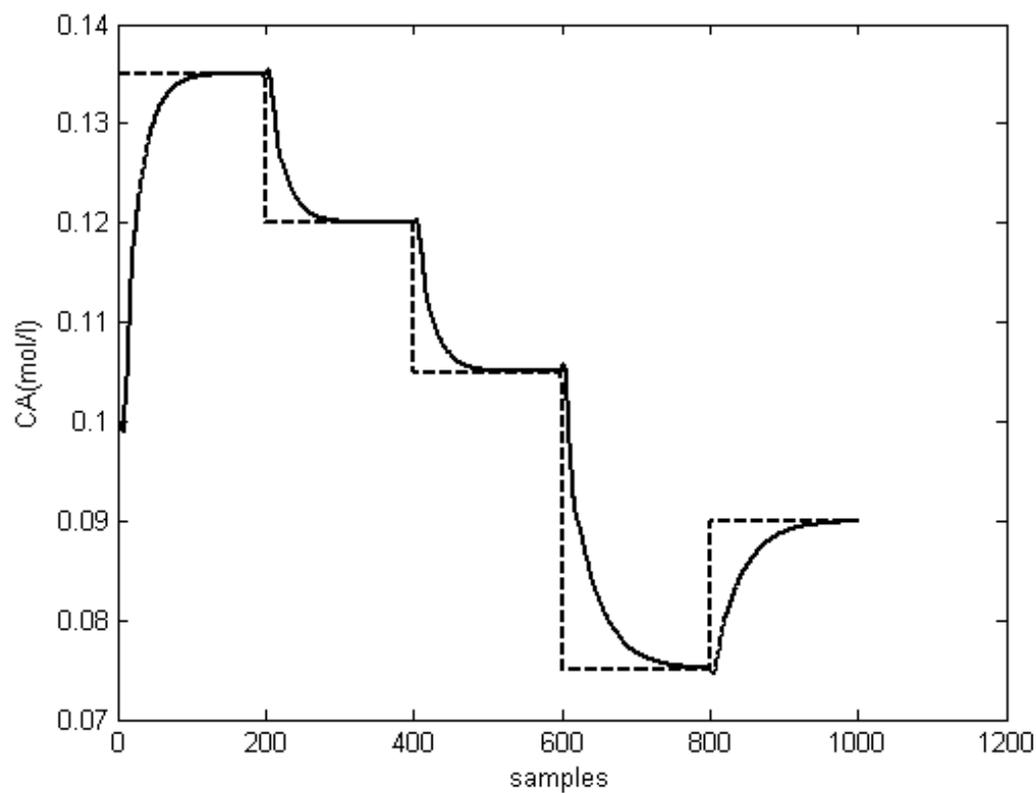


Figure II.5: Réponse de la concentration C_A à des changements de consigne

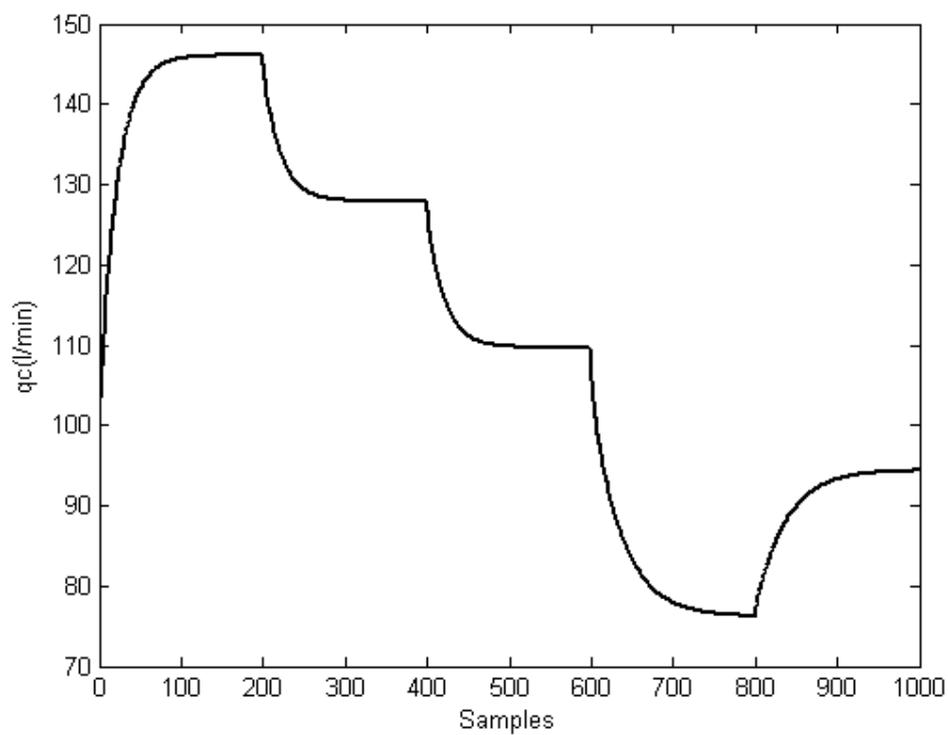


Figure II.6: Le signal de commande

II.4.2. Exemple 2 Système mécanique masse- ressort -Amortisseur:

Dans ce deuxième exemple [7], nous considérons un système mécanique non linéaire masse-ressort -Amortisseur (mass-spring-damper mechanical system), donné par la figure II.7

Le processus est modélisé par les équations suivantes:

$$M\ddot{x}(t) + g(x(t), \dot{x}(t)) + f(x(t)) = \phi(\dot{x}(t))u(t) \quad (\text{II. 30})$$

Avec

$$g(x(t), \dot{x}(t)) = c_1 x(t) + c_2 \dot{x}(t) \quad (\text{II.31})$$

$$f(x(t)) = c_3 x(t) + c_4 x(t)^3 \quad (\text{II.32})$$

$$\phi(\dot{x}(t)) = 1 + c_5 \dot{x}(t)^2 \quad (\text{II.33})$$

Où M , $x(t)$, u sont la masse, son déplacement et la force, respectivement.

$f(x(t))$, $g(x(t), \dot{x}(t))$, $\phi(\dot{x}(t))$ la non linéarité dû à élasticité du ressort, l'amortisseur et l'entrée. Les paramètres sont donnés par

$$M = 1.0, c_1 = 0, c_2 = 1, c_3 = 0.01, c_4 = 0.1, \text{ and } c_5 = 0.13.$$

Le modèle est alors donné par:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_2 - 0.01x_1 - 0.1x_1^3 + (1 + 0.13x_2^2)u \end{aligned} \quad (\text{II.34})$$

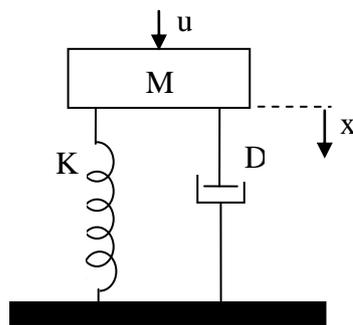


Figure II.7 Le système Masse ressort amortisseur

II.4.2.1 Modélisation floue

Le modèle non linéaire ci-dessus est utilisé pour produire les séquences temporelles des données d'entrées-sorties. La période d'échantillonnage est fixée à 0.01seconde, ainsi que la plage de variation de $x(t)$ est de $[-1.5, 1.5]$.

Le modèle flou contient trois règles exprimées par:

$$R^1 : IF \ x1(n) \ IS \ A^1$$

$$THEN \ x1^1(n) = a_{11}x1(n-1) + a_{12}x1(n-2) + b_{11}u(n) + b_{12}u(n-1)$$

$$R^2 : IF \ x1(n) \ IS \ A^2$$

$$THEN \ x1^2(n) = a_{21}x1(n-1) + a_{22}x1(n-2) + b_{21}u(n) + b_{22}u(n-1)$$

$$R^3 : IF \ x1(n) \ IS \ A^3$$

$$THEN \ x1^3(n) = a_{31}x1(n-1) + a_{32}x1(n-2) + b_{31}u(n) + b_{32}u(n-1)$$

Les ensembles flous A^i , $i=1, 2,3$ sont définis par la figure II.8

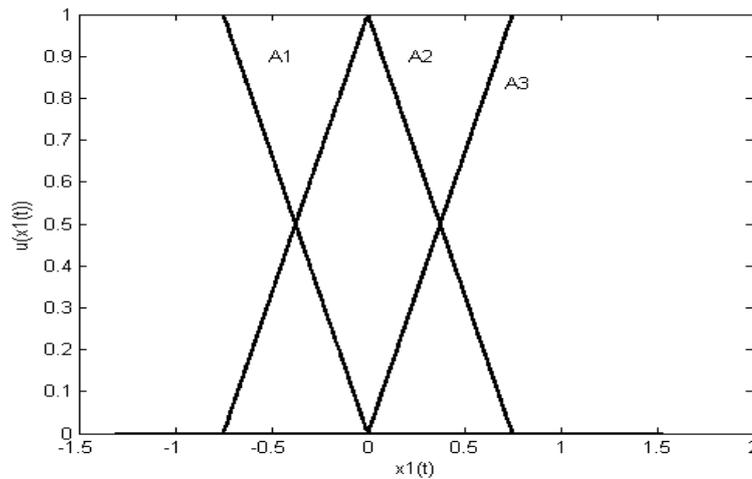


Figure II.8: Les ensembles flous A^1 à A^3 pour la règle R^1 à R^3 , respectivement

L'état initiale est donnée par $[x_1(0), x_2(0)]^T = [1.5, 0]^T$.

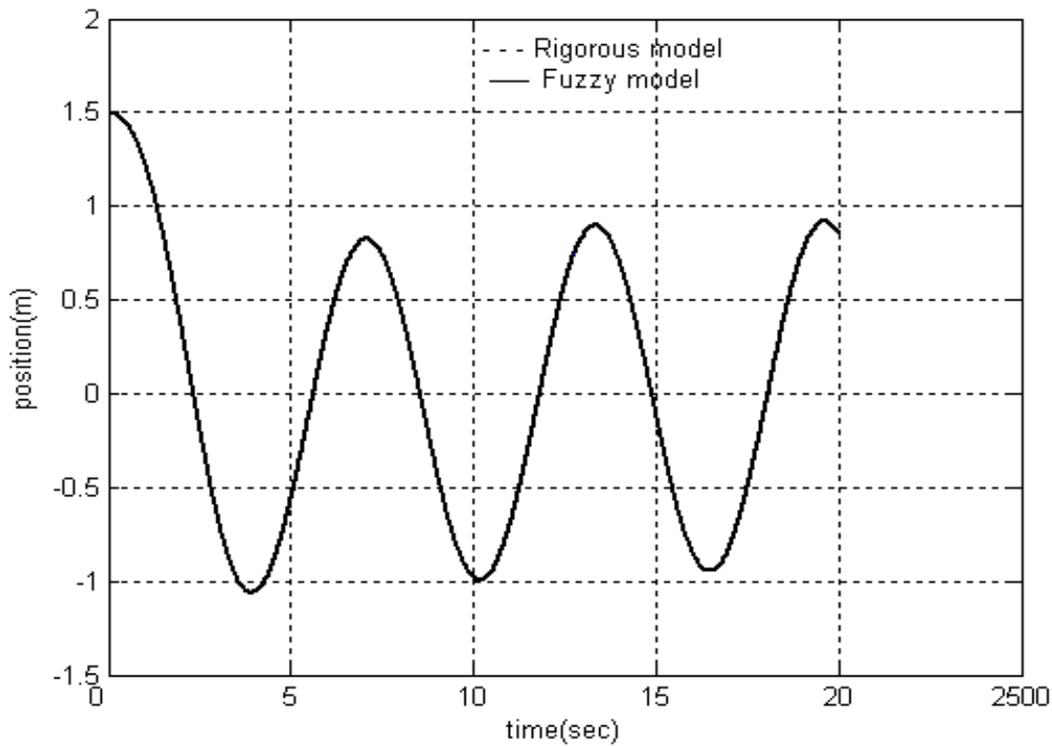


Figure II.9: La réponse en boucle ouverte

Dans la figure II.9 le modèle non linéaire est représenté par un trait continu, alors que le modèle flou par un trait discontinu.

Le vecteur des paramètres de la i^{th} règle obtenu moyennant la méthode des moindres carrées récurrents, est donné par :

$$\begin{array}{lll}
 a_{11} = -1.7956 & a_{21} = -1.9422 & a_{31} = -1.9792 \\
 a_{21} = 0.7952 & a_{22} = 0.9422 & a_{32} = 0.9794 \\
 b_{11} = -0.0817 & b_{21} = -0.0283 & b_{31} = -0.0119 \\
 b_{12} = 0.0831 & b_{22} = 0.0287 & b_{32} = 0.0120
 \end{array}$$

II.4.2.2 La Commande floue hiérarchisée

L'objectif alors de stabiliser le déplacement de la masse à l'origine $x_1(t) = 0$. L'horizon de prédiction et l'horizon de contrôle sont identiques à ceux du premier exemple. Les poids W_{j1} et W_{j2} sont choisis égaux à 0.5 et 60 sur la diagonale, respectivement.

Les résultats de simulation sont présentés sur les figures (II. 10) et (II.11). D'après la figure(II.10) la masse atteint l'équilibre à l'instant $t = 5$ secondes sans dépassement. Le signal de commande indiqué par la figure (II.11) est sans actions brusques.

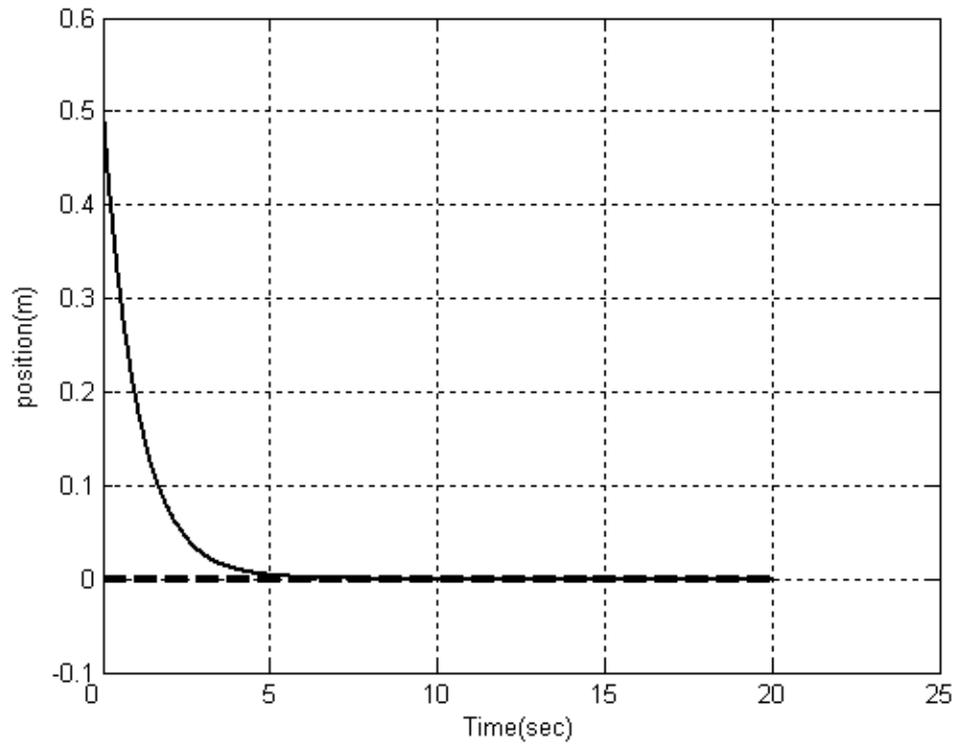


Figure II.10: Position de la masse sous commande

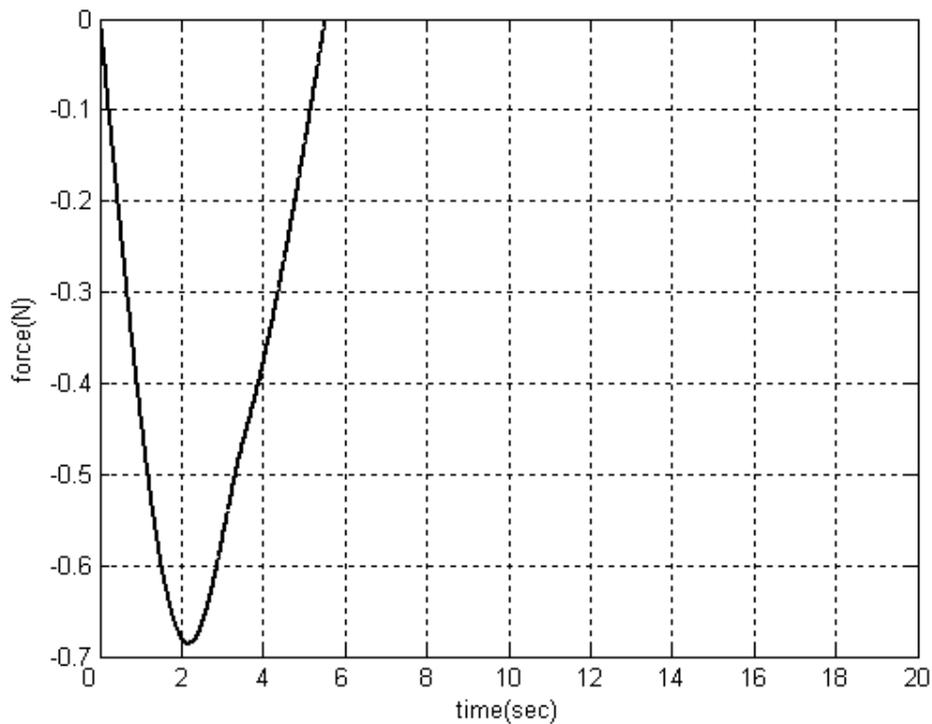


Figure II.11 le signal de commande

II.5 Conclusion:

Ce deuxième chapitre a été consacré au développement d'une technique de commande prédictive basée sur les systèmes flous de Takagi Sugeno et sur une méthode hiérarchisée de décomposition coordination qui s'applique aux systèmes non linéaires. Le système non linéaire est d'abord représenté sous forme de modèle TS composé de sous modèles, ensuite un contrôleur prédictif est obtenu pour chaque modèle. La commande quasi optimale globale est calculée itérativement en utilisant la méthode hiérarchisée. Cette approche a été adoptée dans ce chapitre à la conception de la loi de commande prédictive généralisée GPC. Un ensemble d'exemples de simulation mettent en évidence la validité de cette approche.

Chapitre III

La Commande Prédicative Fonctionnelle

Floue hiérarchisée

Chapitre III : La Commande Prédicative Fonctionnelle Floue hiérarchisée

III. 1. Introduction:

Dans ce chapitre, nous proposons une approche hiérarchisée destinée à la conception d'une loi prédictive fonctionnelle pour la commande des systèmes non linéaires basés sur le modèle flou de Takagi-Sugeno [32].

III. 2 La PFC Floue hiérarchisée:

III.2.1 Le modèle flou:

Considérons un système non linéaire de type SISO (Single Input Single Output). Le modèle Takagi Sugeno [32] du système être décrit par la forme mathématique suivante:

R^i : **IF** $u(k)$ **IS** A^i **THEN**

$$y^i(k+1) = a_1^i(k)y(k) + a_2^i(k)y(k-1) + a_3^i(k)y(k-2) + b^i(k)u(k-D) + r^i, \quad (\text{III.1})$$

Notons que $y^i(k+1)$ est la sortie du $i^{\text{ème}}$ sous-modèle local linéaire correspondant à la $i^{\text{ème}}$ règle floue et $y(k)$, $y(k-1)$, $y(k-2)$, $u(k-D)$ sont les entrées du modèle flou, D représente le retard, A_i sont des ensembles flous des antécédents $u(k)$ est la variable de commande. La sortie du modèle global dynamique est obtenue en fusionnant tous les sous-modèles linéaires locaux moyennant la méthode de défuzzification qui donne lieu [8] [9] :

$$y_p(k+1) = \sum_{i=1}^P \beta_i(k) (a_{1i}y(k) + a_{2i}y(k-1) + a_{3i}y(k-2) + b_i u(k-D) + r_i), \quad (\text{III.2})$$

Avec P le nombre de règles floues et $\beta_i(k)$ le poids normalisé par la formulation de l'équation:

$$\beta_i(k) = \frac{\mu_{A_i}(y(k))}{\sum_{i=1}^P \mu_{A_i}(y(k))} \quad (\text{III.3})$$

$$\sum_i \beta_i(k) = 1 \quad (\text{III.4})$$

Dans ce qui suit, nous développons la conception du contrôle hiérarchisé FPFC.

Les P contrôleurs prédictifs fonctionnels locaux sont calculés au niveau de la couche inférieure et seront coordonnées dans la couche supérieure pour établir une stratégie de contrôle optimal globale. Ces contrôleurs locaux sont basés sur les sous-modèles linéaires sous forme d'équations d'état. De ce fait, il faut réécrire l'équation (III.1) dans l'espace d'état et redéfinir le problème d'optimisation qui devient :

$$X_m^i(k+1) = \tilde{A}_m^i X_m^i(k) + \tilde{B}_m^i u^i(k) + R_m^i + \varepsilon^i(k) \quad (\text{III.5})$$

Le vecteur d'état $x_m(k)$ devient :

$$x_m(k) = \begin{bmatrix} y_m^i(k) \\ y_m^i(k-1) \\ y_m^i(k-2) \end{bmatrix} \quad (\text{III.6})$$

Les matrices $\tilde{A}_m, \tilde{B}_m, \tilde{R}_m$ et \tilde{C}_m deviennent:

$$\tilde{A}_m^i = \begin{bmatrix} \tilde{a}_{11}^i & \tilde{a}_{12}^i & \tilde{a}_{13}^i \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (\text{III.7})$$

$$\tilde{B}_m^i = \begin{bmatrix} \tilde{b}^i \\ 0 \\ 0 \end{bmatrix} \quad (\text{III.8})$$

$$\tilde{C}_m^i = [1 \ 0 \ 0] \quad (\text{III.9})$$

$$(\text{III.10})$$

$\varepsilon^i(k)$ est un terme d'erreur qui sert pour la technique de coordination. Il est déterminé au niveau de la couche supérieure.

III.2.2 La commande fonctionnelle hiérarchisée

III.2.2.1 La couche inférieure:

Au niveau de la couche inférieure nous considérons tous les P sous modèles, pour chaque sous modèle un contrôleur est développé. Afin de déterminer la loi de commande prédictive fonctionnelle locale, nous considérons le modèle de prédiction de chaque sous-système. La prédiction a H pas peut être écrite sous la forme :

$$X_m^i(k+H) = \tilde{C}_m^i (\tilde{A}_m^i X_m^i(k) + (\tilde{A}_m^i{}^{H-1} + \dots + \tilde{A}_m^i + I) \times (\tilde{B}_m^i u(k) + \tilde{R}_m^i)) \quad (\text{III.11})$$

Avec $u(k) = u(k+1) = \dots = u(k+H-1)$

La somme de la matrice \tilde{A}_m en puissance peut être simplifiée par le biais de l'équation:

$$\tilde{A}_m^{iH-1} + \dots + \tilde{A}_m^i + I = (\tilde{A}_m^{iH} - I)(\tilde{A}_m^i - I)^{-1} \quad (\text{III.12})$$

Rappelons que la réponse en boucle fermé doit être similaire à la trajectoire de référence, dont la sortie du modèle de référence est donnée par [6]:

$$\begin{aligned} X_r(k+1) &= A_r X_r(k) + B_r w(k) \\ y_r(k) &= C_r X_r(k) \end{aligned} \quad (\text{III.13})$$

Avec les Matrices A_r , B_r et C_r doivent être choisies vérifiant la condition suivante:

$$C_r(I - A_r)^{-1} B_r = I \quad (\text{III.14})$$

De la même façon, la prédiction à H pas du modèle de référence doit réécrite sous la forme :

$$X_r(k+H) = C_r(A_r^H x_r(k) + (A_r^H - I)(A_r - I)^{-1} B_r w(k)) \quad (\text{III.15})$$

La méthode de commande consiste alors à transférer l'incrément de sortie désiré du processus Δ_p spécifié par la trajectoire de référence, vers un modèle mathématique qui va permettre de calculer une commande qui réalise, sur ce modèle, le même incrément de sortie modèle $\Delta_p = \Delta_m$. Ce transfert de spécification est un élément clé de la méthode PFC.

Notons que l'incrément objectif Δ_p est défini comme la différence entre la prédiction de trajectoire de référence $y_r(k+H)$ et la sortie actuelle du modèle telle que:

$$\Delta_p = y_r(k+H) - y_p^i(k) \quad (\text{III.16})$$

$$\Delta_p = C_r(A_r^H x_r(k) + (A_r^H - I)(A_r - I)^{-1} B_r w(k)) - y_p^i(k) \quad (\text{III.17})$$

Et l'incrément de la sortie du modèle Δ_m est défini par la même manière :

$$\begin{aligned} \Delta_m &= y_m^i(k+H) - y_m^i(k) \\ \Delta_m &= C_m(\tilde{A}_m^H x_m(k) + (\tilde{A}_m^H - I)(\tilde{A}_m - I)^{-1}(\tilde{B}_m^i u(k) + \tilde{R}_m^i)) - y_m^i(k) \end{aligned} \quad (\text{III.18})$$

A partir des équations ci-dessus l'objectif du FPFC est résumé par l'équation :

$$\Delta_m = \Delta_p \quad (\text{III.19})$$

A partir de cette égalité on peut dériver la formule de loi de commande locale $u^i(k)$

$$u^i(k) = \frac{C_r A_r^H x_r(k) + C_r (A_r^H - I)(A_r - I)^{-1} B_r w(k) - y_p^i(k)}{\tilde{C}_m^i (\tilde{A}_m^H - I)(\tilde{A}_m^i - I)^{-1} \tilde{B}_m^i} - \frac{\tilde{C}_m^i \tilde{A}_m^H x_m^i(k) - \tilde{C}_m^i (\tilde{A}_m^H - I)^{-1} \tilde{R}_m^i - y_m^i(k)}{\tilde{C}_m^i (\tilde{A}_m^H - I)(\tilde{A}_m^i - I)^{-1} \tilde{B}_m^i} \quad (\text{III.20})$$

Donc le contrôleur $u^i(k)$ est conçu par l'i-ème sous modèle linéaire. Enfin l'ensemble suivant contient les informations transmises à la couche supérieure.

$$S_{L \rightarrow U}^i = \{y^i(k+H), u^i(k)\} \quad (\text{III.21})$$

IV.2.2.2 La couche supérieure :

L'objectif de la couche supérieure est l'identification du contrôleur optimal global en ajoutant un terme de coordination $\varepsilon^i(k)$, ce dernier représente la différence entre $y(k+H)$ et $y^i(k+H)$ et cela pour chaque sous modèle. Ensuite les informations locales sont transmises de la couche inférieure vers la couche supérieure, par le biais des informations réunies dans l'ensemble $S_{L \rightarrow U}^j$.

Au niveau de la couche supérieure les variables d'erreurs sont évaluées par $e^j(k+H) - y^j(k+H)$

Ces valeurs sont estimés avec la même variable d'erreur calculée à l'itération précédente $\varepsilon^j(k+H)$ et comparée avec la plus petite tolérance ζ

Si la valeur $\sum_{j=1}^P |e^j(k+H) - \varepsilon^j(k+H)| > \zeta$ la commande n'est pas optimale et donc il

faut une modification au niveau la couche inférieure, ceci peut être accompli dans un nouveau processus itératif en envoyant l'ensemble $S_{U \rightarrow L}^j$ pour chaque sous-modèle.

Cet échange d'information entre la couche inférieure et supérieure est répété au cours de la période d'échantillonnage jusqu'à ce que: $\sum_{j=1}^P |e^j(k+H) - \varepsilon^j(k+H)| < \zeta$ est établie.

Dans ce cas, la commande optimale globale résultant des lois de commande optimales locales à la couche inférieure est considérée comme approche optimale. Le signal de commande est alors obtenu par la fusion des lois de commandes locales selon la méthode de défuzzification floue, on obtient:

$$U_{glob} = \sum_{i=1}^P \beta_i u^i(k) \quad (\text{III.22})$$

Avec β_i donnée par (4) et satisfaisant $\sum_i \beta_i(k) = 1$

III. 3. Exemple de simulation

Dans cet exemple, nous considérons le control de la concentration dans un réacteur chimique CSTR (Continuous Stirred Tank Reactor). Le modèle du procédé a été donné au chapitre précédent. L'objectif prescrit dans cet exemple est le contrôle de la concentration $C_A(t)$, en manipulant le taux de d'écoulement du liquide de refroidissement $q_c(t)$ dans le réacteur. Pour la modélisation floue, le modèle non linéaire ci-dessus est utilisé pour constituer des données d'entrées-sorties. On va considérer deux règles floues selon les formules :

$$R^1: \text{IF } q_c \text{ IS } Q^1 \text{ THEN } C_A^1(n) = a_{11}C_A(n-1) + a_{12}C_A(n-2) + a_{13}C_A(n-3) + b_1q_c(n) + r_1$$

$$R^2: \text{IF } q_c \text{ IS } Q^2 \text{ THEN } C_A^2(n) = a_{21}C_A(n-1) + a_{22}C_A(n-2) + a_{23}C_A(n-3) + b_2q_c(n) + r_2$$

De ce fait, le modèle flou est structurellement très simple et ne nécessite que deux règles floues données par R^1 et R^2 , les ensembles flous Q_1 et Q_2 soient présentés par la Figure III.1 Le modèle est réécrit sous forme d'équations d'état.

La Figure III.2 montre la réponse en boucle ouverte avec des changements pour différentes valeurs du débit du liquide de refroidissement

Les paramètres du modèle sont donnés par

$$\begin{array}{ll} a_{11} = -0.9090 & a_{21} = -0.9264 \\ a_{12} = -0.1444 & a_{22} = -0.1383 \\ a_{13} = 0.0615 & a_{23} = 0.0680 \\ b_1 = 0.0002 & b_2 = 0.0008 \\ r_1 = -0.0013 & r_2 = -0.0005 \end{array}$$

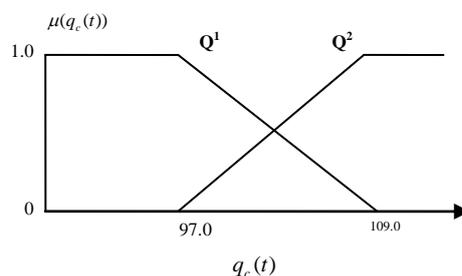


Figure III.1 : Les ensembles flous Q^1 et Q^2

Pour les deux contrôleurs locaux, l'horizon de prédiction et l'horizon de contrôle sont fixées à 10 et 2 respectivement. Les facteurs de pondération W^j_1 et W^j_2 sont tous égaux à 10000 et à 20 sur la diagonale, respectivement. Nous avons étudié la qualité

du contrôleur FPFC par une série de simulations. La réponse dynamique du système est représentée dans la figure III.3 où la consigne est représentée en pointillés, la concentration suit la consigne sans dépassement. La figure III.4 montre que le signal de commande est lisse et sans actions brusques.

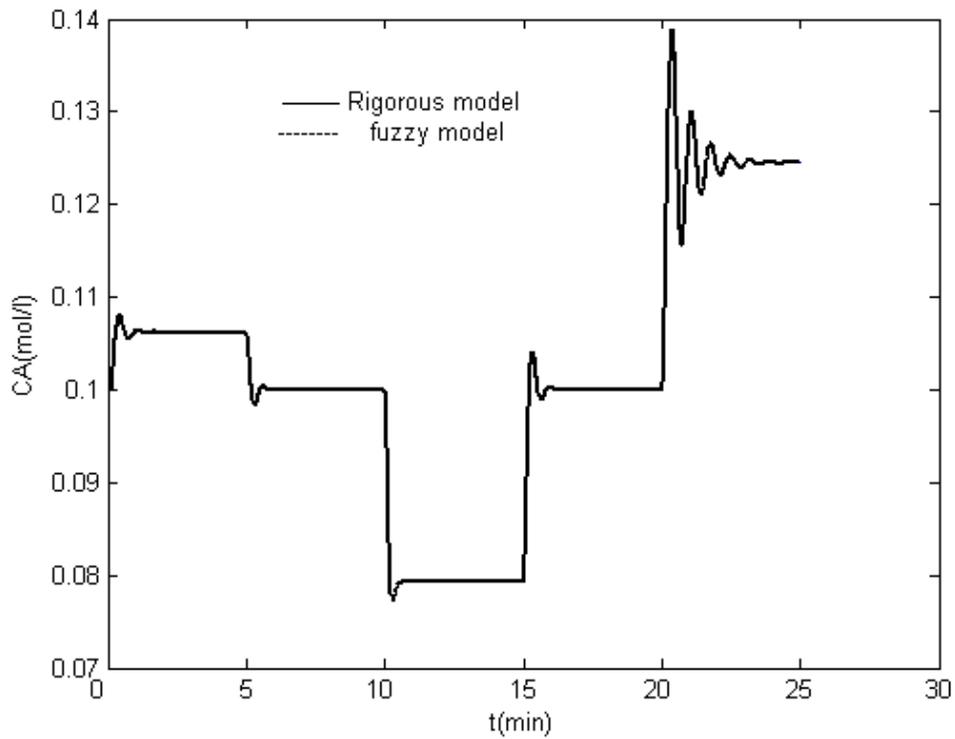


Figure II.2 Réponse en boucle ouverte

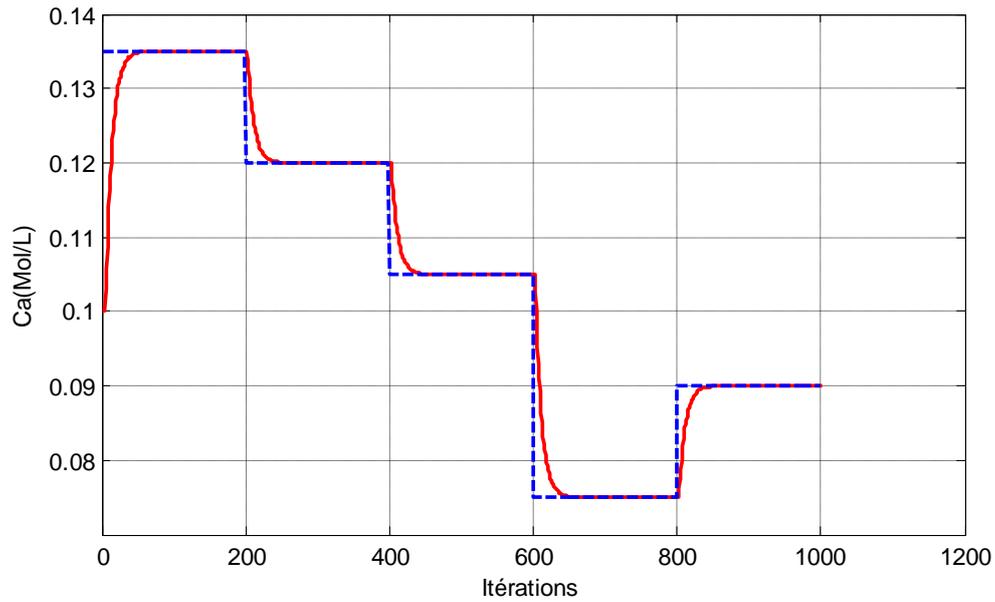


Figure III.3: Evolution de la concentration Ca

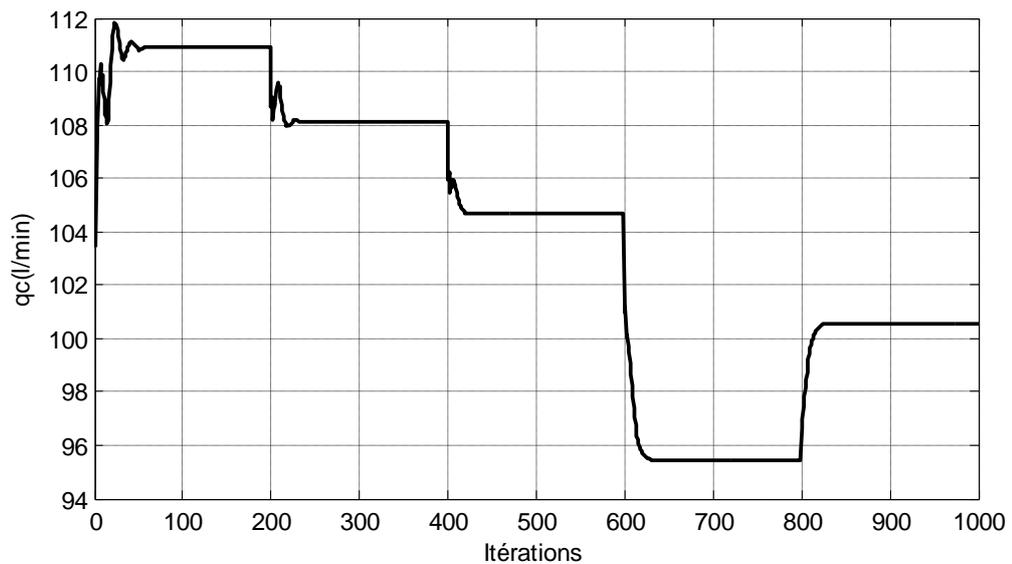


Figure III.4: Le signal de commande

III.4 Conclusion:

Ce chapitre a été consacré à l'élaboration de la loi de commande prédictive fonctionnelle floue hiérarchisée basée sur les mêmes principes que ceux présentés au chapitre précédent. Les résultats de simulation sur un système non linéaire (CSTR) sont encourageants.

Chapitre IV

*Une solution explicite stable et admissible à
la commande prédictive floue à base de
modèle d'état*

Chapitre IV

Une solution explicite stable et admissible à la commande prédictive floue à base de modèle d'état

IV .I Introduction

Dans ce chapitre, nous introduisons une méthode de résolution du problème d'optimisation associé à la commande prédictive floue. Nous commençons par rappeler les principes de base de la prise de décision dans un environnement flou de Bellman Zadeh [1], ensuite nous poserons le problème d'optimisation de MBFPC. Nous considérons la recherche d'une solution stable et admissible.

IV.2 La Commande prédictive Floue

IV. 2.1 Prise de décision dans un environnement flou

Soit X l'ensemble des décisions possibles contenant la solution du problème de la prise de décision floue considérée. Un objectif flou G est un ensemble flou de X caractérisé par sa fonction d'appartenance $\mu_G(x)$. De la même manière que précédemment, une contrainte floue C est un ensemble flou de U caractérisé par sa fonction d'appartenance suivante: $\mu_C(u)$

Considérons un objectif flou et une contrainte floue selon le mécanisme de décision de Bellman et Zadeh [1], la décision floue D résultante est donnée par l'intersection de l'objectif G et de la contrainte C , c'est-à-dire:

$$\mu_D(u) = \mu_G(x) \wedge \mu_C(u) \quad (\text{IV.1})$$

En s'appuyant sur les fonctions d'appartenance définies précédemment, la décision floue sera caractérisée par sa fonction d'appartenance donnée par l'expression suivante: ou \wedge indique l'opérateur minimum

Dans le mécanisme de prise de décision floue, une décision optimale fait référence à la décision avec le plus grand degré d'appartenance (meilleure décision). Cette décision optimale est aussi appelée maximisation de la décision. Sous forme mathématique cette dernière s'exprime comme suit:

$$\mu_D(u^*) = \max(\mu_G(x) \wedge \mu_C(u)) \quad (\text{IV.2})$$

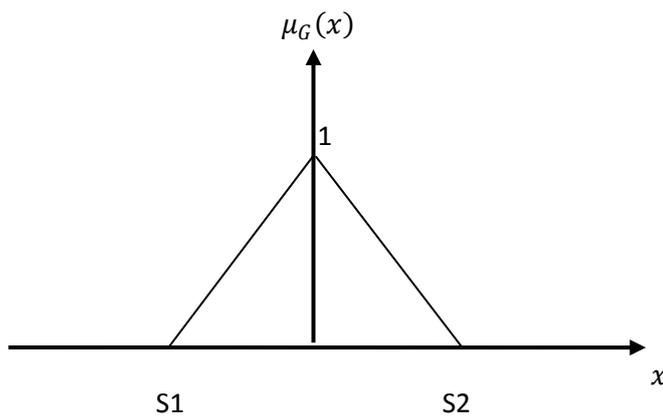


Figure IV.1 : Objectif flou triangulaire

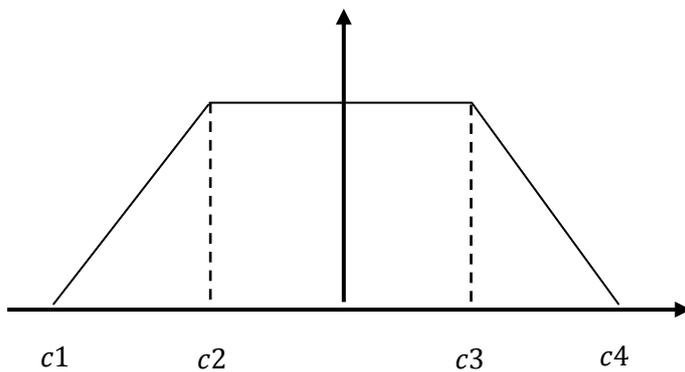


Figure IV.2 : Contrainte floue trapézoïdale

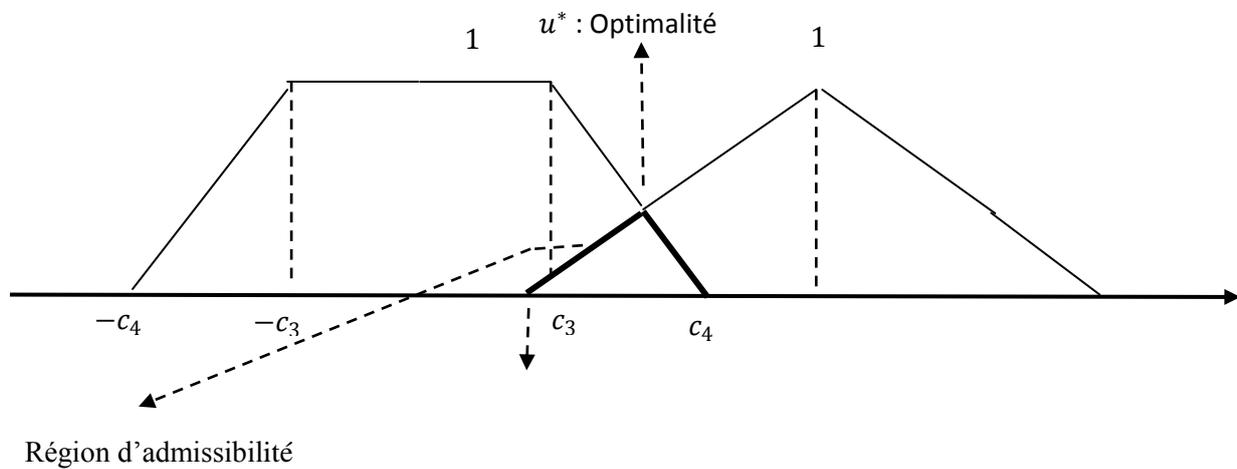


Figure IV.3 : Contraintes et objectif flous : Admissibilité et optimalité

IV.2.2 Commande prédictive floue à base de modèle d'état :

On considère un système à commander déterministe sous forme d'état :

$$x(k+1) = f(x(k), u(k)) \quad (\text{IV.3})$$

$x(k) \in R^n$, $u(k) \in R^m$ Sont respectivement l'état et la commande sujets à des contraintes $x \in X$, $u \in U$, X est un ensemble fermé et U un ensemble compact contenant l'origine en leur intérieur.

Les prédictions futures sur ce système sont alors données par

$$x(k+j) = f(x(k+j-1), u(k+j-1)) \quad (\text{IV.4})$$

Des objectifs flous sont imposés sur les états futurs et des contraintes floues sont imposées sur la séquence future de commandes.

Supposons que la commande est sujet à des contraintes de saturation U_{\min} et U_{\max} nous pouvons interpréter ces contraintes par une phrase du type:” “ la commande ne doit pas être trop grande par rapport à U_{\min} ni trop petite par rapport à U_{\max} ». Cette phrase peut alors

être transformée en une contraintes floue définies par une fonction d'appartenance telle que schématisée par la figue IV.1. L'objectif flou peut être défini par une phrase du type 'l'erreur sur les états doit être petite » qui peut être représentée par une fonction d'appartenance telle que donnée par la figure IV.2.

Pour définir la commande prédictive floue, on impose qu'à chaque instant de prédiction $j=1 \dots N$, les commandes $u(k+j-1)$ sont sujettes à des contraintes floues C^j avec une fonction d'appartenance $\mu_{C^j}(x)$ et les variables d'état $x(k+j)$ sont sujettes à des objectifs flous G^j avec une fonction d'appartenance $\mu_{G^j}(x)$.

On définit alors la fonction coût pour la commande prédictive floues par:

$$J_N(u(k), x(k)) = \mu_{C^1}(u(k)) \wedge \mu_{G^1}(x(k+1)) \wedge \dots \wedge \mu_{C^N}(u(k+N-1)) \wedge \mu_{G^N}(x(k+N)) \quad (\text{IV.5})$$

Cette fonction est positive et inférieure à l'unité.

A partir de ce qui précède, le problème de la MBFPC est de trouver la séquence optimale de commande u qui résout le problème :

$$\text{Max } J_N(u(k), x(k)) \quad (\text{IV.6})$$

$$\text{Sous } \hat{x}(k+j) = f(x(k+j-1), u(k+j-1)) \quad j = 1..N$$

IV.3 La solution explicite admissible

IV. 3.1 Position du problème

L'objectif du problème est de déterminer la région contenant les solutions admissibles du problème posé au paragraphe précédent. En se référant à la figure IV.3, il est clair que cette région est définie par le chevauchement de toutes les contraintes et objectifs flous du problème considéré, rechercher l'ensemble des solutions admissibles revient donc à déterminer la région où les conditions de chevauchements sont satisfaites. Le théorème suivant définit ces conditions.

IV.3.2 Principal résultat : conditions d'admissibilité

Considérons le problème ci-dessus avec pour objectif la régulation à l'origine. Supposons qu'il existe autour de l'origine un ensemble invariant E (voir annexe B) défini par :

$$M_1 \leq Sx \leq M_2 \quad (IV.8)$$

Et caractérisé par :

$$u = -Kx_k \quad (IV.9)$$

$$U_{min} \leq u \leq U_{max} \quad (IV.10)$$

$$x_k, x_{k+1} \in E, \quad (IV.11)$$

On peut alors énoncer le théorème suivant :

Théorème

Une condition nécessaire pour qu'une séquence de commandes admissibles $u_j, j=0 \dots N-1$ transfère l'état courant x_0 à l'intérieur de l'ensemble invariant est que :

$$\max \left(U_{min}, V_{min_1}^{(j)}, \dots, V_{min_i}^{(j)}, \dots, V_{min_n}^{(j)} \right) \leq u_j^f \leq \min \left(U_{max}, V_{max_1}^{(j)}, \dots, V_{max_i}^{(j)}, \dots, V_{max_n}^{(j)} \right)$$

$$V_{min_i}^{(j)} = \frac{m_{1i} - \sum_{l=0}^{N-j-2} sa^{(l)} b_i U_{max} - [SA^{N-j}]_i x_j}{sa^{N-j-1} b_i}, V_{min_i}^{(j)} = \frac{m_{2i} - \sum_{l=0}^{N-j-2} sa^{(l)} b_i U_{min} - [SA^{N-j}]_i x_j}{sa^{N-j-1} b_i} \quad (IV.12)$$

Et

$$V_{min_i}^{(N-1)} = \frac{m_{2i} - [ZA]_i x_{N-1}}{sb_i} \quad V_{max_i}^{(N-1)} = \frac{m_{2i} - [ZA]_i x_{N-1}}{sb_i} \quad (IV.12 \text{ bis})$$

Démonstration

Pour démontrer qu'IV.12 est une condition nécessaire nous procédons par récurrence en démarrant de la région finale et en faisant marche arrière. Nous commençons par démontrer IV.12 bis ensuite nous montrons qu'elle est valable pour $N-2$, ensuite supposons qu'elle est valable pour $N-k$, et nous démontrons qu'elle est valable pour $N-k-1$.

Etape $N-1$

Soit $x_N \in E$ alors $M_1 \leq Sx_N \leq M_2$, en revenant un pas en arrière on obtient :

$$M_1 \leq SAx_{N-1} + SBu_{N-1} \leq M_2 \quad (IV.13)$$

$$M_1 - SAx_{N-1} \leq SBu_{N-1} \leq M_2 - SAx_{N-1} \quad (IV.14)$$

En supposant, sans perte de généralités que tous les éléments de SB sont positifs, on obtient facilement de IV.13 que :

$$V_{\min_i}^{(N-1)} \leq u_{N-1} \leq V_{\max_i}^{(N-1)} \quad i = 1 \dots n - 1, sb_i \neq 0 \quad (IV.15)$$

Avec :

$$V_{\min_i}^{(N-1)} = \frac{m_{2i} - [ZA]_i x_{N-1}}{sb_i} \quad V_{\max_i}^{(N-1)} = \frac{m_{2i} - [ZA]_i x_{N-1}}{sb_i} \quad (IV.16)$$

Où m_{1i} sont les éléments de M_1 , m_{2i} sont les éléments de M_2 , $[ZA]_i$ est la ligne i de SA et sb_i sont les éléments du vecteur SB . De plus, on a la zone d'admissibilité de la commande par rapport aux contraintes :

$$U_{\min} \leq u_{N-1} \leq U_{\max} \quad (IV.17)$$

Pour qu'il existe une solution admissible il faut que IV.15 et IV.17 soient satisfaites simultanément. L'ensemble des commandes admissibles est alors défini par l'intersection de ces inégalités, ce qui s'exprime par l'inégalité suivante :

$$\max(U_{\min}, V_{\min_1}^{(N-1)}, \dots, V_{\min_i}^{(N-1)}, \dots, V_{\min_n}^{(N-1)}) \leq u_{N-1}^f \leq \min(U_{\max}, V_{\max_1}^{(N-1)}, \dots, V_{\max_i}^{(N-1)}, \dots, V_{\max_n}^{(N-1)}) \quad (IV.18)$$

L'ensemble des états pour lesquels cette relation est satisfaite sera noté CF_{N-1} , si cette inégalité est satisfaite, il existe au moins un u_{N-1} qui transfère $x_{N-1} \in CF_{N-1}$ à $x_N \in E$. A partir de V.18, x_{N-1} est contenu dans la région plus grande C_{N-1} définie par :

$$\frac{m_{2i} - [SA]_i x_{N-1}}{sb_i} \geq U_{\min} \text{ et } \frac{m_{1i} - [SA]_i x_{N-1}}{sb_i} \leq U_{\max} \quad i = 1 \dots n - 1, sb_i \neq 0 \quad (IV.19)$$

Après de simples manipulations, on obtient la définition de cette région sous forme matricielle:

$$M_1 - SBU_{\max} \leq SAx_{N-1} \leq M_2 - SBU_{\min} \quad (IV.20)$$

Notons que $CF_{N-1} \subset C_{N-1}$, ce qui implique que x_{N-1} satisfaisant (IV.20) ne satisfait pas nécessairement (IV.20).

Etape N-2 :

En revenant un pas en arrière, on obtient :

$$M_1 - SBU_{max} \leq SA^2 x_{N-2} + SABu_{N-2} \leq M_2 - SBU_{min} \quad (IV.21)$$

En supposant sans perte de généralités tous les éléments SAB positifs, il découle que :

$$V_{min_i}^{(N-2)} \leq u_{N-2} \leq V_{max_i}^{(N-2)} \quad i = 1 \dots n - 1, sab_i \neq 0 \quad (IV.22)$$

Avec

$$V_{min_i}^{(N-2)} = \frac{m_{1i} - sb_i U_{max} - [SA^2]_i x_{N-2}}{sab_i} \quad \text{et} \quad V_{max_i}^{(N-2)} = \frac{m_{2i} - sb_i U_{max} - [SA^2]_i x_{N-2}}{sab_i} \quad (IV.23)$$

Où sab_i sont les éléments de SAB . De la même façon que ci dessus, l'ensemble des commandes admissibles est défini par la satisfaction simultanée des inégalités, soit :

$$\max_i (U_{min}, V_{min_i}^{(N-2)}, \dots, \dots) \leq u_{N-2}^f \leq \min_i (U_{max}, V_{max_i}^{(N-2)}, \dots, \dots) \quad (IV.24)$$

L'ensemble des états pour lesquels la commande existe CF_{N-2} . Si cette inégalité est satisfaite, il existe au moins un u_{N-2}^f qui transfert $x_{N-2} \in CF_{N-2}$ à $x_{N-1} \in C_{N-1}$. A partir de IV.24, x_{N-2} est contenu dans la région C_{N-2} définie par :

$$\frac{m_{1i} - sb_i U_{max} - [SA^2]_i x_{N-2}}{sab_i} \leq U_{max} \quad \text{Et} \quad \frac{m_{2i} - sb_i U_{max} - [SA^2]_i x_{N-2}}{sab_i} \geq U_{min} \quad (IV.25)$$

Après de simples manipulations on obtient sous forme matricielle :

$$M_1 - SBU_{max} - SABU_{max} \leq SA^2 x_{N-2} \leq M_2 - SBU_{min} - SABU_{min} \quad (IV.26)$$

Notons ici aussi que $CF_{N-2} \subset C_{N-2}$, ce qui implique que x_{N-1} satisfaisant (IV.26) ne satisfait pas nécessairement (IV.26).

Etape N-k

Supposant maintenant qu'à l'étape N-k, il existe au moins une commande admissible u_{N-k}^f qui transfert x_{N-k} à x_{N-k+1} et telle que :

$$\max (U_{min}, V_{min_1}^{(N-k)}, \dots, V_{min_i}^{(N-k)}, \dots, V_{min_n}^{(N-k)}) \leq u_{N-k}^f \leq \min (U_{max}, V_{max_1}^{(N-k)}, \dots, V_{max_i}^{(N-k)}, \dots, V_{max_n}^{(N-k)}) \quad (IV.27)$$

Avec

$$V_{\min_i}^{(N-k)} = \frac{m_{1i} - \sum_{j=0}^{k-2} sa^{(j)} b_i U_{\max} - [SA^k]_i x_{N-k}}{sa^{(k-1)} b_i} \quad V_{\max_i}^{(N-k)} = \frac{m_{2i} - \sum_{j=0}^{k-2} sab_i U_{\max} - [SA^k]_i x_{N-k}}{sab_i} \quad (\text{IV.28})$$

$$i = 1 \dots n - 1, sab_i \neq 0$$

Où $za^{(k-1)}b_i$ éléments du vecteur $SA^{k-1}B$ supposé tous positifs sans perte de généralité.

L'ensemble des états pour lesquels IV.28 est satisfaite est noté CF_{N-k} . Il découle de (IV.27) que x_{N-k} est contenu dans la région C_{N-k} :

$$M_1 - \sum_{j=0}^{k-1} SA^j B \leq SA^k x_{N-k} \leq M_2 - \sum_{j=0}^{k-1} SA^j B \quad (\text{IV.29})$$

Avec $CF_{N-k} \subset C_{N-k}$

En revenant un pas en arrière on obtient :

$$M_1 - \sum_{j=0}^{k-1} SA^j B \leq SA^k (Ax_{N-k-1} + Bu_{N-k-1}) \leq M_2 - \sum_{j=0}^{k-1} SA^j B \quad (\text{IV.30})$$

Ou bien :

$$M_1 - \sum_{j=0}^{k-1} SA^j B - SA^{k+1} x_{N-k-1} \leq SA^k Bu_{N-k-1} \leq M_2 - \sum_{j=0}^{k-1} SA^j B - SA^{k+1} x_{N-k-1} \quad (\text{IV.31})$$

Nous supposons ici aussi que les éléments du vecteur $SA^k B$ sont tous positifs alors on peut écrire :

$$V_{\min_i}^{(N-k-1)} \leq u_{N-k-1} \leq V_{\max_i}^{(N-k-1)} \quad i = 1 \dots n - 1, sa^{(k)} b_i \neq 0 \quad (\text{IV.32})$$

Avec :

$$V_{\min_i}^{(N-k-1)} = \frac{m_{1i} - \sum_{j=0}^{k-1} sa^{(j)} b_i U_{\max} - [SA^{k+1}]_i x_{N-k-1}}{sa^{(k)} b_i}$$

$$V_{\max_i}^{(N-k-1)} = \frac{m_{2i} - \sum_{j=0}^{k-1} sa^{(j)} b_i U_{\max} - [SA^{k+1}]_i x_{N-k-1}}{sa^{(k)} b_i} \quad (\text{IV.33})$$

Les différents éléments étant définis comme précédemment.

L'ensemble des commandes admissibles est défini par la satisfaction simultanée des inégalités, soit :

$$\max \left(U_{\min}, V_{\min_1}^{(N-k-1)}, \dots, V_{\min_i}^{(N-k-1)}, \dots, V_{\min_n}^{(N-k-1)} \right) \leq u_{N-k-1}^f \leq$$

$$\min \left(U_{\max}, V_{\max_1}^{(N-k-1)}, \dots, V_{\max_i}^{(N-k-1)}, \dots, V_{\max_n}^{(N-k-1)} \right)$$

L'ensemble des états pour lesquels la commande existe CF_{N-k-1} . et $CF_{N-k-1} \subset C_{N-k-1}$.

Ceci complète la démonstration de la condition nécessaire. La condition n'étant pas suffisante, il n'y a pas de garantie de trouver une solution admissible, ce problème est général pour la commande prédictive. Cependant avec des horizons de prédiction très grands et des tests en simulation on peut arriver, pour un problème donné, à garantir l'admissibilité.

Nous avons supposé que les éléments de SB, SAB,... sont positifs, les calculs relatifs à ces éléments étant réalisés hors ligne, ils n'influent pas sur la démonstration, il suffit de changer le sens des inégalités.

IV.4 Résultat de simulation

Nous considérons l'application de la commande explicite obtenue ci-dessus à un exemple qu'on trouve dans [2]. Le système à commander est défini par :

$$x(k+1) = \begin{bmatrix} 0.7326 & 0.0861 \\ 0.1722 & 0.9909 \end{bmatrix} x(k) + \begin{bmatrix} 0.0609 \\ 0.0064 \end{bmatrix} u(k) \quad (\text{IV.34})$$

Avec $-2 \leq u(k) \leq +2 \quad \forall k$

L'ensemble invariant pour ce problème est donné par [2] :

$$\begin{pmatrix} -2 \\ -2 \end{pmatrix} \leq \begin{pmatrix} 5.9920 & 6.8863 \\ 1.5379 & -6.8291 \end{pmatrix} x \leq \begin{pmatrix} 2 \\ 2 \end{pmatrix} \quad (\text{IV.35})$$

Une commande stabilisante est $u = (-5.9220 \quad -6.8883)x$ (IV.36)

L'horizon est $N=20$. Les conditions initiales : $x = (1 \quad 1)^T$. Les calculs hors ligne sont effectués en fonction des signes des éléments de S, SAB,... Pour illustration, nous donnons les premiers éléments :

$$SB1 = 0.4047 ; SB2 = 0.0499$$

$$SAB1 = 0.37896 ; SAB2 = -0.0492$$

$$SA2B1 = 0.3531 ; SA2B2 = -0.1218$$

$$SA3B1 = 0.3277 ; SA3B2 = -0.1735$$

$$SA4B1 = 0.3031 ; SA4B2 = -0.2088$$

$$SA5B1 = 0.2795 ; SA5B2 = -0.2312$$

$$SA6B1 = 0.2571 ; SA6B2 = -0.2437$$

$$SA7B1 = 0.2359 ; SA7B2 = -0.2487$$

$$SA8B1 = 0.2160 ; SA8B2 = -0.2480$$

$$SA9B1 = 0.1975 ; SA9B2 = -0.2431$$

$$SA10B1 = 0.1803 ; SA10B2 = -0.2351$$

On peut choisir la commande dans le domaine d'admissibilité défini par IV. 12. A partir de cela, trois tests ont été effectués correspondant à trois choix:

$$\textbf{Test 1: } u_j^f = \max (U_{min}, V_{min_1}^{(j)}, \dots, V_{min_i}^{(j)}, \dots, V_{min_n}^{(j)})$$

$$\textbf{Test 2: } u_j^f = \min (U_{max}, V_{max_1}^{(j)}, \dots, V_{max_i}^{(j)}, \dots, V_{max_n}^{(j)})$$

$$\textbf{Test 3: } u_j^f = (\min(\dots) + \max(\dots))/2$$

Pour comparaison, nous avons implémenté au test 4, la commande de [2]. Les figures IV.4, IV.5 et IV.6 donnent respectivement les états, les signaux de commande et les trajectoires dans l'espace d'état pour les quatre tests effectués. Les commandes pour les tests 1, 2 et 3, en traits discontinus et pointillés, sont sensiblement les mêmes ainsi que les réponses des états. Elles subissent cependant une variation importante lors de l'entrée dans l'ensemble invariant. La commande de [2], en traits continus, subit des variations moins accentuées au prix d'une réponse des états plus lente.

Le temps de calcul pour une itération sur une machine IBM compatible 1.4GHz est de 0.000325 secondes tandis que pour la solution de [2] il est de 0.000006 secondes.

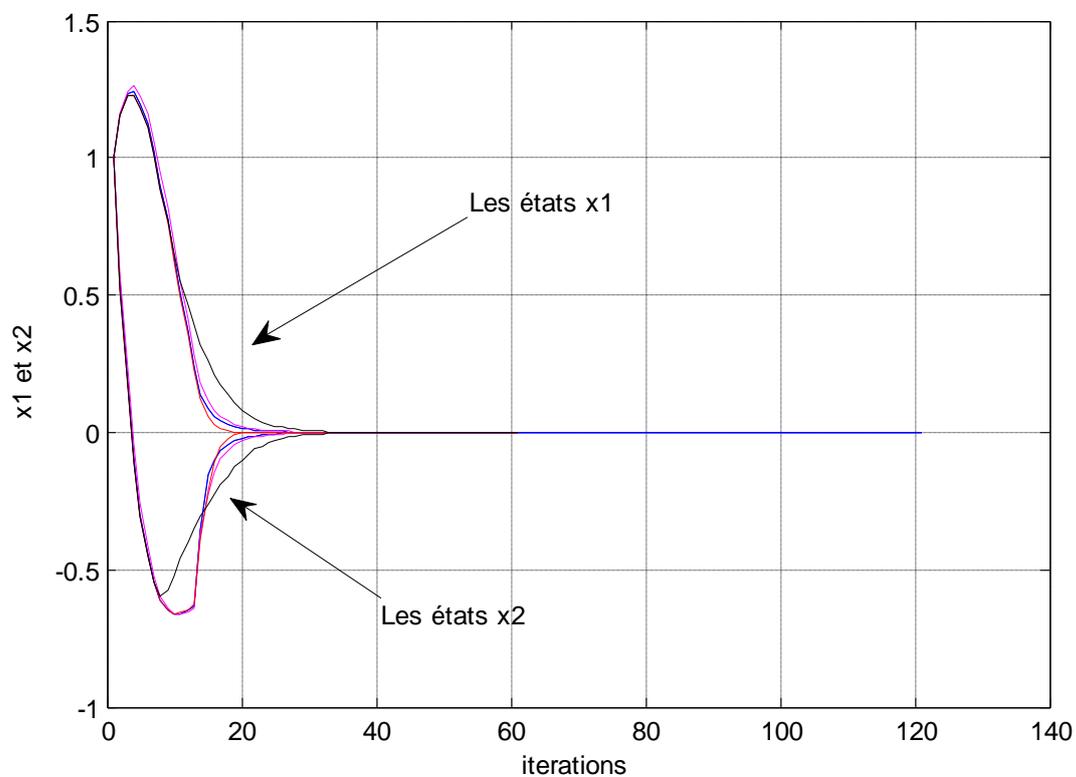


Figure IV.4. Les états pour les quatre tests

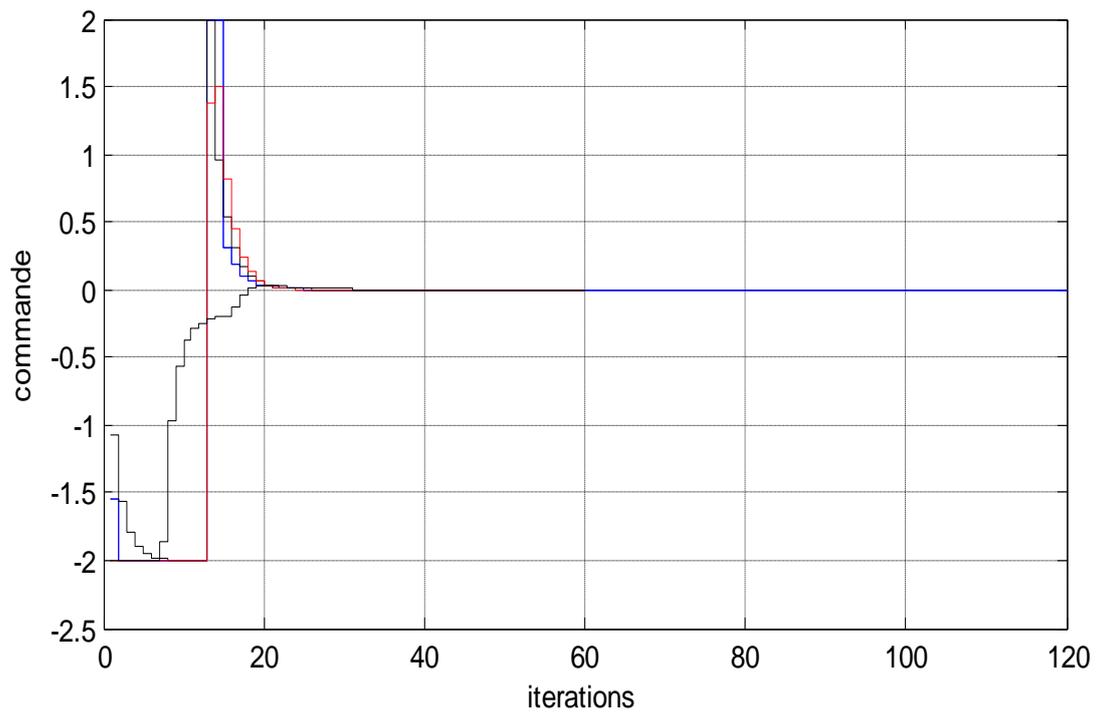


Figure IV.5: Les signaux de commande pour les quatre tests

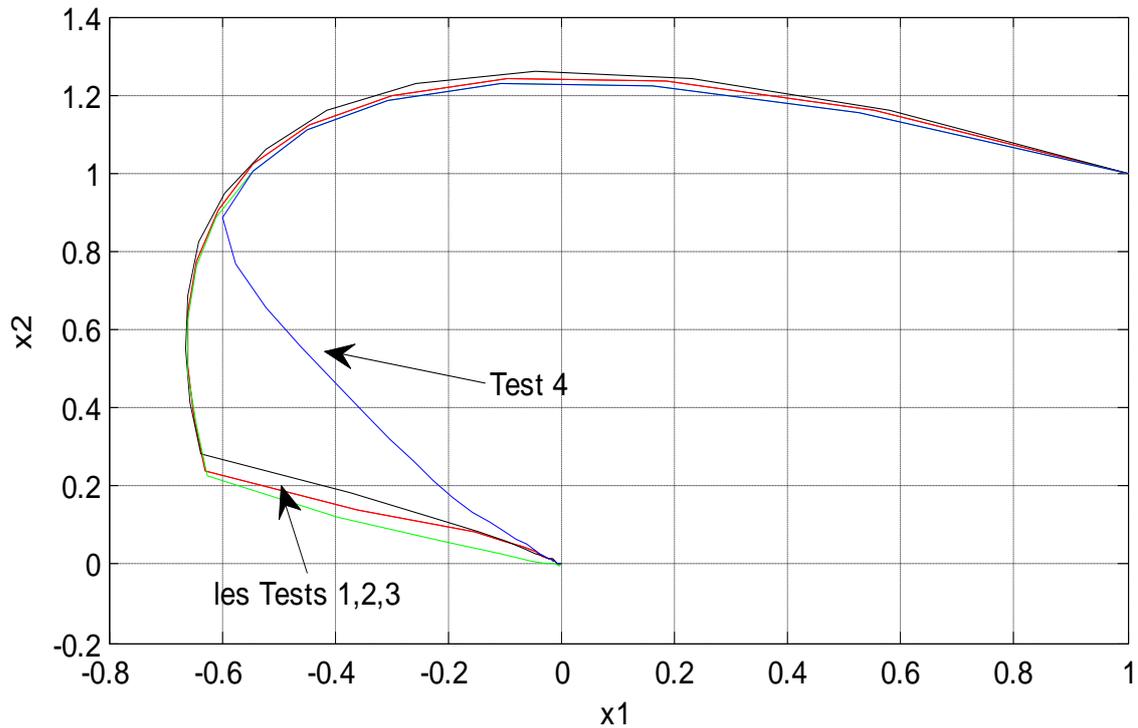


Figure IV.6: Les trajectoires de phase pour les quatre tests

IV.5 Conclusion :

Dans ce chapitre nous avons élaboré une solution explicite admissible introduite par la nouvelle méthodologie de résolution du problème d'optimisation de la commande prédictive floue à base de modèle d'état. Cette approche est basée sur les notions d'objectifs et contraintes flous introduites par Bellman et Zadeh. La stabilité est garantie en utilisant la théorie des ensembles invariants. La solution explicite subdivise l'espace d'état en régions, pour chaque région, l'ensemble des lois de commande admissible est déterminée. Les résultats de simulation sur un problème donné par la référence [2] sont encourageants.

Conclusion Générale

Conclusion Générale

Dans ce travail nous avons considéré des méthodologies de résolution de la commande prédictive à base de modèle. Dans un premier temps, nous avons développé des techniques de commande prédictive linéaire qui s'appliquent aux systèmes non linéaires. La procédure a utilisé des systèmes d'inférences floues de type Takagi Sugeno, moyennant la technique hiérarchisée.

Le système non linéaire global se décompose en sous systèmes linéaires où chaque sous système est généré par une règle floue. Les contrôleurs prédictifs linéaires ont été calculés pour chaque modèle linéaire local. Ces contrôleurs locaux ont été combinés afin d'élaborer le contrôleur global en utilisant l'approche hiérarchisée. L'approche hiérarchique destinée à la conception d'une loi de commande prédictive se compose deux couches coordonnées entre elles par le biais d'un terme d'erreur appelé terme de coordination. Ce calcul nous a permis d'obtenir un algorithme de commande garantissant un quasi optimalité de la loi de commande prédictive.

Cette approche hiérarchique a été adoptée dans le cas de notre conception de la loi de commande prédictive généralisée GPC puis fonctionnelle de type PFC.

Dans un deuxième temps, nous avons élaboré une solution explicite admissible de la commande prédictive floue qui se base sur les notions d'objectifs et contraintes flous introduites par Bellman et Zadeh [1], utilisant des ensembles invariants.

Pour cela nous avons choisi un état final du système à l'intérieur de l'ensemble invariant. Les lois de commande explicite admissible sont calculées en utilisant la programmation dynamique pour atteindre l'état initial. Ainsi si une séquence admissible existe alors la stabilité du système est garantie. La solution explicite subdivise l'espace d'état en régions, pour chaque région, l'ensemble des lois de commande admissible est déterminée. Les résultats de simulation sur un problème de référence sont encourageants.

ANNEXE A

ANNEXE A

LES SYSTEMES FLOUS

I. LA THEORIE DE LA LOGIQUE FLOUE :

I.1 Définition de la logique floue :

Par sa nature approximative non quantitative, la logique floue permet de traiter des variables qui ne sont plus numériques avec des valeurs bien définies, mais elle manipule des variables linguistiques, avec des variables linguistiques associées à des sous-ensembles flous qui décrivent leur répartition sur un univers de discours.

I.2 Les bases de la logique floue :

a) Univers de discours :

Un univers de discours continu est un sous-ensemble qui décrit dans le cas général le domaine de variation d'une variable donnée.

On utilise souvent des univers de discours normalisés $[-1,1]$; il existe alors un coefficient de normalisation k entre l'univers réel et l'univers normalisé (parfois on considère univers de discours discrétisés).

b) Définition des sous-ensembles flous (fuzzy sub set) :

Un sous-ensemble flou noté F défini sur l'univers de discours U est caractérisé par une fonction caractéristique appelée fonction d'appartenance de U notée μ_f et qui prend ses valeurs dans l'intervalle $[0,1]$ à savoir :

$$\mu_f: U \rightarrow [0,1]$$

$$u \rightarrow \mu_f(u)$$

Ainsi, si on définit plusieurs ensembles flous sur un univers de discours donné, chaque élément de U aura un degré d'appartenance à un sous-ensemble flou avec :

$$F = \{(u, \mu_f(u)/u \in U)\}$$

- Lorsque U est un univers de discours continu un sous-ensemble flou F peut être décrit comme :

$$\blacksquare F = \phi_u \mu f(u)/u \quad (\text{A.1})$$

- Lorsque U est un univers de discours discret, un sous-ensemble flou est représenté par :

$$F = \sum_{i=1}^n \mu f(ui)/ui \quad (\text{A.2})$$

Les fonctions d'appartenance les plus utilisées sont des fonctions triangulaires ou Gaussiennes (figure A.1)

- ❖ **La fonction triangulaire** : elle est caractérisée par trois paramètres (a, b, c), les sommets du triangle : $\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right)$
- ❖ **La fonction trapézoïdale** : définie par quatre paramètres (a, b, c, d) : $\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right)$
- ❖ **La fonction Gaussienne** : définie par m et σ le centre et l'épaisseur de la fonction Gaussienne respectivement : $\mu(x) = \exp\left(-\frac{(x-m)^2}{\sigma^2}\right)$

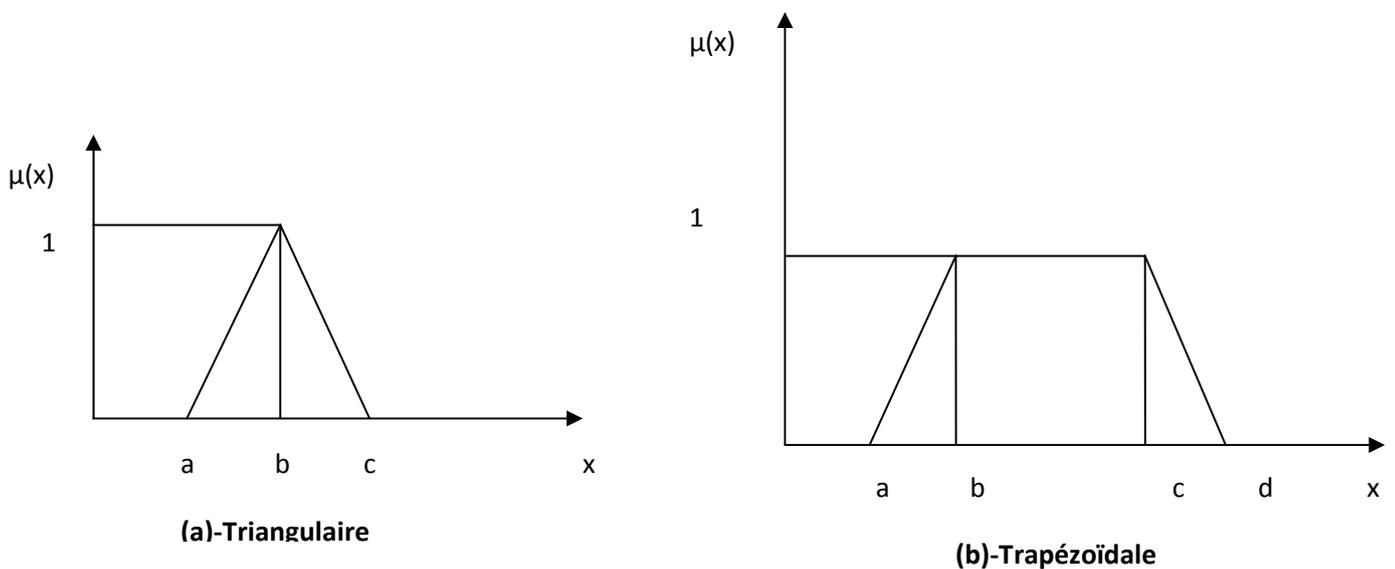


Figure A.1 : Formes des fonctions d'appartenance

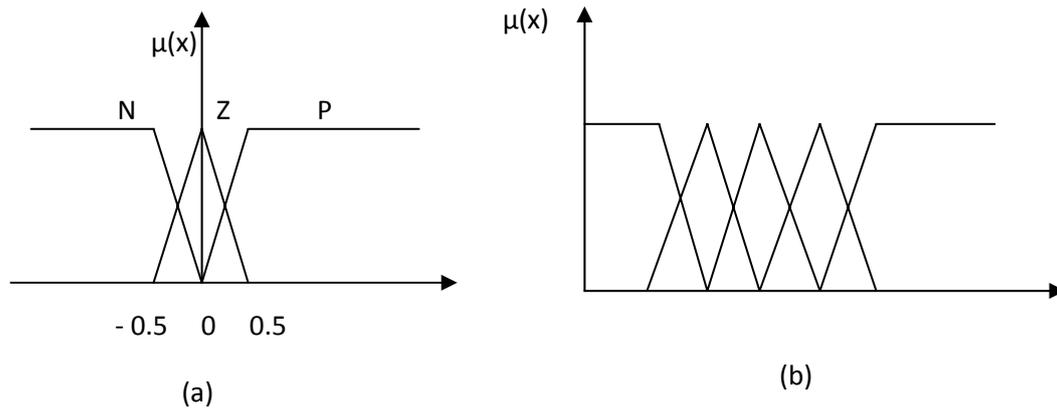


Figure A.2: Exemples d'ensembles flous sur un univers de discours

C) Définition d'une variable linguistique :

En logique floue les variables ne sont plus des variables numériques avec des valeurs définies sur un univers de discours.

Exemple : la variable linguistique « V. vitesse » au quelle associé l'univers de discours U

$v = [80, 160]$, cette variable prend des valeurs linguistiques données par $V = \{lent, modéré, rapide\}$ qui constitue 3 sous ensembles représentés par la fonction d'appartenance

$\mu_{Lent}(u)$; $\mu_{modéré}(u)$; $\mu_{rapide}(u)$, avec $u \in \mu$ c'est l'élément générique.

Ces fonctions représentent le degré d'appartenance pour chaque μ .

II. STRUCTURE GENERALE D'UN SYSTEME FLOU :

La structure générale d'un système floue se compose de 4 éléments de base : la fuzzification, la base des règles, le moteur d'inférence flou et la défuzzification comme c'est montré dans la figure (A.2)

II.1 La fuzzification :

C'est l'interface numériques-flou, elle transforme les données numériques en données linguistiques.

II.2 La base des connaissances :

Elle comporte la base de données et la base des règles.

- ❖ Base de données : type et caractéristiques des fonctions d'appartenances, univers de discours, nombre des variables linguistiques, et le nombre d'ensembles flous associés à chaque variable linguistique.
- ❖ Base des règles : c'est une collection de règles floues de type :

SI (l'ensemble des conditions sont satisfaites) *ALORS* (un ensemble de conséquences peuvent être inférées).

II.3 Moteur d'inférence (la décision) :

C'est le mécanisme qui interprète une règle floue *SI-ALORS* ;

On distingue deux types d'opérateurs, l'opérateur de conjonction *ET* qui lie les différentes variables de prémisse, l'opérateur de disjonction *OU* qui lie les règles formant la base de connaissance.

Parmi les mécanismes d'inférence les plus utilisés on distingue :

- ❖ Mécanisme d'inférence **Max-Min**
- ❖ Mécanisme d'inférence **Max-Pro**
- ❖ Mécanisme d'inférence **Som-Pro**

II.4 Défuzzification :

C'est l'interface flou-réel, elle transforme les données linguistiques en données numériques.

Il existe plusieurs stratégies de défuzzification :

- ❖ Défuzzification par la méthode de centre de gravités
- ❖ Défuzzification par la méthode de hauteurs pondérées
- ❖ Défuzzification par valeur maximum...etc.

La plus utilisée est la défuzzification par la méthode de centre de gravité.

$$y_{Numérique} = \frac{\sum_{i=1}^l m_i \mu_{ci}}{\sum_{i=1}^l \mu_{ci}}$$

m_i sommet des fonctions d'appartenance (fuzzy set) de C_i , l : le nombre des règles floues.

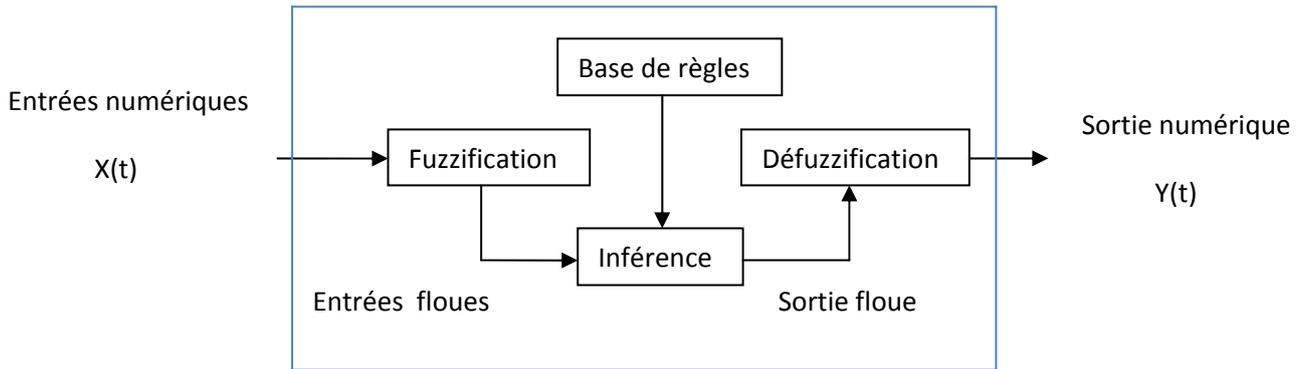


Figure A.3: Système flou

III. LES MODELES FLOUS :

Le modèle de raisonnement approximatif flou est généralement formé d'un ensemble de règles qui décrit le comportement du système ces règles auront la structure suivante :

SI (antécédent, prémisse, condition) \Longrightarrow *ALORS* (conséquences)

La modélisation floue donc est déduite d'un raisonnement élaboré des états de processus et d'une liste des règles décrivant la manière selon laquelle le modèle doit fonctionner pour réaliser les performances désirées.

Plusieurs modèles de raisonnement ont été développés, les plus courants sont :

- Le modèle de **MAMDANI-ZADEH** (1974).
- Le modèle de **TAKAGI-SUGENO** (1985).

III.1 Le modèle de MAMDANI :

Un modèle flou de **MAMDANI** ne traite pas une relation mathématique bien définie mais utilise des inférences avec plusieurs règles en se basant sur des variables linguistiques. Ces inférences sont de la forme :

SI (condition, antécédent ou prémisse) *ALORS* (actions, conclusion ou conséquence).

L'antécédent et la conséquence (conclusion) sont des opérations floues dans lesquelles on trouve des variables linguistiques liées par des opérateurs de la logique floue, ainsi l'ensemble

des opérations affectées par l'opérateur de la logique floue, ne porte pas sur les valeurs réelles mais sur les valeurs qualificatives exprimées par des termes linguistiques ou par code approprié. Dans le cas général :

SI (X_1 est A_1) et (X_2 est A_2)..... (X_n est A_n) **ALORS** Y est B (ou bien plusieurs sorties).

III.2 Le modèle de SUGENO :

Le modèle de raisonnement de **TAKAGI-SUGENO** est défini par un ensemble de règles qui sont constituées d'antécédent symbolique (prémises) et de conséquences ou conclusions sous forme d'équation linéaire, les deux premières étapes la fuzzification des entrées et l'inférence sont présentées comme dans le modèle de **MAMDANI** mais la défuzzification est remplacée une procédure qui calcule directement la sortie de chaque règle floue, en fonction des états du système. L'action ou bien la sortie globale revient à une somme pondérée de toutes les sorties de chaque règles floue.les règles sont du type :

$$R^j : \text{SI } (X_1 \text{ est } A_1, X_2 \text{ est } A_2, \dots, X_k \text{ est } A_k) \text{ ALORS } y^j = a_0^j + a_1^j x_1 + \dots + a_k^j x_k \quad (\text{A.3})$$

Avec $j=1 \dots N$; où N est le nombre de règles.

y^j : est la variable de conséquence pour chaque règle.

X_1 - X_k : est la variable de prémisse (antécédents).

A_1 - A_k : les ensembles flous.

$a_1^j - a_k^j$: Paramètres constant des conséquences.

SUGENO a proposé un algorithme pour la déduction de la valeur finale de y , cet algorithme est décrit comme suit :

On suppose qu'on a R^j ($j=1 \dots n$) implications et on pose $X_i = X_i^0$ ($i=1 \dots k$) et ils sont des singletons. La valeur de la sortie y peut être inférée selon les étapes suivantes :

1) Pour chaque implication R^j , y^j est calculée par :

$$y^j = a_1^j x_1^0 + \dots + a_k^j x_k^0 \quad (\text{A.5})$$

2) La valeur réelle de $y=y^j$ est calculée par :

$$|y = y^j| = |x_1^0 \text{ est } A_1^j \text{ ET } \dots \dots x_k^0 \text{ est } A_k^j|^{\wedge} |R^j|$$

$$= (A_1^j(x_1^0) \wedge \dots \wedge A_k^j(x_k^0)) \wedge |R^j| \quad (\text{A.6})$$

$|*|$ Représente la valeur de position et \wedge représente l'opérateur « min » et $A(x_i^0)$ est le degré d'appartenance de X^0 , pour simplification $|R^j| = 1$

$$\text{Alors : } |y = y^j| = (A_1^j(x_1^0) \wedge \dots \wedge A_k^j(x_k^0)) \wedge |R^j| \quad (\text{A.7})$$

3) La valeur réelle de la sortie est calculée par :

$$y = \frac{\sum_{j=1}^n |y=y^j| y^j}{\sum_{j=1}^n |y=y^j|} \quad (\text{A.8})$$

ANNEXE B

ANNEXE B

Ensembles invariants

Le concept de l'invariance joue un rôle important dans le domaine de la théorie de la commande [2] [3] [5] : synthèse de commande sous contraintes, analyse de stabilité et de robustesse etc. Etant donné un système dynamique, un sous-ensemble de l'espace d'état est dit positivement invariant s'il vérifie la propriété suivante [3]:

Pour tout x_k appartenant à cet ensemble à un moment donné implique son inclusion dans le futur et dans le passé (donc pour une évolution directe ou inverse également).

Ensembles invariants ellipsoïdaux :

Considérons le système discret :

$$x_{k+1} = Ax_k \quad (\text{B.1})$$

Soit une fonction de Lyapunov pour ce système

$$V(x_k) = x_k^T P x_k \quad \text{Avec} \quad A^T P A - P = -Q \quad (\text{B.2})$$

Où $P = P^T > 0$ et $Q = Q^T > 0$ sont des matrices symétriques et définies positives, alors l'ensemble ellipsoïdal ε :

$$\varepsilon = \{x \in \mathbb{R}^n : x^T P x \leq 1\} \quad (\text{B.3})$$

Est un ensemble invariant pour le système.

Ensembles invariants polyédraux:

Une famille importante des ensembles convexes est la famille polyédrale. L'avantage principal de ces ensembles est le fait qu'ils forment un foyer fermé par rapport aux opérations mentionnées. Leur inconvénient majeur est la complexité de sa représentation qui n'est pas fixée par la dimension de l'espace

Définition (ensemble polyèdre) :

Un ensemble polyèdre convexe est donné par la forme:

$$P(F, g) = \{x : Fx \leq g\} = \{x : F_i x \leq g_i, i = 1, 2, \dots, s\} \quad (\text{B.4})$$

Où F_i : $i^{\text{ème}}$ ligne de la matrice $F(s \times n)$ et g_i est l' $i^{\text{ème}}$ élément du vecteur $g(s \times 1)$

Un ensemble polyèdre comprend l'origine si et seulement si $g \geq 0$ et comprend les origines comme un point intérieur si et seulement si $g > 0$ ($g_i > 0, \forall i$). Nous allons utiliser la notation $\bar{1}$ c'est à dire le vecteur de toutes les composantes égale à 1 $\bar{1}^T = [1, \dots, 1]$

Alors que l'ensemble polyèdre comprend l'origine peut ainsi représenté par la forme:

$$P(F, 1) = P(F) = \{x : Fx \leq \bar{1}\} = \{x : F_i x \leq 1, i = 1, 2, \dots, s\} \quad (\text{B.5})$$

Qui peut être accompli à partir de (3,15) en divisant les deux côtés de chaque inégalité par $g_i > 0$.

Un ensemble convexe polyèdre 0-symétrique peut être toujours représenté par la forme:

$$\bar{P}(F, g) = \{x : -g \leq Fx \leq g\} = \{x : -g_i \leq F_i x \leq g_i, i = 1, 2, \dots, s\} \quad (\text{B.6})$$

Encore, si $\bar{p}(F, g)$ contient le 0 comme point intérieure, donc la normalisation nous donne la représentation tel que:

$$\bar{P}(F, 1) = \bar{P}(F) = \{x : -\bar{1} \leq Fx \leq \bar{1}\} = \{x : -1 \leq F_i x \leq 1, i = 1, 2, \dots, s\} \quad (\text{B.7})$$

Le polyèdre \bar{P} peut être représenté par ses sommets:

$$\bar{P} = \left\{ x = T\alpha, \bar{1}^T = \sum_{i=1}^s \alpha_i = 1, \alpha \geq 0 \right\} \quad (\text{B.8})$$

Avec $T \in \mathfrak{R}^{n \times s}$ et $\alpha \in \mathfrak{R}^s$. Dans la représentation (B.8) chaque colonne de la matrice T est un sommet de l'ensemble \bar{P} .

Le théorème suivant [3] établit les conditions de la positive invariance pour l'ensemble P défini par (B.7)

Théorème: Les conditions suivantes sont équivalentes:

(i) l'ensemble polyèdre contractif \bar{P} est positivement invariant pour le système discret (B.1) $x_{k+1} = Ax_k$

(ii) Il existe un réel positif $\lambda \leq 1$, et une matrice non négative F telle que:

$$H\bar{1} \leq \lambda\bar{1} \quad (\text{B.9})$$

$$HF = FA \quad (\text{B.10})$$

(iii) Il existe un réel positif $\lambda \leq 1$, et une matrice non négative P telle que:

$$\bar{1}^T P \leq \lambda\bar{1}^T \quad (\text{B.11})$$

La condition (ii) est basée sur le lemme suivant:

Lemme1: Etant données deux polyèdres :

$$\bar{P}_1 = \{x \in \mathfrak{R}^n : H_1 x \leq k_1\} \quad (\text{B.12})$$

$$\text{Et } \bar{P}_2 = \{x \in \mathfrak{R}^n : H_2 x \leq k_2\} \quad (\text{B.13})$$

Alors $\bar{P}_1 \subseteq \bar{P}_2$ si et seulement s'il existe une matrice non-négative H tel que

$$HH_1 = H_2$$

$$\text{Et } Hk_1 = Hk_2$$

L'ensemble \bar{P} est positivement invariant si et seulement si $x \in \bar{P}$ implique $Ax \in \lambda \bar{P}$, ainsi $FAx \leq \lambda \bar{1}$. Cela veut dire que le polyèdre R doit être inclus dans le polyèdre $\{x \in \mathfrak{R}^n : FAx \leq \lambda \bar{1}\}$.

L'application du lemme 1 nous conduit à la condition (ii).

Pour la démonstration de la condition (iii) on considère un vecteur. Ce vecteur peut être exprimé par la combinaison convexe $x \in \bar{P}$, ce vecteur peut être exprimé par la combinaison convexe des sommets de \bar{P} ; ce qui veut dire $x = T\alpha$, donc le polyèdre \bar{P} est invariant si et seulement si :

$$\forall x_j \in \text{vert}(\bar{P}), \quad Ax_j \subseteq \lambda \bar{P} \quad (\text{B.14})$$

Ou bien d'une façon équivalente :

$$\forall x_j \in \text{vert}(\bar{P}), \quad Ax_j = Tp_j \quad (\text{B.15})$$

Avec p_j vecteur non-négatif tel que :

$$\sum_{i=1}^s p_{i,j} = \bar{1}^T p_j \leq \lambda \leq 1 \quad (\text{B.16})$$

Avec p_{ij} est la $i^{\text{ème}}$ composante de p_j . En posant $P = [p_{ij}]$ on obtient la condition (iii)

On peut généraliser ce résultat comme suit:

L'ensemble \bar{P} est invariant par rapport a la commande pour le système :

$$x_{k+1} = Ax_k + Bu_k \quad (\text{B.17})$$

Si et seulement s'il existe un positif $\lambda \leq 1$, une matrice $P \in \mathfrak{R}^{s \times s} \geq 0$, et une matrice $U \in \mathfrak{R}^{m \times s} \geq 0$ telles que

$$AT + BU = TP \quad (\text{B.18})$$

$$\text{Et } \bar{1}P \leq \lambda \bar{1}^T$$

Les colonnes de U sont les commandes u_k associées aux colonnes de T c'est-à-dire les sommets de x_k de \bar{P} .

BIBLIOGRAPHIE

REFERENCES

- [1] Bellman R. and L.A. Zadeh, "Decision Making In a Fuzzy Environment", *Management Science*, Vol. 17, No, December, 1970.
- [2] Bemporad A., M. Morari, V. Dua, E. Pistikopoulos, "The explicit solution of model predictive control via multiparametric quadratic programming", *Proceedings of the American Control Conference*, Chicago, Illinois, pp.872-876, 2000.
- [3] H. Benaoukli, "*Faisabilité et ensembles invariant pour la commande prédictive*", thèse de doctorat soutenue le 08/10/2009.
- [4] Blanchini F., "Set invariance in control", *Automatica*, pp.1747-1767, 1999
- [5] Blanchini F. and S. Miani, *Set theoretic Methods in Control* Birkhauser Boston 2008
- [6] Boucher P. and D.Dumur, *La commande prédictive. Avancées et perspectives*, Lavaoisier, 2006.
- [7] Bourebia O. and K. Belarbi, "Fuzzy generalized predictive control for nonlinear systems with coordination technique", *International Review of Automatic Control (IREACO)*, Vol.1, N.2, July 2008.
- [8] O.Bourebia and K. Belarbi, " Optimization of fuzzy predictive functional control using coordination technique", 4th International Conference on computer integrated Manufacturing, Setif, Algeria, 2007.
- [9] O.Bourebia and K. Belarbi," Fuzzy Predictive functional Control of nonlinear systems", *International Conference on Electrical Engineering Design and Technologies ICEEDT'07*, 2007.
- [10] Burns R., *Advanced Control Engineering* , Linacre House, Jordan Hill, Oxford, 2001.
- [11] Camacho E. F., C. Bordons, *Model predictive control*, Ed. Springer-Verlag, London, 2004.
- [12] Cannon M., V. Deshmukh, B. Kouvaritakis , "Non linear model predictive control with polytypic invariant sets", In *Proceeding of 15th IFAC word Congress*, Barcelona, Spain, 2002.
- [13] Chemachema M., F. Megri, K. Belarbi, "A fuzzy dynamic programming solution to constrained non linear predictive control", *Proceedings of IEEE SMC02*, Hammamet, Tunisia, 6-9 October 2002.

- [14] Clarke D.W., C. Mohtadi and P. S. Tufs, "Generalized predictive control, Part I. The basic algorithm," *Automatica*, vol. 23, pp. 137-148, **1987**.
- [15] D. Dumur D., "Contribution à l'analyse et au développement d'algorithmes de commande prédictive", Thèse Supélec, 2002.
- [16] J. J. Espinosa, M. L. Hadjili, V. Wertz and J. Vandewalle, "Predictive control using fuzzy models: comparative study," *Proc. European Cont. Conf.*, pp. F0547, **1999**.
- [17] M.Fiacchini, T.Alamo, EF.Camacho, "On the computation of local invariant sets for nonlinear systems", in *Proceeding of 46th IEEE Conference on Decision and Control*, New Orleans, Dec **2007**.
- [18] Fletcher .R, *Practical Methods of Optimization 2: Constrained Optimization* , John Wiley and Sons, Chichester, 1981.
- [19] Gilbert. . G., K. TinTan, "Linear Systems with State and Control Constraints: The theory and Application of Maximal Output Admissible Sets", *IEEE Transactions on Automatic Control*, Vol.36, N° 9, pp. 1008-1020, September **1991**.
- [20] Huang Y. L., H. H. Lou, J.P. Gong and T. F. Edgar, "Fuzzy model predictive control," *IEEE Trans. Fuzzy systems*, vol. 8, N° 6, pp. 665-678, **2000**.
- [21] Keerthi S. S. and E. G. Gilbert, "Optimal infinite horizon feedback laws for a general class of constrained discrete time systems: Stability and moving-horizon approximations", *Journal of Optimization Theory and Applications*, Vol. 57, pp. 265–293, 1988.
- [22] Maciejowski J. M., *Predictive control with constraints*, Prentice Hall, Pearson Education Limited, Harlow, England, **2002**.
- [23] Mayne D.Q., J.B. Rawlings, C.V. Rao, "Constrained model predictive control: Stability and optimality", *Automatica* , N°36, pp.789-814,**2000**.
- [24] Mollov S. and T.van den Boom," Robust Stability Constraints for Fuzzy Model Predictive Control", *IEEE Transaction on Fuzzy Systems*, Vol.10,N°.1, pp. 50-64, February **2002**.
- [25] Nakamori Y., K. Suzuki and T. Yamanaka, "Model predictive control using fuzzy dynamic models," *Proc. IFSA '91 Brussels*, vol. 135, Brussels, Belgium, July **1991**.
- [26] H. N. Nounou and K. M. Passino, " Fuzzy model predictive control: techniques, stabilities issues and examples," *Proc. IEEE Int. Symp. Intelligent control, intelligent systems and Semiotics*, pp. 423-428, **1999**.

-
- [27] J. Rawlings, E. Meadows, K. Muske, "Nonlinear model predictive control: A tutorial and survey", Preprints of IFAC Symposium on Advanced Control of Chemical Processes (ADCHEM'94), Kyoto, Japan, 25-27 May, pp.203-214, **1994**.
- [28] J. Richalet," Modélisation et identification des processus", R7 140, Techniques de l'ingénieur traité mesures et contrôle, vol R7, **1983**.
- [29] J. Richalet, A. Rault , J.L. Testud et J. Papon," Model predictive heuristic control: application to industrial process", Automatica , Vol 14, N° 5, pp 413- 428, **1978**.
- [30] Roubos J. A., S. Molloy, R. Babuska and H. B. Verbruggen, "Fuzzy Model-Based Predictive control using Takagi-Sugeno Models," Int. J. of Approximate Reasoning, vol. 22, pp. 3-30, **1999**.
- [31] Sousa M.," Model Predictive Control Using Fuzzy Decision Function", IEEE Transaction on Systems. MAN AND CYBERNETICS –Part B, Vol.31, N°1, pp. 54-65, February **2001**.
- [32] Takagi T. and M. Sugeno, "Fuzzy identification of systems and its application to modelling and control," IEEE Trans.Syst., Man, Cybern. Vol.SMC-15, pp. 116-132, Jan./Feb.**1985**.
- [33] K.Tanaka, H.Wang, *Fuzzy control systems design and analysis. A Linear Matrix inequality Approach*, A Wiley-Interscience Publication, **2001**.
- [34] K.Tanaka, M.Sugeno, "Stability analysis and design of fuzzy control systems". Fussy Sets and Systems, vol.45, pp. 135-156, **1992**.
- [35] Zeng J. X., J.A.Keane, "Approximation Capabilities of Hierarchical Fuzzy Systems", IEEE Transactions On Fuzzy Systems, Vol. 13, N°5, pp.659-672, October **2005**.

Abstract

In this first part of this work, a fuzzy generalized predictive control (FGPC) and fuzzy functional control are introduced to design a control system for a nonlinear system. A hierarchical predictive control design approach based on the Takagi–Sugeno modeling methodology is described. A classical two-layered iterative optimization process is used. In the second part, the concept of positive invariance is used to obtain necessary conditions of asymptotic stability for the fuzzy model predictive control with fuzzy objective and constraints and to derive an explicit feasible solution.

Index Terms— fuzzy logic, model predictive control, invariant set

Résumé

Dans la première partie de ce travail, nous considérons la commande prédictive généralisée et la commande prédictive fonctionnelle à base de modèle flou pour les systèmes non linéaires. Une approche hiérarchisée basée sur le modèle de Takagi–Sugeno est développée. Une optimisation itérative classique à deux couches est utilisée. Dans la seconde partie, le concept des ensembles invariants est utilisé pour obtenir des conditions nécessaires de stabilité pour la commande prédictive à base de contraintes et objectifs flous et de déterminer une solution explicite admissible.

Mots clés — Logique floue, commande prédictive à base de modèle, ensemble invariant.

ملخص

في المرحلة الأولى لهذا العمل، قدمنا التحكم التنبؤي المعمم و التحكم التنبؤي الوظيفي و ذلك باستعمال النموذج الغامض للأنظمة غير الخطية، حيث قمنا بتطوير طريقة سلمية معتمدة على النموذج الغامض. مستعملين طريقة تحسينية مكررة كلاسيكية بطبقتين .

في المرحلة الثانية ، مبدأ الحيز الثابت استعمل للحصول على شروط لازمة للاستقرار التحكم التنبؤي المعتمد على الهدف و القيد الغامضين ، و ذلك بغية الحصول على حلول واضحة و مقبولة.

الكلمات المفتاحية _ المنطق الغامض، التحكم التنبؤي بنموذج، الحيز الثابت