

République Algérienne Démocratique et Populaire  
Ministère de L'enseignement Supérieur et de la Recherche Scientifique

Université M'hamed Bougara-Boumerdès  
Faculté des Hydrocarbures Et de la Chimie

## Mémoire

Pour Obtenir le diplôme de  
*Magister en Génie Électrique*

Option,

*Automatique et Informatique Industrielle*

Présenté et soutenu publiquement par :

**M. LAMRAOUI Oualid**

Thème

---

# OPTIMISATION DE LA CONDUITE DE PROCESSUS PAR ÉMULATION FLOUE DES STRATÉGIES DE COMMANDE

---

Devant le Jury

KIDOUCHE MADJID	Professeur	Université de Boumerdès	Président
BENTARZI ABDELHAMID	Professeur	Université de Boumerdès	Examinateur
MAIDI AHMED	MC/A	Université de Boumerdès	Examinateur
HABBI HACENE	MC/A	Université de Boumerdès	Encadreur

Le 11 décembre 2014

The ink of the scholar is more holy  
than the blood of the martyr.

Prophet Muhammad

صَلَّى اللهُ عَلَيْهِ وَسَلَّمَ

## REMERCIEMENTS

*Louanges à dieu pour nous avoir permis d'avoir les capacités physiques et mentales, nous permettant de réaliser ce modeste mémoire et nous l'implorons afin que notre travail soit couronné de succès.*

Ce mémoire est le résultat d'un premier travail de recherche. Dans cet effort inédit, j'ai eu le bonheur de trouver sur ma route certaines personnes qui m'ont considérablement aidé dans mon parcours. Qu'elles en soient ici remerciées :

Merci tout d'abord à Monsieur Hacene HABBI, pour l'implication qu'il a pu avoir tout au long de ce mémoire, son suivi, ses conseils, ses directions et éclaircissements, et pour avoir su me transmettre partie de son savoir et sa passion de *l'Intelligence Artificielle*, et cela malgré son emploi du temps chargé.

Je remercie Monsieur KIDOUCHE pour l'honneur qu'il me fait en présidant le jury de ce mémoire et pour l'intérêt qu'il a porté à ce travail, ainsi les membres du jury M. BENTARZI et M. MAIDI d'avoir consacré de leur temps à examiner ma thèse malgré leurs nombreuses responsabilités.

J'exprime ma gratitude à mes deux amis MEFOUED Moumen Cherif et MOUSLI Mohamed Mounir pour leurs efforts et leurs temps consacré pour m'aider et me soutenir.

Enfin, plus que des remerciements pour mes parents qui ont supportés sautes d'humeur, et nuits blanches rythmées au son du clavier pendant ces derniers mois et qui ont su me remotiver quand le besoin s'en faisait sentir, m'aider à ne penser qu'au mémoire et qui sait, plus généralement, me rendre heureux.

## DÉDICACE

À mes très chers parents,

À mes frère et à mes sœurs.

À mes enseignants...

À mes amis et mes collègues...

À ceux qui enseignent honnêtement...

À ceux qui veulent s'instruire sans ennui...

Je dédie ce travail.

## RÉSUMÉ

La théorie de commande multivariable offre des outils intéressants pour la conception de contrôleurs se basant sur des critères d'optimalité et de robustesse. Cependant, l'application de certaines de ces méthodologies conduit dans certains cas (systèmes de grandes dimensions, synthèse  $\mathcal{H}_\infty, \dots$ ) à des réalisations d'ordre élevé nécessitant le recours à des simplifications structurelles (notamment de réduction d'ordre ou d'approximation) avant toute éventuelle implémentation. Ces simplifications ne sont bien évidemment pas sans effets. En fait, dans bon nombre de cas, des dégradations de performances ou même des situations de désadaptation du contrôleur au processus à régler peuvent être observées. L'émulation par approches alternatives inspirées de *l'intelligence Artificielle* (AI) de ce type de lois de commande multivariable conduisant à des configurations réalisables représente une piste intéressante à étudier et à mettre en œuvre. Dans le présent travail, nous nous sommes plus particulièrement intéressés aux systèmes d'inférence floue issus de la théorie de la logique floue pour étudier cette problématique.

L'objectif de ce travail est de synthétiser des systèmes de commande à base de règles floues émulant des lois de commande optimale et robuste d'ordre élevé à partir d'une base de données dont la réalisation pratique est souvent difficile voire impossible à mettre en œuvre. Pour ce faire, une méthodologie de synthèse de lois de commande multivariable basée sur la coalescence floue est proposée. L'objectif étant d'émuler le comportement dynamique d'un système de commande multivariable par la génération automatique de bases de règles floues susceptibles de le décrire à partir de données.

Afin de mettre en œuvre cette approche, deux différentes configurations de commande multivariable sont examinées : le contrôle  $\mathcal{H}_\infty$  avec loop-shaping et le contrôle LQG. Pour démontrer la performance de la méthodologie d'émulation floue proposée, une application à la commande d'un générateur de turbine à vapeur est traitée. Plusieurs simulations sont effectuées dans différentes conditions de fonctionnement.

**Mots-clés** : Émulation floue, coalescence floue, contrôleur de procédé multivariable, générateur de turbine à vapeur.

## مُلخَص

النظرية الحديثة للتحكم الأمثل و المتين تقدم أدوات جديدة بالاهتمام بهدف تصميم وحدات تحكم على أساس معايير متينة و مثالية. و مع ذلك فقد يؤدي التطبيق لهذه الوسائل في بعض الحالات (نظم كبيرة، حساب  $H_\infty$  ، الخ) إلى وحدات تحكم ضخمة تتطلب استخدام طرق التبسيط الهيكلية (نذكر تحديد النظام و الطرق التقريبية) قبل أي تطبيق. هذه التبسيطات ليست بالطبع من دون آثار، في الواقع، و في كثير من الأحيان، نلاحظ تدهور للأداء او حتى مشاكل في وحدة التحكم قد تؤدي إلى مخلفات كارثية. إستعمال المحاكات بواسطة بعض النهج المستوحات من الذكاء الإصطناعي لهذا النوع من قوانين التحكم قد يؤدي إلا تطبيقات ممكنة تبدو وسيلة تدعو للإهتمام. من بين هذه التطبيقات نذكر النظم الغامضة التي تعتبر الأكثر إستعمالاً في المجال الصناعي بسبب سهولة تطبيقها، الحجم المعقول لذاكرة التخزين، و كذا زمن استجابة نموذجي.

الهدف من هذا العمل هو تصميم نظام تحكم على أساس قواعد غامضة بهدف محاكاة قوانين التحكم المثالية و المتينة التي غالباً ما يكون من الصعب تطبيقها فعلياً و ذلك عن طريق قاعدة قياسات بإستخدام خوارزمي ترتيب غامض لحساب هذه القوئد الغامضة.

يهدف تجريب الطريقة المطروحة في هذه الدراسة، لقد قمنا بإختيار نظامي التحكم  $H_\infty$  و  $LQG$  لتصميم نظم تحكم غامضة محاكية لها، عدت إختبارات قد أجريت على النظام مولد البخار - تربين بهدف دراسة جودة و نوعية المحاكات و مقارنتها مع وحدات التحكم الأصلية.

كلمات المفتاح : المحاكات الغامضة، الترتيب الغامض، وحدات تحكم كبيرة، مولد البخار - تربين.

## ABSTRACT

This work addresses the problem of designing multivariable data-driven fuzzy controller for a complex systems. The proposed design methodology aims to construct Takagi-Sugeno-type fuzzy controllers based on learning data extracted from high-order control configurations. More precisely, fuzzy emulator of both multivariable loop-shaping  $\mathcal{H}_\infty$  robust and LQG controllers are developed using subtractive clustering technique. To check the effectiveness of the designed methodology, a control study dealing with the problem of designing multivariable control strategies for a drum-type boiler-turbine system is presented. The aim is to build a suitable approximation for LQG and  $\mathcal{H}_\infty$  controllers designed separately for the considered application. This problem is tackled using fuzzy inference systems designed through clustering techniques. Simulations are performed under different situations with taking into account the stringent control objectives imposed on boiler system operation.

**Keywords:** Fuzzy emulation, fuzzy clustering, multivariable control systems, Boiler-turbine system.

# TABLE DES MATIÈRES

Liste des figures . . . . .	x
Liste des tableaux . . . . .	xii
Introduction générale . . . . .	1
<b>I. Éléments d'analyse et de conduite des systèmes complexes . . . . .</b>	<b>3</b>
I.1 Introduction . . . . .	3
I.2 Caractérisation d'un système complexe . . . . .	4
I.3 Éléments d'analyse des systèmes complexes . . . . .	4
I.3.1 Linéarisation . . . . .	4
I.3.2 Découplage . . . . .	6
I.3.3 Matrice des Gains Relatifs (RGA) . . . . .	7
I.3.4 Comportement à non-minimum de phase . . . . .	8
I.3.5 Norme $\mathcal{H}_\infty$ . . . . .	9
I.3.5.1 Décomposition en valeurs singulières . . . . .	9
I.3.5.2 Directions d'entrée et de sortie . . . . .	9
I.3.6 Factorisation coprime . . . . .	10
I.4 Stratégies de commande multivariable . . . . .	11
I.4.1 La commande LQG . . . . .	11
I.4.1.1 Contrôle linéaire quadratique intégral (LQI) . . . . .	13
I.4.1.2 Filtre de Kalman . . . . .	13
I.4.1.3 Contrôleur LQG . . . . .	14
I.4.2 La commande $\mathcal{H}_\infty$ . . . . .	14
I.4.2.1 Stabilisation robuste . . . . .	14
I.4.2.2 Procédure de conception du contrôleur $\mathcal{H}_\infty$ . . . . .	16
I.5 Implémentation des stratégies de commande . . . . .	18
I.6 Conclusion . . . . .	19
<b>II. Systèmes de commande à base de règles floues . . . . .</b>	<b>20</b>
II.1 Introduction . . . . .	20
II.2 Systèmes de commande à inférence floue . . . . .	21
II.2.1 Concepts de base . . . . .	21
II.2.1.1 Variables linguistiques et fonctions d'appartenance . . . . .	21
II.2.1.2 Opérateurs flous . . . . .	22
II.2.1.3 Activation des règles . . . . .	23
II.2.1.4 Accumulation . . . . .	23
II.2.2 Configuration générale d'un contrôleur flou . . . . .	24
II.2.3 Principe d'un système de commande à inférence floue . . . . .	25
II.3 Modèles de systèmes flous . . . . .	26



II.3.1	Modèle flou de type Mamdani . . . . .	26
II.3.2	Modèle flou de Takagi-Sugeno . . . . .	26
II.3.3	Modèle relationnel flou . . . . .	28
II.4	Synthèse des systèmes de commande flous . . . . .	29
II.5	Algorithmes de coalescence floue . . . . .	30
II.5.1	Algorithme des k-moyennes . . . . .	31
II.5.2	Algorithme des c-moyennes floues . . . . .	32
II.5.3	Algorithme de coalescence soustractive . . . . .	33
II.6	Conclusion . . . . .	35
<b>III. Émulation par systèmes d'inférence floue de lois de commande multivariable . . . . .</b>		
III.1	Introduction . . . . .	36
III.2	Contrôle multivariable . . . . .	37
III.3	Stratégie d'émulation floue des contrôleurs multivariables . . . . .	38
III.3.1	Détermination de la structure de l'émulateur . . . . .	38
III.3.2	Identification des paramètres . . . . .	39
III.3.2.1	Identification des prémisses des règles . . . . .	39
III.3.2.2	Estimation des conséquences des règles . . . . .	40
III.3.3	Considérations sur la mise en œuvre de la stratégie d'émulation floue . . . . .	41
III.3.3.1	Choix du signal d'excitation . . . . .	41
III.3.3.2	Construction de la base d'apprentissage . . . . .	42
III.3.4	Validation de l'émulateur flou . . . . .	43
III.4	Conclusion . . . . .	43
<b>IV. Application à la commande multivariable d'un générateur de turbine à vapeur . . . . .</b>		
IV.1	Introduction . . . . .	44
IV.2	Modélisation du système GV-turbine . . . . .	45
IV.3	Commande du système GV-turbine . . . . .	47
IV.4	Synthèse des stratégies de commande multivariable . . . . .	48
IV.4.1	Commande $\mathcal{H}_\infty$ . . . . .	49
IV.4.2	Contrôle PID multivariable . . . . .	52
IV.4.3	Émulation floue du contrôleur $\mathcal{H}_\infty$ . . . . .	54
IV.4.3.1	Synthèse de l'émulateur flou . . . . .	54
IV.4.3.2	Validation de l'émulateur flou . . . . .	56
IV.4.3.3	Résultats de simulation . . . . .	57
IV.4.3.4	Analyse des résultats . . . . .	58
IV.4.4	Commande LQG . . . . .	74
IV.4.4.1	Synthèse du contrôleur linéaire quadratique . . . . .	74
IV.4.4.2	Synthèse du filtre de Kalman . . . . .	75
IV.4.4.3	Détermination du contrôleur final . . . . .	75
IV.4.5	Émulation floue du contrôleur LQG . . . . .	76
IV.4.5.1	Synthèse de l'émulateur flou . . . . .	76
IV.4.5.2	Validation de l'émulateur flou . . . . .	77
IV.4.5.3	Résultats de simulation . . . . .	78
IV.4.5.4	Analyse des résultats . . . . .	79
IV.5	Conclusion . . . . .	95

<b>Conclusion générale</b> . . . . .	96
<b>Annexe I. Signal d'excitation</b> . . . . .	97
<b>Bibliographie</b> . . . . .	100

# LISTE DES FIGURES

I.1	Configuration d'une commande par rétroaction. . . . .	6
I.2	Configuration générale de la commande LQG. . . . .	12
I.3	Structure du contrôleur LQG. . . . .	14
I.4	Problème de la stabilisation robuste $\mathcal{H}_\infty$ . . . . .	15
I.5	Système compensé en boucle. . . . .	16
I.6	Implémentation pratique de la commande $\mathcal{H}_\infty$ avec loop-shaping. . . . .	17
II.1	Exemple de fonctions d'appartenance. . . . .	22
II.2	Fonction d'appartenance de type singleton. . . . .	22
II.3	Configuration d'une boucle de commande floue. . . . .	24
II.4	Structure de base d'un contrôleur flou. . . . .	25
II.5	Raisonnement flou de Mamdani basé sur la méthode d'inférence max-min. . . . .	27
II.6	Système d'inférence floue de Takagi et Sugeno. . . . .	27
II.7	Système flou dynamique. . . . .	28
II.8	Critère d'acceptation et de rejet des centres de clusters. . . . .	34
III.1	Configuration de base d'un système multivariable en boucle fermée. . . . .	37
III.2	Signal d'excitation binaire pseudo-aléatoire. . . . .	42
IV.1	Centrale thermique à combustibles fossiles. . . . .	45
IV.2	Illustration des variables du système GV-turbine intervenant dans le modèle de Bell et Astörm. . . . .	46
IV.3	Schéma-bloc d'un actionneur de vanne. . . . .	48
IV.4	Allures des valeurs singulières du modèle nominal. . . . .	49
IV.5	Allures des valeurs singulières du système augmenté $GW_1$ . . . . .	51
IV.6	Configuration du contrôle $\mathcal{H}_\infty$ appliqué au système GV-turbine. . . . .	51
IV.7	Allures des valeurs singulières de $GK_\infty$ et $GK_{PI}$ . . . . .	53
IV.8	Schéma de configuration d'un PI avec un anti-windup. . . . .	53
IV.9	Configuration du contrôleur PI multivariable. . . . .	54
IV.10	Les ensembles flous gaussiens associés à chacune des variables d'entrée. . . . .	55
IV.11	Performance du contrôleur flou par apport au contrôleur $\mathcal{H}_\infty$ original (données d'apprentissage). . . . .	56
IV.12	Performance du contrôleur flou par apport au contrôleur $\mathcal{H}_\infty$ original (données de test). . . . .	56
IV.13	Réponses du processus non-perturbé. . . . .	59
IV.14	Réponses du processus en présence de perturbations, Cas 1. . . . .	60
IV.15	Réponses du processus en présence de perturbations, Cas 2. . . . .	61
IV.16	Réponses du processus en présence de perturbations, Cas 3. . . . .	62
IV.17	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) + 10\%$ . . . . .	63

IV.18	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) - 10\%$ .	64
IV.19	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) + 20\%$ .	65
IV.20	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) - 20\%$ .	66
IV.21	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(120, 120, 0)$ .	67
IV.22	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(150, 90, 0)$ .	68
IV.23	Réponses du processus à la demande de puissance électrique, 1 <sup>er</sup> cas.	69
IV.24	Réponses du processus à la demande de puissance électrique, 2 <sup>e</sup> cas.	70
IV.25	Réponses du processus à la demande de puissance électrique, 3 <sup>e</sup> cas.	71
IV.26	Réponses du processus au test de variations paramétriques, 1 <sup>er</sup> cas.	72
IV.27	Réponses du processus au test de variations paramétriques, 2 <sup>er</sup> cas.	73
IV.28	Les ensembles flous gaussiens associés à chacune des variables d'entrée.	77
IV.29	Performance du contrôleur flou par apport au contrôleur LQG original (données d'apprentissage).	77
IV.30	Performance du contrôleur flou par apport au contrôleur LQG original (donnés de validation).	78
IV.31	Réponses du processus non perturbé.	80
IV.32	Réponses du processus en présence de perturbations, Cas 1.	81
IV.33	Réponses du processus en présence de perturbations, Cas 2.	82
IV.34	Réponses du processus en présence de perturbations, Cas 3.	83
IV.35	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) + 10\%$ .	84
IV.36	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) - 10\%$ .	85
IV.37	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) + 20\%$ .	86
IV.38	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(r^0) - 20\%$ .	87
IV.39	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(120, 120, 0)$ .	88
IV.40	Réponses du processus suite à un changement de consigne de $\#(r^0)$ à $\#(150, 90, 0)$ .	89
IV.41	Réponses du processus à la demande de puissance électrique, 1 <sup>er</sup> cas.	90
IV.42	Réponses du processus à la demande de puissance électrique, 2 <sup>e</sup> cas.	91
IV.43	Réponses du processus à la demande de puissance électrique, 3 <sup>e</sup> cas.	92
IV.44	Réponses du processus au test de variations paramétriques, 1 <sup>er</sup> cas.	93
IV.45	Réponses du processus au test de variations paramétriques, 2 <sup>er</sup> cas.	94
A I.1	Signal d'excitation utilisé pour le contrôleur $\mathcal{H}_\infty$ .	97
A I.2	Signal d'excitation utilisé pour le contrôleur LQG.	98

# LISTE DES TABLEAUX

III.1 Exemple de combinaisons des signaux d'excitation . . . . .	42
IV.1 Points de fonctionnements typiques pour le modèle de Bell et Aström. . .	48
IV.2 Caractéristiques des actionneurs. . . . .	48
IV.3 Paramètres des prémisses de l'émulateur flou du contrôleur $\mathcal{H}_\infty$ . . . . .	55
IV.4 Paramètres des prémisses de l'émulateur flou du contrôleur LQG. . . . .	77
A I.1 Combinaisons des signaux d'excitation utilisées pour le contrôleur $\mathcal{H}_\infty$ . .	98
A I.2 Combinaisons des signaux d'excitation utilisées pour le contrôleur LQG. .	99

# INTRODUCTION GÉNÉRALE

Depuis plusieurs décennies, la théorie de commande des systèmes ne cesse de réaliser d'importants progrès, tant sur le plan fondamental que sur le plan des applications. Cette évolution est devenue de plus en plus marquée surtout avec l'avènement de nouvelles techniques d'identification et de commande de processus, en l'occurrence les techniques de l'Intelligence Artificielle telles que la logique floue, les réseaux de neurones artificiels, les techniques d'optimisation méta-heuristiques, etc. L'intégration de ces techniques dites « *Intelligentes* » en automatique a considérablement contribué à la mise en œuvre d'une multitude de stratégies de commande qui se fondent, pour quelques unes d'entre-elles, sur la théorie de commande classique. En effet, il est parfois judicieux d'envisager à mettre en œuvre des mécanismes qui assistent les stratégies de commande classique ou multivariable installées plutôt que de les remplacer intégralement.

La théorie de commande multivariable constitue l'une des plus importantes bases de l'automatique. Depuis son introduction, la manipulation de systèmes de production complexes est rendue plus aisée par le développement de nouveaux outils de conception basés sur le concept de « *système multivariable* ». Ainsi, des approches d'analyse et de commande multivariables se sont introduites, dédiées dans la plupart à l'appréhension et l'optimisation de la conduite des systèmes de grande dimension ou plus généralement les systèmes complexes.

Parmi ces techniques, on peut éventuellement citer les méthodes de commande optimale et de commande robuste telle que la commande linéaire quadratique (LQ) et la synthèse  $\mathcal{H}_\infty$ . Ces stratégies de commande multivariable sont dotées d'outils intéressants et performants pour la conception de contrôleurs multivariables basée sur des critères d'optimalité et de robustesse. Cependant, l'application de ces méthodologies conduit pour certaines classes de systèmes à des réalisations d'ordre élevé nécessitant le plus souvent le recours à des simplifications structurelles (notamment de réduction d'ordre) avant toute éventuelle implémentation pratique. Les transformations opérées sur les synthèses originales peuvent être très concluantes pour certaines applications, comme elles peuvent induire des effets non négligeables sur les performances des boucles de commande en provoquant, par exemple, d'éventuelles dégradations.

L'émulation par systèmes d'inférence floue des lois de commande multivariable constitue la problématique abordée dans ce mémoire. L'objectif étant de développer une méthodologie de conception basée sur la logique floue qui permet d'émuler efficacement le comportement dynamique de systèmes de commande multivariable d'ordre élevé dont l'implémentation pratique est difficile, voire impossible à mettre en œuvre.

Le présent travail revient sur les différentes étapes de développement d'une telle approche d'émulation floue et retrace un certain nombre de considérations pour son

implémentation. Une étude en simulation traitant le problème de commande multivariable d'un générateur de turbine à vapeur est présentée afin de démontrer l'utilité et la performance de la méthodologie de commande présentée.

Le mémoire est composé de quatre (04) chapitres.

Le premier chapitre décrit brièvement quelques éléments d'analyse et de conduite des systèmes multivariables. Nous y présentons certaines méthodes de synthèse de lois de commande multivariable telles que la commande linéaire quadratique gaussienne (LQG) et la commande  $\mathcal{H}_\infty$ . La problématique d'implémentation de ce type de commande est également soulevée, une méthode d'approximation conduisant à une réalisation à base de contrôleurs PID est explicitée.

Le chapitre II introduit des notions et des outils de manipulation des systèmes d'inférence floue. Les principales méthodes d'obtention de ces modèles y sont présentées.

Le chapitre III est consacrée intégralement à la méthodologie d'émulation floue. On y présente les principales orientations et les étapes de développement d'émulateurs flous de lois de commande multivariable.

Le chapitre IV est dédié à la validation de la méthodologie développée sur un problème de commande multivariable d'un générateur de turbine à vapeur, où l'on propose de concevoir des systèmes de commande floue à partir de l'émulation des contrôleurs LQG et  $\mathcal{H}_\infty$ , en vue d'optimiser la conduite du système non-linéaire.

# CHAPITRE I

## Éléments d'analyse et de conduite des systèmes complexes

### I.1 Introduction

Dans la théorie des systèmes, la notion de système complexe est souvent perçue comme étant multiforme, supportant une définition plutôt vague et ramifiée. Cependant, tenter simplement de réussir une première définition nous renvoie à l'évidence qu'un système complexe est souvent difficile à comprendre, modéliser et contrôler en utilisant les méthodes traditionnelles. Cependant, plusieurs chercheurs s'accordent sur un certain nombre de phénomènes qui peuvent caractériser un système complexe. Parmi ces caractéristiques, on peut citer la présence de dynamiques non-linéaires, de variations paramétriques ou structurelles, d'incertitudes ou encore d'interconnexions. D'un point de vue structurel, un système complexe peut être composé de plusieurs sous-systèmes interconnectés qui influencent mutuellement l'évolution de ses variables. Dans certains cas, l'effet du couplage domine la dynamique du système, d'autant plus que les couplages puissent être non-linéaires. Une source supplémentaire de complexité est la grande dimension des espaces d'état et des paramètres qui conduit à des représentations d'ordre élevé auxquelles seront logiquement associées des configurations de commande d'ordre supérieur.

Quelque soit la caractérisation que l'on associe à un système complexe donné, il est clair que ce dernier est conçu pour servir un but et avoir un certain comportement. La présence de phénomènes complexes complique éventuellement l'analyse et la conduite du système. Devant ce constat, il revient à la stratégie de commande adoptée (selon un cahier des charges) d'assurer un fonctionnement désirable, même en présence de ces phénomènes. Une bonne conduite est généralement synonyme d'une bonne maîtrise du système et de ses conditions d'exploitation ; un objectif que l'on vise toujours à atteindre à travers la mise en place de configurations de commande diverses, à commencer par les traditionnels régulateurs PID et les lois de commande à retour d'état, jusqu'à même tenter des mécanismes plus élaborés tels que le contrôle optimal et le contrôle robuste. D'autres nouvelles techniques et contributions plus récentes sont également développées, dans le seul souci de défier les problèmes liés à la conduite des systèmes complexes.

Dans ce chapitre, nous commençons par illustrer quelques éléments liés à l'analyse



des systèmes complexes. Ensuite, nous présentons deux différentes méthodologies de commande, à savoir la commande  $\mathcal{H}_\infty$  pour la catégorie des commandes robustes et la commande quadratique pour la catégorie des commandes optimales. Les limitations de leur mise en œuvre y seront également discutées.

## I.2 Caractérisation d'un système complexe

Pour avancer une première définition d'un système complexe, il nous est avéré convenable de rappeler celle de Newman [37] qui stipule que « *A complex system is a system composed of many interacting parts, which displays collective behaviour that does not follow trivially from the behaviours of the individuals.* ». Cette définition s'applique aux systèmes d'un large éventail de disciplines scientifiques car en fait les sciences de la complexité sont nécessairement fondées sur des recherches interdisciplinaires. Cette définition grossière fait principalement allusion aux éventuelles interconnexions et interactions qui peuvent se présenter pour former une entité ou structure à caractère globalement non-linéaire au sein duquel un ensemble de sous-systèmes évoluent pour accomplir une tâche spécifique.

Pour caractériser cette complexité, il est important de soulever les éléments suivants :

- **L'ordre d'un système** : C'est un indicateur important de complexité, qui est étroitement lié à la dimension des espaces d'état et des paramètres, ainsi qu'au nombre de sous-systèmes qui le forment. Cet élément caractéristique est déterminant quant à la configuration de commande que le concepteur doit adopter pour assurer sa conduite. En effet, certaines configurations multivariables intègrent des contrôleurs dont l'ordre est plus élevé que celui des systèmes commandés.
- **La non-linéarité** : Elle est de sources diverses. Elle peut par exemple traduire la conséquence d'observer un changement radical dans l'évolution du système suite à de petits changements dans ses variables. D'autres phénomènes peuvent expliquer le caractère non-linéaire d'un système [28].
- **Les interactions** : Les systèmes complexes sont composés essentiellement de plusieurs sous-systèmes *interconnectés* mutuellement, l'interaction peut traduire soit un échange de l'énergie, de la matière ou de l'information.
- **Les incertitudes** : Elles sont de type structurel ou paramétrique pour parler d'incertitudes de modèle, et peuvent être liées aux imprécisions et aux incomplétudes dans les connaissances disponibles. La présence d'incertitudes pose de sérieux problèmes à l'appréhension même du système complexe [40].

## I.3 Éléments d'analyse des systèmes complexes

### I.3.1 Linéarisation

La modélisation des systèmes nous montre que la plupart des systèmes que l'on considère comme étant complexes possèdent *naturellement* une dynamique non-linéaire. Par

conséquent, les équations différentielles régissant l'évolution des variables du système ne sont pas linéaires. La linéarisation d'un système non-linéaire autour d'un point de fonctionnement spécifique a pour but de simplifier sa représentation pour ramener son analyse à un domaine restreint conformément aux exigences de conduite requises [38, 41].

Considérons la représentation d'état à temps continu non-linéaire suivante :

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), t) \\ y(t) = g(x(t), u(t), t) \end{cases} \quad (\text{I.1})$$

où,  $x(t)$  le vecteur d'état de dimension  $n_s$ ,  $u(t)$  le vecteur de commande de dimension  $n_u$ ,  $y(t)$  désigne le vecteur de sortie de dimension  $n_y$ , et  $f$  et  $g$  sont des fonctions vectorielles réelles non-linéaires sur  $\mathfrak{R}^{n_s} \times \mathfrak{R}^{n_u} \times \mathfrak{R} \rightarrow \mathfrak{R}^{n_s}$  et  $\mathfrak{R}^{n_s} \times \mathfrak{R}^{n_u} \times \mathfrak{R} \rightarrow \mathfrak{R}^{n_y}$ , respectivement.

Autour d'un point d'équilibre  $(x_0, u_0)$  donné, le modèle (I.1) peut être réécrit sous la forme :

$$\begin{cases} \delta\dot{x} = f((x_0 + \delta x), (u_0 + \delta u), t) \\ \delta y = g((x_0 + \delta x), (u_0 + \delta u), t) - y_0 \end{cases} \quad (\text{I.2})$$

où  $y_0$  est la valeur de sortie initiale.

Le développement en série de *Taylor* appliqué aux équations (I.2) en négligeant les ordres supérieurs à 1 permet d'écrire :

$$\begin{cases} \delta\dot{x} = \left. \frac{\partial f}{\partial x} \right|_{\substack{x=x_0 \\ u=u_0}} \delta x + \left. \frac{\partial f}{\partial u} \right|_{\substack{x=x_0 \\ u=u_0}} \delta u \\ \delta y = \left. \frac{\partial g}{\partial x} \right|_{\substack{x=x_0 \\ u=u_0}} \delta x + \left. \frac{\partial g}{\partial u} \right|_{\substack{x=x_0 \\ u=u_0}} \delta u \end{cases} \quad (\text{I.3})$$

On aboutit alors à une représentation d'état non stationnaire de la forme :

$$\begin{cases} \delta\dot{x} = A(t) \delta x + B(t) \delta u \\ \delta y = C(t) \delta x + D(t) \delta u \end{cases} \quad (\text{I.4})$$

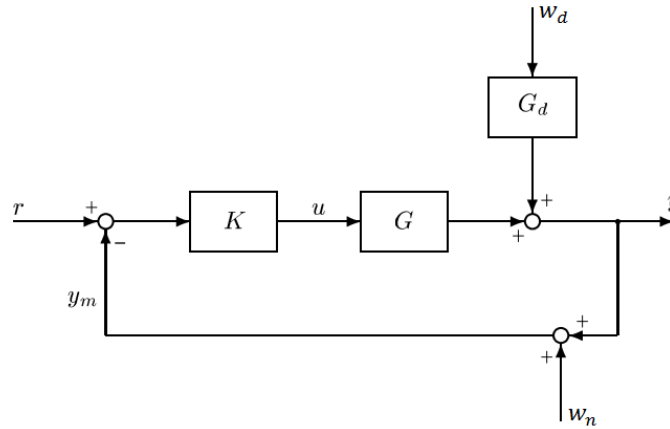
avec  $A(t)$ ,  $B(t)$ ,  $C(t)$  et  $D(t)$  des matrices de dimensions appropriées et qui sont calculées par :

$$\begin{aligned} A(t) &= \left. \frac{\partial f}{\partial x} \right|_{\substack{x=x_0 \\ u=u_0}} , & B(t) &= \left. \frac{\partial f}{\partial u} \right|_{\substack{x=x_0 \\ u=u_0}} \\ C(t) &= \left. \frac{\partial g}{\partial x} \right|_{\substack{x=x_0 \\ u=u_0}} , & D(t) &= \left. \frac{\partial g}{\partial u} \right|_{\substack{x=x_0 \\ u=u_0}} \end{aligned} \quad (\text{I.5})$$

Le système linéaire variant dans le temps (I.4) se ramène à un système linéaire stationnaire (Invariant dans le temps) lorsque les matrices  $A(t) = A$ ,  $B(t) = B$ ,  $C(t) = C$  et  $D(t) = D$  sont alors constantes.

### I.3.2 Découplage

Dans la perspective de commander un système dont l'état est caractérisé par plusieurs variables, le but du découplage est de transformer la fonction de transfert ou la représentation d'état afin de pouvoir commander chaque sortie indépendamment des autres [22, 44]. Considérons un système en boucle fermée dont le schéma est présenté par la Figure I.1.



**Figure I.1** Configuration d'une commande par rétroaction.

Dans cette figure, on note les éléments suivants :

- $G$  modèle du processus à contrôler,
- $G_d$  modèle de perturbation,
- $K$  contrôleur,
- $r$  entrée de référence,
- $w_d$  perturbation,
- $w_n$  bruit de mesure,
- $y$  sortie du système,
- $y_m$  sortie mesurée,
- $u$  signal de commande.

En introduisant un pré-compensateur  $W_1(s)$ , la transmittance directe s'écrit :

$$G_s(s) = G(s)W_1(s) \quad (\text{I.6})$$

Nous obtenons un *découplage* lorsque le compensateur  $W_1$  est choisi de telle sorte que  $G_s = GW_1$  est diagonale sur une plage de fréquences choisies. Les différents cas possibles pour un découpleur sont les suivants :

- a. **Découpleur dynamique** : Dans ce cas  $G_s(s)$  est une matrice de transfert diagonale pour toutes les fréquences. Par exemple avec  $G_s(s) = I$  et  $G(s)$  est une matrice de transfert carrée, nous obtenons  $W_1 = G^{-1}(s)$ .
- b. **Découpleur statique** : Dans ce cas,  $G_s(0)$  est diagonale, que l'on pourra obtenir par la sélection d'un pré-compensateur constant  $W_1 = G^{-1}(0)$ .

- c. **Découplage approximatif à une fréquence**  $\omega_0$  :  $G_s(j\omega_0)$  est diagonale que possible, généralement par le choix d'un pré-compensateur constant  $W_1 = G_0^{-1}$  où  $G_0$  est une approximation réelle de  $G(j\omega_0)$ . Nous pouvons calculer  $G_0$ , par exemple en utilisant l'algorithme de *Kouvaritakis* [41]. Un bon choix consiste à prendre  $\omega_0$  dans la bande passante du système car l'effet sur la performance en réduisant les interactions est appréciable.

L'idée d'utiliser un découpleur est intéressante, mais il y a plusieurs difficultés [41] :

- Le découplage peut être très sensible aux erreurs de modélisation et aux incertitudes.
- L'obligation de découplage et l'utilisation d'un contrôleur à base d'inverseur peut ne pas être souhaitable pour compenser les perturbations.
- Si le système possède des zéros instables alors l'utilisation d'un découpleur introduit généralement des zéros instables supplémentaires dans la boucle fermée.

### I.3.3 Matrice des Gains Relatifs (RGA)

La matrice de gain relatif (RGA) est une mesure fréquemment utilisée dans le contrôle des systèmes multivariables [41, 10]. Nous supposons que  $u_j$  et  $y_i$  désignent une entrée et une sortie particulière pour le système multivariable  $G(s)$ , et que notre tâche est d'utiliser  $u_j$  pour contrôler  $y_i$ . Deux cas extrêmes peuvent se présenter :

- Les autres boucles sont ouvertes : Toutes les autres entrées sont constantes ;  $u_k = 0$ ,  $\forall k \neq j$ .
- Les autres boucles sont fermées : Toutes les autres sorties sont constantes ;  $y_k = 0$ ,  $\forall k \neq i$ .

Dans ce dernier cas, il est supposé que les autres boucles sont fermées avec un contrôle parfait. Un contrôle parfait est possible seulement à l'état d'équilibre, mais ceci est valable pour des fréquences appartenant à la bande passante de chaque boucle. Par l'évaluation de l'effet  $\partial y_i / \partial u_j$  sur l'entrée  $u_j$  et la sortie  $y_i$ , pour les deux cas précédents, on écrit :

- Les autres boucles sont ouvertes :

$$\left( \frac{\partial y_i}{\partial u_j} \right)_{u_k=0, k \neq j} \triangleq g_{ij}. \quad (\text{I.7})$$

- Les autres boucles sont fermées :

$$\left( \frac{\partial y_i}{\partial u_j} \right)_{y_k=0, k \neq i} \triangleq \hat{g}_{ij}. \quad (\text{I.8})$$

Notons ici que  $g_{ij} = [G]_{ij}$  est l'élément de l'ordre  $ij$  dans la matrice  $G$ , tandis que  $\hat{g}_{ij}$  est l'inverse de l'élément de l'ordre  $ji$  de la matrice  $G^{-1}$  ;  $\hat{g}_{ij} = 1 / [G^{-1}]_{ji}$ .

Le rapport entre les gains en (I.7) et (I.8), correspondant aux deux cas extrêmes, est une mesure utile de l'interaction. Ce rapport représente bien le gain relatif de l'ordre  $ij$  défini comme suit :

$$\lambda_{ij} \triangleq \frac{g_{ij}}{\hat{g}_{ij}} = [G]_{ij} [G^{-1}]_{ji} \quad (\text{I.9})$$

On définit alors la matrice des gains relatifs d'une matrice carrée inversible  $G$  par :

$$\text{RGA}(G) = \Lambda(G) \triangleq G \times (G^{-1})^T \quad (\text{I.10})$$

où  $\times$  représente la multiplication élément par élément (le produit de Hadamard ou de Schur).

Les propriétés de la RGA pour l'analyse des systèmes de commande sont nombreuses. Citons à présent certaines d'entre-elles :

- Les systèmes dont les éléments de la RGA possèdent de grandes valeurs ( $\gg 1$ ) autour de la fréquence de coupure sont fondamentalement difficiles à contrôler en raison de la sensibilité aux entrées incertaines. En particulier, un découpleur ne devrait pas être utilisé dans ce cas.
- Si le signe d'un élément de la RGA change de  $s = 0$  à  $s = \infty$ , alors il y a un zéro instable dans le système.
- La définition de la RGA peut être généralisée à des matrices non carrées en utilisant la pseudo-inverse.

### I.3.4 Comportement à non-minimum de phase

Un système est dit à minimum de phase, si tous ses pôles et ses zéros se trouvent dans la partie gauche du plan complexe. En conséquence, un système est dit à non-minimum de phase (NMP), s'il possède des zéros situés dans le demi-plan droit [21]. Souvent les zéros situés dans le demi-plan droit sont appelés des zéros instables. Les systèmes à non-minimum de phase ont un grand retard de phase, ce qui explique le nom « non-minimum de phase ».

La conception de contrôleurs pour les systèmes à non-minimum de phase présente plusieurs difficultés et limites. Dans certains cas, des modifications sur la manipulation de processus (variables d'entrée, variables des sortie...) nécessitent d'être effectuées. Dans ce qui suit, on soulève quelques points importants :

- **Réponse dynamique inverse** : Dans ce cas, le système montre une réponse dans la direction opposée à l'endroit où initialement il finit par se stabiliser, ce qui implique une augmentation inévitable du temps de réponse.
- **Risque d'instabilité avec les gains élevés** : Il est bien connu que si le gain de rétroaction augmente vers l'infini, les pôles en boucle fermée migrent vers les positions des zéros en boucle ouverte. Ainsi, la présence de zéros instables va impliquer l'instabilité du système.
- **Limitation de bande passante** : Les zéros instables imposent une limite maximale sur la bande passante désirable. En plus, ils limitent les performances désirables de la boucle de rétroaction en limitant le produit des gains en boucle ouverte dans la bande passante.

### I.3.5 Norme $\mathcal{H}_\infty$

L'objectif principal, généralement retenu en commande de processus, est d'atteindre certaines spécifications de performance en assurant la stabilité interne du système. Les performances désirées peuvent être exprimées en terme de normes établies pour certains signaux. Afin de définir la norme la plus appropriée à la situation traitée (la commande  $\mathcal{H}_\infty$ ), il convient d'abord de présenter l'espace robuste  $\mathcal{H}_\infty$  (Hardy Space) [16, 9]. Pour toutes fonctions complexes  $G(s)$  analytiques et bornées dans le demi-plan droit ( $Re(s) > 0$ ) de  $\mathfrak{C}^{n_y \times n_u}$  associant la variable complexe  $s = j\omega$ , il existe un nombre réel  $b \in \mathfrak{R}$  tel que,

$$\|G(s)\| \leq b, \quad Re(s) > 0 \quad (\text{I.11})$$

La valeur minimale de la borne  $b$  représente ce que nous appelons la norme  $\mathcal{H}_\infty$  de  $G$ , notée  $\|G\|_\infty$ , et d'une manière équivalente,

$$\|G\|_\infty \triangleq \sup_{\omega} \{\|G(s)\| : Re(s) > 0\} \quad (\text{I.12})$$

Nous disons alors qu'une fonction complexe  $G$  est définie dans l'espace  $\mathcal{H}_\infty$ , si elle est analytique et bornée sur  $Re(s) > 0$  [47].

#### I.3.5.1 Décomposition en valeurs singulières

Toute matrice  $G(j\omega)$  peut être décomposée en valeurs singulières [46], et on écrit :

$$G = Y\Sigma U^H \quad (\text{I.13})$$

où  $\Sigma$  est une matrice de dimension  $n_y \times n_u$  avec  $k = \min(n_y, n_u)$  valeurs singulières non négatives  $\sigma_i$  placées dans un ordre décroissant le long de sa diagonale principale. Les valeurs singulières sont les racines carrées positives des valeurs propres  $\lambda_i$  de  $G^H G$ ;  $G^H$  est la transposée conjuguée complexe de  $G$ .

$$\sigma_i(G(j\omega)) = \sqrt{\lambda_i(G^H G)} \quad (\text{I.14})$$

$Y$  est une matrice unitaire de  $n_y \times n_y$  qui représente les vecteurs singuliers associés aux sorties,  $y_i$ .  $U$  est une matrice unitaire de  $n_u \times n_u$  qui représente les vecteurs singuliers associés aux entrées  $u_i$ .

#### I.3.5.2 Directions d'entrée et de sortie

Les colonnes de  $Y$ , noté  $y_i$ , représentent les directions de sortie du système. Elles sont orthogonales et de longueur unitaire (Orthonormales), soit :

$$\begin{cases} \|y_i\| = 1 \\ y_i^H y_i = 1, y_i^H y_j = 0 \text{ avec } i \neq j \end{cases} \quad (\text{I.15})$$

De même, les colonnes de  $U$ , notées  $u_i$ , sont orthogonales et d'une longueur unitaire, et représentent les directions d'entrée. Ces directions d'entrée et de sortie sont reliées par les valeurs singulières selon l'équation :

$$Gu_i = \sigma_i y_i \quad (\text{I.16})$$

Par conséquent, on obtient :

$$\sigma_i(G) = \|Gu_i\| = \frac{\|Gu_i\|}{\|u_i\|} \quad (\text{I.17})$$

Le plus grand gain pour n'importe quelle direction d'entrée est égal à la valeur singulière maximale

$$\bar{\sigma}(G) \equiv \sigma_1(G) = \max_{d \neq 0} \frac{\|Gd\|}{\|d\|} = \frac{\|Gu_1\|}{\|u_1\|} \quad (\text{I.18})$$

Le plus petit gain pour toute direction d'entrée est égal à la valeur singulière minimale. Exception faite pour les matrices avec un nombre d'entrées supérieur au nombre de sorties ( $n_u > n_y$ ), où nous pouvons toujours choisir une entrée différente de zéro  $d$  dans l'espace nul de  $G$  tel que  $Gd = 0$ . On écrit alors :

$$\underline{\sigma}(G) \equiv \sigma_k(G) = \min_{d \neq 0} \frac{\|Gd\|}{\|d\|} = \frac{\|Gu_k\|}{\|u_k\|} \quad (\text{I.19})$$

où  $k = \min\{n_y, n_u\}$ . Pour n'importe quel vecteur  $d$ , nous avons :

$$\underline{\sigma}(G) \leq \frac{\|Gd\|}{\|d\|} \leq \bar{\sigma}(G(j\omega)) \quad (\text{I.20})$$

**Definition.** Soit  $G$  une matrice de transfert stable. La norme  $\mathcal{H}_\infty$  de  $G$  est définie par :

$$\|G\|_\infty = \sup_{\omega \in \Re} \{\bar{\sigma}(G)\}$$

Notons que  $\bar{\sigma}(G(j\omega)) \equiv \bar{\sigma}(\omega)$  peut être utilisée pour l'interprétation du gain maximal du système pour une fréquence  $\omega$ . C'est pour cette raison que le graphe de  $\bar{\sigma}(\omega)$  avec  $\omega \in \Re$  peut être regardé comme étant une généralisation du diagramme de Bode pour les systèmes multivariables.

### I.3.6 Factorisation coprime

La factorisation coprime (FC) peut représenter un système dans le domaine temporel (Représentation d'état) et dans le domaine fréquentiel (Matrice de transfert). Nous définissons la FC droite du système  $G(s)$  comme étant :

$$G(s) = N_r(s)M_r^{-1}(s) \quad (\text{I.21})$$

avec  $(N_r(s), M_r(s)) \in \mathcal{H}_\infty \times \mathcal{H}_\infty$ ,  $M_r(s)$  est carrée. La stabilité implique que  $N_r$  devrait contenir tous les zéros instables de  $G$ , et  $M_r$  devrait contenir tous les pôles instables de  $G$ . Ceci signifie qu'il existe  $(U_r, V_r) \in \mathcal{H}_\infty \times \mathcal{H}_\infty$  satisfaisant l'identité de Bezout

$$U_r N_r + V_r M_r = I \quad (\text{I.22})$$

De la même façon, une factorisation coprime gauche de  $G$  peut être définie comme étant,

$$G(s) = M_l^{-1}(s)N_l(s) \quad (\text{I.23})$$

où  $(M_l(s), N_l(s)) \in \mathcal{H}_\infty \times \mathcal{H}_\infty$ , et  $M_l(s)$  est carrée. Ceci signifie qu'il existe  $(U_l, V_l) \in \mathcal{H}_\infty \times \mathcal{H}_\infty$  satisfaisant l'identité de *Bezout*

$$N_l U_l + M_l V_l = I \quad (\text{I.24})$$

La factorisation coprime d'un système  $G(s)$  n'est pas unique, on introduit alors la notion de factorisation coprime normalisée. Soit l'opérateur  $M^*$  défini par  $M^*(s) = M^T(-s)$ .  $G = N_r M_r^{-1}$  est appelée factorisation coprime normalisée si,

$$M_r^* M_r + N_r^* N_r = I \quad (\text{I.25})$$

Dans le cas d'une réalisation d'état minimale, i.e.

$$G \stackrel{s}{=} \left[ \begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] \quad (\text{I.26})$$

La réalisation minimale de l'espace d'état d'une factorisation coprime gauche normalisée est donnée par l'équation (I.27),

$$[N_l(s)M_l(s)] \stackrel{s}{=} \left[ \begin{array}{c|cc} A + HC & B + HD & H \\ \hline R^{-1/2}C & R^{-1/2}D & R^{-1/2} \end{array} \right] \quad (\text{I.27})$$

avec

$$H \triangleq -(BD^T + ZC^T)R^{-1}, \quad R \triangleq I + DD^T \quad (\text{I.28})$$

La matrice  $Z$  est la solution positive unique de l'équation algébrique de Riccati qui s'exprime par :

$$(A - BS^{-1}D^T C)Z + Z(A - BS^{-1}D^T C)^T - ZC^T R^{-1}CZ + BS^{-1}B^T = 0 \quad (\text{I.29})$$

et

$$S \triangleq I + D^T D. \quad (\text{I.30})$$

## I.4 Stratégies de commande multivariable

### I.4.1 La commande LQG

La commande optimale, en s'appuyant sur les travaux de filtrage optimal de Wiener [41, 2], a atteint la maturité dans les années 1960 avec l'introduction de la commande linéaire quadratique gaussienne (LQG). Pour la synthèse de la commande LQG, il est nécessaire d'obtenir un modèle linéaire qui représente le processus autour d'un point de fonctionnement donné. Soit le modèle linéaire suivant [41] :

$$\begin{cases} \dot{x} = A x + B u + w_d \\ y = C x + D u + w_n \end{cases} \quad (\text{I.31})$$

où  $w_d$  et  $w_n$  désignent les perturbations et les bruits de mesure, respectivement. Ces grandeurs sont généralement supposés non corrélés, stochastiques, et gaussiennes de moyenne nulle ayant pour densités spectrales les matrices constantes  $W$  et  $V$ , respectivement, c'est-à-dire :



$$\begin{aligned}
 E\{w_d(t)w_d(\tau)^T\} &= W\delta(t - \tau) \\
 E\{w_n(t)w_n(\tau)^T\} &= V\delta(t - \tau)
 \end{aligned}
 \tag{I.32}$$

et

$$E\{w_d(t)w_n(\tau)^T\} = 0, \quad E\{w_n(t)w_d(\tau)^T\} = 0
 \tag{I.33}$$

où  $E$  est l'opérateur d'espérance et  $\delta(t - \tau)$  est la fonction delta.

Le problème de la commande LQG formulé dans ce contexte est de trouver la commande optimale  $u(t)$  qui minimise le critère,

$$J = E \left\{ \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T [x^T Q x + u^T R u + x_i^T Q_i x_i] dt \right\}
 \tag{I.34}$$

où  $Q$ ,  $R$  et  $Q_i$  sont des matrices de pondération constantes choisies de façon appropriée (paramètres de conception) telle que  $Q = Q^T \geq 0$ ,  $R = R^T > 0$  et  $Q_i = Q_i^T \geq 0$ .  $x_i$  est l'intégrale de l'erreur de poursuite  $r - y$ .

La solution au problème LQG est connue sous le théorème de séparation. La démarche consiste à déterminer d'abord le contrôle optimal au problème linéaire quadratique intégral (LQI). La solution à ce problème peut être écrite en termes d'une loi de commande par retour d'état :

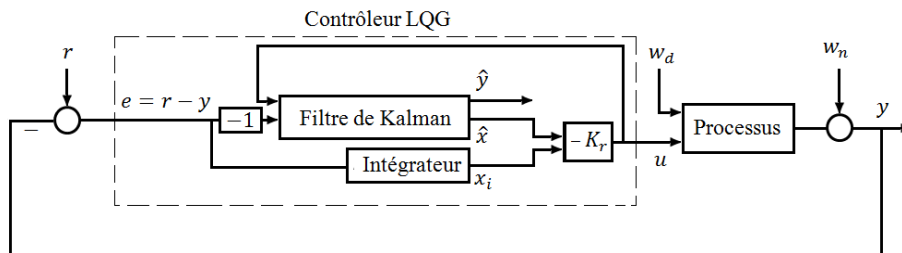
$$u(t) = -K_r \begin{bmatrix} x(t) \\ x_i(t) \end{bmatrix}
 \tag{I.35}$$

avec  $K_r$  est une matrice constante. L'étape suivante consiste à trouver un estimateur optimal  $\hat{x}$  pour le vecteur d'état  $x$ , de sorte que le critère  $E\{[x - \hat{x}]^T [x - \hat{x}]\}$  soit minimisé.

L'estimateur optimal des états du système est donné par un filtre de Kalman, qui s'exprime indépendamment de  $Q$ ,  $R$  et  $Q_i$ . La solution nécessaire pour le problème LQG est obtenue alors en remplaçant  $x$  par  $\hat{x}$ , ce qui donne

$$u(t) = -K_r \begin{bmatrix} \hat{x}(t) \\ x_i(t) \end{bmatrix}
 \tag{I.36}$$

La Figure I.2 illustre le schéma d'une configuration de commande LQG. Les étapes de synthèse sont décrites par la suite.



**Figure I.2** Configuration générale de la commande LQG.

### I.4.1.1 Contrôle linéaire quadratique intégral (LQI)

Étant donné le système suivant, où tous les états sont connus :

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (\text{I.37})$$

et

$$\dot{x}_i = r - y \quad (\text{I.38})$$

À partir des équations (I.37) et (I.38), on établit le modèle augmenté suivant :

$$\begin{bmatrix} \dot{x} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} B \\ -D \end{bmatrix} u + \begin{bmatrix} 0 \\ I \end{bmatrix} r \quad (\text{I.39})$$

Les matrices d'évolution et de commande du système augmenté sont alors données par :

$$A_a = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}, \quad B_a = \begin{bmatrix} B \\ -D \end{bmatrix} \quad (\text{I.40})$$

Le problème à résoudre est celui de la recherche d'une commande optimale  $u(t)$  qui permet au système de suivre une référence  $r$  en minimisant le critère :

$$J_{LQI} = \int_0^\infty \left[ \begin{bmatrix} x^T & x_i^T \end{bmatrix} Q \begin{bmatrix} x \\ x_i \end{bmatrix} + u^T R u \right] dt \quad (\text{I.41})$$

La solution optimale (pour tout état initial) est donnée par :  $u(t) = -K_r \begin{bmatrix} x \\ x_i \end{bmatrix}$ , où

$$K_r = R^{-1} B_a^T X \quad (\text{I.42})$$

avec  $X = X^T$  est l'unique solution réelle et semi-définie positive de l'équation algébrique de Riccati

$$A_a^T X + X A_a - X B_a R^{-1} B_a^T X + Q = 0 \quad (\text{I.43})$$

### I.4.1.2 Filtre de Kalman

Le filtre de Kalman a la structure d'un estimateur d'état ordinaire ou d'un observateur [41, 23], comme le montre la Figure I.3,

$$\dot{\hat{x}} = A\hat{x} + (B - K_f D)u + K_f(e - (C\hat{x} + Du)) \quad (\text{I.44})$$

Le choix optimal de  $K_f$ , qui minimise le critère  $E\{[x - \hat{x}]^T [x - \hat{x}]\}$  est donné par :

$$K_f = Y C^T V^{-1} \quad (\text{I.45})$$

où  $Y = Y^T \geq 0$  est la solution unique de l'équation algébrique de Riccati

$$Y A_a^T + A_a Y - Y C^T V^{-1} C^T Y + W = 0 \quad (\text{I.46})$$

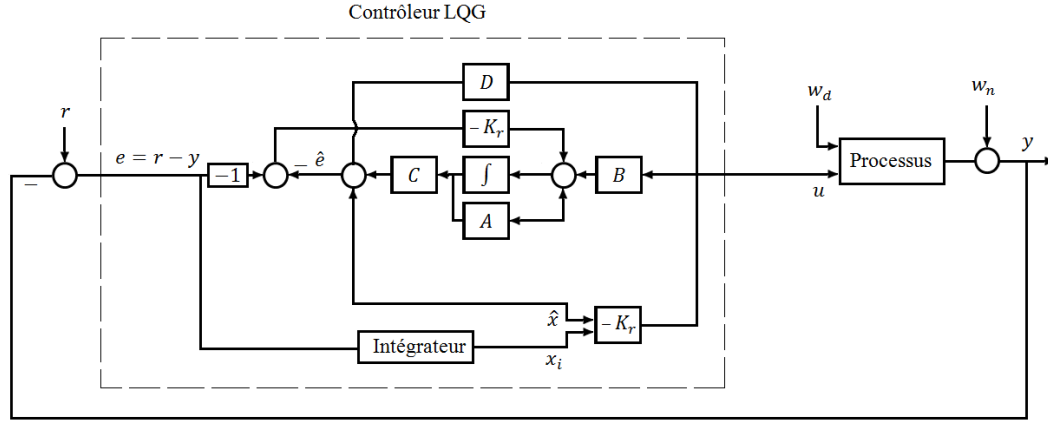


Figure I.3 Structure du contrôleur LQG.

### I.4.1.3 Contrôleur LQG

La solution finale du problème de commande LQG résultant des deux étapes de synthèse précédentes conduit à un contrôleur linéaire donné par la matrice de fonction de transfert suivante :

$$K_{LQG}(s) = \left[ \begin{array}{cc|c} A - BK_x - K_f C + K_f D K_x & -BK_i + K_f D K_i & -K_f \\ 0 & 0 & I \\ \hline -K_x & -K_i & 0 \end{array} \right] \quad (\text{I.47})$$

avec  $K_r = [K_x \ K_i]$

Notons que l'existence des matrices de gain  $K_f$  et  $K_r$  conduisant à une loi de commande LQG stable est conditionnée par la stabilisabilité et la détectabilité des réalisations d'état  $(A_a, B_a, Q^{1/2})$  et  $(A, W^{1/2}, C)$  [41].

## I.4.2 La commande $\mathcal{H}_\infty$

La commande  $\mathcal{H}_\infty$  avec la mise en forme de la boucle (loop-shaping) a été proposée par McFarlane et Glover en 1990 [41, 15, 19]. Elle est basée sur la théorie de stabilisation robuste  $\mathcal{H}_\infty$  combinée avec la procédure classique de loop-shaping. Il s'agit essentiellement d'un processus de conception en deux étapes. Tout d'abord, le système en boucle ouverte est augmenté avec un pré et un post-compensateur pour donner la forme désirée pour les valeurs singulières de la réponse fréquentielle en boucle ouverte. Ensuite, le système résultant est stabilisé en utilisant l'optimisation  $\mathcal{H}_\infty$ .

### I.4.2.1 Stabilisation robuste

Nous allons considérer la stabilisation du système  $G$  auquel on associe une factorisation coprime gauche. Soit :

$$G = M^{-1}N \quad (\text{I.48})$$

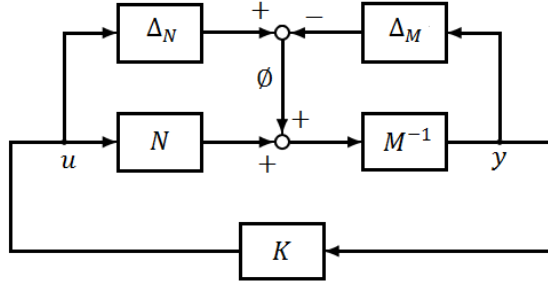
Le modèle du système perturbé  $G_p$  peut alors s'écrire comme suit :

$$G_p = (M + \Delta_M)^{-1}(N + \Delta_N) \quad (\text{I.49})$$

où  $\Delta_M$ ,  $\Delta_N$  sont des fonctions de transfert inconnues qui représentant l'incertitude dans le modèle nominal  $G$ . L'objectif de la stabilisation robuste est de stabiliser non seulement le modèle nominal  $G$ , mais une famille de systèmes perturbés définis par :

$$G_p = \left\{ (M + \Delta_M)^{-1}(N + \Delta_N) : \|[\Delta_N \ \Delta_M]\|_\infty < \epsilon \right\} \quad (\text{I.50})$$

où  $\epsilon > 0$  désigne la marge de stabilité. Le problème de la stabilisation robuste est de maximiser cette marge de stabilité pour des systèmes représentés par des factorisations coprimes normalisées.



**Figure I.4** Problème de la stabilisation robuste  $\mathcal{H}_\infty$ .

Pour le système en boucle fermée de la Figure I.4, la propriété de robustesse en stabilité est garantie si et seulement si le système nominal en boucle fermée est stable et

$$\gamma \triangleq \left\| \begin{bmatrix} K \\ I \end{bmatrix} (I - GK)^{-1} M^{-1} \right\|_\infty \leq \frac{1}{\epsilon} \quad (\text{I.51})$$

où  $\gamma$  est la norme  $\mathcal{H}_\infty$  de  $\phi$  à  $\begin{bmatrix} u \\ y \end{bmatrix}$  et  $(I - GK)^{-1}$  est la fonction de sensibilité du système en boucle fermée.

D'après [41], la marge de stabilité maximale est obtenue pour une valeur minimale de  $\gamma$ , soit

$$\gamma_{min} = \epsilon_{max}^{-1} = (1 + \rho(XZ))^{\frac{1}{2}} \quad (\text{I.52})$$

où  $\rho$  désigne le rayon spectral (valeur propre maximale). Pour une réalisation d'état minimale de  $G$ ,  $Z$  est la solution unique et définie positive de l'équation algébrique de Riccati

$$(A - BS^{-1}D^T C)Z + Z(A - BS^{-1}D^T C)^T - ZC^T R^{-1} CZ + BS^{-1}B^T = 0 \quad (\text{I.53})$$

où  $R = I + DD^T$ ,  $S = I + D^T D$  et  $X$  est la solution unique et définie positive de l'équation algébrique de Riccati

$$(A - BS^{-1}D^T C)^T X + X(A - BS^{-1}D^T C) - XBS^{-1}B^T X + C^T R^{-1} C = 0 \quad (\text{I.54})$$

Le contrôleur pour lequel

$$\left\| \begin{bmatrix} K \\ I \end{bmatrix} (I - GK)^{-1} M^{-1} \right\|_{\infty} \leq \gamma \quad (\text{I.55})$$

pour tout  $\gamma > \gamma_{min}$ , s'exprime par la représentation d'état suivante

$$K \stackrel{s}{=} \left[ \begin{array}{c|c} A + BF + \gamma^2(L^T)^{-1}ZC^T(C + DF) & \gamma(L^T)^{-1}ZC^T \\ \hline B^T X & -D^T \end{array} \right] \quad (\text{I.56})$$

avec

$$F = -S^{-1}(D^T C + B^T X) \quad (\text{I.57})$$

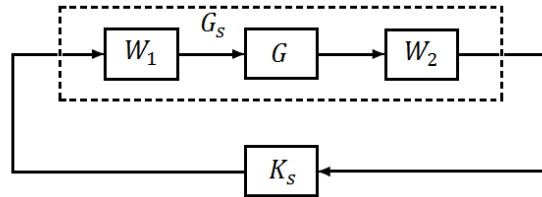
$$L = (1 - \gamma^2)I + XZ \quad (\text{I.58})$$

#### I.4.2.2 Procédure de conception du contrôleur $\mathcal{H}_{\infty}$

La stabilisation robuste nécessite une phase de conditionnement préalable à opérer sur le modèle du système. Pour ce faire, McFarlane et Glover (1990) ont proposé d'introduire des pré- et post-compensations dans le système pour ajuster les allures des valeurs singulières en boucle ouverte avant de réaliser la stabilisation robuste.

Désignons par  $W_1$  et  $W_2$  les pré- et post-compensateurs à introduire dans la boucle. Le système compensé  $G_s$  est donné par

$$G_s = W_2 G W_1 \quad (\text{I.59})$$



**Figure I.5** Système compensé en boucle.

Comme le montre la Figure I.5, le contrôleur  $K_s$  est synthétisé en résolvant le problème de la stabilisation robuste du système compensé  $G_s$  avec une factorisation coprime gauche normalisée  $G_s = M_s^{-1}N_s$ . Le contrôleur à concevoir pour le système  $G$  sera  $K = W_1 K_s W_2$ .

Le choix des pré- et post-compensateurs  $W_1$  et  $W_2$  est une étape primordiale et déterminante quant à la performance du contrôleur [41]. En se basant sur plusieurs études une procédure systématique a été établie pour la synthèse d'un contrôleur  $\mathcal{H}_{\infty}$  robuste. On peut la résumer aux étapes suivantes :

- **Étape 1** : Soit  $W_1 = W_p W_a W_g$ , sélectionner les pré- et post-compensateurs diagonaux  $W_p$  et  $W_2$  de sorte que les valeurs singulières de  $W_2 G W_p$  répondent aux

exigences de synthèse spécifiées. Plus précisément, on opte souvent pour un gain élevé dans les basses fréquences, un roll-off d'environ  $20dB$  (une pente d'environ  $-1$ ) dans la bande passante désirée, avec un roll-off plus élevé à des fréquences élevées.  $W_2$  est généralement choisi comme une constante, il reflète l'importance relative des sorties à contrôler.  $W_p$  contient la mise en forme dynamique : l'action intégrale, pour une performance en basses fréquences ; une avance de phase pour réduire le roll-off à croisement ; et un retard de phase pour augmenter le roll-off en hautes fréquences. Toutes ces spécifications doivent être placées dans  $W_p$  si désiré. Notons que le paramétrage des pré- et post-compensateurs doit être effectué tout en évitant d'introduire des modes instables dans  $G_s$ .

- **Étape 2** : Aligner les valeurs singulières à une largeur de bande désirée en utilisant un nouveau poids constant  $W_a$  cascadié avec  $W_p$ . Il s'agit d'un découpleur constant qui ne devrait pas être employé si le système présente un grand RGA (voir Section I.3.3).
- **Étape 3** : Dans certains cas, une matrice de pondération  $W_g$  cascadiée avec  $W_a$  peut être introduite pour assurer un contrôle sur les actionneurs.  $W_g$  est diagonale et ajustée de sorte que les limites des taux de variation des commandes ne soient pas dépassées pour des changements de référence et/ou des perturbations agissant sur les sorties du système.
- **Étape 4** : Stabilisation robuste du système compensé  $G_s = W_2GW_1$ , où  $W_1 = W_pW_aW_g$ , tel que décrit dans la Section I.4.2.1. D'abord, calculer la marge de stabilité maximale  $\epsilon_{max} = 1/\gamma_{min}$ . Si la marge est trop petite,  $\epsilon < 0.25$ , alors on passe à l'étape 1 et on modifie les matrices de pondération.

La Figure I.6 illustre la configuration pratique d'une commande  $\mathcal{H}_\infty$  avec mise en forme de la boucle. Comme on peut le constater, un filtre est introduit dans la boucle de commande afin d'éviter les grands dépassements qui peuvent survenir lorsque les références excitent directement la dynamique de  $K_s$ . Ce filtre constant est appelé à maintenir un gain statique unité en présence d'intégrateur dans  $G$  ou  $W_1$ .

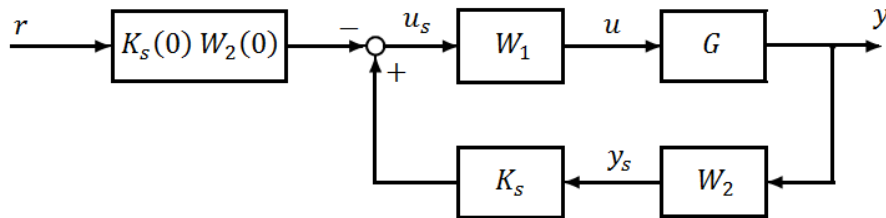


Figure I.6 Implémentation pratique de la commande  $\mathcal{H}_\infty$  avec loop-shaping.

## I.5 Implémentation des stratégies de commande

Pour des raisons pratiques, on cherche souvent à concevoir des solutions de conduite optimale employant des contrôleurs à structure (réalisation) minimale. En effet, l'implémentation effective d'une loi de commande est limitée par la taille et par le temps de calcul d'autant plus qu'un contrôleur simple est plus facile à mettre en œuvre et à comprendre [31].

Comme montré précédemment, les contrôleurs LQG et  $\mathcal{H}_\infty$  sont déterminés à partir des modèles augmentés ou généralisés, l'ordre de ces contrôleurs est généralement plus élevé que celui du système commandé. La difficulté de mettre en œuvre des solutions de commande d'ordre élevé est sérieusement posée dans un grand nombre d'applications industrielles. Dans de nombreux cas, le dispositif de commande d'ordre supérieur peut être approximé par un contrôleur de structure simplifiée ou d'ordre réduit, tout en admettant un recul de performances plus ou moins acceptable. Étant donné que 90% des contrôleurs industriels réellement installés sont de type PID, il est donc rationnel de constater que dans la plupart des cas, des approximations de type PID sont envisagées pour rendre possible l'implémentation des contrôleurs d'ordre élevé [20, 43]. Nous présentons dans la suite une des méthodes d'approximation employée dans ce sens.

Considérons un contrôleur  $K(s)$  représenté dans l'espace d'état par le modèle dynamique :

$$\begin{cases} \dot{x} &= A_k x + B_k y \\ u &= C_k x + D_k y \end{cases} \quad (\text{I.60})$$

Le problème de la recherche d'une approximation PID pour  $K(s)$  peut se ramener à celui de la recherche d'une transformation  $T$  telle que :

$$T A_k T^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & a_2 \end{bmatrix} \quad (\text{I.61})$$

où  $a_2$  est une matrice constante inversible et non singulière. Cette transformation peut être calculée en utilisant la décomposition de la matrice  $A_k$  en éléments propres.

Sur la base de  $T$ , les matrices  $C_k$  et  $B_k$  sont décomposées comme suit :

$$C_k T = [c_1 \ c_2], \quad T^{-1} B_k = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad (\text{I.62})$$

Une approximation de la forme  $K_{PID}(s) = K_p + \frac{K_i}{s} + K_d s$ , peut être obtenue en tronquant l'expansion de McLaurain du contrôleur [26],

$$\begin{aligned} K(s) &= C_k (sI - A_k)^{-1} B_k + D_k \\ &= [c_1 \ c_2] \left( sI - \begin{bmatrix} 0 & 0 \\ 0 & a_2 \end{bmatrix} \right)^{-1} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} + D_k \\ &= \frac{c_1 b_1}{s} + (D_k - c_2 a_2^{-1} b_2) - (c_2 a_2^{-2} b_2) s + O(|s|) \end{aligned} \quad (\text{I.63})$$

L'approximation PID pour  $K(s)$  dans les basses fréquences s'obtient alors en établissant les actions proportionnelle, intégrale et dérivée selon les équations :

$$K_p = (D_k - c_2 a_2^{-1} b_2), K_i = c_1 b_1, K_d = -(c_2 a_2^{-2} b_2) \quad (\text{I.64})$$

Un paramétrage final du contrôleur PID doit souvent être opéré après la procédure de réduction. Il peut être effectué selon les points suivants :

- Afin d'assurer un minimum de phase, les signes des gains proportionnels, intégraux et dérivatifs devraient être les mêmes.
- L'action dérivée doit être associée à un filtre de la forme  $\frac{1}{1 + \alpha s}$ .
- Selon le cas étudié, il est préférable d'éliminer l'action dérivée, afin d'éviter toute nuisance à la stabilité du système commandé.
- Les petits termes dans les matrices des gains proportionnels, intégraux et dérivatifs, ainsi que les termes non-réalisables peuvent être négligés.

La validation de l'approximation PID est une étape primordiale qui doit obligatoirement suivre l'étape de synthèse. Elle a pour objectif d'évaluer les performances du contrôleur réduit et les comparer avec celles du contrôleur d'ordre élevé.

## I.6 Conclusion

Ce chapitre passe en revue certains éléments et notions utilisés en analyse des systèmes et retrace quelques méthodes de synthèse de lois de commande optimale et robuste. Le point fondamental soulevé concerne la mise en œuvre effective de ces stratégies de commande pour des systèmes d'ordre élevé. Dans certaines configurations, les contrôleurs optimaux ou robustes résultants possèdent des réalisations d'ordre élevé nécessitant une étape de réduction ou d'approximation avant toute démarche d'implémentation. A titre d'illustration, une méthode d'approximation conduisant à un contrôle de type PID est examinée. Les limites d'une telle approche peuvent être rapidement dégagées, car il s'agit tout d'abord d'un calcul approché qui influence directement les performances du contrôleur original. On pourra éventuellement réaliser une bonne approximation du contrôleur dans des conditions bien spécifiques. Cependant, une dégradation totale des performances peut être observée, ce qui nécessite un réajustement du contrôleur réduit. C'est ainsi que l'on introduit la position du problème d'implémentation des lois de commande d'ordre élevé, une problématique qui sera traitée le long de ce mémoire sur la base du concept de système d'inférence floue.



# CHAPITRE II

## Systemes de commande à base de règles floues

### II.1 Introduction

La logique floue, en tant que théorie à part entière, s'est graduellement dotée d'outils et de concepts permettant la manipulation et le traitement de connaissances vagues, exprimées linguistiquement selon un raisonnement purement approché. Le raisonnement *humain* qui projette en fait *la connaissance humaine* n'est rien d'autre qu'une image de ce *raisonnement approché* que l'on tente à formuler en utilisant la logique floue. D'un point de vue perceptuel, la connaissance humaine sur un problème technique particulier peut être classée en deux catégories : la connaissance *consciente* et la connaissance *inconsciente*. La connaissance consciente peut être exprimée explicitement dans les mots, tandis que la connaissance inconsciente réfère à des situations où les experts savent quoi faire mais ne peuvent pas les exprimer exactement en mots. Cette classification a permis d'établir deux orientations distinctes lors de la formulation de la connaissance. La première approche est systématique, car il suffit de demander aux experts d'exprimer leurs connaissances en termes de règles linguistiques IF-THEN sur la base desquelles un système flou peut être conçu. La deuxième approche nécessite l'emploi de mécanismes appropriés pour tout d'abord structurer la connaissance inconsciente, avant de passer aux étapes de formulation et de traitement. C'est ainsi que l'on peut introduire la problématique générale de conception des systèmes à base de règles floues, et des contrôleurs flous en particulier. Selon le type de connaissances disponibles, le problème de synthèse des systèmes de commande floue peut se ramener d'une part à une procédure de formulation du protocole de commande à partir de connaissances qualitatives et/ou quantitatives (connaissance consciente). D'autre part, la conception de contrôleurs flous peut être formulée comme un problème d'extraction de connaissances à partir de données numériques (mesures entrée/sortie), ce qui traduit la manipulation de l'autre catégorie de connaissance, en l'occurrence la connaissance inconsciente.

Ce chapitre est dédié à l'étude des systèmes de commande à base de règles floues. Nous reviendrons plus précisément sur les différentes notions et concepts liés à la commande par logique floue, la configuration des contrôleurs flous et quelques approches de synthèse.

## II.2 Systèmes de commande à inférence floue

### II.2.1 Concepts de base

La logique floue a été inventée par Zadeh en 1965 [7, 34] comme une extension de la logique booléenne. Alors que la logique classique attribue à une variable, soit la valeur 1 pour *vrai* ou la valeur 0 pour *faux*, la logique floue permet d'assigner à une variable une valeur dans l'intervalle  $[0, 1]$ . Cette extension est motivée par la nature du raisonnement humain que l'on exprime souvent d'une manière vague et incertaine.

Afin d'illustrer la notion de *raisonnement approché*, un exemple simple sera introduit. La logique floue permet d'exprimer les relations d'un processus sous forme de règles. Par exemple, la règle suivante formule l'effet de refroidissement éolien, dont la température ressentie par une personne ne dépend pas uniquement de la température réelle de l'environnement, mais aussi du vent :

$$\text{If } \textit{température} = \textit{basse} \textit{ and } \textit{vent} = \textit{fort} \textbf{ Then } \textit{sensation} = \textit{très froid} \quad (\text{II.1})$$

où les composantes de cette règle sont désignées par :

<i>température, vent, sensation</i>	variables linguistiques,
<i>basse, haute, très froid</i>	termes linguistiques,
<i>and</i>	opérateur, ici : conjonction,
<i>température = basse, vent = fort</i>	valeurs linguistiques,
<i>température = basse and vent = fort</i>	prémisse de la règle,
<i>sensation = très froid</i>	conséquence de la règle.

Les variables linguistiques *température* et *vent* sont souvent désignées comme des entrées et la variable linguistique *sensation* comme une sortie. Un ensemble complet de règles peut être établi à partir de toutes les combinaisons de *température* et de *vent* :

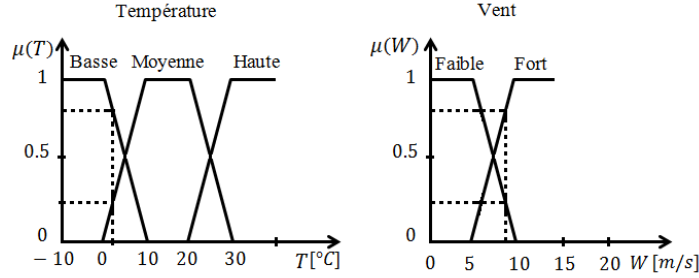
$$\begin{aligned}
 \text{R1 : If } \textit{temp.} = \textit{basse} \quad \textit{and} \quad \textit{vent} = \textit{fort} \quad \textbf{Then} \quad \textit{sens.} = \textit{très froid} \\
 \text{R2 : If } \textit{temp.} = \textit{moyenne} \quad \textit{and} \quad \textit{vent} = \textit{fort} \quad \textbf{Then} \quad \textit{sens.} = \textit{froid} \\
 \text{R3 : If } \textit{temp.} = \textit{haute} \quad \textit{and} \quad \textit{vent} = \textit{fort} \quad \textbf{Then} \quad \textit{sens.} = \textit{moyen} \\
 \text{R4 : If } \textit{temp.} = \textit{basse} \quad \textit{and} \quad \textit{vent} = \textit{faible} \quad \textbf{Then} \quad \textit{sens.} = \textit{froid} \\
 \text{R5 : If } \textit{temp.} = \textit{moyenne} \quad \textit{and} \quad \textit{vent} = \textit{faible} \quad \textbf{Then} \quad \textit{sens.} = \textit{moyen} \\
 \text{R6 : If } \textit{temp.} = \textit{haute} \quad \textit{and} \quad \textit{vent} = \textit{faible} \quad \textbf{Then} \quad \textit{sens.} = \textit{chaud}
 \end{aligned} \quad (\text{II.2})$$

Le nombre total de règles (ici 6) dépend de la finesse ou de la résolution choisie des ensembles flous. De toute évidence, la précision du système flou dépend de cette propriété, qui est appelée **granularité**. Il a été montré qu'un système flou peut approximer toute relation d'entrée/sortie à un degré de précision arbitraire si la granularité est réduite ; en d'autres termes les systèmes flous sont des approximateurs universels [36].

#### II.2.1.1 Variables linguistiques et fonctions d'appartenance

Dans la base des règles, les variables linguistiques sont exprimées sous la forme d'ensembles flous. Dans l'exemple ci-dessus, la *température* et le *vent* sont des variables

linguistiques d'entrée et sont étiquetées par les termes linguistiques *basse*, *moyenne*, *haute*, *faible* et *fort*. Ces termes linguistiques sont définis par leurs fonctions d'appartenance (MFs). La Figure II.1 montre une définition possible de ces fonctions d'appartenance. Ces MFs définissent le degré d'appartenance d'une *température* ou du *vent* aux ensembles flous. Cette procédure, qui calcule pour une entrée donnée, le degré d'appartenance aux ensembles flous est appelée la **fuzzification**.

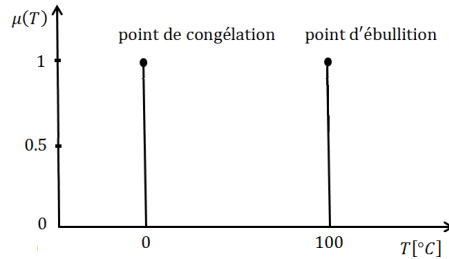


**Figure II.1** Exemple de fonctions d'appartenance.

Elle peut être vue comme une transformation non-linéaire des entrées. Une autre propriété intéressante des MFs réside dans le fait que les degrés d'appartenance aux ensembles flous sont établis de manière à remplir la condition :  $\mu_{basse}(T) + \mu_{moyenne}(T) + \mu_{haute}(T) = 1$ , et  $\mu_{faible}(W) + \mu_{fort}(w) = 1$ , ou plus généralement :

$$\sum_{i=1}^M \mu_i(u) = 1 \quad \text{pour tout } u, \quad (\text{II.3})$$

où  $M$  désigne le nombre de MFs pour la variable linguistique  $u$ . Bien qu'il n'est pas nécessaire que les MFs soient normalisées, cette propriété est souvent utilisée car elle facilite l'interprétation des déductions. Dans la Figure II.1, les fonctions d'appartenances sont de forme triangulaire et trapézoïdale, d'autres types de MFs peuvent être utilisés, à l'exemple de la fonction singleton montrée en Figure II.2.



**Figure II.2** Fonction d'appartenance de type singleton.

### II.2.1.2 Opérateurs flous

Les opérateurs de logique floue sont une extension des opérateurs booléens. Quelques définitions sont données dans la suite.

L'opérateur de *négation* d'une proposition linguistique «  $T = basse$  » est défini par :

$$Not(\mu_i(T)) = 1 - \mu_i(T) \quad (II.4)$$

Pour la *conjonction* de deux propositions linguistiques «  $T = basse$  » et «  $W = fort$  », des t-normes sont utilisées dont les plus courantes sont :

$$\begin{aligned} \text{Min :} & \quad \mu_i(T) \text{ and } \mu_j(W) = \min[\mu_i(T), \mu_j(W)] \\ \text{Produit :} & \quad \mu_i(T) \text{ and } \mu_j(W) = \mu_i(T)\mu_j(W) \\ \text{Différence bornée :} & \quad \mu_i(T) \text{ and } \mu_j(W) = \max[0, \mu_i(T) + \mu_j(W) - 1] \end{aligned} \quad (II.5)$$

Pour la disjonction de deux propositions linguistiques, des t-conormes (ou s-normes) sont employées à l'exemple de :

$$\begin{aligned} \text{Max :} & \quad \mu_i(T) \text{ or } \mu_j(W) = \max[\mu_i(T), \mu_j(W)] \\ \text{Somme algébrique :} & \quad \mu_i(T) \text{ or } \mu_j(W) = \mu_i(T) + \mu_j(W) - \mu_i(T)\mu_j(W) \\ \text{Somme bornée :} & \quad \mu_i(T) \text{ or } \mu_j(W) = \max[1, \mu_i(T) + \mu_j(W)] \end{aligned} \quad (II.6)$$

### II.2.1.3 Activation des règles

À l'aide d'opérateurs flous, il est possible de combiner les degrés d'appartenance de toutes les propositions linguistiques de la prémisse de la règle. En reprenant l'exemple de la Figure II.1, pour  $T = 3^\circ C$  et  $W = 11m/s$ , on obtient les degrés d'appartenance suivants :  $\mu_{basse}(T) = 0.7$ ,  $\mu_{moyenne}(T) = 0.3$ ,  $\mu_{haute}(T) = 0$ ,  $\mu_{faible}(W) = 0.4$ , et  $\mu_{fort}(W) = 0.6$ . Ainsi, pour la règle R1 la conjonction entre  $\mu_{basse}(T) = 0.7$  et  $\mu_{fort}(W) = 0.6$  donnerait 0.6 pour l'opérateur *min*, 0.42 pour l'opérateur *produit*, et 0.30 pour l'opérateur de *différence bornée*. Il est évident que le résultat d'un système flou dépend fortement du choix spécifique des opérateurs. La combinaison des degrés d'appartenance de toutes les propositions linguistiques est appelée *degré d'activation de la règle*, car il exprime la façon dont une prémisse de la règle correspond à une valeur d'entrée spécifique (ici  $T = 3^\circ C$  et  $W = 11m/s$ ). Notons que seules les règles avec un degré d'activation supérieur à zéro sont pertinentes, toutes les autres sont inactives. Lors de la conception du système flou, il faut veiller à ce que la totalité de l'espace d'entrée possible soit couvert par des règles afin d'éviter la situation où toutes les règles floues sont inactives.

### II.2.1.4 Accumulation

Après que le degré d'activation est calculé pour toutes les règles, les conséquences doivent être évaluées et accumulées pour générer une sortie pour le système flou. Pour la plupart des applications, cette sortie représente généralement un sous-ensemble flou, qui doit être *défuzzifié* afin d'obtenir une valeur de sortie numérique. Notons que la défuzzification n'est pas nécessaire si la sortie du système flou est utilisée comme entrée pour un autre système flou (règle de chaînage dans les systèmes hiérarchiques flous) ou si elle est directement présentée à un opérateur. Par exemple, la sortie sensation ne peut pas être facilement quantifiée car elle est, d'une mesure qualitative subjective, et donc la défuzzification pour une valeur numérique n'est pas nécessairement raisonnable,

puisque la procédure exacte pour ces dernières étapes de déduction floue dépend du type spécifique des conséquences des règles floues.

## II.2.2 Configuration générale d'un contrôleur flou

Le succès de la commande floue trouve une grande partie de son origine dans sa capacité à traduire une stratégie de contrôle d'un opérateur qualifié en un ensemble de règles linguistiques IF-THEN facilement interprétables. L'utilisation de la commande floue est particulièrement intéressante lorsqu'on ne dispose pas de modèle mathématique précis du processus à commander ou lorsque ce dernier présente de trop fortes non-linéarités ou imprécisions.

En général, les contrôleurs flous sont utilisés dans les structures de commande en boucle fermée (Figure II.3). Les variables de sortie du système à commander et les consignes définissent les variables d'entrée du contrôleur flou. Les variables de sortie du contrôleur flou sont les commandes appliquées au processus. La mise en échelle (normalisation/dénormalisation) des grandeurs d'entrée/sortie permet d'adapter le traitement des signaux. Par convention, la plage de variation des variables d'entrée/sortie est comprise entre  $-1$  et  $+1$ . Les opérations de normalisation et dénormalisation sont optionnelles.

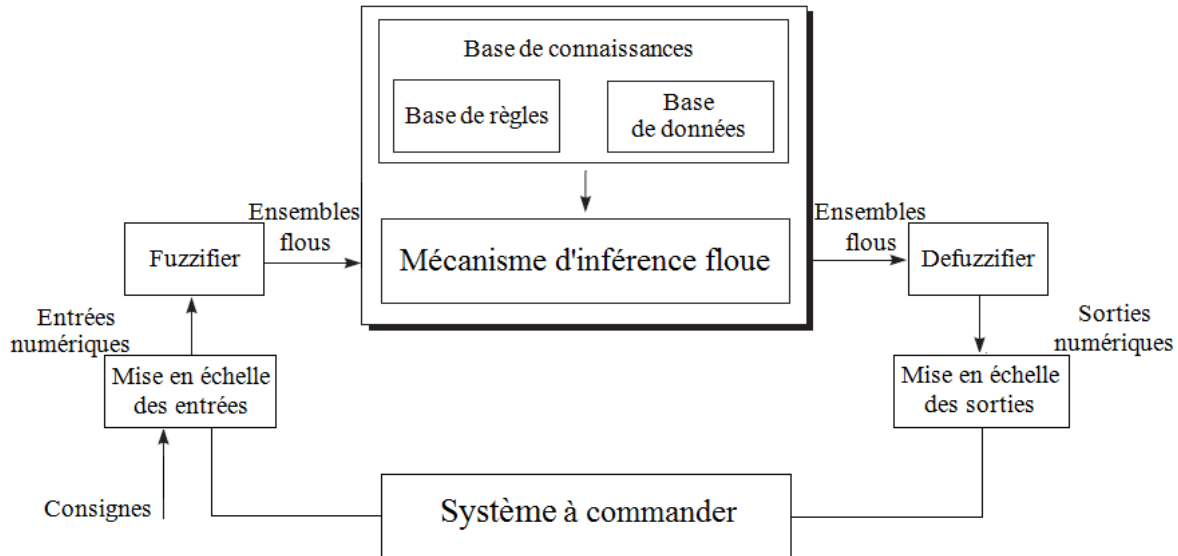


Figure II.3 Configuration d'une boucle de commande floue.

La mise en œuvre d'une commande floue fait apparaître trois grands modules (Figure II.3). Le premier module traite les entrées du système. On définit tout d'abord un univers de discours, un partitionnement de cet univers en classes pour chaque entrée, et des fonctions d'appartenance pour chacune de ces entrées. La première étape, appelée *fuzzification*, consiste à attribuer à la valeur réelle de chaque entrée, au temps  $t$ , sa fonction d'appartenance à chacune des classes préalablement définies, donc à transformer l'entrée réelle en un sous-ensemble flou. Le deuxième module consiste en l'application de règles de type IF-THEN. Ces règles vont permettre de passer d'un degré d'appartenance d'une entrée au degré d'appartenance d'une commande. Ce module est constitué d'une *base de règles* et d'un *mécanisme d'inférence* qui permet le calcul. Le troisième

est le dernier module, il décrit l'étape de *défuzzification* qui est l'inverse de la première transformation. Il permet de passer d'un degré d'appartenance d'une commande à la détermination de la valeur à donner à cette commande.

### II.2.3 Principe d'un système de commande à inférence floue

Les contrôleurs flous sont des systèmes conçus à partir de connaissances basées sur des concepts de sous-ensembles flous, les règles floues IF-THEN, et le raisonnement flou. Un système flou typique, se compose de quatre éléments : la base de connaissances ou la base des règles floues, le mécanisme d'inférence, l'interface de fuzzification, et l'interface de défuzzification [17]. La Figure II.4 montre le schéma de principe d'un système flou [17].

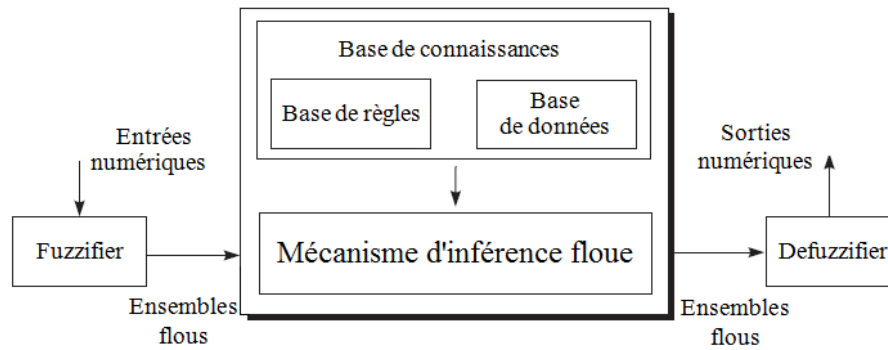


Figure II.4 Structure de base d'un contrôleur flou.

La base de connaissances dans un système flou contient les règles floues IF-THEN et une base de données. La base de données est la partie déclarative de la base de connaissances qui décrit la définition des ensembles flous et la partition de l'espace flou d'entrée et de sortie. La base des règles rassemble l'ensemble des règles floues de type IF-THEN décrivant linguistiquement le protocole de commande établi par un expert. Le mécanisme d'inférence, qui est l'élément central d'un système flou, se chargera d'appliquer chacune des règles floues. L'interface de fuzzification est un mécanisme pour transformer une valeur d'une variable réelle en un ensemble flou. Il est bien connu que la valeur crisper d'une entrée peut être transformée en un ensemble flou sous la forme d'un singleton. D'autre part, l'interface de défuzzification est un mécanisme permettant de transformer un ensemble flou sur un univers de discours de sortie à une variable d'une valeur réelle. L'objectif de défuzzification est d'extraire une valeur crisper qui interprète le mieux l'ensemble flou. Il existe plusieurs méthodes pour défuzzifier, parmi les plus utilisées, on peut citer la méthode du centre de gravité, qui est définie par :

$$z^* = \frac{\int \mu_s(z)zdz}{\int \mu_s(z)dz} \quad (\text{II.7})$$

où  $\mu_s(z)$  est la fonction d'appartenance de sortie agrégée. L'avantage de la méthode du centre de gravité est sa plausibilité intuitive, tandis que l'inconvénient est son coût élevé de calcul. Pour réduire la charge de calcul de cette méthode, d'autres stratégies de défuzzification ont été développées pour des applications spécifiques, telles que le bissecteur de la surface, la moyenne des maximums, le plus grand des maximums [36, 45].

Il a été démontré qu'un système flou est en fait une transformation non-linéaire de son espace d'entrée à son espace de sortie à partir d'un ensemble de règles floues IF-THEN linguistiques.

## II.3 Modèles de systèmes flous

Les modèles flous sont classés en trois classes particulières selon leur nature structurelle : les modèles flous de type Mamdani, les modèles flous de type Takagi-Sugeno et les modèles relationnels flous. Ces trois types de systèmes flous sont largement utilisés dans les applications de commande floue. Il convient de noter que les principales différences entre les trois systèmes flous se situent dans les conséquences de leurs règles floues.

### II.3.1 Modèle flou de type Mamdani

Le système flou de Mamdani a été proposé comme première tentative pour contrôler une machine à vapeur avec une chaudière [36]. Une règle floue typique dans un système flou de Mamdani a la forme :

$$\mathbf{IF} \ x_1 \text{ is } A_1, \dots \text{ and } x_k \text{ is } A_k \ \mathbf{THEN} \ u \text{ is } B \quad (\text{II.8})$$

où  $x = (x_1, \dots, x_k)$  et  $u$  sont des variables linguistiques d'entrée et de sortie, respectivement.  $A_1, \dots, A_k$  sont des ensembles flous de l'antécédent, et  $B$  est un ensemble flou de la conséquence. Le raisonnement dans système flou de Mamdani est basé sur la méthode d'inférence max-min. La Figure II.5 présente une illustration de cette méthode d'inférence floue avec deux règles. En employant différents opérateurs pour les normes et les conormes triangulaires, on pourra établir d'autres mécanismes d'inférence pour ce type de systèmes flous [17].

### II.3.2 Modèle flou de Takagi-Sugeno

Un autre type de systèmes flous, connu sous le nom de système flou de Takagi-Sugeno (TS), a été proposé par Takagi et Sugeno en 1985 [17], dans le but de développer une approche systématique pour l'approximation d'une fonction non-linéaire. Une règle floue typique d'un système flou TS a la forme :

$$\mathbf{IF} \ x_1 \text{ is } A_1, \dots \text{ and } x_k \text{ is } A_k \ \mathbf{THEN} \ u = f(x_1, \dots, x_k) \quad (\text{II.9})$$

où  $x = (x_1, \dots, x_k)$  et  $u$  sont des variables linguistiques d'entrée et de sortie, respectivement.  $A_1, \dots, A_k$  sont des ensembles flous de l'antécédent, et  $u = f(x_1, \dots, x_k)$  est un polynôme de la variable d'entrée  $x$ , qui peut être n'importe quelle fonction décrivant la sortie du système à l'intérieur de la région déterminée par l'antécédent de la règle. Lorsque  $f = (x_1, \dots, x_k)$  est un polynôme du premier ordre, le modèle flou résultant est alors dit du premier ordre. Ce dernier a été initialement proposé par Takagi et Sugeno (1985) et Sugeno et Kang (1988) [17]. Si  $f$  est une fonction constante, le système flou TS est alors d'ordre zéro, qui peut être considéré comme un cas particulier du système flou Mamdani, où la conséquence de chaque règle est un singleton. La sortie d'un système flou TS est obtenue par la moyenne pondérée des sorties crispes de règles floues. La Figure II.6 donne un exemple d'inférence floue pour un système flou TS avec deux règles [17].

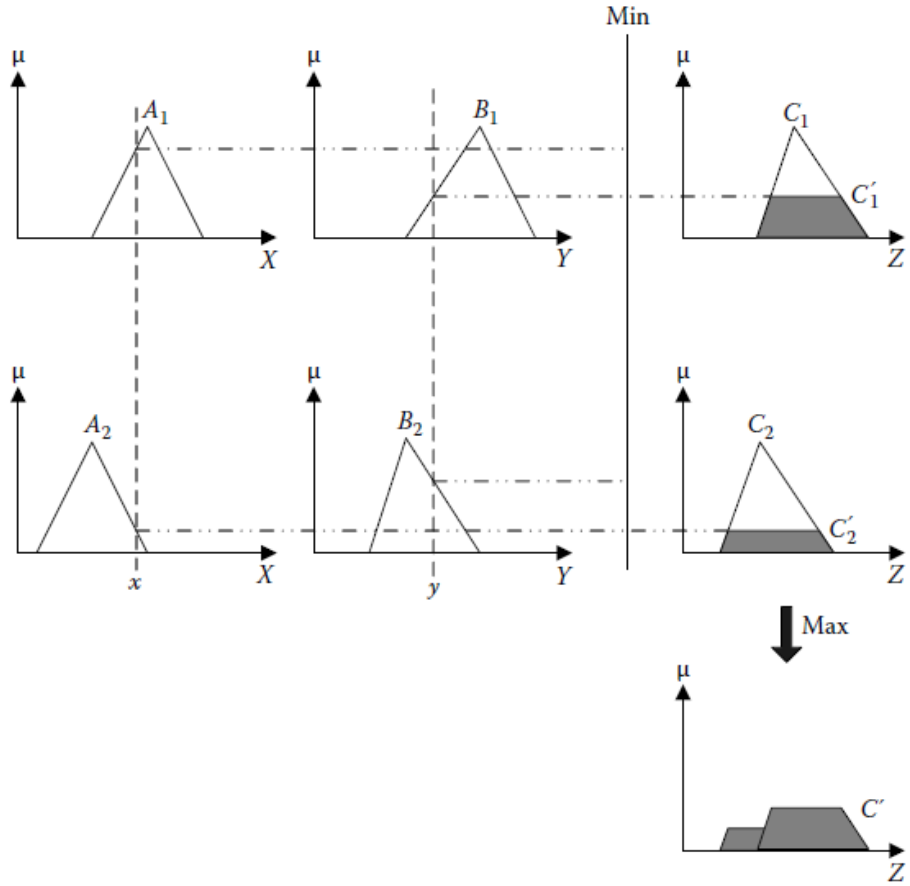


Figure II.5 Raisonnement flou de Mamdani basé sur la méthode d'inférence max-min.

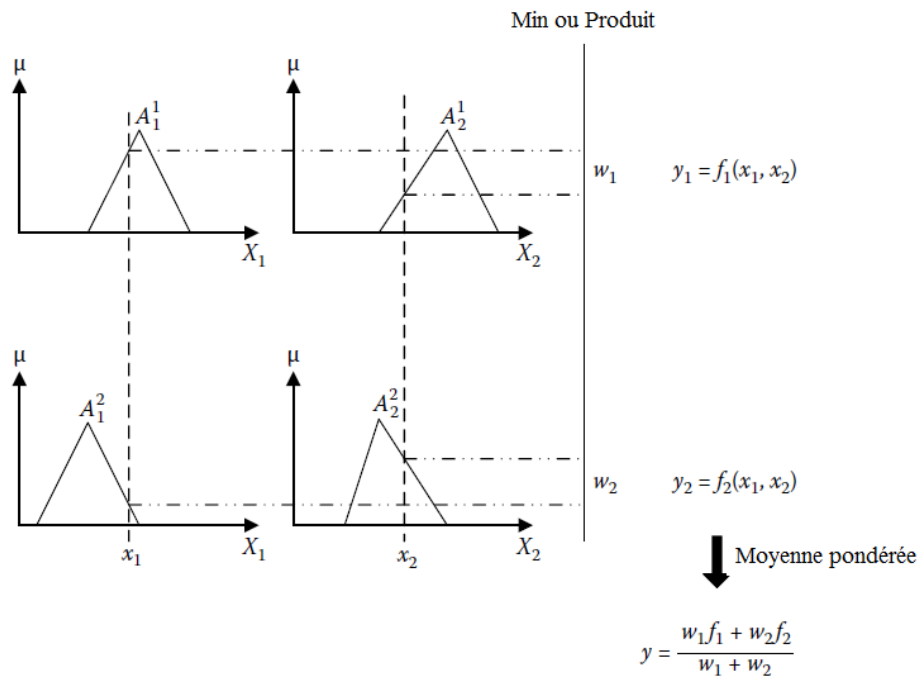


Figure II.6 Système d'inférence floue de Takagi et Sugeno.



Plusieurs approches sont proposées pour développer des modèles flous dynamiques basés sur le concept du modèle flou TS. Certaines approches utilisent le concept des états du système. En effet, représentation d'état est utile lorsque la connaissance préalable nous permet de modéliser le système à partir des lois physiques telles que le bilan de masse et d'énergie. Dans la littérature, cette approche est appelée une modélisation de type boîte-blanche. L'avantage de cette approche réside dans le fait que la structure du modèle est liée à la structure du système réel, d'où les paramètres du modèle sont souvent physiquement interprétables. Ceci n'est généralement pas le cas dans les modèles d'entrée-sortie. En outre d'autres modèles dynamiques TS ont des conclusions représentées par un modèle linéaire ARX (AutoRegressive with eXogenous input) construit à partir des mesures d'entrée-sortie dont les paramètres sont généralement différents dans chaque règle [7],

**IF**  $u(k)$  is  $A_1 \dots$  and  $u(k - k_u + 1)$  is  $A_{k_u}$  and  $x(k)$  is  $B_1 \dots$  and  $x(k - k_x + 1)$  is  $B_{k_x}$

$$\text{Then } u(k + 1) = \sum_{j=1}^{k_u} a_j u(k - j + 1) + \sum_{i=1}^{k_x} b_i x(k - i + 1) + c \quad (\text{II.10})$$

En ce sens, nous pouvons constater que le comportement dynamique est pris en charge par des filtres dynamiques externes ajoutés au système flou, voir Figure II.7. Les filtres dynamiques sont de simples générateurs des entrées et des sorties retardées [7].

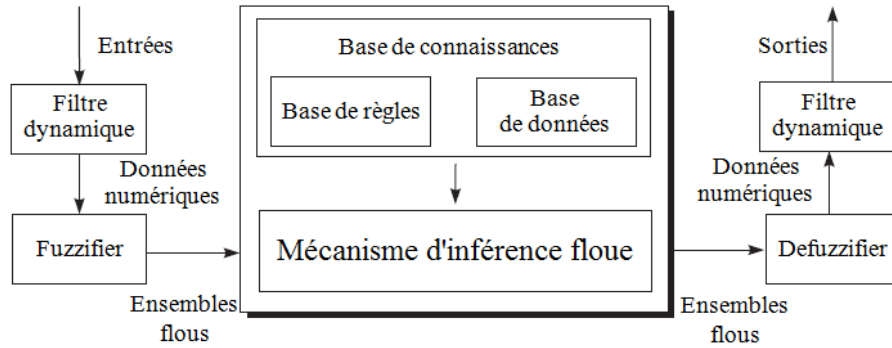


Figure II.7 Système flou dynamique.

### II.3.3 Modèle relationnel flou

Un modèle relationnel flou [7] décrit des associations entre les termes linguistiques définis dans les domaines d'entrée et de la sortie du système à l'aide des relations floues. Les éléments individuels de la relation représentent la force de l'association entre les sous-ensembles flous. Considérons d'abord un modèle flou linguistique composé d'un ensemble de règles de la forme :

$$\text{If } x_1 \text{ is } A_{i,1}, \dots \text{ and } x_k \text{ is } A_{i,k} \text{ Then } u \text{ is } B_i, \quad i = 1, 2, \dots, m_c. \quad (\text{II.11})$$

Désignons par  $\mathcal{A}_j$  l'ensemble des termes linguistiques définis pour la variable d'antécédent  $x_j$  :

$$\mathcal{A}_j = \{A_{j,l} \mid l = 1, 2, \dots, N_j\}, \quad i = 1, 2, \dots, k. \quad (\text{II.12})$$

où

$$\mu_{A_{j,l}}(x_j) : X_j \rightarrow [0, 1]. \quad (\text{II.13})$$

De même, l'ensemble des termes linguistiques définis pour la variable de conséquence  $u$  est désigné par :

$$\mathcal{B} = \{Bl \mid l = 1, 2, \dots, M\}, \quad (\text{II.14})$$

où

$$\mu_{Bl}(u) : U \rightarrow [0, 1]. \quad (\text{II.15})$$

avec,  $X_j$  est le domaine de  $x_j$ , et  $U$  est l'univers de discours de la variable  $u$ .

Dans une représentation par modèle relationnel flou, la base de règles II.11 peut être représentée par une relation crisper  $\mathcal{S}$  entre les termes linguistiques de prémisse  $\mathcal{A}_j$  et les termes linguistiques de conséquence  $\mathcal{B}$  :

$$\mathcal{S} : \mathcal{A}_1 \times \mathcal{A}_2 \times \dots \times \mathcal{A}_k \times \mathcal{B} \rightarrow \{0, 1\} \quad (\text{II.16})$$

En notant  $\mathcal{A} = \mathcal{A}_1 \times \dots \times \mathcal{A}_k$ , l'espace cartésien des termes linguistiques d'antécédent, l'équation (II.16) peut être réécrite sous la forme :  $\mathcal{S} : \mathcal{A} \times \mathcal{B} \rightarrow \{0, 1\}$ .

Le modèle relationnel flou n'est rien d'autre que le prolongement de la relation crisper  $\mathcal{S}$  à une relation floue  $\mathcal{R} = [r_{i,j}]$  :

$$\mathcal{R} : \mathcal{A} \times \mathcal{B} \rightarrow [0, 1]. \quad (\text{II.17})$$

Chaque règle contient tous les termes de conséquence possibles, chacune avec son propre facteur de pondération, donné par l'élément  $r_{i,j}$  de la relation floue (II.17). Cette pondération permet au modèle d'être affiné plus facilement, par exemple pour ajuster les données. Dans les modèles relationnels flous, la relation représente des associations entre les termes linguistiques individuels. Les éléments  $r_{i,j}$  décrivent les associations entre les combinaisons des termes linguistiques d'antécédent et les termes linguistiques de conséquence. Cela implique que les termes de conséquence ne sont pas exactement égaux aux termes linguistiques prédéfinis, mais sont donnés par leurs combinaisons pondérées.

## II.4 Synthèse des systèmes de commande flous

Typiquement, la construction de modèles flous peut se faire à partir de deux sources d'information différentes, telles que la connaissance préalable et les données (mesures du procédé). La connaissance préalable peut être plutôt d'une nature qualitative ou heuristique, issue de la connaissance des *experts*, i.e., des designers de processus, des

opérateurs. Dans ce sens, les modèles flous peuvent être vus comme des systèmes experts flous [29, 39].

D'un autre côté, pour certains procédés, des données sont disponibles sous forme d'enregistrements de l'opération du procédé ou bien il est possible de réaliser des expériences d'identification afin d'obtenir les données appropriées du comportement du système. La construction de modèles flous à partir de données implique des méthodes basées sur la logique floue et le raisonnement approché, mais aussi des idées issues du domaine de l'analyse de données et de l'identification conventionnelle de systèmes. L'utilisation de techniques et d'algorithmes pour la construction de modèles flous à partir de données est habituellement appelée *identification floue*.

Deux approches principales pour l'intégration de la connaissance et des données dans un modèle flou peuvent être distinguées :

- La connaissance experte exprimée sous une forme verbale est traduite en une collection de règles IF-THEN. De cette manière, une certaine structure de modèle est créé. Les paramètres de cette structure (fonctions d'appartenance, paramètres des conséquents) peuvent être affinés en utilisant les données d'entrée-sortie. Les algorithmes particuliers d'ajustement exploitent le fait qu'au niveau du calcul, le modèle flou peut être vu comme une structure par couches (réseaux), similaire aux réseaux de neurones artificiels, pour laquelle des algorithmes standard d'apprentissage peuvent être appliqués [1, 7].
- Aucune connaissance a priori sur le système appréhendé n'est disponible. Ceci conduit à opter pour une méthode de conception de systèmes flous à partir de données. Il est prévu que les règles extraites et les fonctions d'appartenance peuvent fournir une interprétation a posteriori du comportement du système. Un expert peut confronter ces informations avec ses propres connaissances, peut modifier les règles, ou concevoir des expériences supplémentaires afin d'obtenir des données plus informatives. Cette approche peut être appelée *extraction de règles*. La coalescence floue est l'une des techniques que l'on applique souvent dans ce cas précis [1, 30].

## II.5 Algorithmes de coalescence floue

Les techniques de coalescence (clustering) floue recherche des groupes de données nommées *clusters*. Un cluster peut être défini comme un groupe de données construit sur la base d'une mesure de similitude appropriée. Les formes de cluster les plus communes sont les sphères et les ellipses. Dans ce cas, on représente le cluster par son centre. Dans la littérature, on recense plusieurs algorithmes de coalescence floue. Formulés dans la plupart comme étant des problèmes d'optimisation, ils ont pour objectif de générer des modèles de représentation flous directement à partir des données. Plus précisément, il est question d'extraire les paramètres et la base des règles floues par la génération des partitions floues et des prototypes de clusters représentés le plus souvent par leurs centres. Un contrôleur flou étant représenté par un modèle flou, la synthèse d'un système de commande à base de règles floues peut être entièrement établie à partir de données d'entrée/sortie en utilisant une méthode de clustering flou. Nous examinons dans la suite certains de ces algorithmes.

### II.5.1 Algorithme des k-moyennes

L'algorithme des k-moyennes discuté dans cette section est la méthode de coalescence la plus commune et la plus simple. Elle peut être considérée comme la base de toutes les approches développées dans ce domaine. L'algorithme des k-moyennes repose sur la minimisation de la fonction objectif :

$$f_p = \sum_{j=1}^{m_c} \sum_{i \in S_j} \|z(i) - c_j\|^2 \rightarrow \min_{c_j} \quad (\text{II.18})$$

où  $m_c$  est le nombre de clusters, et  $c_j$  les centres des clusters (prototypes). Les ensembles  $S_j$  contiennent les échantillons de données qui appartiennent au cluster  $j$ , c'est-à-dire, ceux situés le plus près du centre du cluster  $c_j$ . En exprimant l'appartenance des données aux clusters associés, la fonction objectif  $f_p$  à minimiser peut être réécrite sous la forme :

$$f_p = \sum_{j=1}^{m_c} \sum_{i=1}^N \mu_{ji} \|z(i) - c_j\|^2, \quad (\text{II.19})$$

où  $\mu_{ji} = 1$  si l'échantillon de données  $z(i)$  est associé (appartient) au cluster  $j$  et  $\mu_{ji} = 0$  en cas de non appartenance.

L'algorithme des k-moyennes [36] se résume aux étapes suivantes :

- **Étape 1** : Initialiser aléatoirement les centres des  $m_c$  clusters  $c_j$ ,  $j = 1 \dots m_c$ .
- **Étape 2** : Attribuer tous les échantillons de données aux centres de clusters les plus proches.
- **Étape 3** : Calculer le centre de gravité (moyenne) de chaque cluster puis réajuster chaque centre de cluster par rapport à son centre de gravité en utilisant l'expression :

$$c_j = \frac{\sum_{i \in S_j} z(i)}{N_j}, \quad (\text{II.20})$$

où  $N_j$  est le nombre d'éléments de l'ensemble  $S_j$ , ( $\sum_{j=1}^{m_c} N_j = N$ )

- **Étape 4** : Si un centre de cluster se trouve déplacé dans l'étape précédente, alors passer à l'étape 2, sinon mettre fin à l'algorithme.

Plusieurs normes de distances peuvent être employées. On peut éventuellement utiliser la norme euclidienne qui fournit une mesure de distance de la forme :

$$D_{ij}^2 = \|z(i) - c_j\|^2 = (z(i) - c_j)^T (z(i) - c_j) \quad (\text{II.21})$$

Cette norme peut être étendue à la norme quadratique générale de *Mahalanobis* qui est définie par [36] :

$$D_{ij,\Sigma}^2 = \|z(i) - c_j\|_{\Sigma}^2 = (z(i) - c_j)^T \Sigma (z(i) - c_j) \quad (\text{II.22})$$

La matrice de covariance  $\Sigma$  est donnée par :

$$\Sigma = \begin{bmatrix} 1/\sigma_1^2 & 0 & \dots & 0 \\ 0 & 1/\sigma_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/\sigma_P^2 \end{bmatrix} \quad (\text{II.23})$$

où  $P$  désigne la dimension de l'espace d'entrée. La norme de Mahalanobis se ramène à la norme euclidienne lorsque la matrice de covariance est égale à la matrice identité.

## II.5.2 Algorithme des c-moyennes floues

L'algorithme des c-moyennes floues est la version floue de l'algorithme classique des k-moyennes décrit ci-dessus. La fonction objectif à minimiser s'exprime par :

$$f_p = \sum_{j=1}^{m_c} \sum_{i=1}^N \mu_{ji}^\nu \|z(i) - c_j\|_\Sigma^2, \quad \text{avec} \quad \sum_{j=1}^{m_c} \mu_{ji} = 1. \quad (\text{II.24})$$

où  $\mu_{ji}$  désigne le degré d'appartenance d'un échantillon de données  $z(i)$  au cluster  $j$  de centre  $c_j$ . Le paramètre  $\nu$ , détermine le degré de flou des clusters. Il est choisi dans l'intervalle  $[1, \infty[$ . Dans le cas où on a pas une connaissance préalable sur sa valeur, on prend  $\nu = 2$ .

Le degré d'appartenance  $\mu_{ji}$  d'un échantillon de données  $z(i)$  associé au cluster  $j$  est défini par

$$\mu_{ji} = \frac{1}{\sum_{l=1}^{m_c} \left( D_{ij,\Sigma}^2 / D_{il,\Sigma}^2 \right)^{\frac{1}{\nu-1}}}. \quad (\text{II.25})$$

avec

$$D_{ij,\Sigma}^2 = \|z(i) - c_j\|_\Sigma^2 = (z(i) - c_j)^T \Sigma (z(i) - c_j) \quad (\text{II.26})$$

À partir des équations (II.25) et (II.26), il est évident de constater que si l'échantillon de données se rapproche du centre du cluster ( $D_{ij,\Sigma}^2 \rightarrow 0$ ), le degré d'appartenance de ce cluster se rapproche de 1 ( $\mu_{ji} \rightarrow 1$ ). En revanche, si  $D_{ij,\Sigma}^2 \rightarrow \infty$  alors  $\mu_{ji} \rightarrow 0$ . Aussi, il est clair que les degrés d'appartenance (II.25) remplissent la contrainte formulée dans l'équation (II.24).

Lors du calcul du degré d'appartenance à l'aide de l'équation (II.25), il est important de tenir compte des deux considérations suivantes :

- Dans le cas où l'échantillon de données  $z(i)$  se trouve exactement sur un centre du cluster  $c_l$ , différent du cluster  $j$  ( $l \neq j$ ) alors  $\mu_{ji} = 0$ .
- Dans le cas où l'échantillon de données  $z(i)$  se trouve exactement sur le centre du cluster  $c_j$ , alors  $\mu_{ji}$  peut être choisi arbitrairement en respectant la contrainte  $\sum_{j=1}^{m_c} \mu_{ji} = 1$ .

L'algorithme des c-moyennes [36] se résume à la structure algorithmique suivante :

- **Étape 1** : Initialiser aléatoirement les centres des  $m_c$  clusters  $c_j$ ,  $j = 1 \dots m_c$ .
- **Étape 2** : Calculer la distance  $D_{ij,\Sigma}^2$  de tous les échantillons de données  $z(i)$  pour chaque centre de cluster  $c_j$  selon (II.26).
- **Étape 3** : Calculer les degrés d'appartenance aux clusters  $c_j$  pour chaque échantillon de données  $z(i)$  en utilisant l'équation (II.25).
- **Étape 4** : Actualiser les centres des clusters en utilisant l'équation :

$$c_j = \frac{\sum_{i=1}^N \mu_{ji}^\nu z(i)}{\sum_{i=1}^N \mu_{ji}^\nu}, \quad (\text{II.27})$$

- **Étape 5** : Dans le cas où un centre de cluster se trouve déplacé de façon significative, soit plus d'un seuil  $\epsilon$ , alors passer à l'étape 3, sinon fin de l'algorithme.

La forme du cluster doit être fixée par l'utilisateur a priori. Elle ne peut être ni adaptée aux données ni différente pour chaque cluster individuel.

### II.5.3 Algorithme de coalescence soustractive

L'algorithme de coalescence soustractive a été initialement développé par *Chiu* dans [11, 13, 32]. Dans cette méthode, les échantillons de données doivent être mis à l'échelle sur l'intervalle  $[0, 1]$ . À chaque échantillon de données  $z(i) = (x(i), u(i))$  on associe un potentiel  $P(i)$ , en fonction de son emplacement par rapport à tous les autres points de données :

$$P(i) = \sum_{j=1}^n e^{\alpha \|z(i) - z(j)\|^2} \quad (\text{II.28})$$

où

$$\alpha = \frac{\gamma_a}{r_a^2} \quad (\text{II.29})$$

$\|\cdot\|$  désigne la distance euclidienne,

$\alpha$  est un paramètre de conception,

$\gamma_a$  est une constante fixée généralement à 4,

$r_a$  est une constante positive appelée rayon du cluster.

Après avoir évalué le potentiel de chaque point de données, on choisit le point de données avec le plus grand potentiel en tant que premier centre de cluster. Soit  $c_1$  le premier centre choisi et  $P_1$  sa valeur potentielle. Le potentiel  $P(i)$  est réévalué pour chaque point de données  $z(i)$  selon l'expression :

$$P(i) = P(i) - P_1 e^{-\beta \|z(i) - c_1\|^2} \quad (\text{II.30})$$

où

$$\beta = \frac{\gamma_a}{r_b^2} \quad (\text{II.31})$$

avec  $r_b$  un constant positif. Cela revient à soustraire une quantité de potentiel pour chaque point de données en fonction de sa distance par rapport au premier centre de

cluster. Le potentiel des points de données au voisinage du premier centre du cluster sera largement réduit, par conséquent, ces données ne seront donc probablement pas choisies pour former le prochain centre du cluster. Le constant  $r_b$  est effectivement le rayon définissant la zone qui aura des réductions de potentiel. Pour éviter d'obtenir des centres de clusters rapprochés, on choisit  $r_b$  un peu plus grand que  $r_a$ , un bon choix est  $r_b = 1,25r_a$ . Lorsque le potentiel de tous les points de données est révisé selon l'équation (II.30), on sélectionne le point de données avec le potentiel restant le plus élevé pour former le deuxième centre de cluster. Plus généralement, après avoir obtenu le  $k^e$  centre de cluster, le potentiel de chaque point de données est recalculé par la formule :

$$P(i) = P(i) - P_k e^{-\beta \|z(i) - c_k\|^2} \quad (\text{II.32})$$

où  $c_k$  est l'emplacement du  $k^e$  centre de cluster et  $P_k$  est la valeur de son potentiel. Le processus de détermination de nouveaux centres de clusters et de révision des potentiels est répété jusqu'à ce que le potentiel restant de tous les points de données passe au dessous d'une certaine limite fixée en fonction du potentiel du premier centre  $P_1$ . En plus de ce critère, pour mettre fin à la procédure de coalescence il existe des critères d'acceptation et de rejet des centres qui permettent d'éviter la formation de clusters marginaux [12]. L'algorithme suivant décrit un de ces critères :

- 
1. Si  $P_k^* > \bar{\epsilon}P_1^*$  alors  
 Accepter  $c_k$  comme un centre de cluster.
  2. Sinon si  $P_k^* < \underline{\epsilon}P_1^*$  alors  
 Rejeter  $c_k$  et terminer le processus de coalescence.
  3. Sinon  
 Soit  $d_{min}$  = la plus courte distance entre  $c_k$  et tous les centres des clusters trouvés précédemment.
  4. Si  $\frac{d_{min}}{r_a} + \frac{P_k^*}{P_1^*} \geq 1$  alors  
 Accepter  $c_k$  comme un centre de cluster.
  5. Sinon  
 Rejeter  $c_k$  et changer son potentiel à 0.  
 Ne pas modifier le potentiel d'autres points de données.  
 Sélectionner le point de données ayant le plus grand potentiel  
 comme nouveau centre  $c_k$  et re-test.
  6. Fin si
  7. Fin si
- 

**Figure II.8** Critère d'acceptation et de rejet des centres de clusters.

Ici  $\bar{\epsilon}$  spécifie un seuil pour le potentiel au-dessus duquel nous allons certainement accepter le point de données en tant que centre du cluster,  $\underline{\epsilon}$  définit un seuil en dessous duquel nous allons certainement rejeter le point de données. Les valeurs par défaut pour ces deux seuils sont  $\bar{\epsilon} = 0.5$  et  $\underline{\epsilon} = 0.15$ .

On résume l'algorithme de coalescence soustractive comme suivant :

- **Étape 1** : Calculer le potentiel de chaque point de données en utilisant l'équation (II.28), définir le nombre de centres de cluster  $k = 1$ .
- **Étape 2** : Sélectionner le point avec le plus grand potentiel noté  $c_k$ .
- **Étape 3** : Modifier le potentiel de chaque point de données selon l'équation (II.30).
- **Étape 4** : Si le critère d'arrêt décrit dans la Figure II.8 est atteint alors arrêter le processus, sinon, mettre  $k = k + 1$ , et retourner à l'étape 2.

Dans ce contexte, et pour le reste de cette étude, l'algorithme de coalescence soustractive est choisi pour la conception de notre stratégie de synthèse de systèmes de commande flous. De part la simplicité de mise œuvre de l'algorithme, notre choix est principalement motivé par le fait que l'algorithme de coalescence soustractive permet d'extraire automatiquement le nombre de règles à partir des données sans que ce dernier ne soit explicitement fourni. Ainsi, une procédure systématique d'extraction de la base de règles floues sera développée pour la synthèse des lois de commande.

## II.6 Conclusion

Ce chapitre introduit brièvement certaines notions et outils de manipulation des systèmes d'inférence floue. Comme il peut être constaté, il existe une multitude de modèles de représentation flous, que le concepteur pourra choisir en fonction de l'application appréhendée. Les principales méthodes d'obtention de ces modèles y sont présentées. Les méthodes se basant sur les techniques de coalescence floue sont très intéressantes, mais nécessitent la disponibilité d'une base de données adéquate sur le système à identifier.



# CHAPITRE III

## Émulation par systèmes d'inférence floue de lois de commande multivariable

### III.1 Introduction

Les quelques stratégies de commande se basant sur des critères d'optimalité et/ou de robustesse examinées dans le Chapitre I, en l'occurrence la commande quadratique LQG et la commande  $\mathcal{H}_\infty$ , sont particulièrement dédiées aux systèmes multivariables. Comme il vient d'être montré, les procédures de synthèse de ce type de commande font appel à des représentations dans l'espace d'état dont la dimension est tributaire du système appréhendé. Pour les systèmes de grande dimension, il est évident de constater que les réalisations de commande obtenues sont généralement d'ordre élevé. Cette constatation pourrait même concerner les systèmes de dimension relativement grande (de l'ordre 3, par exemple), car l'augmentation de dynamiques, qu'il est souvent nécessaire d'introduire pour les besoins de robustesse en stabilité ou en performance, aura pour effet d'accroître l'ordre du système commandé, ce qui conduit à l'obtention de contrôleurs multivariables d'ordre élevé. En pratique, les modèles de contrôle de grande dimension posent de sérieux problèmes d'implémentation, surtout que ces derniers sont de structure complexe et multivariable. La mise en œuvre effective de ces techniques de commande nécessite des simplifications structurelles, notamment de réduction d'ordre. Ces manipulations ne sont bien évidemment pas sans incidences sur la boucle de commande. En fait, dans bon nombre de cas, on observe d'éventuelles désadaptations ou dégradation de performances sur les boucles de commande. L'idée de construire de bonnes approximations émulant efficacement les dynamiques représentées par les modèles de commande multivariable de grande dimension constitue toujours une piste intéressante.

Dans ce chapitre, on propose une alternative fondée exclusivement sur les systèmes d'inférence floue. Il s'agit d'une méthodologie d'émulation floue de lois de commande multivariable utilisant des techniques de coalescence floue. Les différents aspects de développement de cette stratégie seront examinés dans la suite.

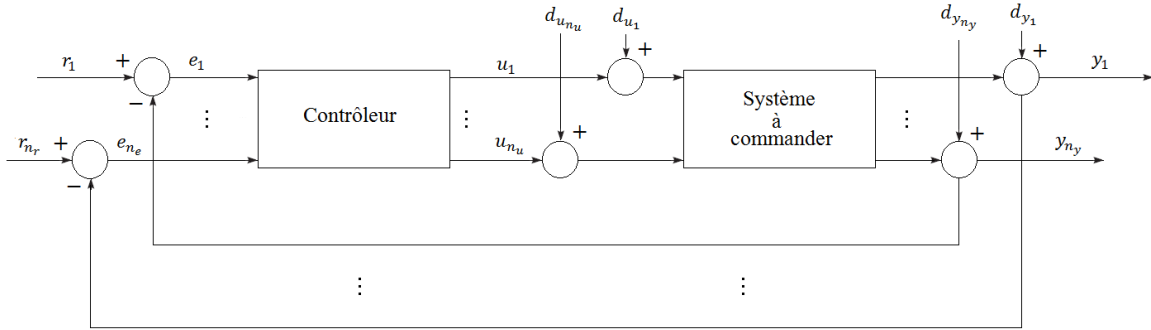
## III.2 Contrôle multivariable

Dans cette étude, on s'intéresse plus particulièrement à l'émulation par systèmes flous des modèles de contrôle multivariable. Ces contrôleurs sont le résultat inévitable de l'application de stratégies de commande multivariable aux systèmes complexes. Une telle catégorie de systèmes peut être généralement représentée par le modèle d'état suivant :

$$\begin{cases} \dot{x}(t) = f(x(t), u(t), t) \\ y(t) = g(x(t), u(t), t) \end{cases} \quad (\text{III.1})$$

où,  $x(t)$  le vecteur d'état de dimension  $n_s$ ,  $u(t)$  le vecteur de commande de dimension  $n_u$ ,  $y(t)$  désigne le vecteur de sortie de dimension  $n_y$ , et  $f$  et  $g$  sont des fonctions vectorielles réelles non-linéaires sur  $\mathfrak{R}^{n_s} \times \mathfrak{R}^{n_u} \times \mathfrak{R} \rightarrow \mathfrak{R}^{n_s}$  et  $\mathfrak{R}^{n_s} \times \mathfrak{R}^{n_u} \times \mathfrak{R} \rightarrow \mathfrak{R}^{n_y}$ , respectivement.

La configuration d'un système multivariable contrôlé en boucle fermée et soumis à des perturbations en entrée  $d_{u_1} \dots d_{u_{n_u}}$  et en sortie  $d_{y_1} \dots d_{y_{n_y}}$  est illustrée par la Figure III.1.



**Figure III.1** Configuration de base d'un système multivariable en boucle fermée.

Comme mentionné auparavant, la conception d'une stratégie de commande pour un système multivariable conduit généralement à un contrôleur multivariable dont l'ordre dépend de celui du système commandé. Admettons pour ce contrôleur une réalisation d'état de la forme :

$$\begin{cases} \dot{x}_c = A_c x_c + B_c e \\ u = C_c x_c + D_c e \end{cases} \quad (\text{III.2})$$

où  $x_c$  est le vecteur d'état de dimension  $n_c$ ,

$e$  est le vecteur d'erreur de dimension  $n_e$ ,

$r$  est le vecteur de référence de dimension  $n_r$ , et les matrices  $A_c$ ,  $B_c$ ,  $C_c$  et  $D_c$  sont invariantes dans le temps.

### III.3 Stratégie d'émulation floue des contrôleurs multivariables

L'approche d'approximation que nous proposons dans la présente contribution est valable pour n'importe quel modèle de commande multivariable. Or, l'utilité majeure réside dans son application à des systèmes de commande multivariable d'ordre élevé, car il s'agit plutôt de résoudre une situation problématique d'implémentation. Pour ce faire, nous avons choisi de recourir au concept de modélisation floue en posant le problème d'émulation du comportement dynamique d'un contrôleur multivariable donné comme un problème d'identification par logique floue de son modèle de représentation. Plus précisément, il est question de concevoir un système d'inférence floue à partir de données d'entrée/sortie en utilisant les techniques de coalescence floue. La procédure de conception de l'émulateur flou n'est guère systématique, car elle doit être menée de manière à garantir une bonne approximation de dynamiques en tenant compte d'un certain nombre de spécifications et paramètres de conception. Elle est généralement établie à travers deux principales étapes : (1) Détermination de la structure de l'émulateur flou, et (2) identification de ses paramètres.

#### III.3.1 Détermination de la structure de l'émulateur

Un émulateur flou d'un contrôleur multivariable est tout simplement un contrôleur flou dont la loi de commande qu'il génère constitue une approximation du contrôleur multivariable original. Il est donc clair que la détermination d'une structure pour l'émulateur flou revient à choisir un modèle de représentation flou pour ce système à base de règles. Parmi l'ensemble des modèles flous présentés dans le Chapitre I, il nous est avéré utile de faire appel au modèle flou de Takagi-Sugeno (TS), vu sa capacité d'approximation universelle. Dans cette phase, il est également fondamental de procéder à la sélection des variables de prémisses des règles et les variables de conséquence. Comme il s'agit d'émuler le comportement d'un contrôleur multivariable, les entrées de l'émulateur flou seront naturellement choisies comme étant les variables d'erreur et les variables d'erreur retardées, alors que les sorties de l'émulateur ne sont rien d'autre que les variables de commande et les variables de commande retardées. Parmi toutes ces variables, on doit éventuellement retenir celles qui sont les plus significatives, et ce en fonction de la configuration de commande multivariable originale. Plus généralement, l'émulateur flou à concevoir peut être décrit par un modèle TS dont la forme générale de la  $l^e$  règle est exprimée par :

$$\begin{aligned}
 &\text{If } e_1(k) \text{ is } E_{1l}^0 \text{ and } e_2(k) \text{ is } E_{2l}^0 \dots \text{ and } e_1(k-1) \text{ is } E_{1l}^1 \text{ and } e_2(k-1) \text{ is } E_{2l}^1 \dots \\
 &\quad \text{and } e_1(k-n_{e_1}) \text{ is } E_{1l}^{n_{e_1}} \text{ and } e_2(k-n_{e_2}) \text{ is } E_{2l}^{n_{e_2}} \dots \\
 &\quad \text{and } u_1(k-1) \text{ is } U_{1l}^0 \text{ and } u_2(k-1) \text{ is } U_{2l}^0 \dots \\
 &\quad \text{and } u_1(k-2) \text{ is } U_{1l}^1 \text{ and } u_2(k-2) \text{ is } U_{2l}^1 \dots \\
 &\quad \text{and } u_1(k-n_{u_1}) \text{ is } U_{1l}^{n_{u_1}} \text{ and } u_2(k-n_{u_2}) \text{ is } U_{2l}^{n_{u_2}} \dots \\
 &\text{then } u_1(k) = f_{1l}(e_1(k), \dots, e_1(k-n_{e_1}), \dots, u_1(k-1), \dots, u_1(k-n_{u_1}), \dots) \\
 &\text{and } u_2(k) = f_{2l}(e_1(k), \dots, e_1(k-n_{e_1}), \dots, u_1(k-1), \dots, u_1(k-n_{u_1}), \dots) \dots \quad (\text{III.3})
 \end{aligned}$$

où  $e_i(k), \dots, e_i(k-n_{e_i})$   $i = 1 \dots n_e$  et  $u_j(k-1), \dots, u_j(k-n_{u_j})$ ,  $j = 1 \dots n_u$ , indiquent la  $i^e$  variable d'erreur et ses valeurs retardées et les  $j^e$  commandes retardées, respecti-

vement.  $E_{il}^0, \dots, E_{il}^{n_{e_i}}$ , et  $U_{jl}^0, \dots, U_{jl}^{n_{u_j}}$  sont les ensembles flous associés, et  $f_{il}$  est une fonction arbitraire. Pour tout vecteur d'entrée  $x(k) = (e_1(k), \dots, e_1(k - n_{e_1}), \dots, u_1(k - 1), \dots, u_1(k - n_{u_1}))$ , la  $i^e$  valeur de sortie du modèle flou TS est calculé comme suit :

$$u_i(k) = \frac{\sum_{l=1}^{m_c} \tau_l \times f_{il}(x(k))}{\sum_{l=1}^{m_c} \tau_l} \quad (\text{III.4})$$

et

$$\tau_l = \prod_{i=1}^{n_e} \prod_{m=1}^{n_{e_i}} (\mu_{E_{il}^m}(e_i(k - m))) \times \prod_{j=1}^{n_u} \prod_{p=0}^{n_{u_j}} (\mu_{U_{jl}^p}(u_j(k - p - 1))) \quad (\text{III.5})$$

où  $m_c$  est le nombre de règles floues,  $\mu_{E_{il}^m}$  et  $\mu_{U_{jl}^p}$  sont les fonctions d'appartenance de sous-ensembles flous de la prémisse  $E_{il}^m$  et  $U_{jl}^p$ , respectivement, et  $\tau_l$  est le degré d'activation de la  $l^e$  règle.

Le modèle à base de règles floues (III.3) est entièrement défini par l'ensemble des paramètres suivants : Le nombre de règles floues  $m_c$ , les paramètres des fonctions d'appartenance associées aux sous-ensembles flous de la prémisse  $E_{il}^m$  et  $U_{jl}^p$ , que les les paramètres des conséquences des règles floues.

### III.3.2 Identification des paramètres

Dans cette phase, les paramètres des prémisses et des conséquences des règles floues sont à identifier. Plus précisément, il s'agit du nombre de règles  $m_c$ , des ensembles flous des prémisses  $E_{il}^m$  et  $U_{jl}^p$ , et les paramètres des fonctions linéaires  $f_{il}$  figurant dans les conséquences des règles, avec  $l = 1 \dots m_c$ ,  $i = 1 \dots n_e$ ,  $j = 1 \dots n_u$ ,  $m = 1 \dots n_{e_i}$ , et  $p = 1 \dots n_{u_j}$ .

#### III.3.2.1 Identification des prémisses des règles

Afin d'identifier les ensembles flous associés aux prémisses des règles, la méthode de coalescence soustractive introduite dans le Paragraphe II.5.3 est utilisée. Cet algorithme est appliqué dans l'espace combiné d'entrée/sortie, c'est-à-dire, chaque mesure  $z(k)$  est donnée par  $z(k) = (x(k), u(k))$  avec  $k = 1, \dots, N$ , où  $x(k) = (e_1(k), \dots, e_1(k - n_{e_1}), \dots, u_1(k - 1), \dots, u_1(k - n_{u_1}), \dots)$  et  $u(k) = (u_1(k), u_2(k), \dots)$ . Ainsi, un potentiel  $P(i)$  est attribué à la mesure  $z(i)$  en fonction de sa position par rapport à tous les autres points de données  $z(k)$ . Ce potentiel est calculé par :

$$P(i) = \sum_{k=1}^N e^{\alpha \|z(i) - z(k)\|^2} \quad (\text{III.6})$$

où  $\alpha$  est un paramètre définissant le rayon du cluster et  $N$  est le nombre de données d'apprentissage. Après que le potentiel de chaque point de données ait été calculé, le point de données ayant le plus haut potentiel est choisi comme premier centre de cluster. À partir de cette position initiale du cluster, un processus de révision des potentiels et de création de nouveaux centres est effectué de manière itérative jusqu'à ce qu'un critère d'arrêt donné est rempli.

Considérons à présent un ensemble de  $m_c$  centres  $\{c_1, c_2 \dots c_{m_c}\}$  détectés dans un espace de dimension  $n$ . Notons par  $n_x$  la dimension du vecteur d'entrée et par  $n_u =$

$n - n_x$  la dimension du vecteur de sortie (de commande). Le vecteur des centres  $c_i$  peut être décomposé en deux parties :  $c_i^x$  et  $c_i^u$ , où  $c_i^x$  contient les premiers  $n_x$  éléments de  $c_i$  (C'est-à-dire, les coordonnées des centres de clusters dans l'espace d'entrée) et  $c_i^u$  regroupe les derniers  $n_u$  éléments (C'est-à-dire, les coordonnées des centres de clusters dans l'espace de sortie). Ainsi, chaque centre de cluster  $c_i$  peut être perçu comme une règle que l'on exprime linguistiquement sous la forme :

$$R_l : \mathbf{If} \text{ l'entrée est proche de } c_l^x \mathbf{ then} \text{ la sortie est proche de } c_l^u \quad (\text{III.7})$$

Étant donné le vecteur d'entrée  $x(k)$ , le degré d'activation de la  $l^e$  règle est défini par :

$$\mu_l = e^{-\alpha \|x(k) - c_l^x\|^2} \quad (\text{III.8})$$

où  $\alpha$  est la constante définie dans l'équation (II.29). La variable de sortie (commande) sera alors calculée par :

$$u_k = \frac{\sum_{l=1}^{m_c} \mu_l c_l}{\sum_{l=1}^{m_c} \mu_l} \quad (\text{III.9})$$

L'équation (III.7) qui traduit une règle floue selon le formalisme de Mamdani, peut être transformée en une règle floue selon le formalisme de Takagi-Sugeno, et prend alors la forme suivante :

$$R_l : \mathbf{If} \ x_1 \text{ is } A_{1l} \mathbf{ and} \ x_2 \text{ is } A_{2l} \dots \mathbf{ then} \ u = a_l^0 + a_l x \quad (\text{III.10})$$

où  $[a_l^0 \ a_l]$  avec  $a_l = [a_l^1 \dots a_l^{n_x}]$  désigne le vecteur des paramètres de la conclusion de la règle  $l$ , et  $A_{jl}$  est une fonction d'appartenance gaussienne associée à la  $j^e$  entrée dans la règle  $l$ . Elle s'exprime par :

$$A_{jl}(x_j) = e^{-\frac{1}{2} \left( \frac{x_j - c_l^x}{\sigma_l^j} \right)^2} \quad (\text{III.11})$$

avec  $(\sigma_l^j)^2 = 1/(2\alpha)$ .

Bien que le nombre de clusters (ou règles) est automatiquement déterminé par cette méthode, il faut noter que l'utilisateur spécifie le paramètre  $r_a$  (Le rayon d'influence d'un centre de cluster) qui affecte fortement le nombre de clusters générés. Une grande valeur de  $r_a$  se traduit généralement par un petit nombre de clusters, tandis qu'une petite valeur  $r_a$  peut produire un grand nombre de clusters. Par conséquent, nous pouvons considérer  $r_a$  en tant que paramètre de conception à choisir en fonction de la résolution souhaitée du modèle, et qui peut être ajusté en fonction de la complexité et de la capacité de la généralisation du modèle.

### III.3.2.2 Estimation des conséquences des règles

L'étape qui suit consiste à estimer les paramètres des conséquences des règles floues. Ceci peut être réalisé en appliquant simplement la méthode des moindres carrés utilisant les coordonnées des centres de clusters dans l'espace de sortie [42]. Les conclusions du modèle flou TS (III.10) sont linéaires dans les paramètres  $a_l^0$  et  $a_l$ . Par conséquent, ces paramètres peuvent être estimés à partir des données mesurées par la technique des

moindres carrés.

Étant donné les matrices des données d'entrée  $X \in \mathfrak{R}^{N \times n_x}$  et de sortie  $U \in \mathfrak{R}^{N \times n_u}$  :

$$X = [x(1), \dots, x(N)]^T, \quad U = [u(1), \dots, u(N)]^T \quad (\text{III.12})$$

On définit la matrice étendue  $X_e = [X, 1]$ , en ajoutant une colonne unitaire à  $X$ . En outre, on note la matrice  $X'$  sur  $\mathfrak{R}^{N \times C(n_x+1)}$  qu'est composée de produits des deux matrices  $\Gamma_i$  et  $X_e$ ,

$$X' = [\Gamma_1 X_e | \Gamma_2 X_e, \dots, \Gamma_{m_c} X_e] \quad (\text{III.13})$$

où  $\Gamma_i \in \mathfrak{R}^{N \times N}$  est la matrice diagonale dont les éléments sont les degrés d'appartenance normalisés, et  $\gamma_i(x(k))$  est son  $k^e$  élément diagonal. Les paramètres  $a_i^0$  et  $a_i$  sont regroupés dans un vecteur de paramètres unique  $\theta \in \mathfrak{R}^{m_c(n_x+1)}$

$$\theta = [a_1^T, a_1^0, \dots, a_{m_c}^T, a_{m_c}^0]^T \quad (\text{III.14})$$

Soit la matrice des données d'entrée  $X$ , et le  $j^e$  vecteur des données de sortie  $u_j$  avec  $j = 1, \dots, n_u$ , il est bien connu que le vecteur des paramètres  $\theta$  peut être calculé pour la  $j^e$  valeur de sortie comme suit :

$$\theta = [(X')^T X']^{-1} (X')^T u_j \quad (\text{III.15})$$

Il s'agit de la solution optimale des moindres carrés qui donne une erreur minimale de prédiction. Si une estimation précise des paramètres du modèle local est souhaitée, une approche pondérée des moindres carrés appliquée par règle peut être utilisée :

$$[\theta_i^T, \theta_i^0]^T = [X_e^T \Gamma_i X_e]^{-1} X_e^T \Gamma_i u_j \quad (\text{III.16})$$

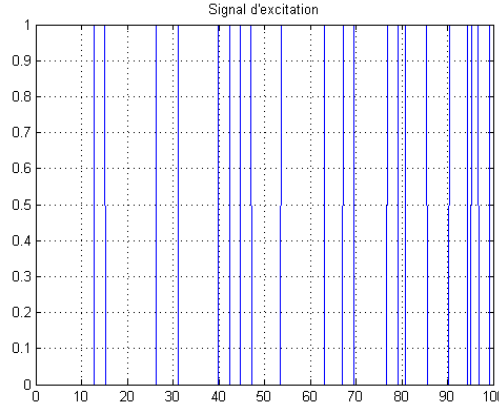
Dans ce cas, les paramètres des règles floues sont estimés indépendamment les uns des autres, et ne sont donc pas *biaisés* par les interactions des règles.

### III.3.3 Considérations sur la mise en œuvre de la stratégie d'émulation floue

#### III.3.3.1 Choix du signal d'excitation

La base de règles décrivant l'émulateur flou est extraite automatiquement à partir de données d'apprentissage d'entrée/sortie. Générer cet ensemble de données est une question clé qui doit être adressée avec une attention particulière, en particulier pour les processus multivariables. Le contrôleur à identifier est excité à l'aide d'un signal d'entrée  $x(k)$ . Il est important de choisir une entrée qui excite tous les modes du contrôleur multivariable en veillant à couvrir la totalité ou presque de la gamme de fréquences manipulées. Par exemple, un signal d'entrée sinusoïdal de pulsation  $\omega_0$  ne pourra décrire l'information qu'il véhicule qu'à cette pulsation  $\omega_0$ . Pour une meilleure caractérisation de dynamiques, il est donc nécessaire d'utiliser des excitations riches en fréquences. En pratique, on utilise souvent des excitations binaires (deux niveaux de signal) avec une durée variable par niveau. Cette durée peut varier linéairement ou être générée de manière pseudo-aléatoire.

Le signal binaire pseudo-aléatoire (SBPA) est un signal à deux niveaux 0 et 1 (Figure III.2) qui peut être généré en utilisant des registres à décalage et l'algèbre de Boole [14]. La génération du signal est caractérisée par deux paramètres :  $n_{sr}$ , le nombre de registres à décalage, et  $T_{sw}$ , le temps de commutation. La séquence se répète après  $N_s T_{sw}$  unités de temps, avec  $N_s = 2^{n_{sr}} - 1$ . L'amplitude du signal d'excitation  $\tilde{e}$  représente aussi un paramètre clé dans la procédure d'identification, car il existe un compromis entre le rapport signal/bruit et la complexité du système à identifier lors du choix de l'amplitude de la SBPA. L'amplitude de l'excitation doit rester supérieure au niveau du bruit. La sélection d'une séquence binaire pseudo-aléatoire adéquate se fait après plusieurs tests appliqués à chaque canal d'entrée en fonctions de combinaisons bien choisies.



**Figure III.2** Signal d'excitation binaire pseudo-aléatoire.

Des combinaisons judicieusement choisies sont composées utilisées afin d'exciter l'ensemble des entrées du contrôleur original. Le tableau III.1 représente un exemple de combinaisons de signaux d'amplitude unitaire pouvant être composées pour un contrôleur à deux entrées.

$\tilde{e}_1$	$\tilde{e}_2$
1	0
0	1
1	1

**Tableau III.1** Exemple de combinaisons des signaux d'excitation

### III.3.3.2 Construction de la base d'apprentissage

La construction de la base de données d'apprentissage dépend de la procédure d'excitation adoptée. On peut par exemple envisager d'exciter simultanément tous les canaux d'entrée du contrôleur multivariable par le même signal, ou combiner les excitations appliquées en considérant chaque canal d'entrée séparément, ou encore deux canaux à la fois. Ainsi, des matrices de données de différentes tailles peuvent être construites. Après chaque test d'excitation s'étalant sur une période égale à  $(N - 1) \times T_{slt}$ , où  $N$  est le nombre de mesures et  $T_{slt}$  est la période de mesure, on obtient les matrices des

mesures d'entrée et de sortie  $X_{new}$  et  $U_{new}$ , suivantes :

$$X_{new} = \begin{bmatrix} x_1(1) & x_2(1) & \dots & x_{n_x}(1) \\ x_1(2) & x_2(2) & \dots & x_{n_x}(2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(N) & x_2(N) & \dots & x_{n_x}(N) \end{bmatrix} \quad (\text{III.17})$$

et

$$U_{new} = \begin{bmatrix} u_1(1) & u_2(1) & \dots & u_{n_u}(1) \\ u_1(2) & u_2(2) & \dots & u_{n_u}(2) \\ \vdots & \vdots & \ddots & \vdots \\ u_1(N) & u_2(N) & \dots & u_{n_u}(N) \end{bmatrix} \quad (\text{III.18})$$

L'actualisation des matrices d'entrée  $D_x$  et de sortie  $D_u$  est effectuée comme suit :

$$D_x = \begin{bmatrix} D_x \\ X_{new} \end{bmatrix}, \quad D_u = \begin{bmatrix} D_u \\ U_{new} \end{bmatrix} \quad (\text{III.19})$$

Ce mécanisme est répété autant de fois jusqu'à arriver à construire la base d'apprentissage adéquate. À chaque essai, des modifications sur les paramètres des signaux d'excitation et leurs combinaisons peuvent être effectuées après évaluation de l'indice de validité des modèles de commande obtenus.

### III.3.4 Validation de l'émulateur flou

Pour évaluer l'efficacité du contrôleur flou multivariable identifié, nous avons utilisé le VAF (Variance Accounted For). Le VAF est souvent utilisé pour vérifier l'exactitude d'un modèle, en comparant la sortie mesurée avec la sortie estimée du modèle. Le VAF entre la sortie mesurée  $u$  et la sortie estimée du modèle  $\hat{u}$  est défini comme suit :

$$\text{VAF} = \left( 1 - \frac{\text{var}(u - \hat{u})}{\text{var}(u)} \right) \quad (\text{III.20})$$

Le VAF de deux signaux identiques est 100%. S'ils sont différents, le VAF sera plus faible. Si  $u$  et  $\hat{u}$  ont plusieurs colonnes, le VAF est calculé pour chaque colonne en  $u$  et  $\hat{u}$  séparément.

## III.4 Conclusion

Dans ce chapitre, nous avons présenté une méthode d'identification destinée à l'émulation des contrôleurs multivariables d'ordre élevé par un modèle flou de Takagi-Sugeno à partir des mesures d'entrée/sortie. Les paramètres de prémisses et de conséquences du modèle flou TS sont identifiés en utilisant l'algorithme de coalescence soustractive. Des considérations d'implémentation de cette stratégie sont également données. Elles concernent plus particulièrement la nature des signaux d'excitation à utiliser, le choix de la structure du modèle de contrôle, ainsi que la construction de la base de données d'apprentissage. Toutes ces considérations sont d'un intérêt majeur pour réussir une éventuelle application. De toutes les manières, la procédure d'émulation étudiée est attrayante et très simple en comparaison avec d'autres procédés d'identification impliquant des techniques d'optimisation qui induisent des calculs longs et complexes.



# CHAPITRE IV

## Application à la commande multivariable d'un générateur de turbine à vapeur

### IV.1 Introduction

Dans ce chapitre, nous proposons d'examiner l'application de la stratégie d'émulation floue de lois de commande décrite dans le chapitre III à un système de génération de vapeur à circulation naturelle (chaudière) accouplé à un turbo-alternateur. L'ensemble composé du générateur de vapeur (GV), de la turbine et de l'alternateur est désigné le long de ce chapitre par « GV-turbine ». Un générateur de vapeur industriel représente une des composantes principales des centrales électriques, thermiques ou des stations de compression de gaz. La conduite du système GV-turbine nécessite une meilleure appréhension du comportement dynamique de chacun des équipements qui le constituent, à savoir le GV, la turbine et l'alternateur. Cet ensemble est généralement caractérisé par des dynamiques fortement non-linéaires, rendues d'autant plus complexes par la présence de fortes interactions entre les différentes variables du système, la dimension physique des sous-systèmes et les contraintes d'exploitation imposées. Le fonctionnement du système GV-turbine implique plusieurs boucles de commande de structures diverses qui assurent la régulation de différentes grandeurs comme la pression dans le ballon supérieur du GV, le niveau d'eau, le débit de vapeur, la puissance électrique, etc. La littérature rapporte plusieurs travaux axés sur l'analyse, la modélisation et la commande du système GV-turbine. Une multitude de techniques classiques ou avancées ont été développées et appliquées. Citons par exemple, le contrôle linéaire quadratique [24], la commande  $\mathcal{H}_\infty$  [26], le contrôle intelligent [25], etc. Notons que la synthèse de lois de commandes robustes et optimales a suscité une attention particulière vue le nombre important d'études réalisées dans ce sens. Toutefois, les contrôleurs qui en résultent sont généralement d'ordre élevé qui nécessitent d'être simplifiés avant de les implémenter. La réduction de l'ordre du contrôleur et la simplification de sa structure influencent bien évidemment les caractéristiques dynamiques ou statiques du système de commande. En effet, cela peut altérer le contrôleur ainsi que les performances du système GV-turbine, surtout si ce dernier est manipulé sur une large zone de fonctionnement. Les difficultés d'implémentation des stratégies de commande multivariable robuste et optimale sont examinées dans la suite à travers l'étude du problème de commande multivariable d'un système GV-turbine. Afin de surmonter cette problématique, une alternative de com-

mande basée sur les systèmes à inférence floue est proposée. Plus précisément, il s'agit de concevoir un système de commande floue qui doit émuler le comportement multivariable du contrôleur original de sorte que les principales performances de l'ensemble du système en boucle fermée soient préservées et/ou optimisées.

## IV.2 Modélisation du système GV-turbine

Une centrale à vapeur produit de l'énergie à partir de combustibles fossiles par l'intermédiaire de plusieurs processus de conversion d'énergie. L'énergie chimique du combustible fossile est transformée en énergie thermique par le GV, puis en énergie mécanique par la turbine, et, enfin, en énergie électrique par l'alternateur (Figure IV.1). En même temps, l'eau est vaporisée et condensée en alternance dans un circuit fermé qui suit un cycle thermodynamique.

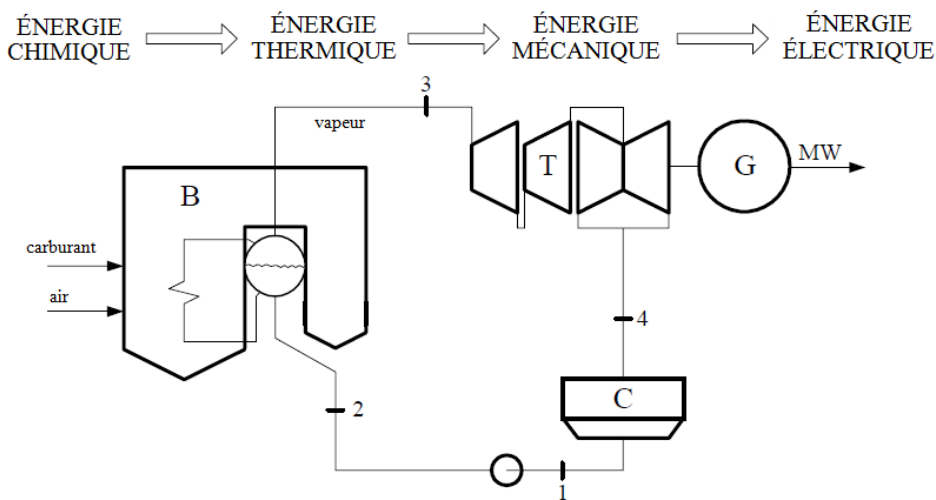
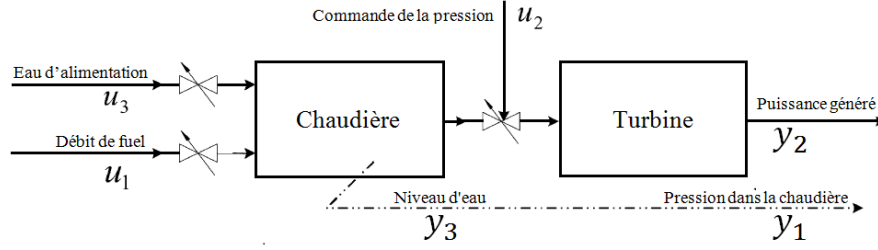


Figure IV.1 Centrale thermique à combustibles fossiles.

B = générateur de vapeur (chaudière), T = ensemble de turbines à vapeur,  
G = générateur électrique, C = condenseur.

Un premier effort de caractérisation de la dynamique complexe d'un système GV-turbine a été présenté dans [4, 6, 5], où un modèle non-linéaire du 2<sup>e</sup> ordre est développé décrivant la pression dans le GV et la sortie électrique. Dans [8] deux autres versions du modèle ont été proposées, par l'intégration du concept de taux d'évaporation [35] qui a permis de simplifier la prédiction du niveau d'eau dans le GV. Dans [3] les résultats d'une étude comparative de plusieurs modèles d'ordre réduit ont été présentés, où le modèle du 3<sup>e</sup> ordre s'est révélé d'une performance acceptable. Le modèle considéré dans ce travail (Figure IV.2) est celui développé par Bell et Astörm dans [3] pour une station électrique portant référence P16-G16 et qui permet de générer une puissance maximale de 160MW. Il s'agit d'un modèle d'état non-linéaire du troisième ordre exprimé par les équations suivantes :



**Figure IV.2** Illustration des variables du système GV-turbine intervenant dans le modèle de Bell et Astörm.

$$\begin{cases} \dot{x}_1 = -0.0018u_2x_1^{9/8} + 0.9u_1 - 0.15u_3 \\ \dot{x}_2 = (0.073u_2 - 0.016)x_1^{9/8} - 0.1x_2 \\ \dot{x}_3 = (141u_3 - (1.1u_2 - 0.19)x_1)/85 \\ y_1 = x_1 \\ y_2 = x_2 \\ y_3 = 0.05(0.13073x_3 + 100a_{cs} + q_e/9 - 67.975) \end{cases} \quad (\text{IV.1})$$

où les variables d'état  $x_1, x_2$  et  $x_3$  désignent la pression dans la chaudière en  $kg/cm^2$ , la puissance électrique en  $MW$ , la densité de fluide en  $Kg/m^3$ , respectivement. Les variables  $u_1, u_2$ , et  $u_3$  sont les positions des vannes de commande (en  $pu$ ) du débit du combustible, du débit de vapeur, et du débit d'eau d'alimentation.  $y_3$  est le niveau d'eau en  $m$ , avec  $a_{cs}$  et  $q_e$  sont donnés par :

$$a_{cs} = \frac{(1 - 0.001538x_3)(0.8x_1 - 25.6)}{x_3(1.0394 - 0.0012304x_1)} (Kg/s) \quad (\text{IV.2})$$

$$q_e = (0.854u_2 - 0.147)x_1 + 45.59u_1 - 2.514u_3 - 2.096 (Kg/s) \quad (\text{IV.3})$$

Au point de fonctionnement nominal,

$$x^0 = \begin{bmatrix} 108 \\ 66.65 \\ 428 \end{bmatrix}, u^0 = \begin{bmatrix} 0.34 \\ 0.69 \\ 0.436 \end{bmatrix}, y^0 = \begin{bmatrix} 108 \\ 66.65 \\ 0 \end{bmatrix},$$

Le modèle d'état linéaire est donné par les matrices :

$$A = \begin{bmatrix} -0.0025087 & 0 & 0 \\ 0.069424 & -0.1 & 0 \\ -0.0066941 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0.9 & -0.34904 & -0.15 \\ 0 & 14.155 & 0 \\ 0 & -1.3976 & 1.6588 \end{bmatrix}, \quad (\text{IV.4})$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0.0063436 & 0 & 0.0047058 \end{bmatrix}, D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.25328 & 0.51240 & -0.01.3967 \end{bmatrix}$$

Les positions des vannes sont limitées dans l'intervalle  $[0, 1]$ , c-à-d,

$$0 \leq u_i \leq 1 \quad (i = 1, 2, 3) \quad (\text{IV.5})$$

### IV.3 Commande du système GV-turbine

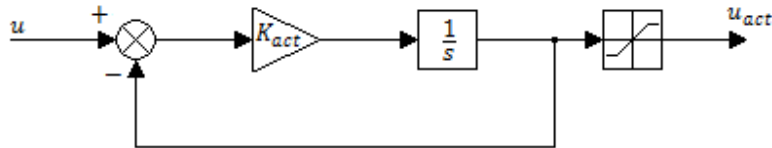
Comme il vient d'être souligné ci-dessus, la bonne conduite de l'ensemble GV-turbine est étroitement liée aux plateformes de régulation et de commande installées. Celles-ci intègrent en pratique une dizaine de boucles de régulation montées suivant différentes configurations. Plus généralement, un système de commande d'un GV-turbine doit nécessairement réaliser une bonne stabilisation des variables significatives du processus autour des conditions d'opération nominales. Il s'agit de contrôler simultanément la pression dans le ballon supérieur du GV, le niveau d'eau et la puissance électrique générée à la sortie de l'alternateur. D'autres grandeurs peuvent intervenir dans les différents schémas de commande. Dans notre étude, nous allons nous limiter à ces trois variables qui sont en fait déterminantes car elles sont explicitement représentées dans le modèle non-linéaire (IV.1). Un contrôleur multivariable pour le système GV-turbine est appelé à réaliser une régulation acceptable de ces trois variables à travers des actions bien adaptées sur les vannes de commande du fuel  $u_1$ , du débit de vapeur  $u_2$  et de l'alimentation en eau  $u_3$ , tout en respectant les limites de variations imposées sur chacun des canaux d'entrée. Il convient de noter que le couplage, d'abord de nature physique, qui existe entre ces différentes variables d'entrée et de sortie nécessite une prise en considération adéquate dans toute démarche de synthèse de lois de commande multivariable. Un autre problème qui caractérise le système GV-turbine est celui du contrôle du niveau d'eau dans le ballon supérieur du GV. La dynamique de l'eau est très complexe, voire difficilement contrôlable surtout pour les générateurs de vapeur industriels de grande dimension. À cette dynamique se voit associer un phénomène complexe nommé « le phénomène de gonflement et de trempe » qui influence considérablement le contrôle du niveau d'eau surtout durant les conditions transitoires. Ce phénomène est provoqué par des variations dans le rapport vapeur/liquide dans la section d'évaporation du générateur de vapeur d'eau. Lorsque la demande d'énergie augmente, le débit de vapeur augmente rapidement, ce qui fait baisser la pression dans le GV. Cette chute de pression stimule l'apparition de bulles d'air provoquant par conséquent l'augmentation du niveau d'eau. Le principe de l'équilibre de masse, exige que l'augmentation du débit de vapeur quittant le GV aura à provoquer une réduction de la masse totale à l'intérieur du GV. Ainsi, en maintenant constant l'apport d'eau d'alimentation, la masse d'eau à l'intérieur du ballon supérieur sera par la suite diminuée, ce qui fait tremper le niveau d'eau. Par contre, si la demande de vapeur d'eau est réduite, on remarque des bulles de vapeur d'eau qui commencent à rétrécir, ainsi le niveau d'eau augmente en raison du gonflement de la masse d'eau et de vapeur dans le ballon supérieur [27]. Le système GV-turbine fonctionne en pratique dans différentes conditions d'exploitation. Ainsi, le point de fonctionnement du système pourrait basculer d'une zone à une autre dans la plage admissible, chose qu'il faut tenir compte lors de la synthèse des commandes. Le tableau IV.1 indique quelques points de fonctionnement pour le modèle de Bell et Aström.

Les actionneurs des vannes de commande du fuel  $u_1$ , du débit de vapeur  $u_2$  et de l'alimentation en eau  $u_3$  sont soumis à des contraintes opératoires associées à leurs

	#1	#2	#3	#4	#5	#6	#7
$x_1^0$	75.60	86.40	97.20	108	118.8	129.6	140.4
$x_2^0$	24.72	36.43	50.39	66.65	83.64	104.41	127.56
$x_3^0$	300	342	385	428	471	513	556
$u_1^0$	0.155	0.208	0.270	0.34	0.412	0.499	0.59
$u_2^0$	0.483	0.552	0.621	0.69	0.759	0.828	0.897
$u_3^0$	0.183	0.256	0.340	0.436	0.535	0.654	0.785
$y_3^0$	-0.97	-0.65	-0.32	0	0.32	0.64	0.98

**Tableau IV.1** Points de fonctionnements typiques pour le modèle de Bell et Aström.

dynamiques et à l'amplitude des signaux qu'ils génèrent. Le modèle d'actionneur de vanne est schématisé à la Figure IV.3. Il s'agit d'un modèle du premier ordre caractérisé par une constante de temps  $\tau_{act} = 1/K_{act}$  avec un bloc de saturation. Les valeurs des constantes de temps indiquées dans le tableau IV.2 sont déterminées à partir de plusieurs simulations.


**Figure IV.3** Schéma-bloc d'un actionneur de vanne.

Vanne	Position	Constante de temps
Fuel	$0 \leq u_1 \leq 1$	0.1
Vapeur	$0 \leq u_2 \leq 1$	0.1
Eau d'alimentation	$0 \leq u_3 \leq 1$	0.1

**Tableau IV.2** Caractéristiques des actionneurs.

## IV.4 Synthèse des stratégies de commande multivariable

La conception de lois de commandes multivariables pour le système non-linéaire GV-turbine constitue toujours l'objet majeur de plusieurs travaux récents [27, 35, 20, 43]. L'une des principales difficultés est le contrôle du niveau d'eau dans le GV, qui doit être maintenu constant quelque soient les fluctuations ou les variations des autres variables contrôlées du système. La dynamique de l'eau est théoriquement à non-minimum de phase, en raison des effets du gonflement et de trempe, et varie considérablement avec

les conditions d'exploitation. La plupart des dispositifs de commande des systèmes GV-turbine actuellement en service sont synthétisés sur la base de la théorie de commande classique. Le système de commande à trois éléments est fréquemment utilisé dans la pratique [33, 18]. Cependant, il est toujours difficile de maintenir cette configuration aussi performante que possible vu les grandes et fréquentes variations des débits d'eau et de vapeur, ce qui entraîne souvent le contrôleur à se désadapter au procédé. Des conceptions plus avancées basées sur la théorie de commande robuste et optimale ont été proposées afin d'optimiser le contrôle de ce système. Dans la suite nous allons nous intéresser à deux types d'entre-elles avant d'examiner leur émulation par systèmes flous.

#### IV.4.1 Commande $\mathcal{H}_\infty$

La procédure de conception du contrôleur  $\mathcal{H}_\infty$  décrite dans le Paragraphe I.4.2 est appliquée au système GV-turbine. Elle est menée en deux étapes successives. La première étape consiste en la mise en forme du gain du système en boucle ouverte afin de répondre aux objectifs spécifiques de synthèse. Ceci peut être réalisé en améliorant le gain et le roll-off dans les trois bandes de fréquences. Pour simplifier la tâche de sélection des paramètres, on a choisi  $W_2 = I_3$  et  $W_g = I_3$ , alors que le pré-compensateur  $W_1$  est déterminé à partir des matrices  $W_p$  et  $W_a$ . Le graphe des valeurs singulières de la boucle ouverte du modèle nominal est représenté dans la Figure IV.4. D'après cette figure, il apparaît clairement que le comportement de la boucle ouverte du GV-turbine est mal conditionné. Nous choisissons alors les deux matrices de manière à garantir les exigences suivantes :

- Un gain élevé dans les basses fréquences ; afin de minimiser l'effet des perturbations.
- Un roll-off d'environ  $20dB$  (Une pente d'environ  $-1$ ) dans la bande passante choisie.
- Un roll-off plus élevé à des fréquences élevées ; pour tronquer les effets du bruit de mesure.

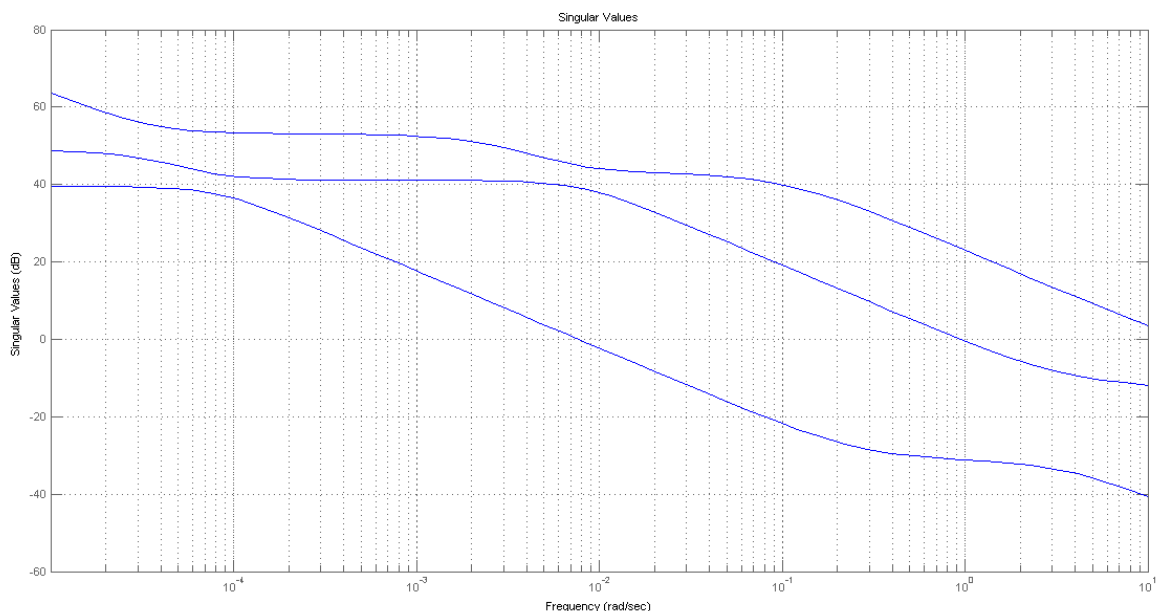


Figure IV.4 Allures des valeurs singulières du modèle nominal.

La matrice  $W_p$  est choisie pour contenir la partie dynamique du compensateur, il est clair qu'elle doit avoir un intégrateur, pour améliorer le roll-off sur certaines gammes de fréquences. La partie numérateur est de la forme  $l_1s + l_2$ , permettant ainsi d'ajuster l'effet intégrateur dans des gammes où il est non désirable.

Dans la deuxième étape, nous appliquons la procédure  $\mathcal{H}_\infty$ , afin de déterminer un contrôleur  $K_\infty$  qui garantit stabilisation du système  $GW_1$ . Après plusieurs simulations, une marge de stabilité acceptable est déterminée. Elle est évaluée à  $\epsilon_{max} = 0.40693$ . Ainsi, un contrôleur  $\mathcal{H}_\infty$  final d'ordre 8 est obtenu. Il est donné par les équations (IV.7), (IV.8), (IV.9), (IV.10). Les valeurs singulières modifiées sont représentées dans la Figure IV.5 où l'on constate clairement que le compensateur  $W_1$  décrit par le modèle (IV.6) induisait un effet efficace sur le comportement dynamique du GV-turbine.

$$W_1 = \begin{bmatrix} 0.001786 & 0.0036787 & 0.019153 \\ -0.0049044 & 0.0071280 & 0 \\ -0.00059152 & 0.0054860 & 0.11492 \end{bmatrix} \begin{bmatrix} 3(1 + \frac{1}{s}) & 0 & 0 \\ 0 & 3(1 + \frac{1}{s}) & 0 \\ 0 & 0 & 3(1 + \frac{1}{s}) \end{bmatrix} \quad (\text{IV.6})$$

$$A_{\mathcal{H}_\infty} = \begin{bmatrix} 0 & 0 & 0 & 0.00956 & -0.08963 & 0.05896 & -0.004356 & -0.005479 \\ 0 & 0 & 0 & -0.001859 & 0.2538 & 0.08116 & -0.005331 & 0.02018 \\ 0 & 0 & 0 & 0.3228 & 0.004761 & -0.02793 & -0.02488 & -0.0003623 \\ 0 & 0 & 0 & -0.5042 & 0.1175 & -0.006866 & -0.2375 & -0.0764 \\ 0 & 0 & 0 & 0.07175 & -2.407 & 0.002872 & -0.005572 & 1.184 \\ 0 & 0 & 0 & -0.001459 & 0.005451 & -0.1971 & 0.01184 & -0.006545 \\ 0 & 0 & 0 & 0.1112 & -0.007643 & 0.009093 & -0.04216 & -0.00348 \\ 0 & 0 & 0 & 0.02166 & -0.6578 & 0.002789 & -0.005557 & -0.408 \end{bmatrix}, \quad (\text{IV.7})$$

$$B_{\mathcal{H}_\infty} = \begin{bmatrix} -0.2291 & -0.00579 & 0.01772 \\ -0.3159 & -0.007983 & 0.01999 \\ 0.1109 & 0.002803 & -0.01307 \\ -0.1067 & 0.1388 & -1.468 \\ 0.03996 & -2.29 & -0.05464 \\ 0.6142 & 0.01544 & -0.04734 \\ -0.03124 & -0.0001901 & -0.05445 \\ -0.0001973 & 0.0934 & -0.002491 \end{bmatrix}, \quad (\text{IV.8})$$

$$C_{\mathcal{H}_\infty} = \begin{bmatrix} 0.0428 & 0.044 & 0.1149 & 0.0374 & 0.007905 & 0.002901 & -0.0033 & 0.00061 \\ -0.117 & 0.085 & 0 & -0.00128 & 0.03226 & 0.00000226 & 0.000057 & 0.00237 \\ -0.014 & 0.066 & 0.6895 & 0.222 & 0.02126 & -0.01475 & -0.017 & 0.00116 \end{bmatrix}, \quad (\text{IV.9})$$

$$D_{\mathcal{H}_\infty} = \begin{bmatrix} -0.01102 & -0.0002786 & 0.0001397 \\ -0.00005511 & -0.000001393 & -0.0003757 \\ 0.05894 & 0.001489 & -0.00795 \end{bmatrix} \quad (\text{IV.10})$$

Le schéma de simulation de la stratégie de commande  $\mathcal{H}_\infty$  implémenté dans l'environnement SIMULINK de MATLAB est illustré par la Figure IV.6.

L'exécution de cette simulation avec des conditions initiales nulles peut provoquer des erreurs de calcul dans la phase initiale. Il faut alors dimensionner les valeurs initiales du contrôleur  $\mathcal{H}_\infty$  avant de l'implémenter. Pour ce faire, on suppose que le vecteur d'entrée  $e$  du contrôleur est initialement nul. Par conséquent, le problème se résume au calcul de la solution  $x_c^0$  du système linéaire (IV.11) sous la contrainte

$$u^0 = C_c x_c^0, \text{ avec } A_c x_c^0 = 0_3 \quad (\text{IV.11})$$

Par l'utilisation de la pseudo inverse, on trouve les valeurs initiales comme suit :

$$x_c^0 = \begin{bmatrix} 0 \\ 8.0668 \\ -0.13810 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{IV.12})$$

Évidemment, il n'est pas facile de réaliser l'implémentation pratique de ce contrôleur multivariable d'ordre élevé. Ainsi, une approximation ou une réduction d'ordre est exigée. Dans la suite, on examine l'application de la méthode d'approximation par modèle de contrôle PID multivariable présentée dans le Paragraphe I.5.

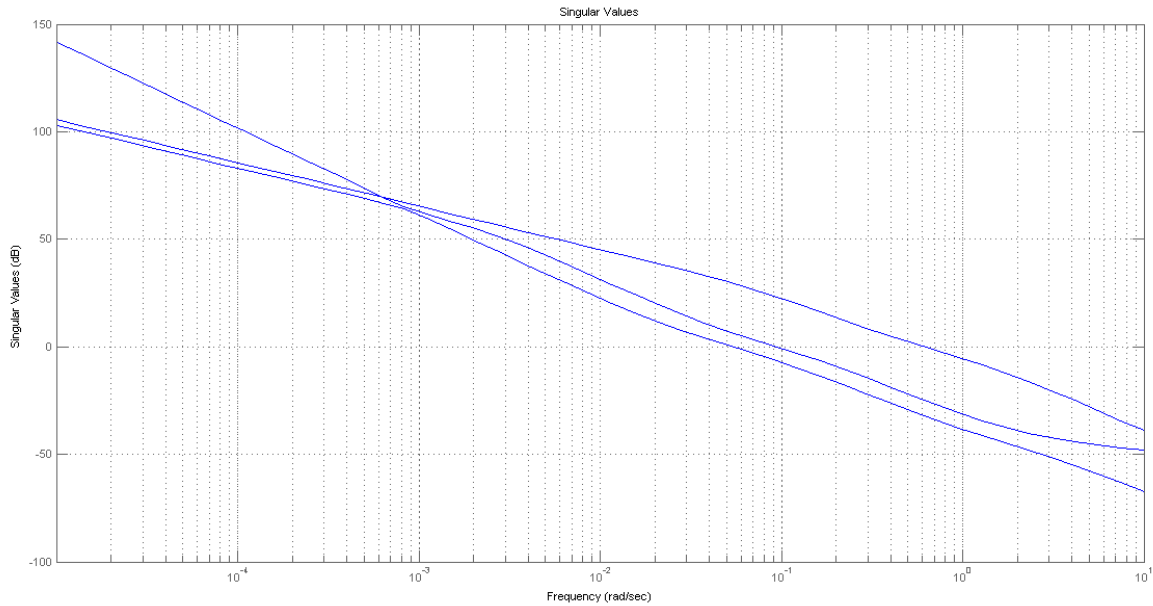


Figure IV.5 Allures des valeurs singulières du système augmenté  $GW_1$ .

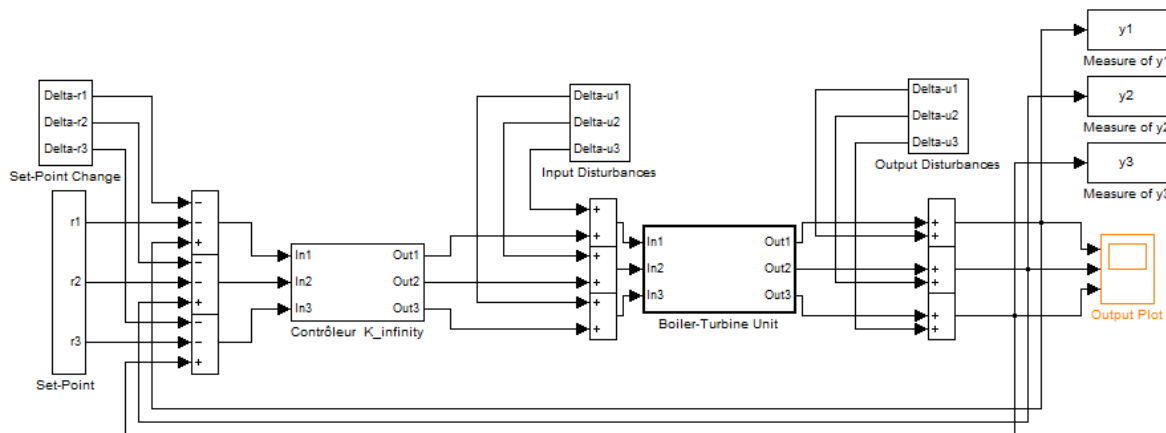


Figure IV.6 Configuration du contrôle  $\mathcal{H}_\infty$  appliqué au système GV-turbine.



## IV.4.2 Contrôle PID multivariable

Pour des considérations pratiques, l'implémentation du contrôleur  $\mathcal{H}_\infty$  d'ordre 8 déterminée ci-dessus ne peut être effectuée de manière aisée. Dans la plupart des cas, une approximation par PID multivariable est calculée à partir du contrôleur  $\mathcal{H}_\infty$  d'ordre élevé original. Par application de la méthode de réduction décrite dans le Paragraphe I.5, nous avons abouti à la réalisation suivante :

$$K_{PID}(s) = K_p + K_i/s + K_d s \quad (\text{IV.13})$$

avec

$$K_p = \begin{bmatrix} -0.086317 & -0.0043676 & -0.9.0306 \\ -0.00092067 & -0.03.4227 & -0.00036657 \\ 0.016433 & 0.0080507 & -5.4193 \end{bmatrix}, K_i = \begin{bmatrix} -0.0044149 & -0.0031139 & -0.025880 \\ 0.00054895 & -0.012759 & -0.000087234 \\ -0.0019507 & -0.0068304 & -0.15493 \end{bmatrix},$$

$$K_d = \begin{bmatrix} 0.53713 & -0.00097536 & 9.7874 \\ 0.006.9111 & 0.020136 & -0.0063916 \\ 1.1528 & -0.039462 & 58.675 \end{bmatrix} \quad (\text{IV.14})$$

Cette approximation peut être simplifiée davantage en considérant l'ensemble des remarques décrites dans I.5, ce qui conduit au contrôleur PI multivariable de la forme :

$$K_{PI} = \begin{bmatrix} 0.08632 + \frac{0.004415}{s} & 0.004368 + \frac{0.003114}{s} & 0.9031 + \frac{0.02588}{s} \\ 0 & 0.03423 + \frac{0.01276}{s} & 0 \\ 0 & 0 & 5.419 + \frac{0.1549}{s} \end{bmatrix} \quad (\text{IV.15})$$

Les allures des valeurs singulières représentées dans la figure IV.7 montrent clairement que le PI multivariable reproduit les mêmes caractéristiques fréquentielles du contrôleur  $\mathcal{H}_\infty$  dans la bande de fonctionnement du système GV-turbine, ce qui permet de préserver les mêmes performances du contrôleur original, notamment en termes de rejet de perturbations et de poursuite de références.

L'action intégrale du contrôleur PI avec une saturation pourrait introduire un effet indésirable dans la boucle de commande. Ce phénomène est identifié sous le nom *Windup*. Les points suivants résument la façon dont ce phénomène émerge :

- La différence entre la référence d'entrée et la sortie du système génère une grande erreur.
- En conséquence, le contrôleur PI agit en appliquant une valeur de sortie en fonction des gains du PI. L'action intégrale commence à accumuler l'erreur, ce qui augmente sa valeur.
- Par la suite, la valeur de sortie du contrôleur PI, principalement en raison de la grandeur d'intégrale accumulée, peut être plus grande que le niveau de la limite de saturation. Sous cette condition, les blocs de saturation fournissent la valeur maximale tolérable au système.

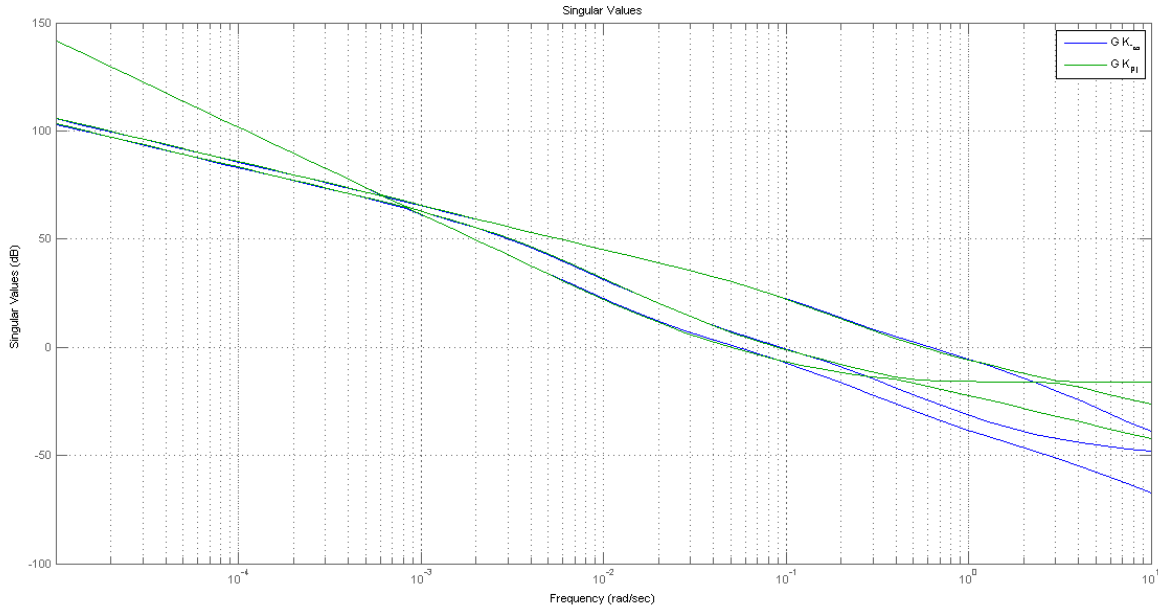


Figure IV.7 Allures des valeurs singulières de  $GK_\infty$  et  $GK_{PI}$ .

- Une fois la sortie réelle du système atteint sa référence, l'erreur est de nouveau à zéro, néanmoins la valeur accumulée reste plus élevée que la limite de saturation.

L'objectif principal de l'utilisation de la configuration *anti-windup* est d'éviter la valeur accumulée de l'action intégrale, donc la sortie de l'intégration sera maintenue dans une gamme limitée. La figure IV.8 représente un anti-windup de poursuite, cette procédure utilise la différence entre l'entrée et la sortie du bloc de saturation avec un gain  $K_{AW}$  pour réduire la valeur de l'intégrateur.

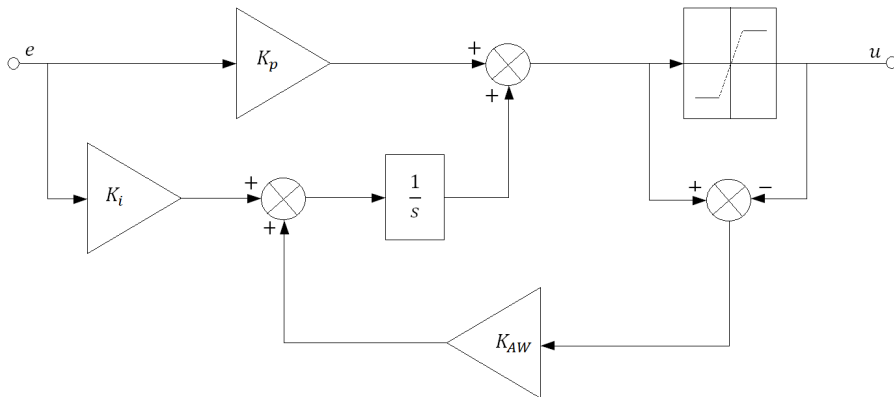


Figure IV.8 Schéma de configuration d'un PI avec un anti-windup.

La configuration du contrôle PI multivariable du système GV-turbine implémentée en simulation est montrée dans la Figure IV.9.

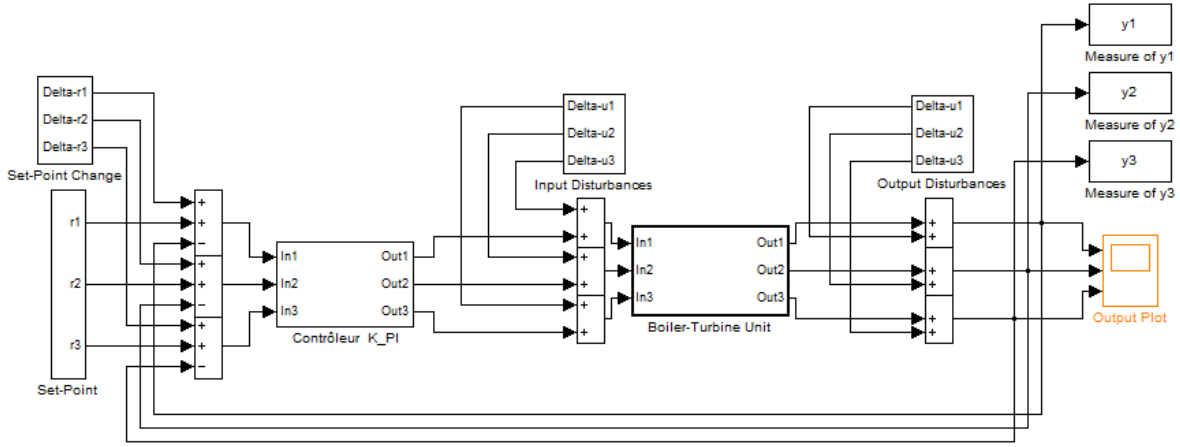


Figure IV.9 Configuration du contrôleur PI multivariable.

### IV.4.3 Émulation floue du contrôleur $\mathcal{H}_\infty$

#### IV.4.3.1 Synthèse de l'émulateur flou

La synthèse d'un émulateur flou pour le contrôleur  $\mathcal{H}_\infty$  décrit par les équations (IV.7), (IV.8), (IV.9), (IV.10) est établie sur la base de la stratégie d'émulation par systèmes d'inférence floue présentée dans le chapitre III. Son application au contrôleur  $\mathcal{H}_\infty$  conçu pour le système GV-turbine se résume aux points suivants :

- Choisir les variables d'entrée et de sortie de l'émulateur flou : Comme variables d'entrée six grandeurs sont retenues qui sont  $e_1(k)$ ,  $e_2(k)$ ,  $e_3(k)$ ,  $u_1(k-1)$ ,  $u_2(k-1)$  et  $u_3(k-1)$ . Les variables de sortie sont  $u_1(k)$ ,  $u_2(k)$  et  $u_3(k)$ .
- Établir la structure du modèle flou à identifier à partir des données d'entrée et de sortie : L'émulateur flou à concevoir est représenté par un modèle flou de type Takagi-Sugeno exprimé linguistiquement par :

$$\begin{aligned}
 & \text{If } e_1(k) \text{ is } E_1 \text{ and } e_2(k) \text{ is } E_2 \text{ and } e_3(k) \text{ is } E_3 \text{ and} \\
 & u_1(k-1) \text{ is } U_1 \text{ and } u_2(k-1) \text{ is } U_2 \text{ and } u_3(k-1) \text{ is } U_3 \\
 & \text{then } u_1(k) = f_1(e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1)) \text{ and} \\
 & u_2(k) = f_2(e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1)) \text{ and} \\
 & u_3(k) = f_3(e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1))
 \end{aligned}$$

où  $e_i(k)$  et  $u_i(k-1)$ ,  $i = 1, 2, 3$  sont la variable d'erreur et la variable de commande retardée, respectivement.  $E_i$  et  $U_i$  sont les ensembles flous associés.  $f_i$  est une fonction arbitraire associée à chaque variable de sortie  $u_i(k)$  et dépend du vecteur d'entrée  $x(k) = (e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1))$ .

- Générer la base de données d'apprentissage : C'est une étape clef qui nécessite une attention particulière. Après plusieurs essais en simulation, les séquences binaires pseudo-aléatoires (PRBS) ont été sélectionnées pour servir de signaux d'excitation. Elles sont appliquées à chaque canal d'entrée selon une combinaison bien adaptée répondant aux paramètres suivants : nombre de registres de décalage  $n_{sr} = 20$ , le temps de commutation  $T_{sw} = 0.8s$ , et l'amplitude du signal d'excitation  $\tilde{e} = \pm 10$  (voir Annexe I). Ainsi, un ensemble de données d'entrée/sortie composé de 26026 échantillons a été généré pour chaque variable de sortie.

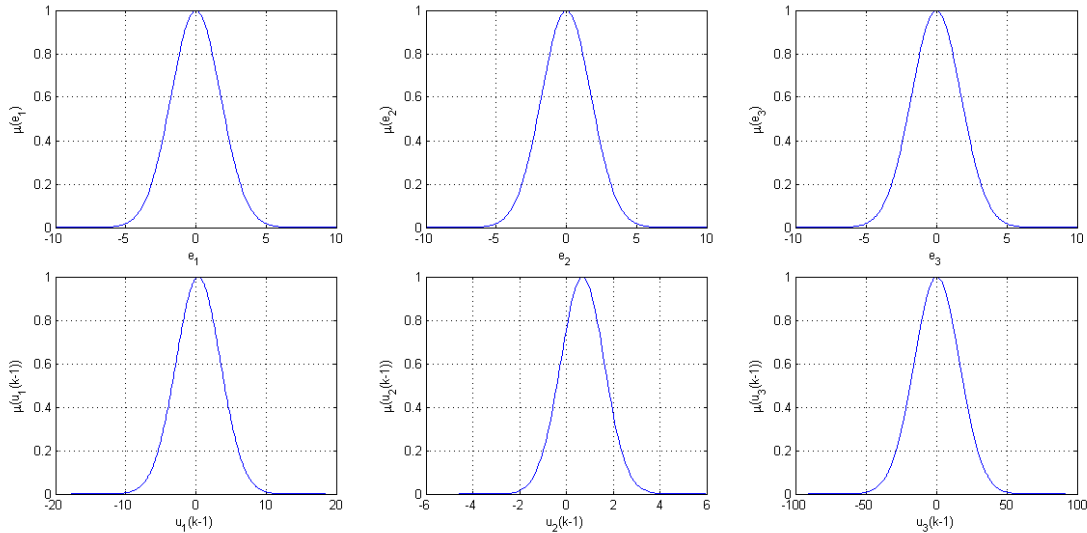
- Déterminer la structure et les paramètres de l'émulateur flou : Cette étape est réalisée à l'aide de l'algorithme de coalescence soustractive qui permet de générer systématiquement les paramètres des prémisses et des conclusions du modèle TS flou ainsi que le nombre de règles. Les paramètres d'application de cet algorithme aux données d'apprentissage construites à partir du modèle de contrôle  $\mathcal{H}_\infty$  sont :  $r_a = 0.25$ ,  $r_b/r_a = 1.25$ ,  $\bar{\epsilon} = 0.5$  et  $\underline{\epsilon} = 0.15$ . Avec ces paramètres, les expressions analytiques des lois de commande  $u_1(k)$ ,  $u_2(k)$  et  $u_3(k)$  que génère l'émulateur flou sont données par :

$$u_1(k) = -0.002033 e_1(k) - 0.0005364 e_2(k) - 0.007117 e_3(k) + 0.9941 u_1(k-1) + 0.0006964 u_2(k-1) + 0.0006909 u_3(k-1) + 0.001231$$

$$u_2(k) = 0.0000499 e_1(k) - 0.002806 e_2(k) - 0.0000412 e_3(k) - 0.0002137 u_1(k-1) + 0.9976 u_2(k-1) + 0.00003398 u_3(k-1) + 0.001706$$

$$u_3(k) = 0.005666 e_1(k) - 0.0005078 e_2(k) - 0.04338 e_3(k) + 0.02479 u_1(k-1) - 0.002584 u_2(k-1) + 0.994 u_3(k-1) - 0.004045$$

Les partitions floues gaussiennes résultantes sont représentées dans la Figure IV.10. Leurs paramètres sont résumés dans le tableau IV.3.



**Figure IV.10** Les ensembles flous gaussiens associés à chacune des variables d'entrée.

$MF$	$E_1$	$E_2$	$E_3$	$U_1$	$U_2$	$U_3$
[Sigma c]	[1.768 0]	[1.768 0]	[1.768 0]	[3.19 0.3402]	[0.9311 0.69]	[16.14 0.4358]

**Tableau IV.3** Paramètres des prémisses de l'émulateur flou du contrôleur  $\mathcal{H}_\infty$ .

### IV.4.3.2 Validation de l'émulateur flou

Pour tester la capacité d'approximation de l'émulateur flou développé pour le contrôleur  $\mathcal{H}_\infty$ , l'indice VAF (Variance Accounted For) est évalué pour chacune des variables de commande. Les valeurs déterminées sont 99.614%, 96.352% et 99.638% pour les positions des vannes de commande du débit du combustible, du débit de vapeur, et du débit d'eau d'alimentation, respectivement. Ainsi, on conclut que le modèle flou reproduit considérablement bien le comportement du contrôleur  $\mathcal{H}_\infty$  original comme le montre la Figure IV.11.

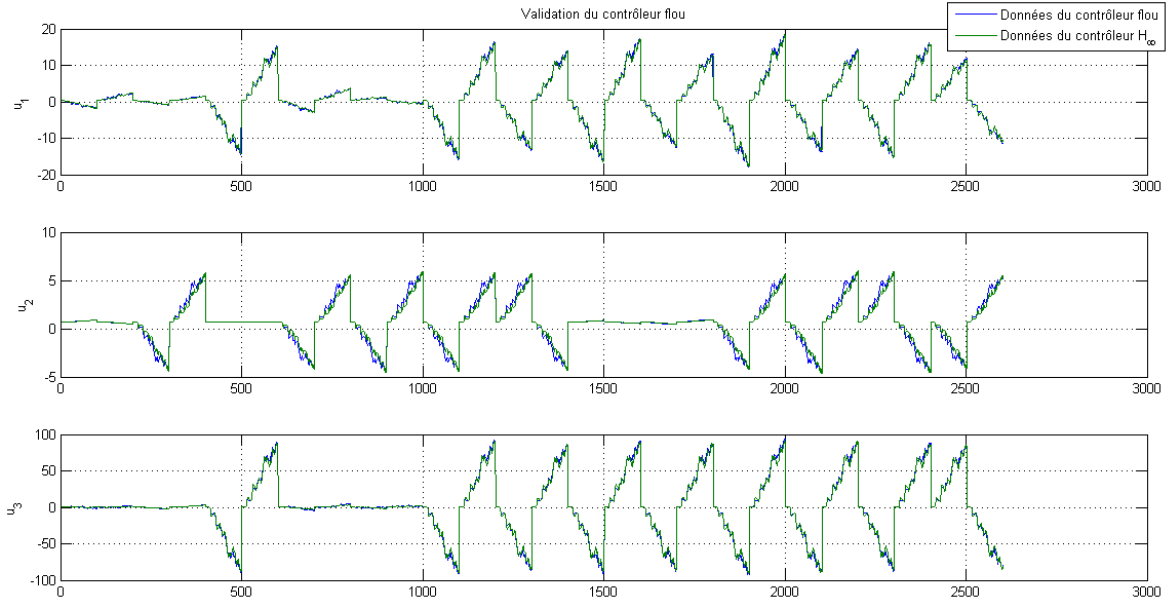


Figure IV.11 Performance du contrôleur flou par apport au contrôleur  $\mathcal{H}_\infty$  original (données d'apprentissage).

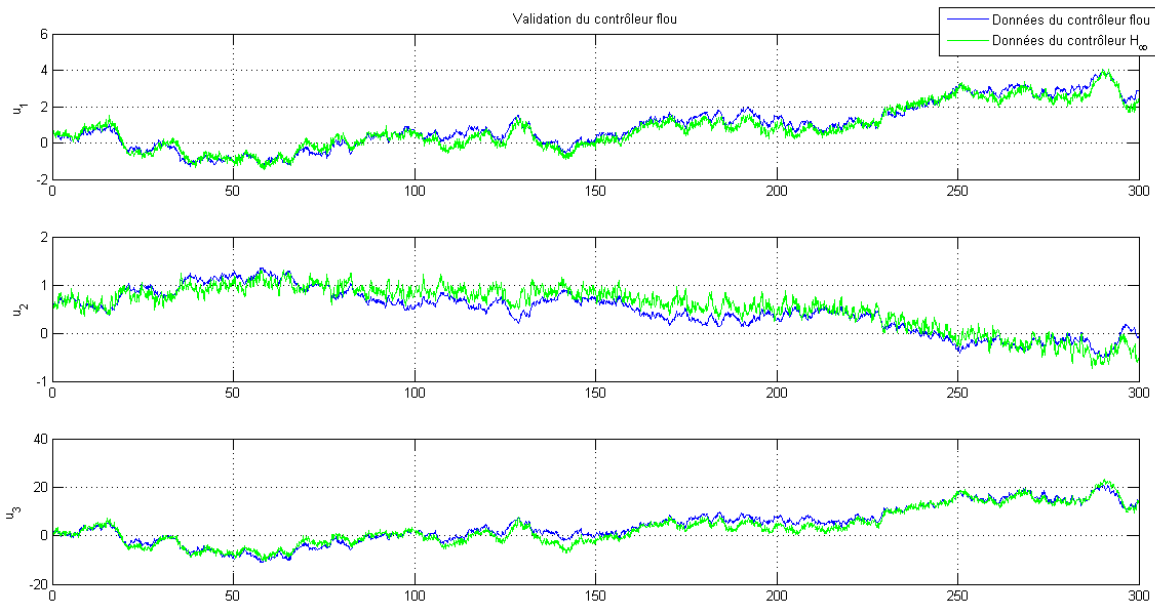


Figure IV.12 Performance du contrôleur flou par apport au contrôleur  $\mathcal{H}_\infty$  original (données de test).

Pour tester la capacité de prédiction de l'émulateur flou, on applique des signaux d'excitation différents de ceux employés dans la phase d'apprentissage. La Figure IV.12 illustre les résultats de la validation. Les réponses des deux contrôleurs sont visiblement concordantes à quelques différences près, réalisant ainsi une capacité de généralisation très appréciable. Les valeurs des VAF calculés pour  $u_1$ ,  $u_2$  et  $u_3$  sont 94.36%, 83.32% et 94.46%, respectivement.

#### IV.4.3.3 Résultats de simulation

Pour évaluer la performance du système flou émulant le contrôleur  $\mathcal{H}_\infty$ , des simulations sont effectuées sur le modèle non-linéaire du système GV-turbine dans différentes situations d'exploitation [25]. Les cas présentés ici visent à montrer et évaluer la capacité d'approximation du contrôleur flou multivariable dans des situations de rejet de perturbation, de poursuite de références et dans des cas de changement du point de fonctionnement et de variations paramétriques.

Avant de considérer chacun de ces cas, il est utile de montrer l'aptitude de l'émulateur flou à reproduire les caractéristiques du contrôleur  $\mathcal{H}_\infty$  en absence de perturbations sur le système GV-turbine. C'est justement ce que montre la Figure IV.13 où l'on voit clairement que toutes les variables contrôlées du système GV-turbine maintiennent l'état d'équilibre autour du point de fonctionnement nominal.

##### a) Rejet de perturbation

Cas 1 : Á  $t = 10s$ ,  $\Delta u_1 = 10\%$ , puis à  $t = 200s$ ,  $\Delta y_1 = 10Kg/cm^2$ , ensuite à  $t = 500s$ ,  $\Delta r_1 = -10Kg/cm^2$ . Les réponses sont représentées dans la Figure IV.14.

Cas 2 : Á  $t = 10s$ ,  $\Delta u_2 = 10\%$ , puis à  $t = 200s$ ,  $\Delta y_2 = 10MW$ , ensuite à  $t = 500s$ ,  $\Delta r_2 = -10MW$ . Les réponses sont représentées dans la Figure IV.15.

Cas 3 : Á  $t = 10s$ ,  $\Delta u_3 = 10\%$ , puis à  $t = 200s$ ,  $\Delta y_3 = 0.1m$ , ensuite à  $t = 500s$ ,  $\Delta r_3 = 0.1m$ . Les réponses sont représentées dans la Figure IV.16.

##### b) Changement du point de fonctionnement

Cas 1 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) + 10\%$ . Les réponses sont représentées dans la Figure IV.17.

Cas 2 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) - 10\%$ . Les réponses sont représentées dans la Figure IV.18.

Cas 3 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) + 20\%$ . Les réponses sont représentées dans la Figure IV.19.

Cas 4 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) - 20\%$ . Les réponses sont représentées dans la Figure IV.20.

### c) Changement de consignes

Cas 1 : Étant en régime nominal, on introduit la consigne :  $\#(120, 120, 0)$ . Les réponses sont indiquées dans la Figure IV.21.

Cas 2 : Étant en régime nominal, on introduit la consigne :  $\#(150, 90, 0)$ . Les réponses sont indiquées dans la Figure IV.22.

### d) Test de demande supplémentaire en puissance électrique

Cas 1 : Étant en régime nominal, on introduit la consigne :  $\#(110, 80, 0)$ . Les réponses sont indiquées dans la Figure IV.23.

Cas 2 : Étant en régime nominal, on introduit la consigne :  $\#(120, 100, 0)$ . Les réponses sont indiquées dans la Figure IV.24.

Cas 3 : Étant en régime nominal, on introduit la consigne :  $\#(130, 120, 0)$ . Les réponses sont indiquées dans la Figure IV.25.

### e) Variations paramétriques

Cas 1 : Dans ce test, on modifie la valeur nominale du débit massique d'évaporation  $q_e = 60.084 Kg/s$ , vers  $q_e/10$ . Les réponses sont données en figure IV.26.

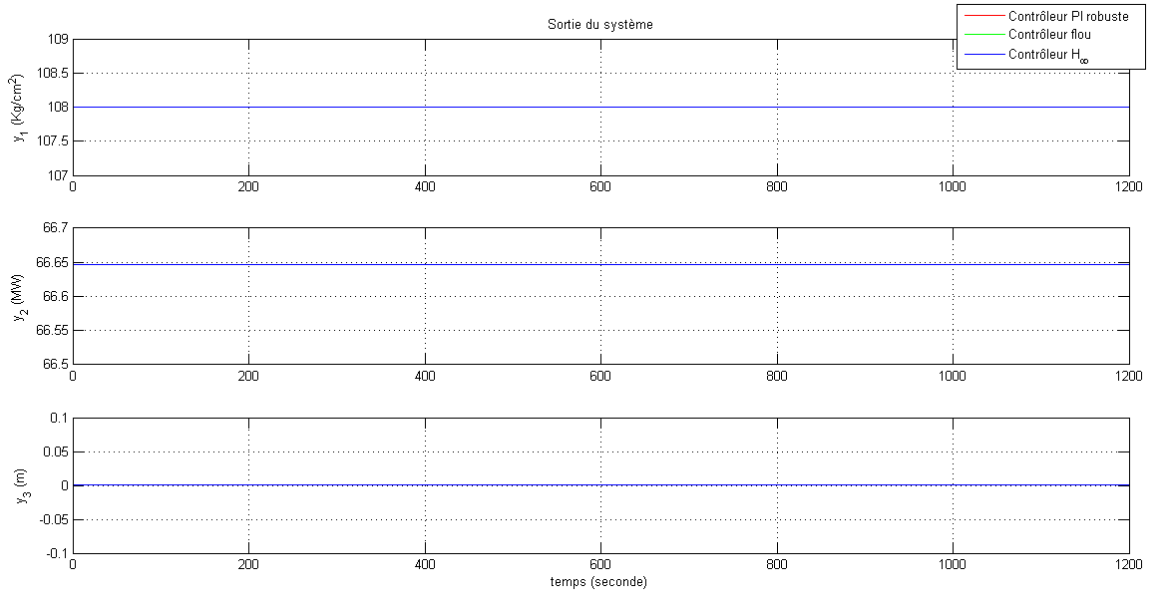
Cas 2 : Dans ce test, on modifie la valeur nominale du facteur de qualité de la vapeur d'eau  $a_{cs} = 0.0535 Kg/s$ , vers  $a_{cs}/2$ . Les réponses sont données en figure IV.27.

#### IV.4.3.4 Analyse des résultats

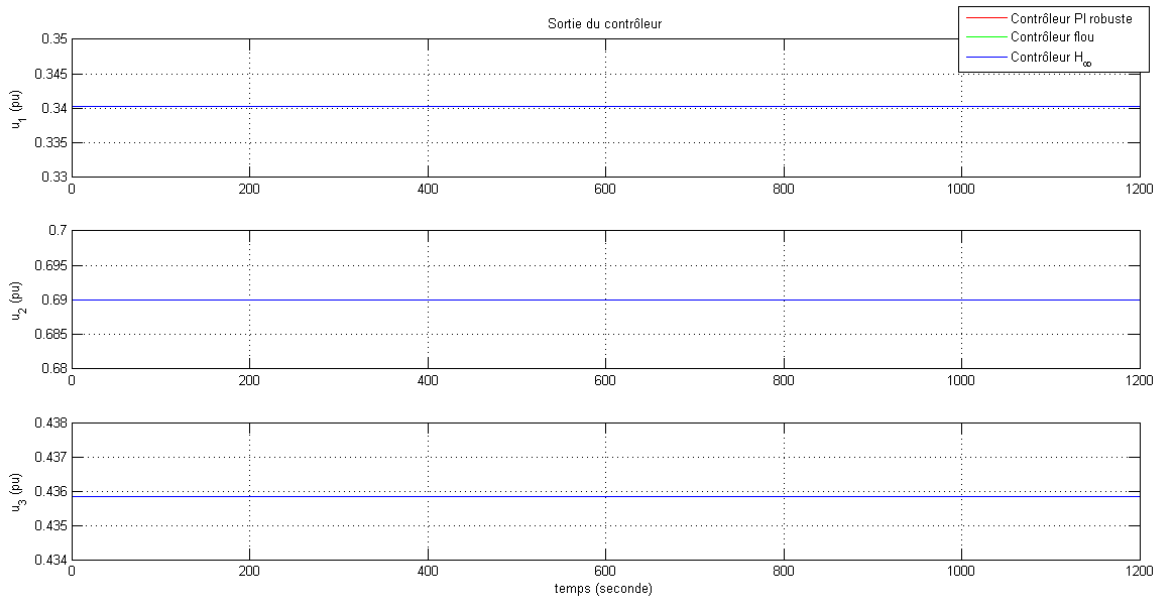
Les tests de rejet de perturbations appliqués à l'entrée et à la sortie du système GV-turbine ont démontré l'aptitude du contrôleur flou à compenser les effets de ces perturbations, comme le montrent les Figures IV.14 à IV.16. Les lois de commande floue déduites à partir de l'émulation du comportement dynamique du contrôleur  $\mathcal{H}_\infty$  arrivent bien à stabiliser le système non linéaire au voisinage du point de fonctionnement nominal. Certaines déviations sont constatées sur les Figures IV.14 et IV.15, mais celles-ci restent très proches du point nominal.

Dans les tests de changement du point de fonctionnement (Figures IV.17 à IV.20), une bonne capacité d'approximation est réalisée. En effet, le contrôleur flou multivariable reproduit considérablement bien le comportement dynamique du contrôleur  $\mathcal{H}_\infty$  original. Cependant, quelques légères fluctuations au niveau de l'évolution de la puissance électrique de sortie et du niveau d'eau sont observées. Elles restent néanmoins dans les limites admissibles sans effets majeurs sur la performance du système de commande.

Les simulations de changement de consignes représentent plus qu'une bonne approximation. Le contrôleur flou émulateur du contrôleur  $\mathcal{H}_\infty$  a montré une robustesse aux changements de références, alors que le contrôleur PI a échoué totalement de garantir la stabilité du système en boucle fermée, comme le montre bien les Figures IV.21 à IV.22.



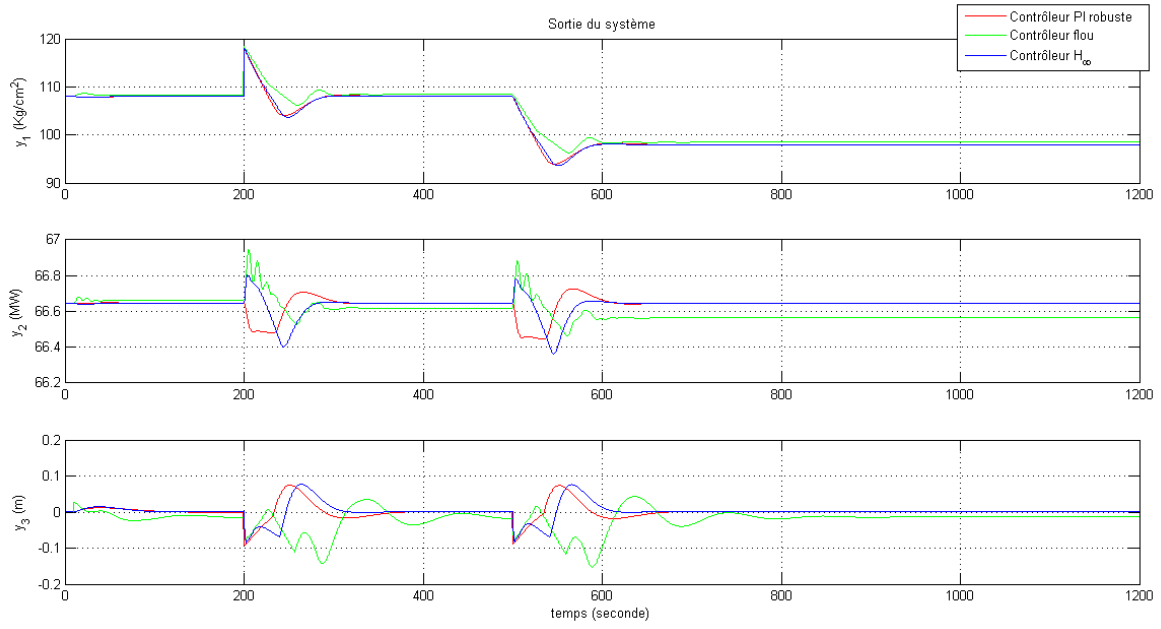
(a) Sortie du système



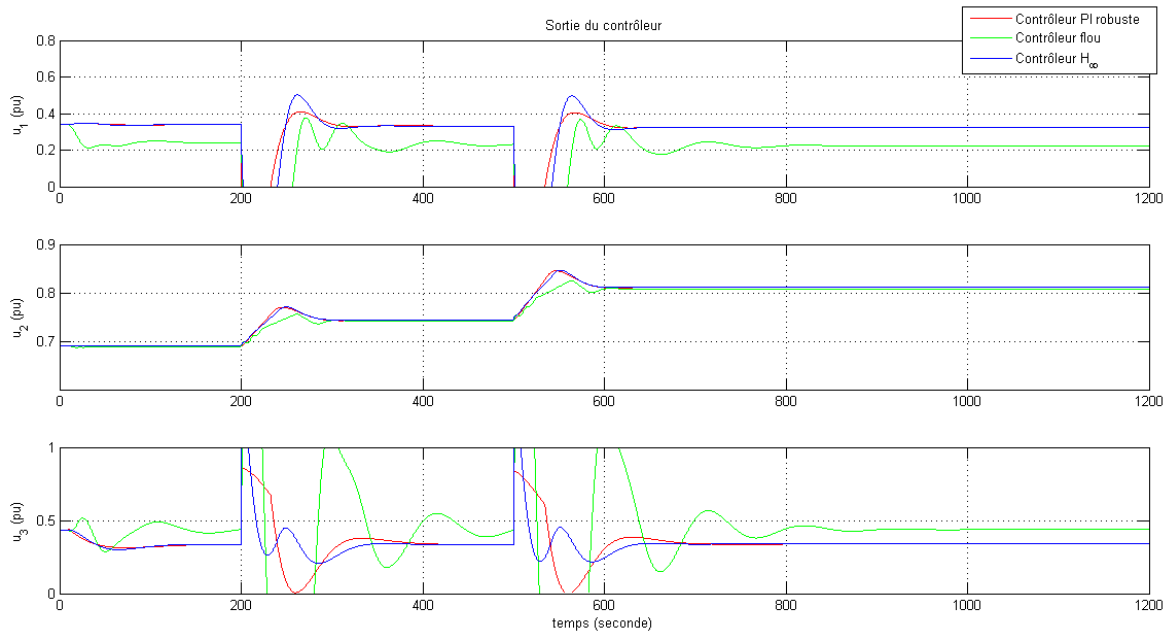
(b) Sortie du contrôleur

Figure IV.13 Réponses du processus non-perturbé.



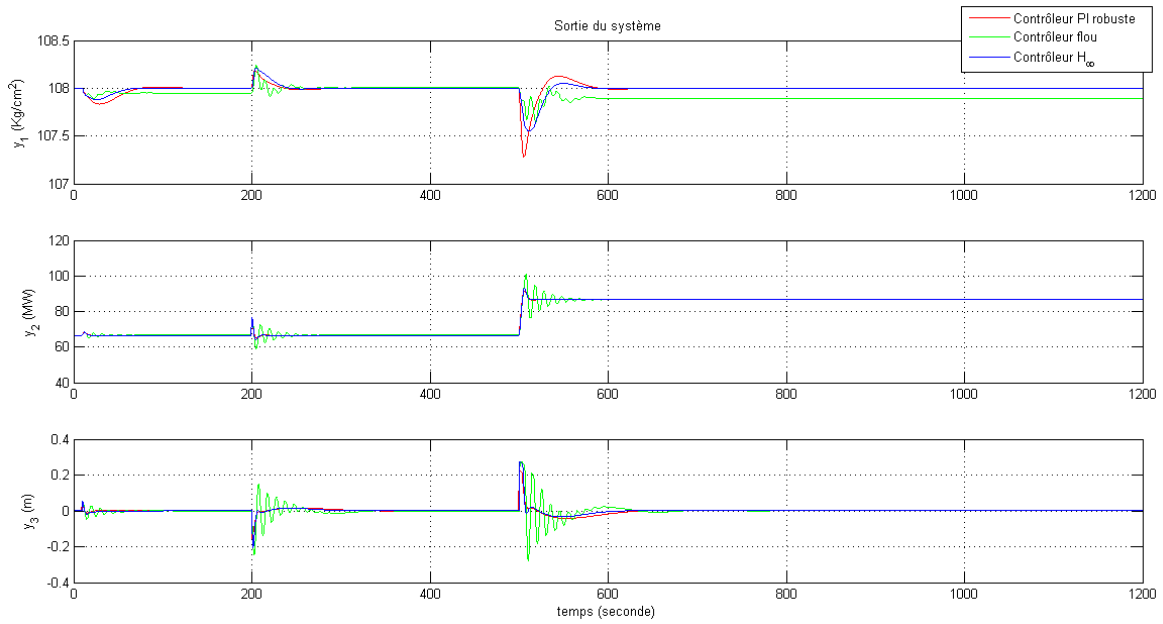


(a) Sortie du système

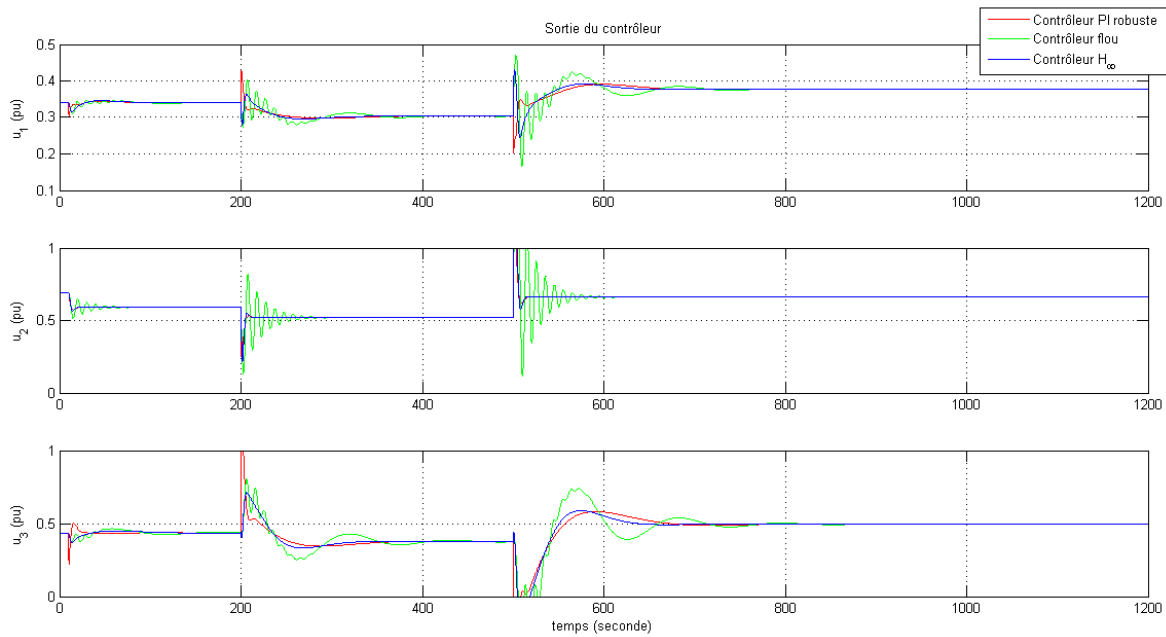


(b) Sortie du contrôleur

Figure IV.14 Réponses du processus en présence de perturbations, Cas 1.

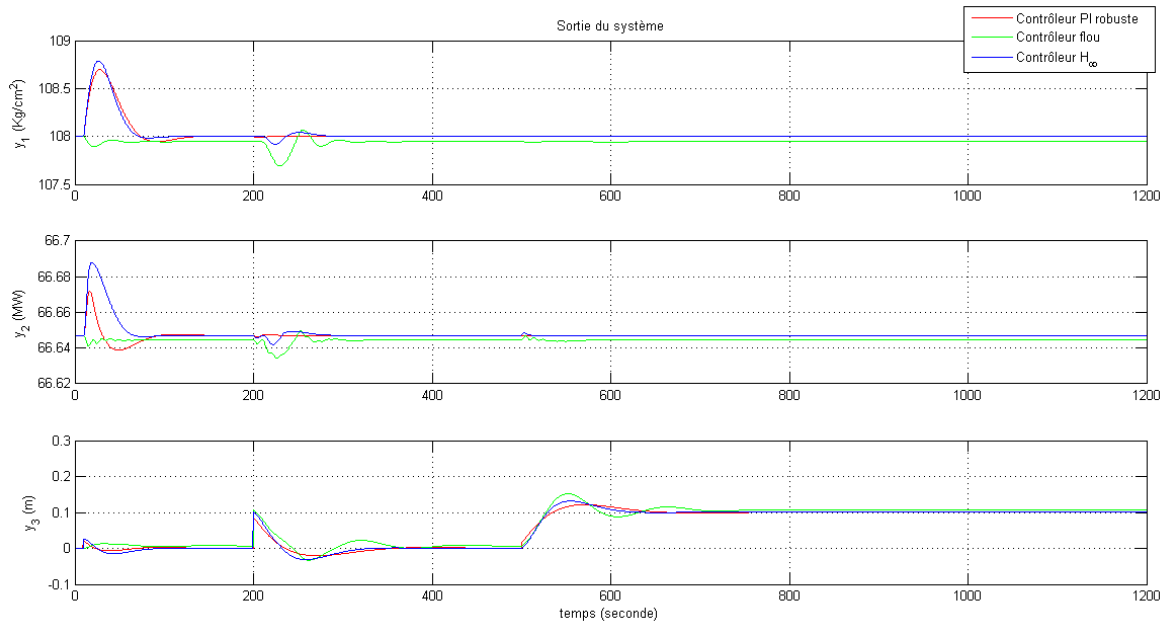


(a) Sortie du système

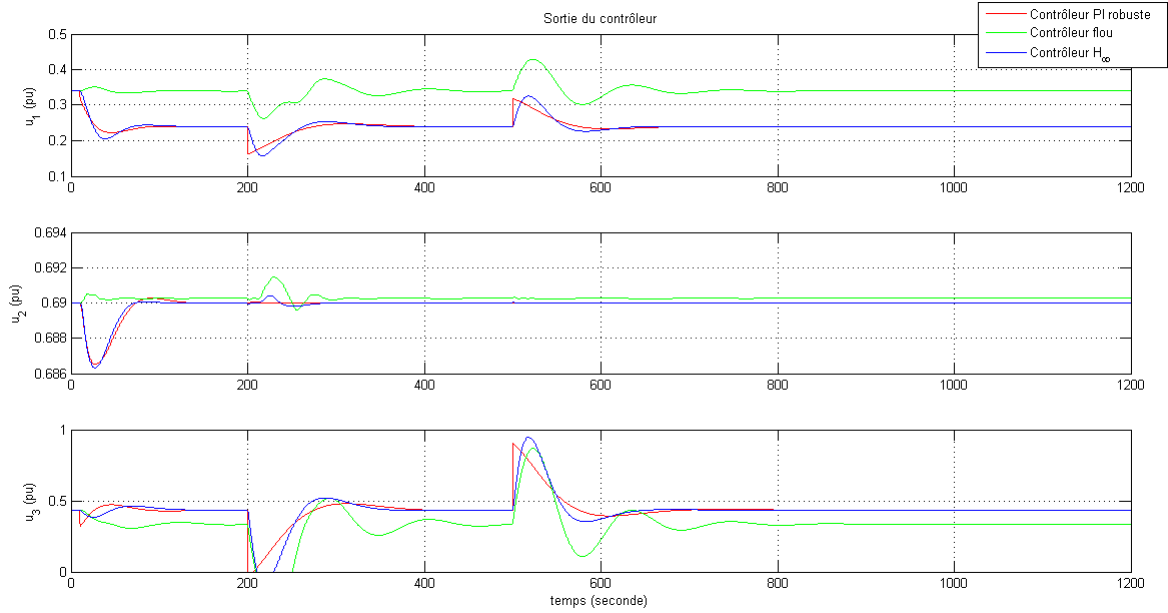


(b) Sortie du contrôleur

Figure IV.15 Réponses du processus en présence de perturbations, Cas 2.

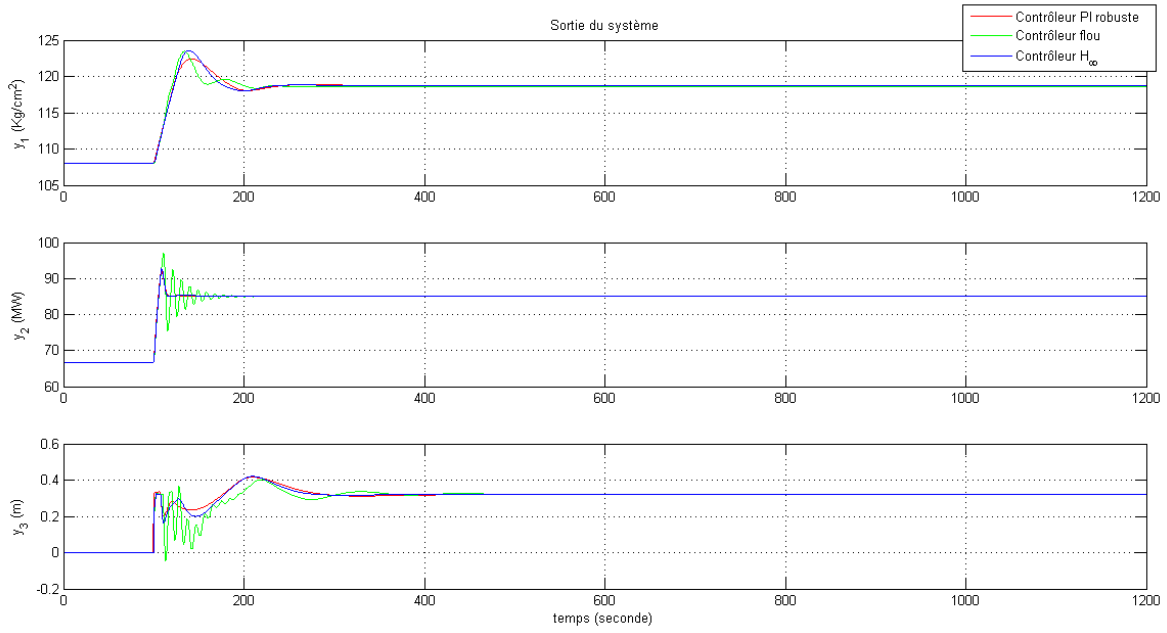


(a) Sortie du système

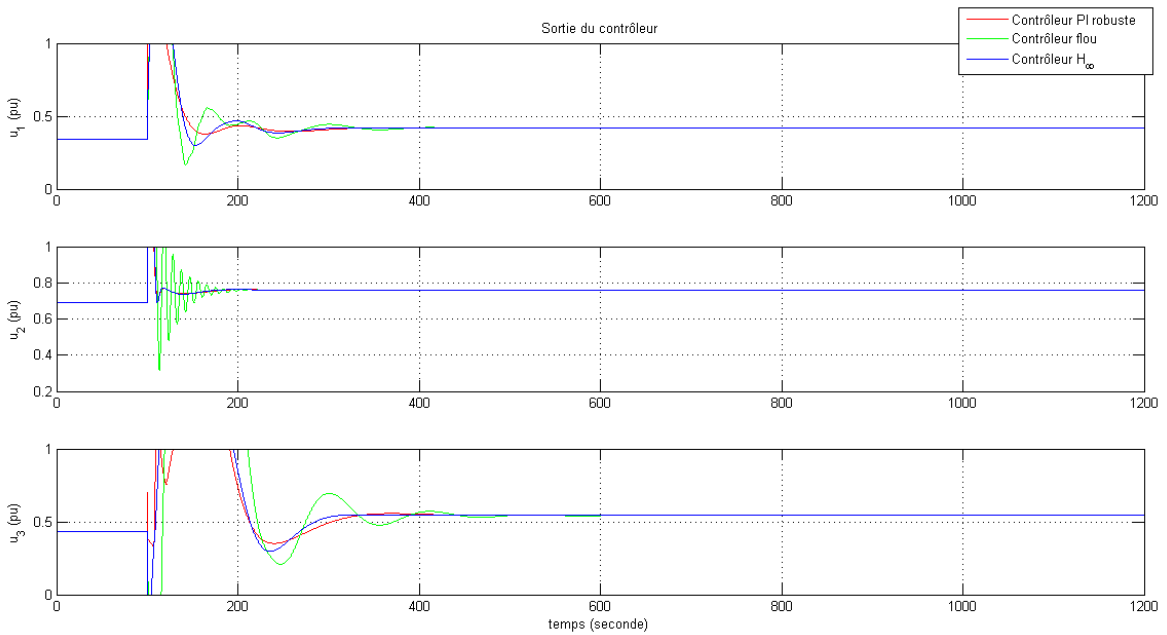


(b) Sortie du contrôleur

**Figure IV.16** Réponses du processus en présence de perturbations, Cas 3.

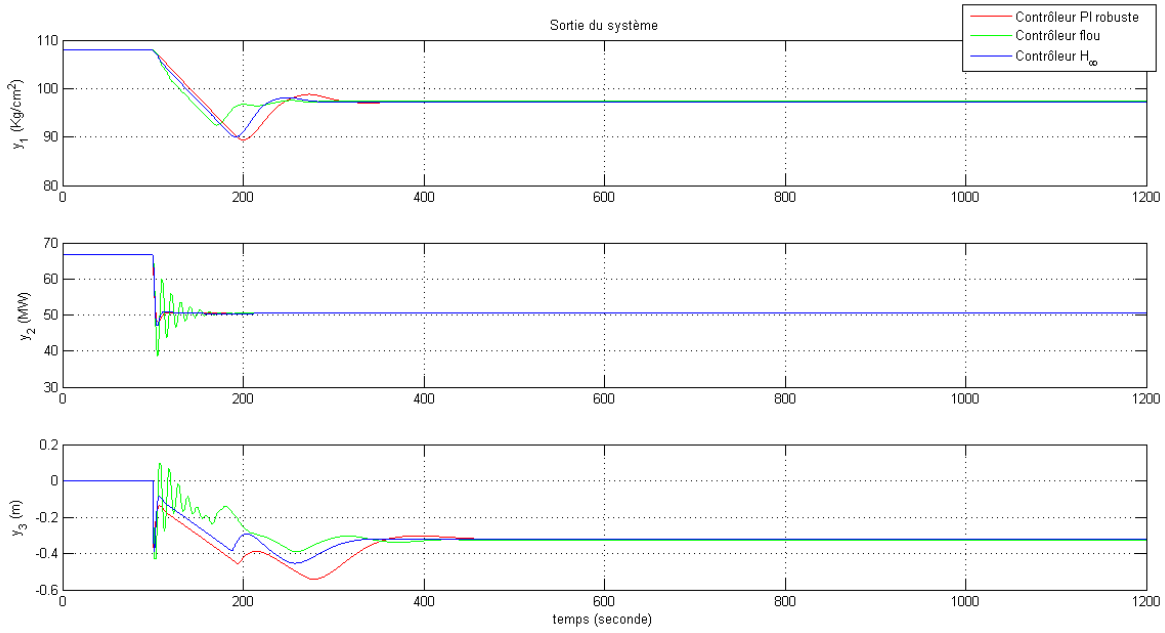


(a) Sortie du système

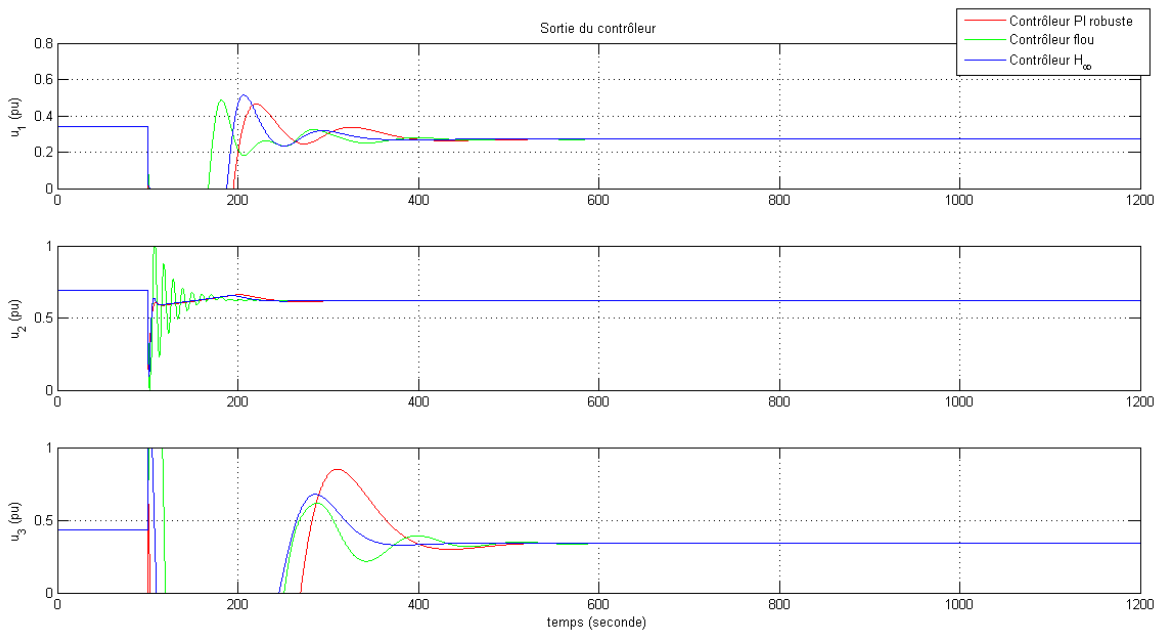


(b) Sortie du contrôleur

**Figure IV.17** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) + 10\%$ .

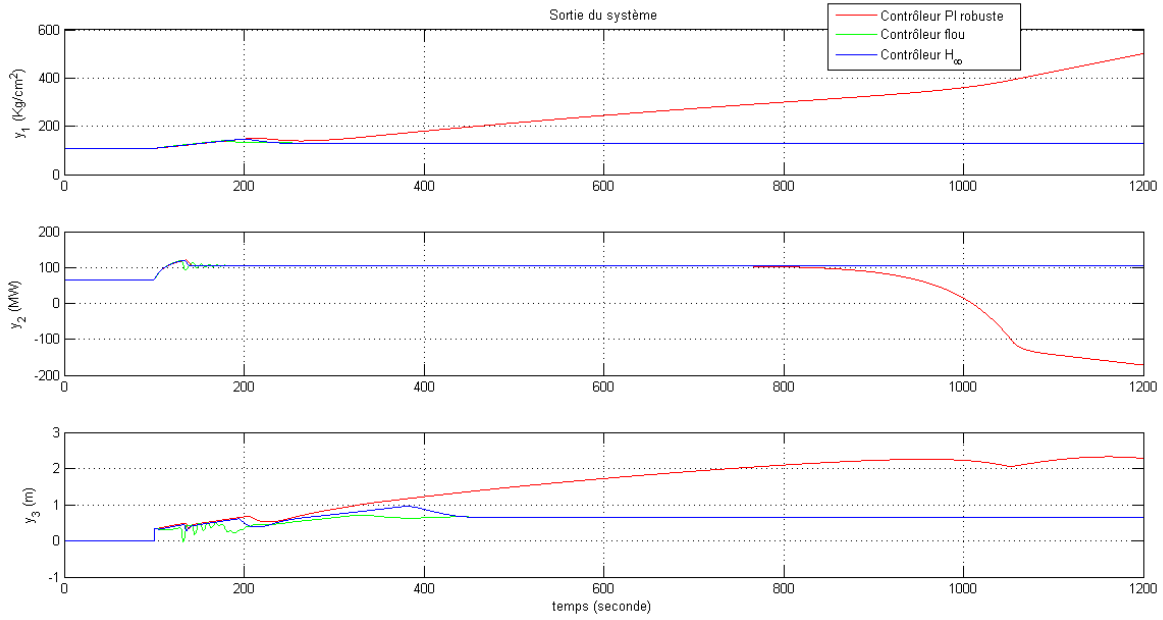


(a) Sortie du système

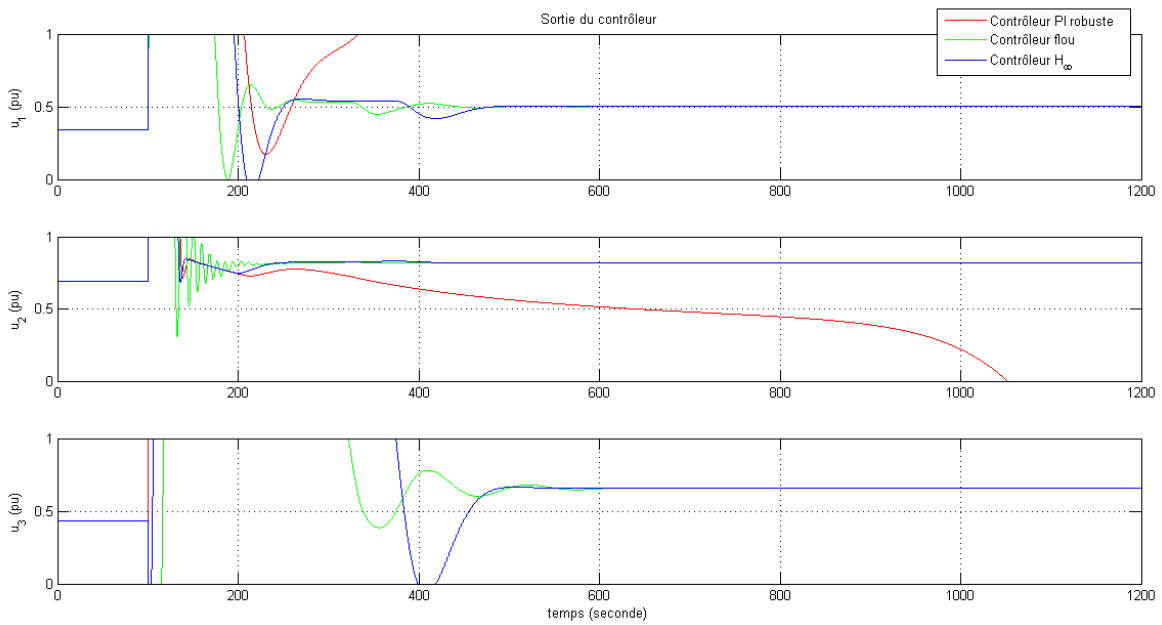


(b) Sortie du contrôleur

**Figure IV.18** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) - 10\%$ .

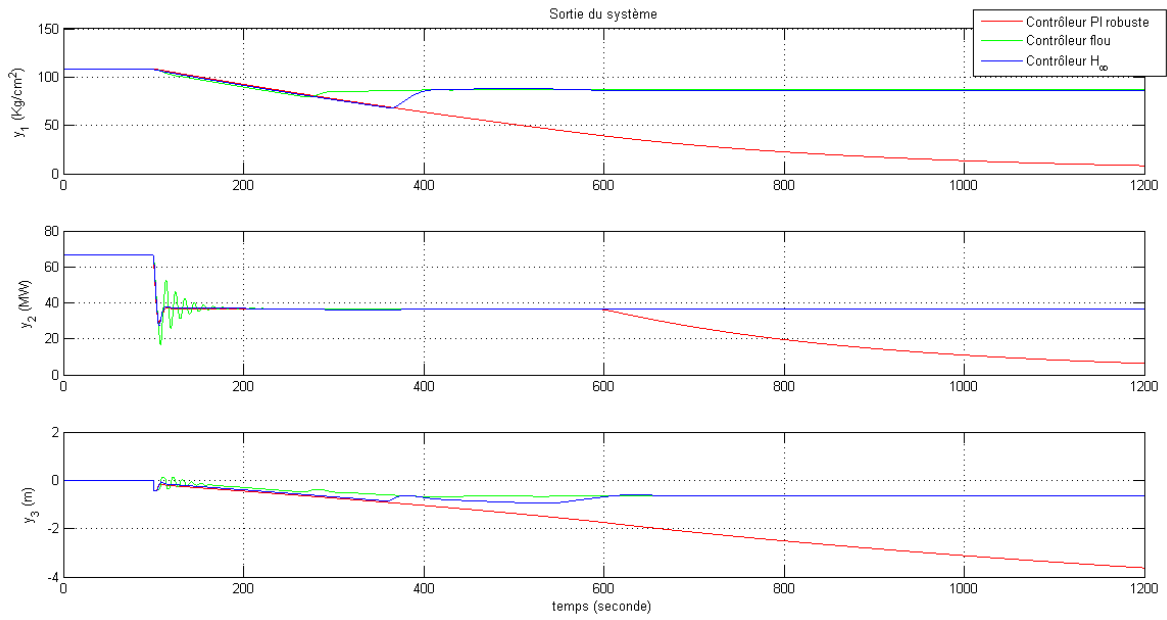


(a) Sortie du système

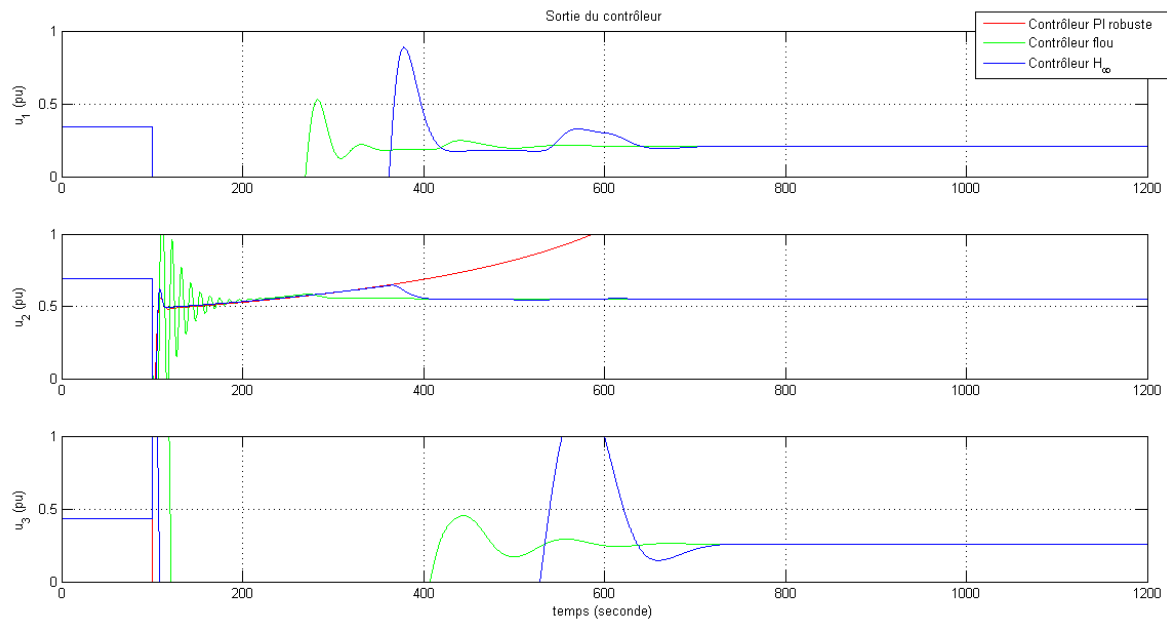


(b) Sortie du contrôleur

**Figure IV.19** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) + 20\%$ .

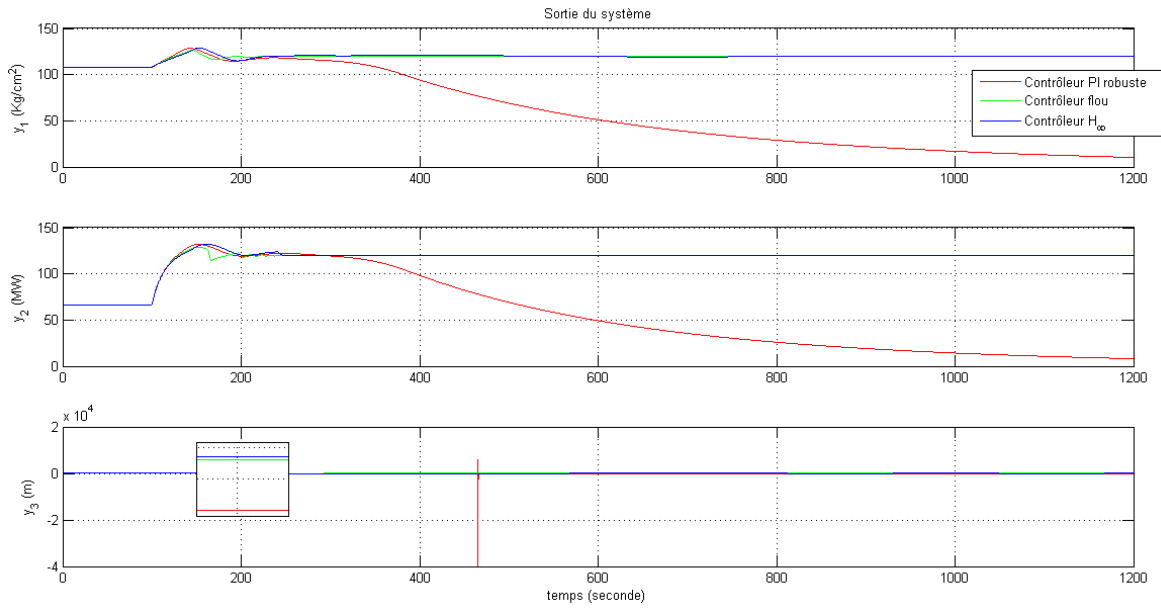


(a) Sortie du système

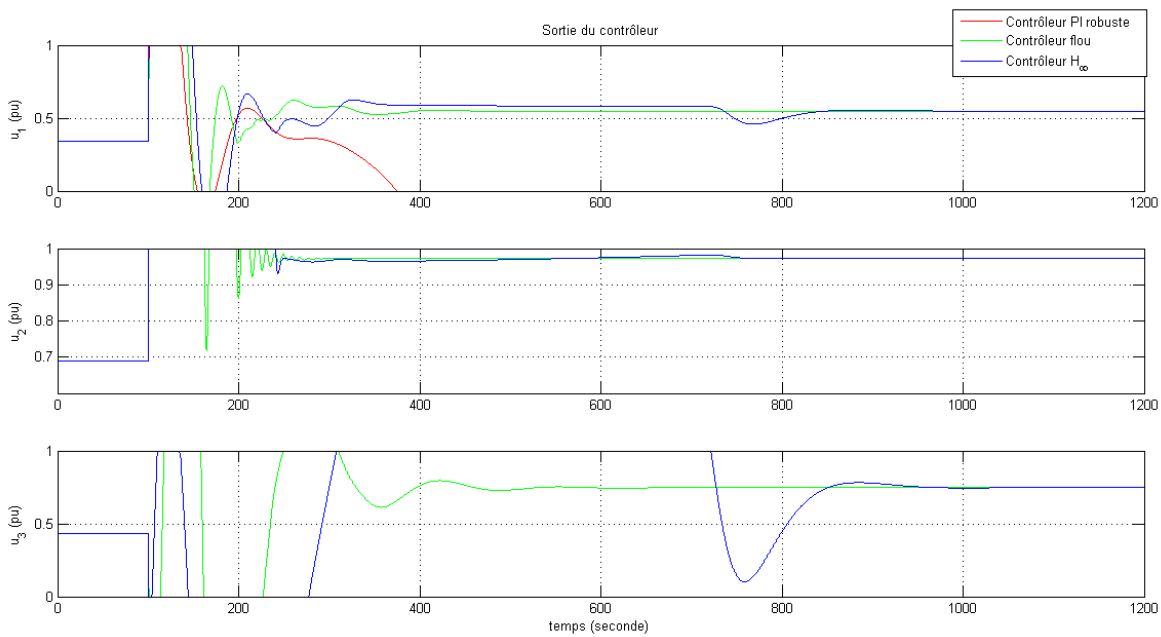


(b) Sortie du contrôleur

**Figure IV.20** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) - 20\%$ .



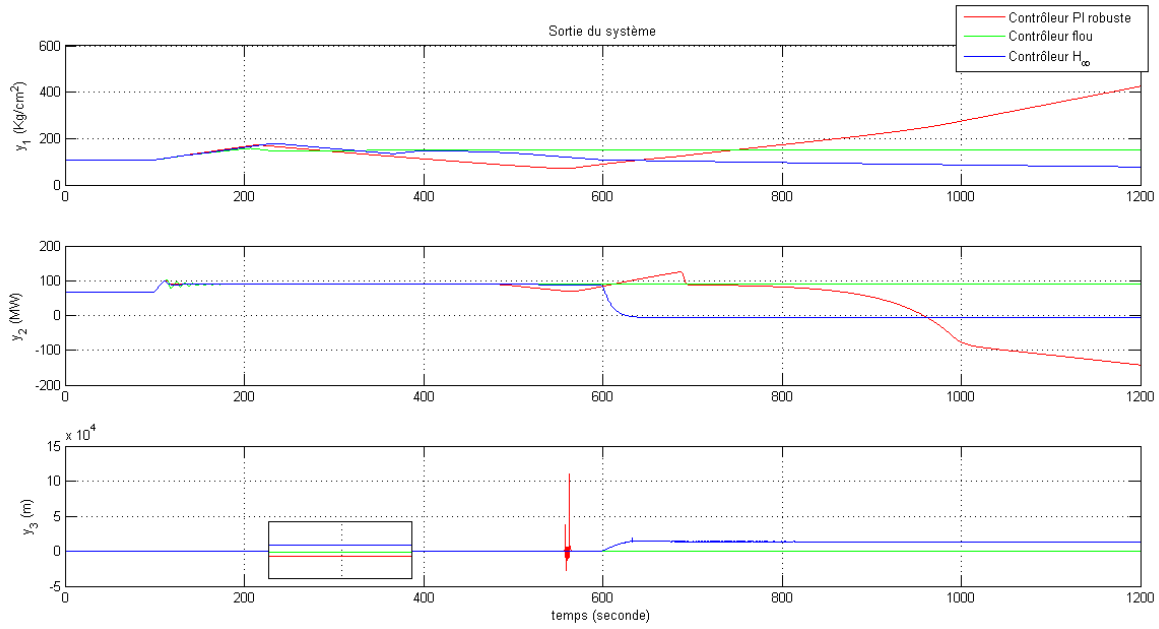
(a) Sortie du système



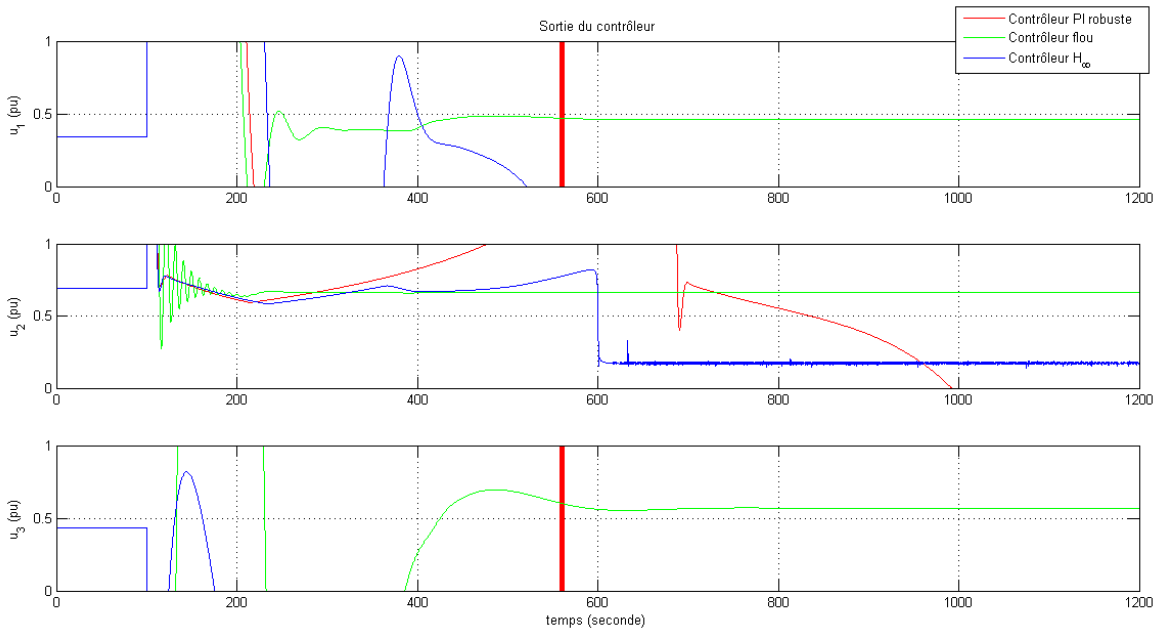
(b) Sortie du contrôleur

**Figure IV.21** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(120, 120, 0)$ .



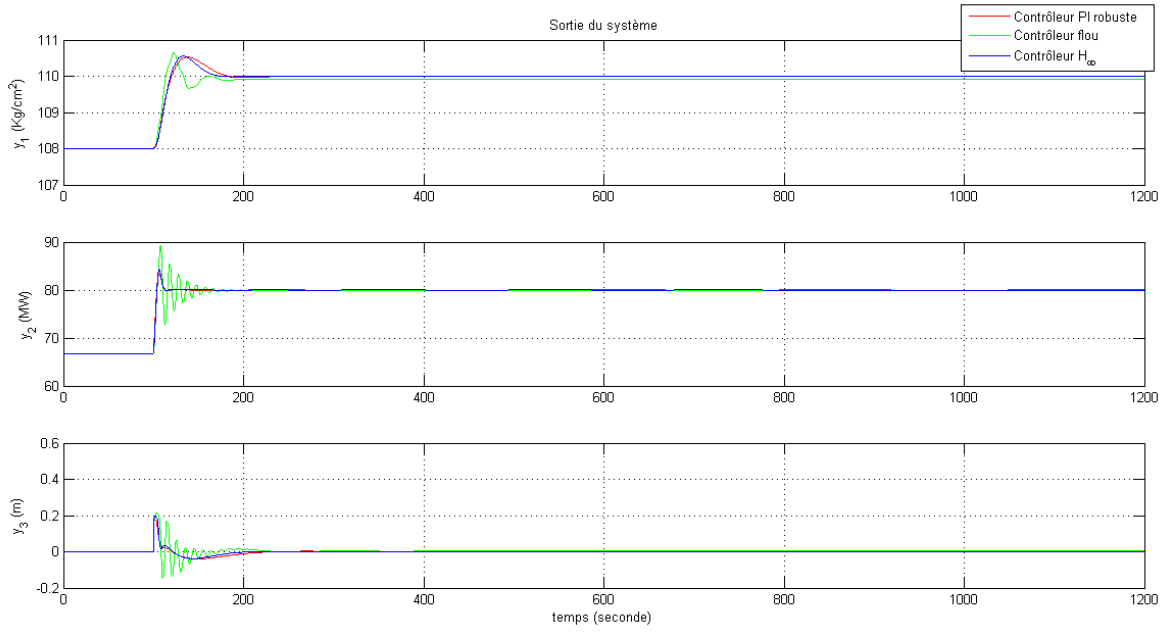


(a) Sortie du système

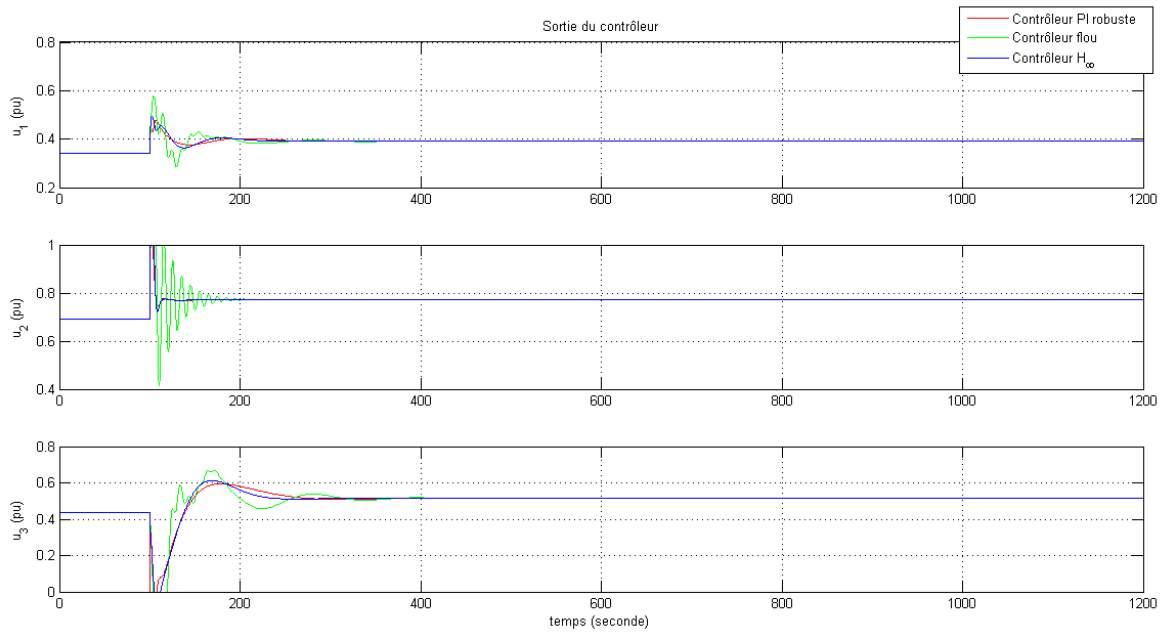


(b) Sortie du contrôleur

**Figure IV.22** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(150, 90, 0)$ .

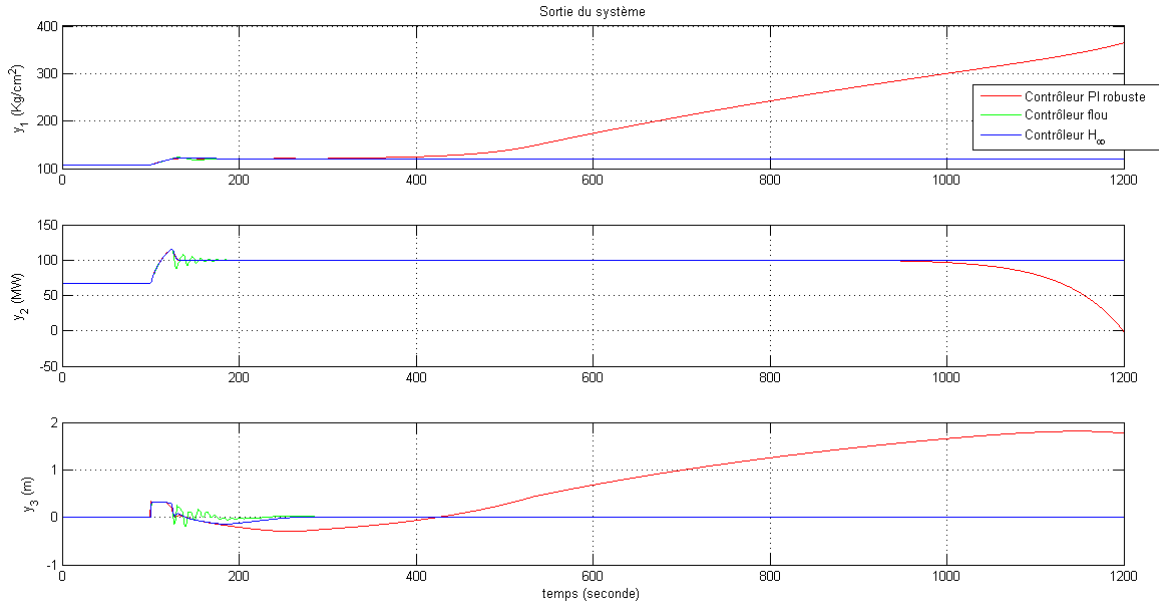


(a) Sortie du système

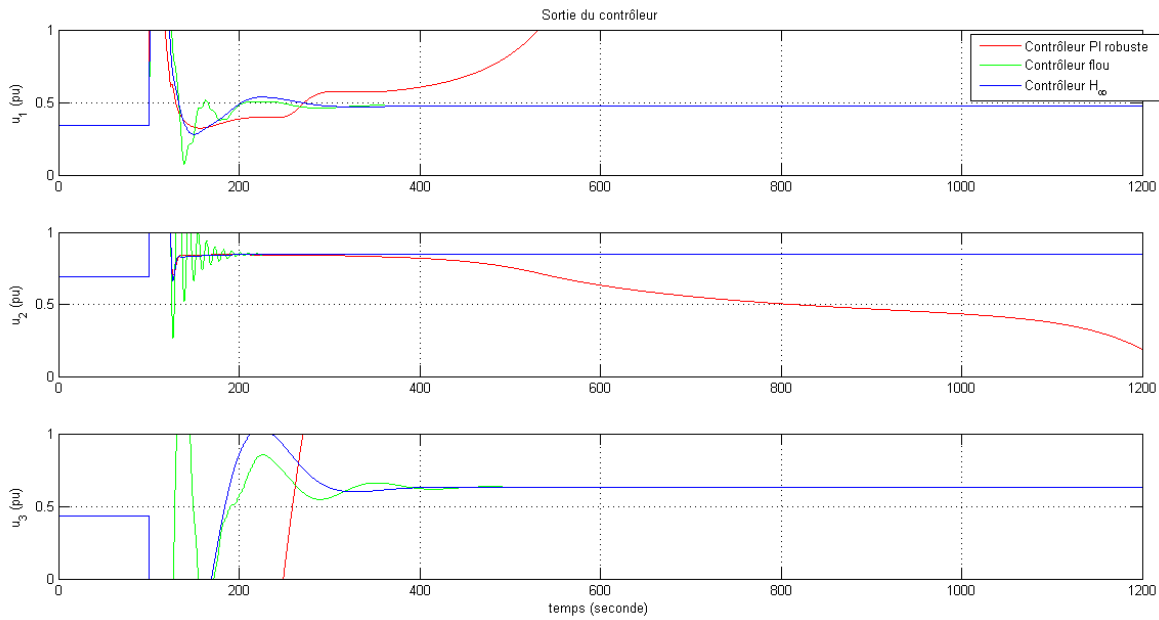


(b) Sortie du contrôleur

Figure IV.23 Réponses du processus à la demande de puissance électrique, 1<sup>er</sup> cas.

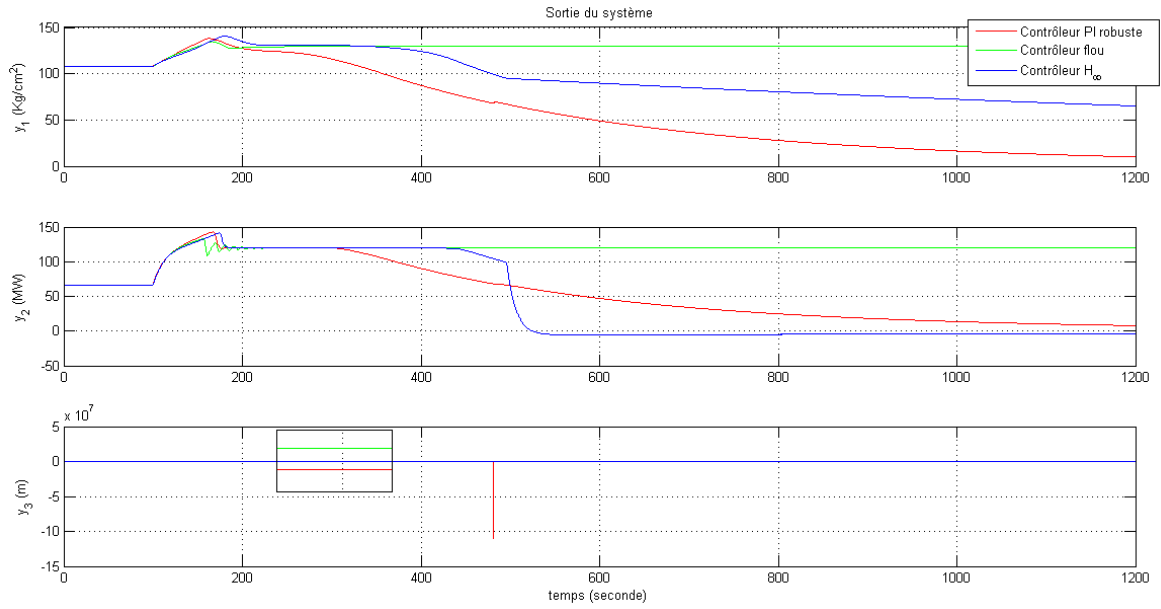


(a) Sortie du système

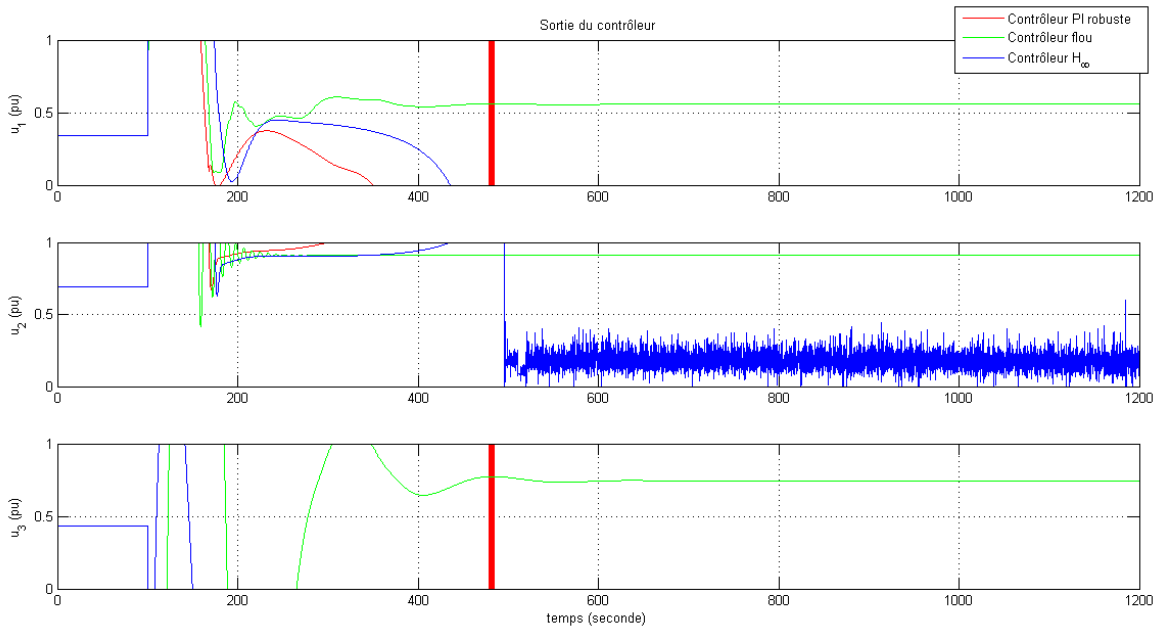


(b) Sortie du contrôleur

**Figure IV.24** Réponses du processus à la demande de puissance électrique, 2<sup>e</sup> cas.

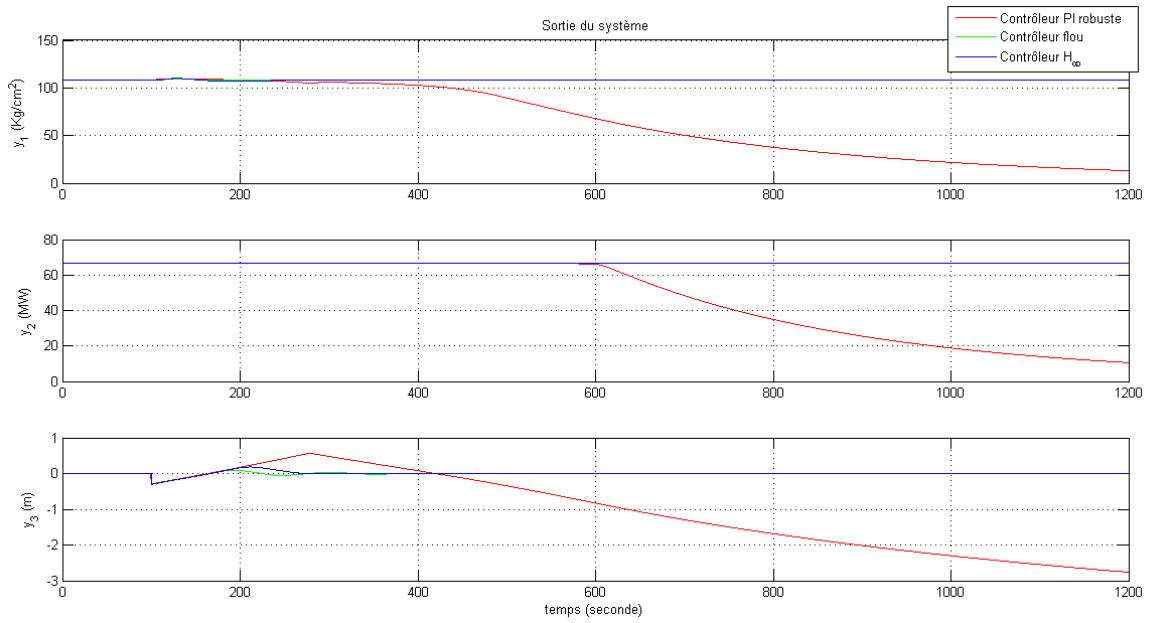


(a) Sortie du système

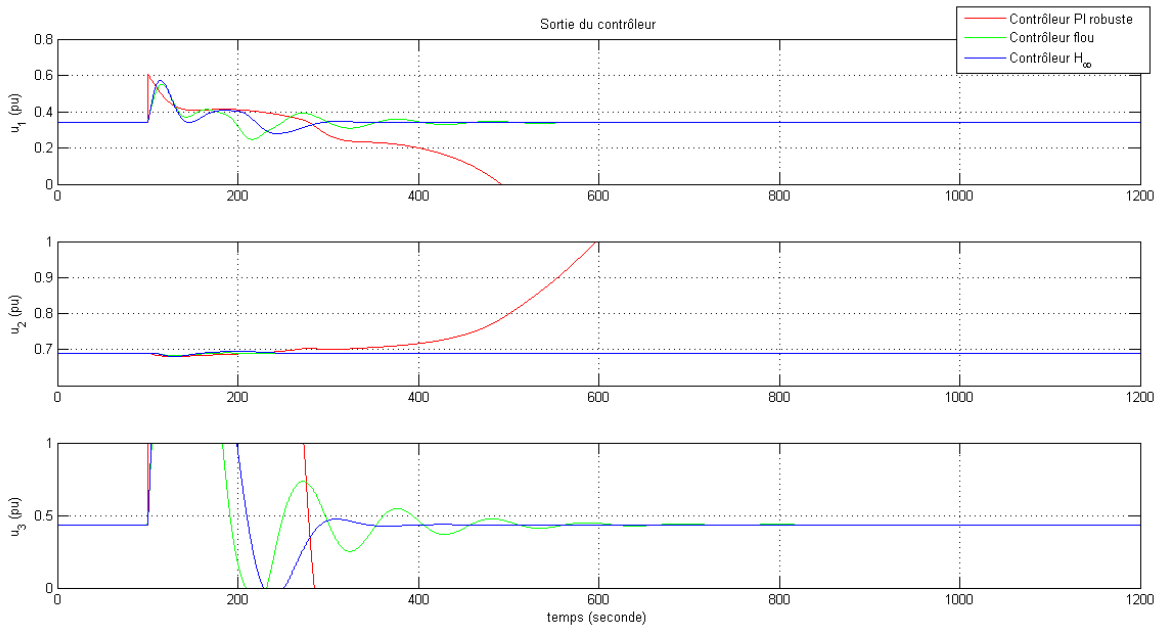


(b) Sortie du contrôleur

**Figure IV.25** Réponses du processus à la demande de puissance électrique, 3<sup>e</sup> cas.

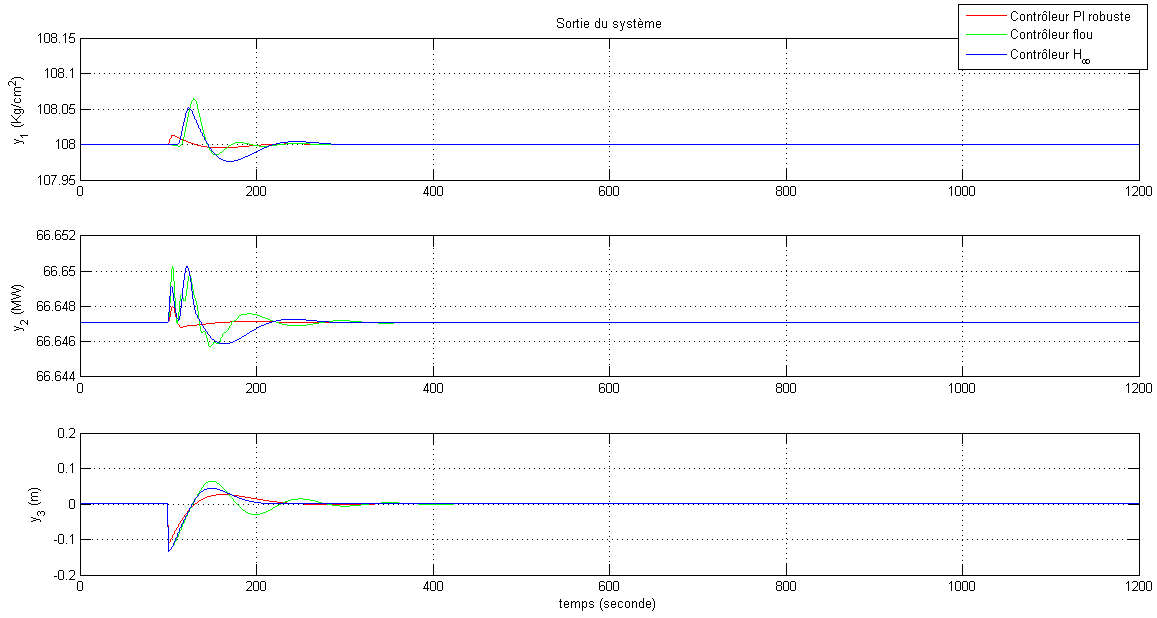


(a) Sortie du système

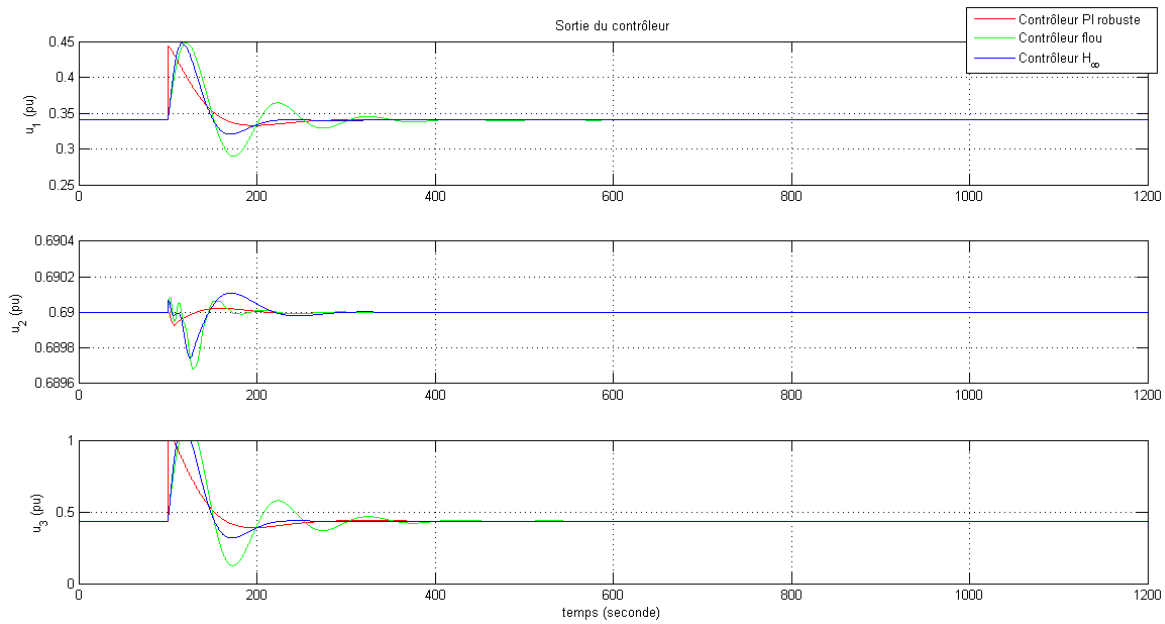


(b) Sortie du contrôleur

**Figure IV.26** Réponses du processus au test de variations paramétriques, 1<sup>er</sup> cas.



(a) Sortie du système



(b) Sortie du contrôleur

Figure IV.27 Réponses du processus au test de variations paramétriques, 2<sup>er</sup> cas.

Les mêmes points ont été remarqués dans les tests de demande supplémentaire en puissance électrique (Figures IV.23 à IV.25) et les tests de variations paramétriques (Figures IV.26 à IV.27). Le contrôleur flou est d'autant plus performant que le contrôleur  $\mathcal{H}_\infty$ . La conduite du système a été même optimisée dans certains cas tel que montré dans la Figure IV.25. De plus l'émulateur du contrôleur  $\mathcal{H}_\infty$  s'est montré suffisamment robuste vis-à-vis des variations paramétriques qui ont conduit le contrôleur PI multivariable à une désadaptation complète.

#### IV.4.4 Commande LQG

On illustre à présent une autre synthèse multivariable à appliquer au système GV-turbine. Il s'agit de la commande LQG décrite dans le Paragraphe I.4. Afin de répondre aux objectifs spécifiques de conception de la commande, nous avons choisi les matrices de covariance associées aux bruits comme suit :

$$W = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, V = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Pour réaliser le compromis entre la performance en poursuite et l'effort du contrôle, on utilise la fonction de coût suivante :

$$J = E \left\{ \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T [x^T Q x + u^T R u + x_i^T Q_i x_i] dt \right\} \quad (\text{IV.16})$$

avec

$$Q = \begin{bmatrix} 0.001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.001 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ et } Q_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Comme mentionné dans I.4.1 trois étapes sont nécessaires pour la conception du contrôleur LQG : (1) Synthèse du contrôleur linéaire quadratique, (2) Calcul du filtre de Kalman, (3) Détermination du contrôleur final.

##### IV.4.4.1 Synthèse du contrôleur linéaire quadratique

La conception du contrôleur linéaire quadratique est basée sur la minimisation du critère suivant :

$$J_{LQI} = \int_0^\infty \left[ [x^T x_i^T] Q \begin{bmatrix} x \\ x_i \end{bmatrix} + u^T R u \right] dt \quad (\text{IV.17})$$

La solution de ce problème peut être obtenue par la résolution de l'équation de Riccati I.43. Le gain du contrôleur optimal  $K_r$  est déterminé par :

$$K_r = \begin{bmatrix} 1.4394 & 0.030521 & 0.010350 & -0.98427 & -0.020361 & -0.17552 \\ 0.0045632 & 0.37185 & -0.00013448 & 0.021387 & -0.99976 & -0.0039569 \\ -0.50059 & -0.041522 & 0.066083 & 0.17540 & 0.0076485 & -0.98447 \end{bmatrix} \quad (\text{IV.18})$$

#### IV.4.4.2 Synthèse du filtre de Kalman

Après la résolution de l'équation de Riccati [I.46](#) et le calcul de la matrice  $Y$ , le gain du filtre de Kalman est déterminé par :

$$K_f = \begin{bmatrix} 0.71291 & 0.69865 & 0.0078655 \\ 0.69865 & 0.68666 & 0.0077114 \\ 0.71042 & 0.69690 & 0.011266 \end{bmatrix} \quad (\text{IV.19})$$

#### IV.4.4.3 Détermination du contrôleur final

Le contrôleur LQG résultant pour le système GV-turbine est décrit par le modèle d'état :

$$A_{K_{LQG}} = \begin{bmatrix} -2.081 & -0.601 & 0.0005265 & 0.9177 & -0.3336 & 0.008659 \\ -0.691 & -6.049 & 0.00188 & -0.3046 & 14.15 & 0.05576 \\ 0.1238 & -0.1061 & -0.1098 & -0.2638 & -1.416 & 1.627 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (\text{IV.20})$$

$$B_{K_{LQG}} = \begin{bmatrix} -0.7129 & -0.6987 & -0.007866 \\ -0.6987 & -0.6867 & -0.007711 \\ -0.7104 & -0.6969 & -0.01127 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (\text{IV.21})$$

$$C_{K_{LQG}} = \begin{bmatrix} -1.439 & -0.03052 & -0.01035 & 0.9843 & 0.02036 & 0.1755 \\ -0.004563 & -0.3719 & 0.0001345 & -0.02139 & 0.9998 & 0.003957 \\ 0.5006 & 0.04152 & -0.06608 & -0.1754 & -0.007648 & 0.9845 \end{bmatrix}, \quad (\text{IV.22})$$

$$D_{K_{LQG}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (\text{IV.23})$$

Les valeurs initiales impliquées en simulation pour le contrôleur LQG sont obtenues selon le même mécanisme décrit par l'équation ([IV.11](#)). Elles sont données par :

$$x_c^0 = \begin{bmatrix} -55.306 \\ 56.793 \\ -3017.0 \\ -18.547 \\ 4.4651 \\ -9.4282 \end{bmatrix} \quad (\text{IV.24})$$



Le contrôleur LQG obtenu est d'ordre 6. Afin de rendre possible son implémentation pratique, on propose d'émuler son comportement par l'utilisation de la stratégie d'émulation floue proposée dans le cadre ce mémoire.

## IV.4.5 Émulation floue du contrôleur LQG

### IV.4.5.1 Synthèse de l'émulateur flou

La stratégie d'émulation floue s'applique selon la même procédure décrite pour l'émulation floue du contrôleur  $\mathcal{H}_\infty$ . On vise à concevoir un modèle de contrôle flou exprimé linguistiquement par :

$$\begin{aligned}
 & \text{If } e_1(k) \text{ is } E_1 \text{ and } e_2(k) \text{ is } E_2 \text{ and } e_3(k) \text{ is } E_3 \text{ and} \\
 & \quad u_1(k-1) \text{ is } U_1 \text{ and } u_2(k-1) \text{ is } U_2 \text{ and } u_3(k-1) \text{ est } U_3 \\
 & \text{then } u_1(k) = f_1(e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1)) \text{ and} \\
 & \quad u_2(k) = f_2(e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1)) \text{ and} \\
 & \quad u_3(k) = f_3(e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1))
 \end{aligned}$$

où  $e_i(k)$  et  $u_i(k-1)$ ,  $i = 1, 2, 3$  sont la variable d'erreur et la variable de commande retardée, respectivement.  $E_i$  et  $U_i$  sont les ensembles flous associés.  $f_i$  est une fonction arbitraire associée à chaque variable de sortie  $u_i(k)$  et qui dépend vecteur d'entrée  $x(k) = (e_1(k), e_2(k), e_3(k), u_1(k-1), u_2(k-1), u_3(k-1))$ .

Des séquences binaires pseudo-aléatoires (SBPA) sont sélectionnées pour l'excitation du contrôleur LQG. Elles sont appliquées à ses entrées par des combinaisons bien choisies avec les paramètres suivants : nombre de registres à décalage  $n_{sr} = 05$ , le temps de commutation  $T_{sw} = 1.8s$ , et l'amplitude du signal d'excitation  $\tilde{e} = \pm 10$  (voir Annexe I). Ainsi, un ensemble de données d'entrée/sortie composé de 12006 échantillons a été générée pour chaque variable de sortie.

Les paramètres de l'algorithme de coalescence floue appliqué aux données d'apprentissage sont :  $r_a = 0.6$ ,  $r_b/r_a = 1.25$ ,  $\bar{\epsilon} = 0.5$  et  $\underline{\epsilon} = 0.15$ . Avec ces paramètres, les expressions analytiques des lois de commande  $u_1(k)$ ,  $u_2(k)$  et  $u_3(k)$  que génère l'émulateur flou sont données par :

$$\begin{aligned}
 u_1(k) &= 1.697 e_1(k) + 2.375 e_2(k) + 1.393 e_3(k) + 0.9924 u_1(k-1) \\
 &+ 0.2046 u_2(k-1) - 0.009965 u_3(k-1) - 0.1343 \quad (\text{IV.25})
 \end{aligned}$$

$$\begin{aligned}
 u_2(k) &= -0.0054 e_1(k) + 0.3529 e_2(k) + 0.01295 e_3(k) + 0.001757 u_1(k-1) \\
 &+ 0.9745 u_2(k-1) - 0.0003731 u_3(k-1) + 0.01717 \quad (\text{IV.26})
 \end{aligned}$$

$$\begin{aligned}
 u_3(k) &= 2.559 e_1(k) + 0.4901 e_2(k) + 6.511 e_3(k) + 0.03792 u_1(k-1) \\
 &- 0.02754 u_2(k-1) + 0.9327 u_3(k-1) + 0.03542 \quad (\text{IV.27})
 \end{aligned}$$

Les ensembles flous des prémisses des règles résultants sont représentés dans la Figure IV.28. Leurs paramètres sont résumés dans le Tableau IV.4.

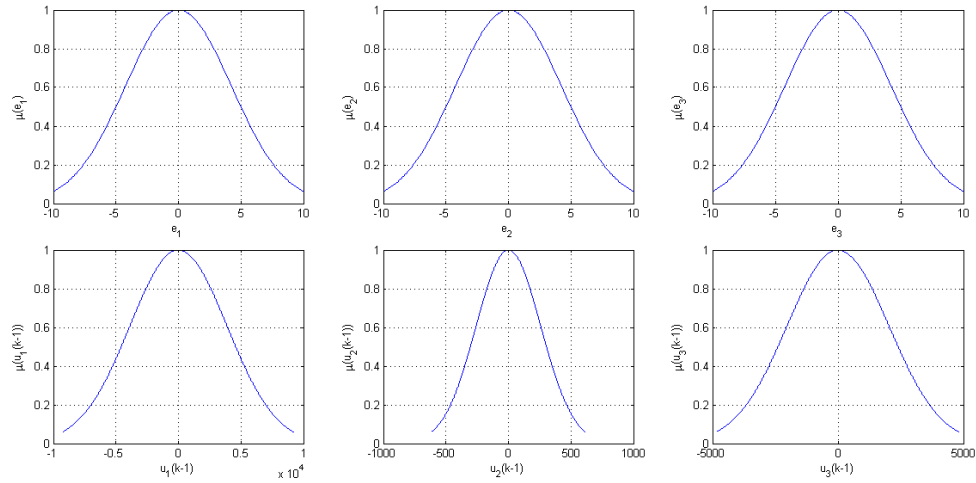


Figure IV.28 Les ensembles flous gaussiens associés à chacune des variables d'entrée.

$MF$	$E_1$	$E_2$	$E_3$	$U_1$	$U_2$	$U_3$
[Sigma c]	[4.243 0]	[4.243 0]	[4.243 0]	[3893 0.3402]	[258.7 0.69]	[2036 0.4358]

Tableau IV.4 Paramètres des prémisses de l'émulateur flou du contrôleur LQG.

#### IV.4.5.2 Validation de l'émulateur flou

L'évaluation de l'indice VAF sur les données d'apprentissage conduit aux valeurs suivantes : 99.570% pour  $u_1$ , 99.351% pour  $u_2$  et 99.599% pour  $u_3$ . La Figure IV.29 illustre le comportement des deux contrôleurs LQG et son émulateur flou obtenu dans la phase d'apprentissage. La bonne adéquation entre les deux réponses démontre la capacité d'approximation de l'émulateur flou.

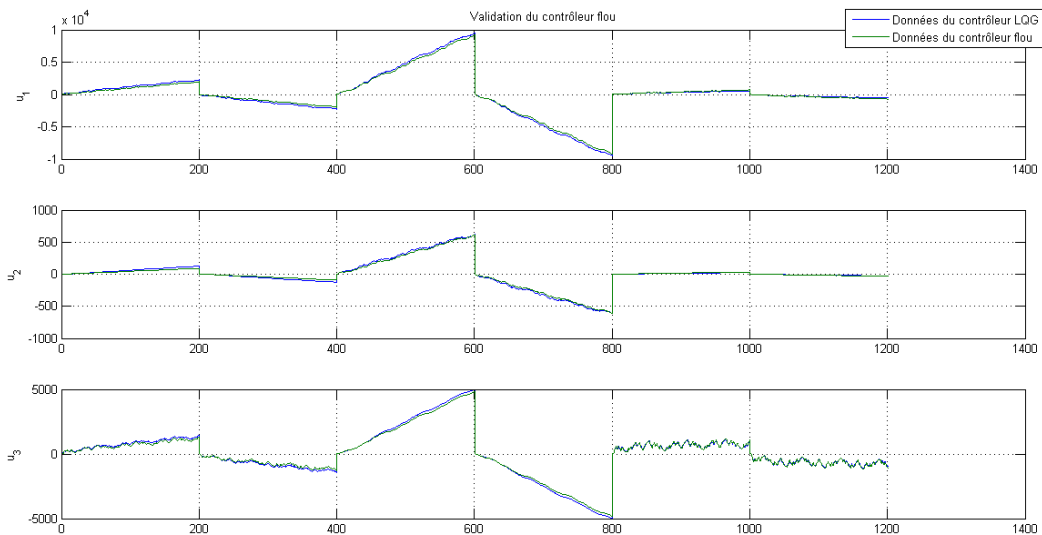
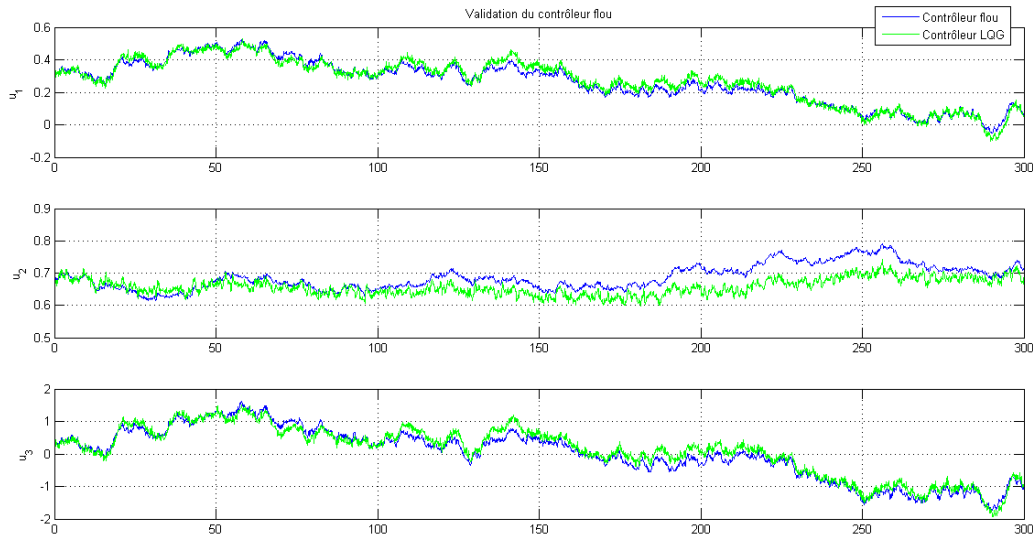


Figure IV.29 Performance du contrôleur flou par apport au contrôleur LQG original (données d'apprentissage).

Un autre test de validation est effectué, par l'emploi de signaux d'excitation différents de ceux employés en apprentissage. La Figure IV.30 démontre la capacité de généralisation du contrôleur flou.



**Figure IV.30** Performance du contrôleur flou par apport au contrôleur LQG original (donnés de validation).

### IV.4.5.3 Résultats de simulation

Dans le but de tester la performance de la stratégie d'émulation floue du contrôleur LQG, plusieurs simulations sont effectuées sur le modèle GV-turbine non-linéaire. Les cas simulés correspondent à des situations de rejet de perturbations, de poursuite de références, de changement du point de fonctionnement, et de variations paramétriques. Les réponses du système GV-turbine en absence de perturbations sont tout d'abord simulées pour les deux stratégies de commande LQG et floue. La Figure IV.31 illustre ces résultats.

#### a) Rejet de perturbation

Cas 1 : Á  $t = 10s$ ,  $\Delta u_1 = 10\%$ , puis à  $t = 200s$ ,  $\Delta y_1 = 10Kg/cm^2$ , ensuite à  $t = 500s$ ,  $\Delta r_1 = -10Kg/cm^2$ . Les réponses sont représentées dans la Figure IV.32.

Cas 2 : Á  $t = 10s$ ,  $\Delta u_2 = 10\%$ , puis à  $t = 200s$ ,  $\Delta y_2 = 10MW$ , ensuite à  $t = 500s$ ,  $\Delta r_2 = -10MW$ . Les réponses sont représentées dans la Figure IV.33.

Cas 3 : Á  $t = 10s$ ,  $\Delta u_3 = 10\%$ , puis à  $t = 200s$ ,  $\Delta y_3 = 0.1m$ , ensuite à  $t = 500s$ ,  $\Delta r_3 = 0.1m$ . Les réponses sont représentées dans la Figure IV.34.

#### b) Changement du point de fonctionnement

Cas 1 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) + 10\%$ . Les réponses sont représentées dans la Figure IV.35.

Cas 2 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) - 10\%$ . Les réponses sont représentées dans la Figure IV.36.

Cas 3 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) + 20\%$ . Les réponses sont représentées dans la Figure IV.37.

Cas 4 : Dans ce cas, on introduit une variation du point de fonctionnement nominal  $\#(y_1^0, y_2^0, y_3^0)$  vers un point de fonctionnement proche  $\#(y^0) - 20\%$ . Les réponses sont représentées dans la Figure IV.38.

### c) Changement de consignes

Cas 1 : Étant en régime nominal, on introduit la consigne :  $\#(120, 120, 0)$ . Les réponses sont indiquées dans la Figure IV.39.

Cas 2 : Étant en régime nominal, on introduit la consigne :  $\#(150, 90, 0)$ . Les réponses sont indiquées dans la Figure IV.40.

### d) Test de demande supplémentaire en puissance électrique

Cas 1 : Étant en régime nominal, on introduit la consigne :  $\#(110, 80, 0)$ . Les réponses sont indiquées dans la Figure IV.41.

Cas 2 : Étant en régime nominal, on introduit la consigne :  $\#(120, 100, 0)$ . Les réponses sont indiquées dans la Figure IV.42.

Cas 3 : Étant en régime nominal, on introduit la consigne :  $\#(130, 120, 0)$ . Les réponses sont indiquées dans la Figure IV.43.

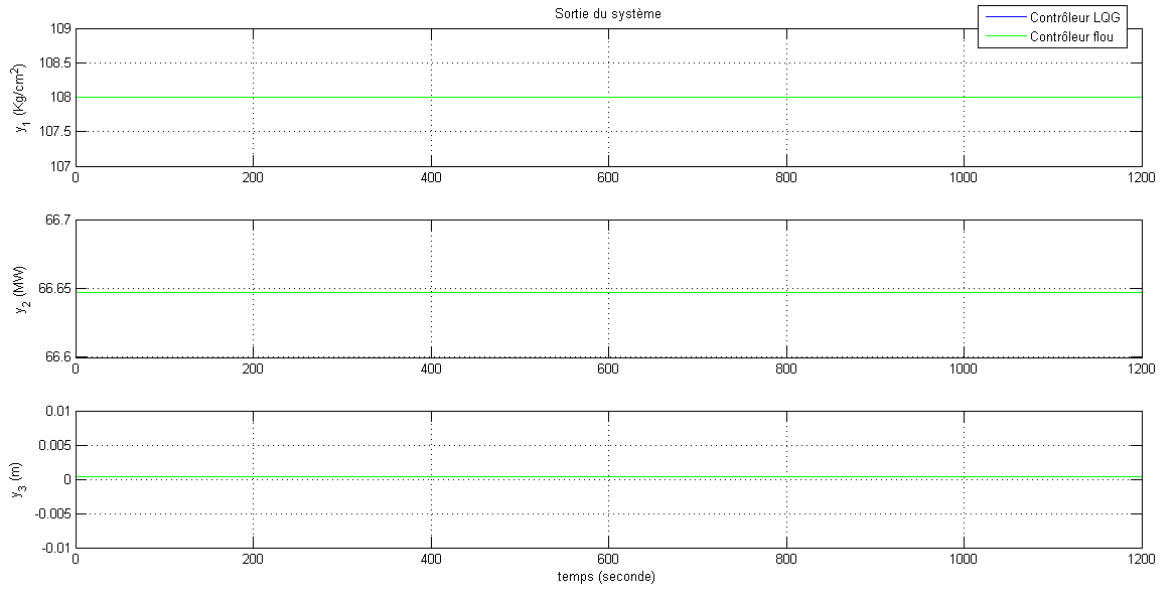
### e) Variations paramétriques

Cas 1 : Dans ce test, on modifie la valeur nominale du débit massique d'évaporation  $q_e = 60.084 Kg/s$ , vers  $q_e/10$ . Les réponses sont données en figure IV.44.

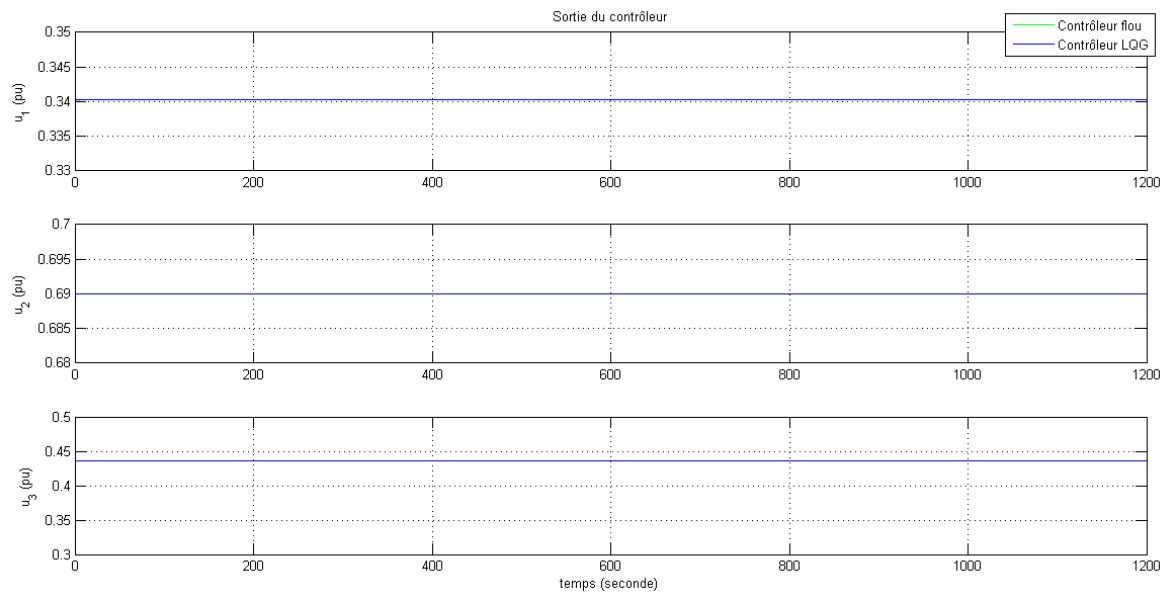
Cas 2 : Dans ce test, on modifie la valeur nominale du facteur de qualité de la vapeur d'eau  $a_{cs} = 0.0535 Kg/s$ , vers  $a_{cs}/2$ . Les réponses sont données en figure IV.45.

#### IV.4.5.4 Analyse des résultats

Afin de démontrer les performance du système de commande floue émulant le contrôleur LQG, on reconduit les mêmes tests établis pour l'émulateur du contrôleur  $\mathcal{H}_\infty$ . Les premiers tests effectués sont ceux de rejet de perturbations, les sorties du système résultantes montrent une bonne stabilisation autour des conditions nominales d'exploitation et ce malgré les amplitudes relativement grandes des perturbations appliquées, tel que montré en Figures IV.32 à IV.34. Quelques fluctuations sont cependant observées sous le contrôle flou, notamment au niveau de la puissance électrique et le niveau d'eau. Elles sont néanmoins tolérables vu qu'elles arrivent à se stabiliser quelques secondes seulement après leur occurrence.

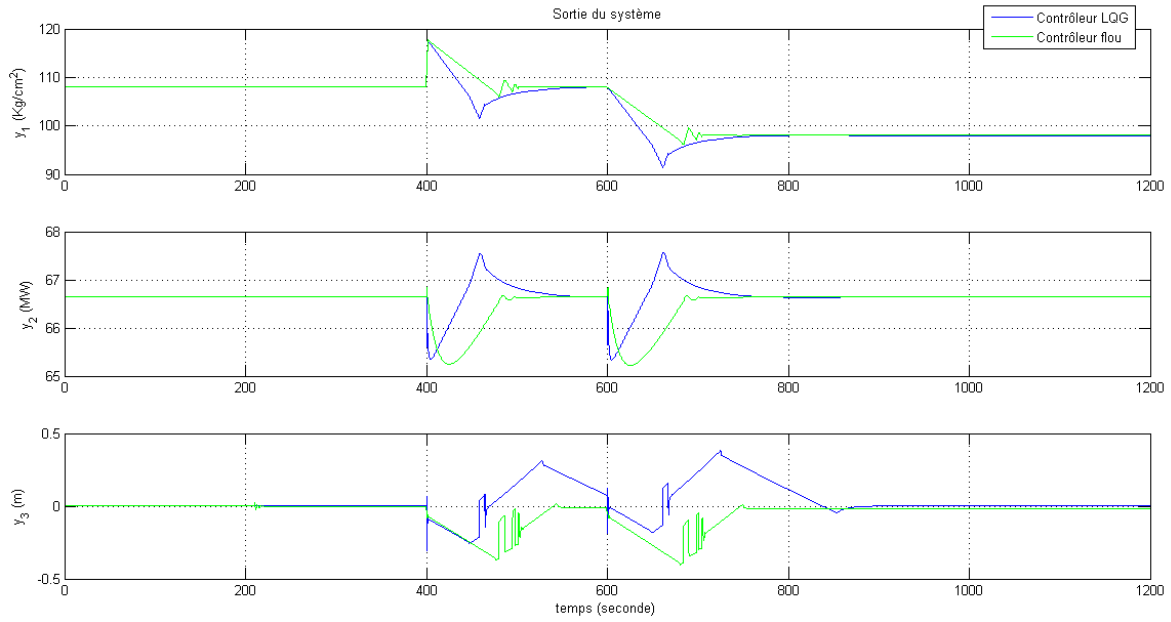


(a) Sortie du système

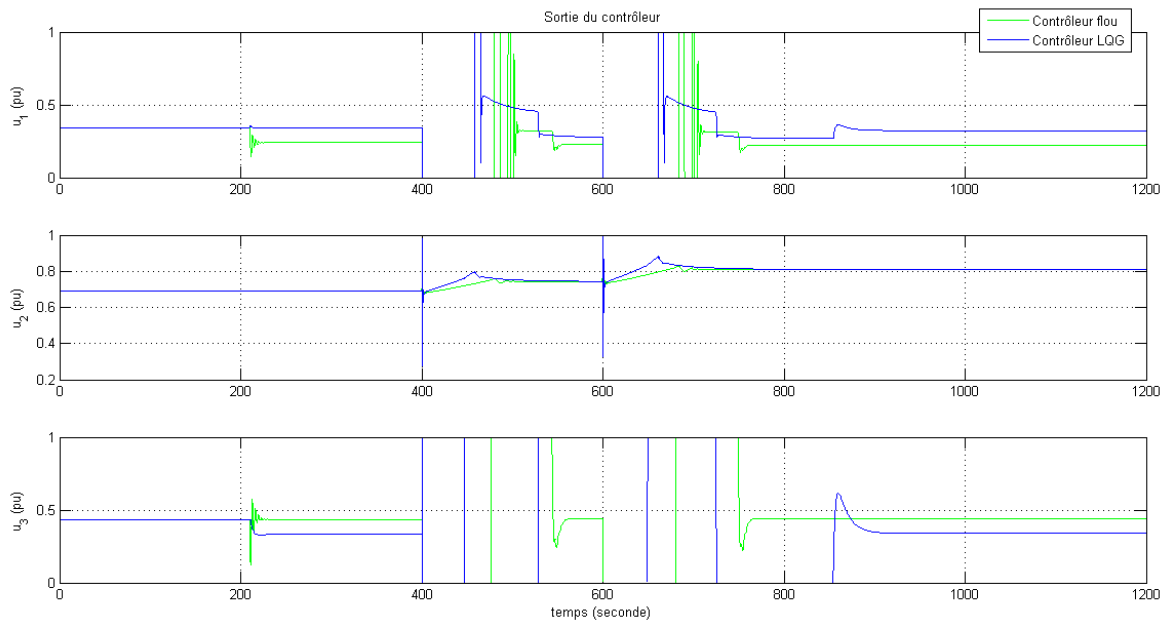


(b) Sortie du contrôleur

Figure IV.31 Réponses du processus non perturbé.

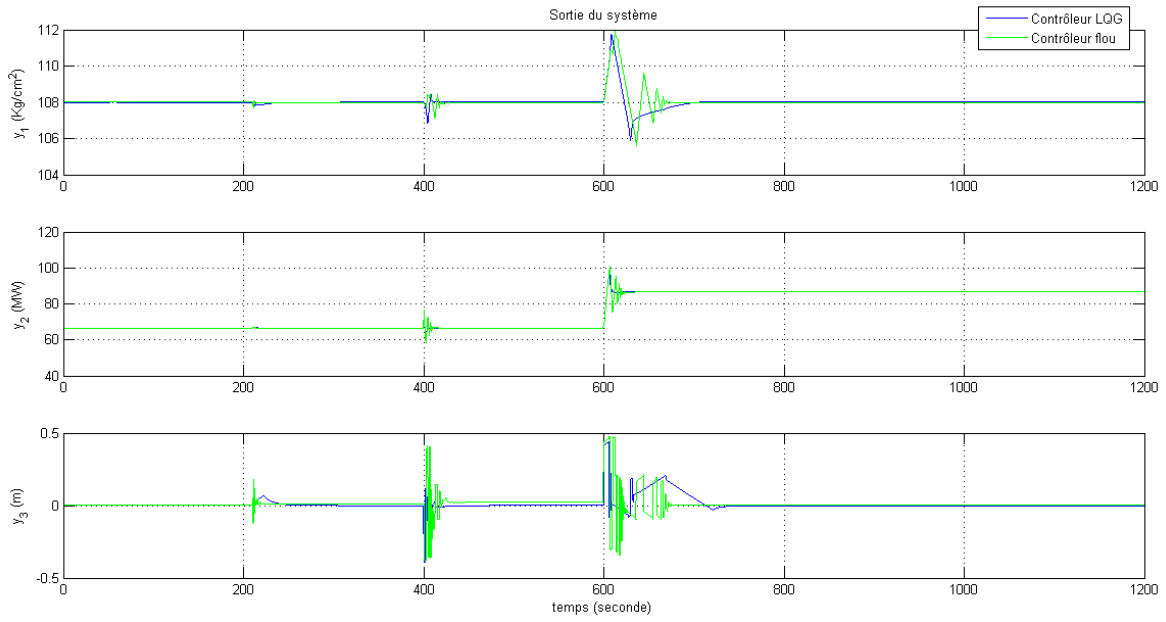


(a) Sortie du système

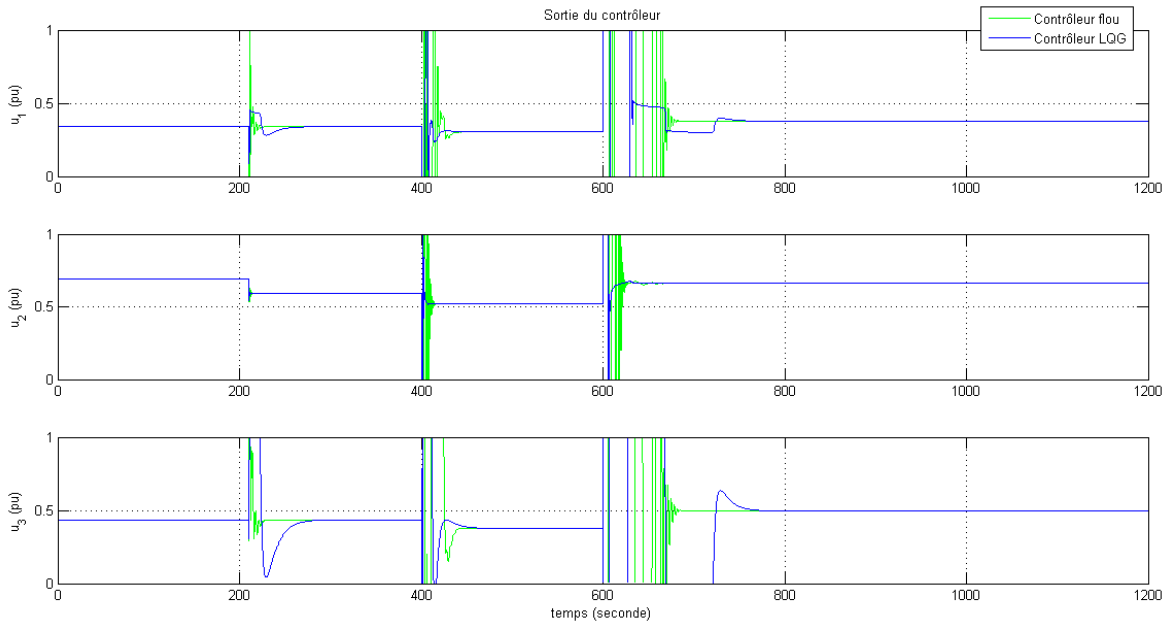


(b) Sortie du contrôleur

Figure IV.32 Réponses du processus en présence de perturbations, Cas 1.

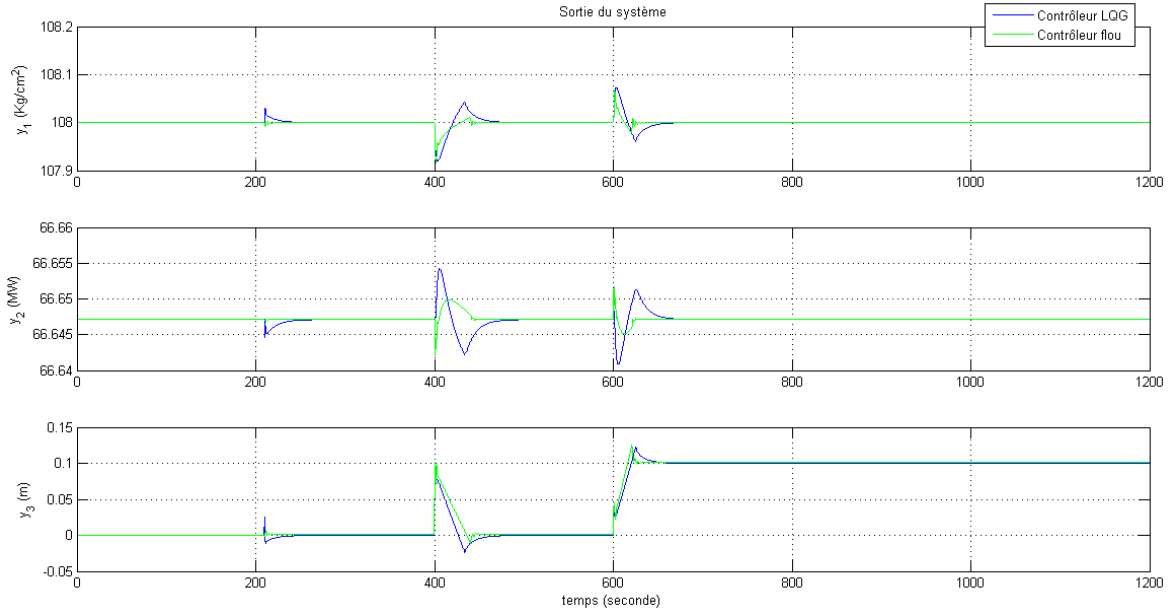


(a) Sortie du système

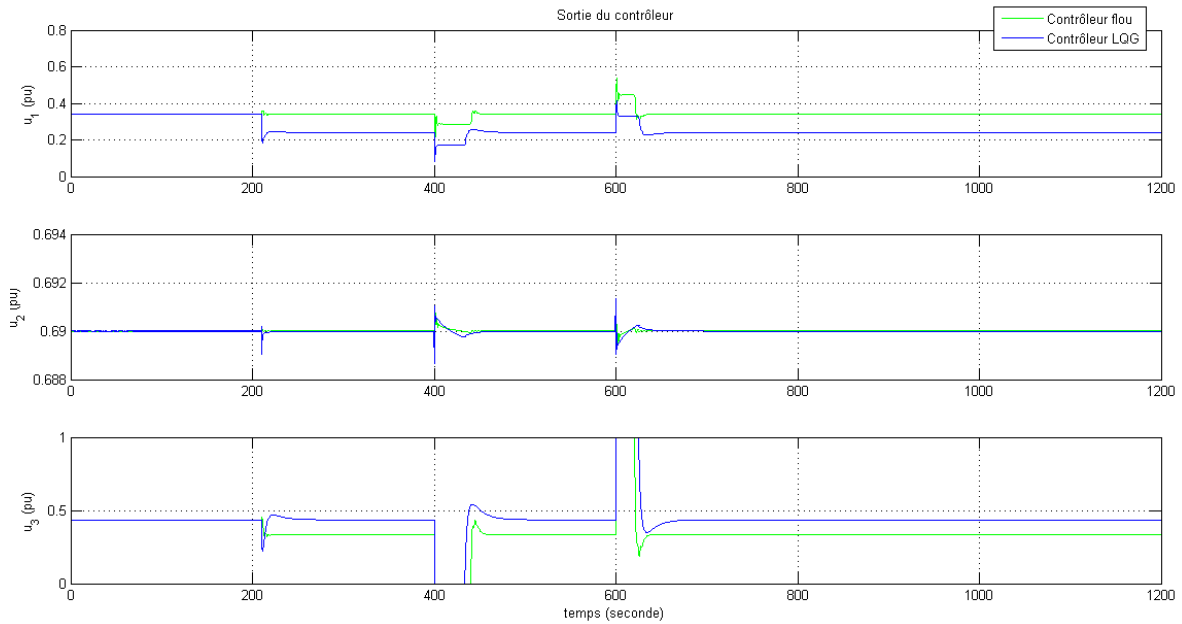


(b) Sortie du contrôleur

**Figure IV.33** Réponses du processus en présence de perturbations, Cas 2.



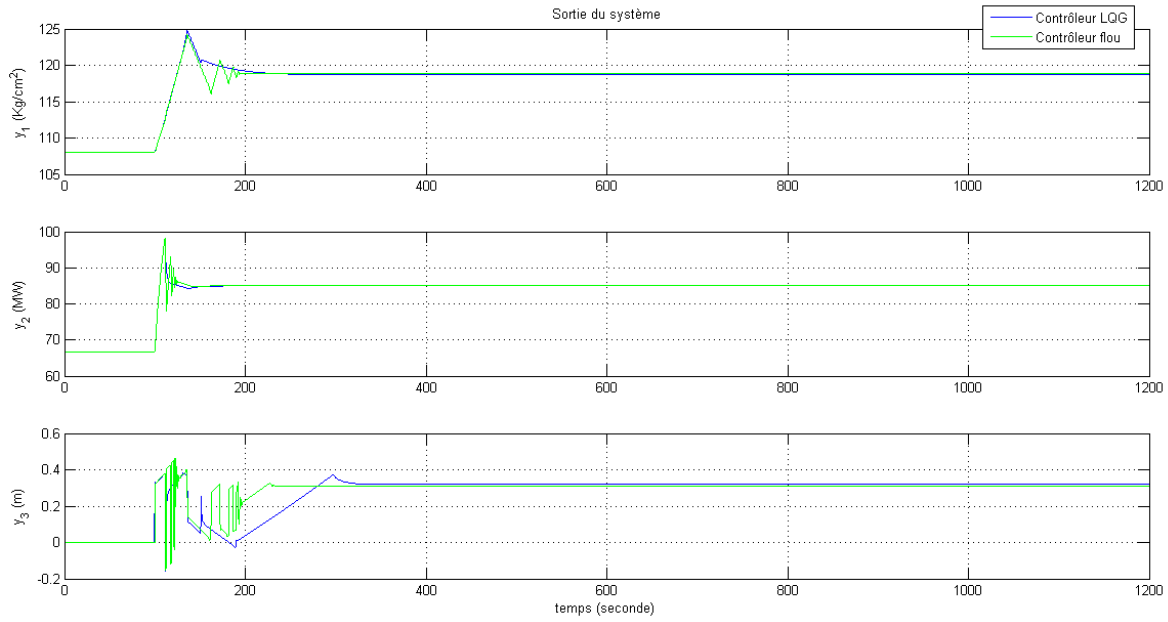
(a) Sortie du système



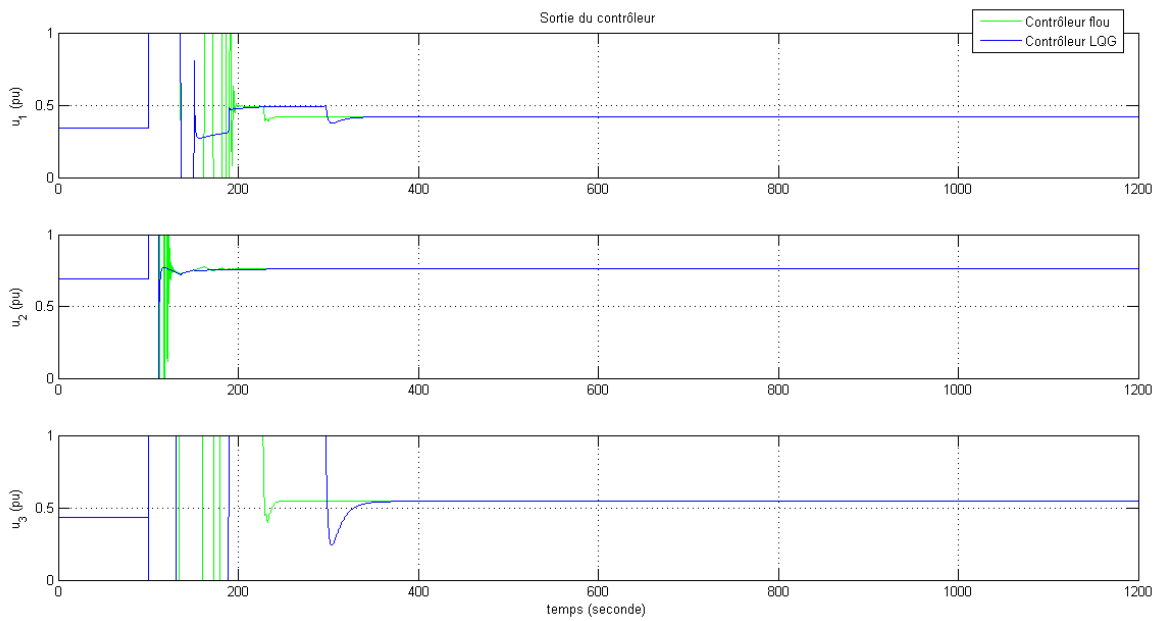
(b) Sortie du contrôleur

**Figure IV.34** Réponses du processus en présence de perturbations, Cas 3.



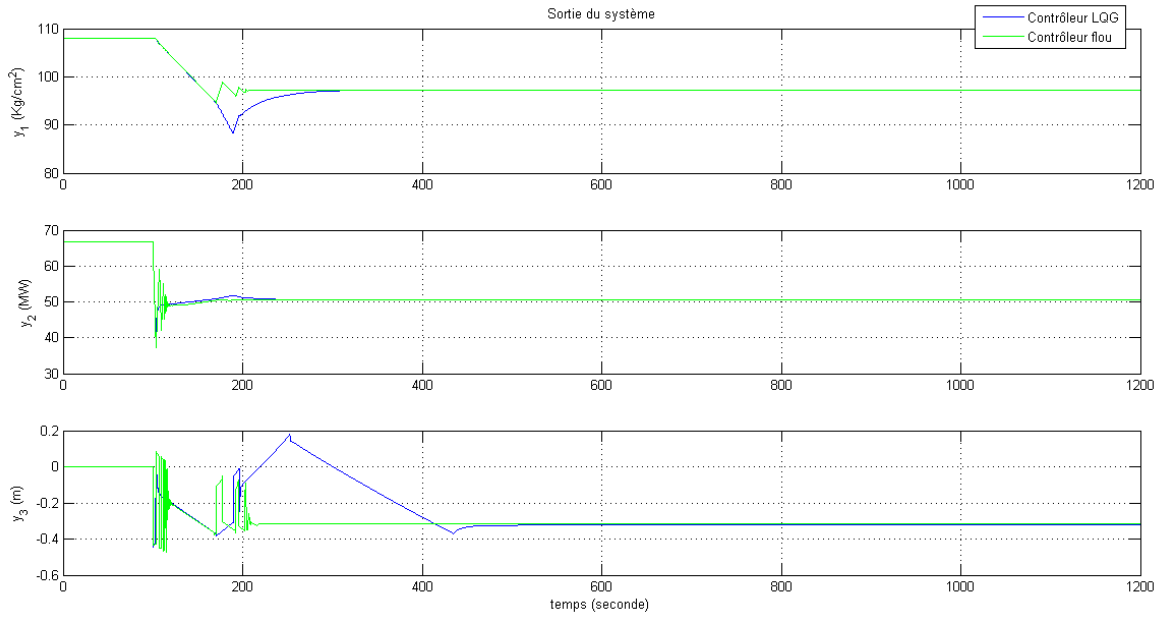


(a) Sortie du système

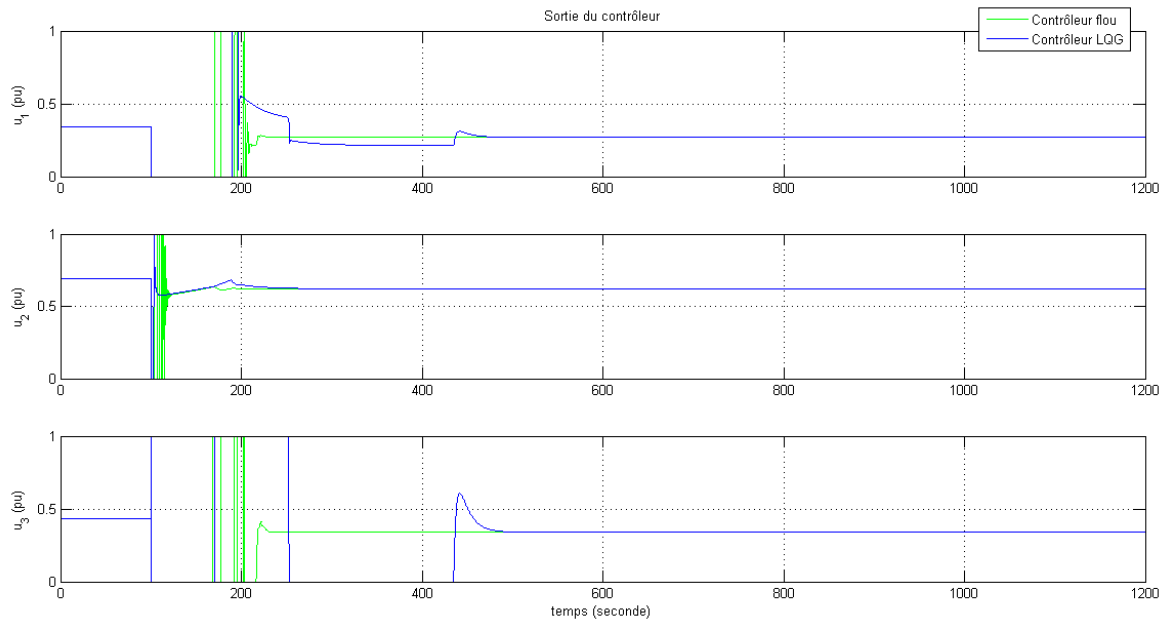


(b) Sortie du contrôleur

**Figure IV.35** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) + 10\%$ .

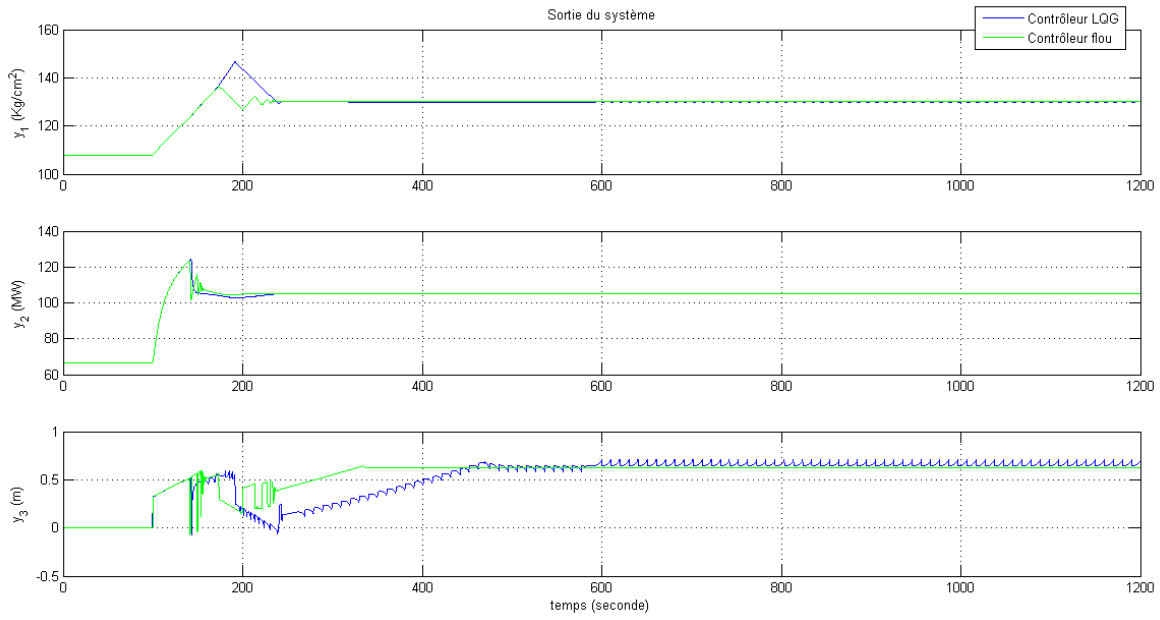


(a) Sortie du système

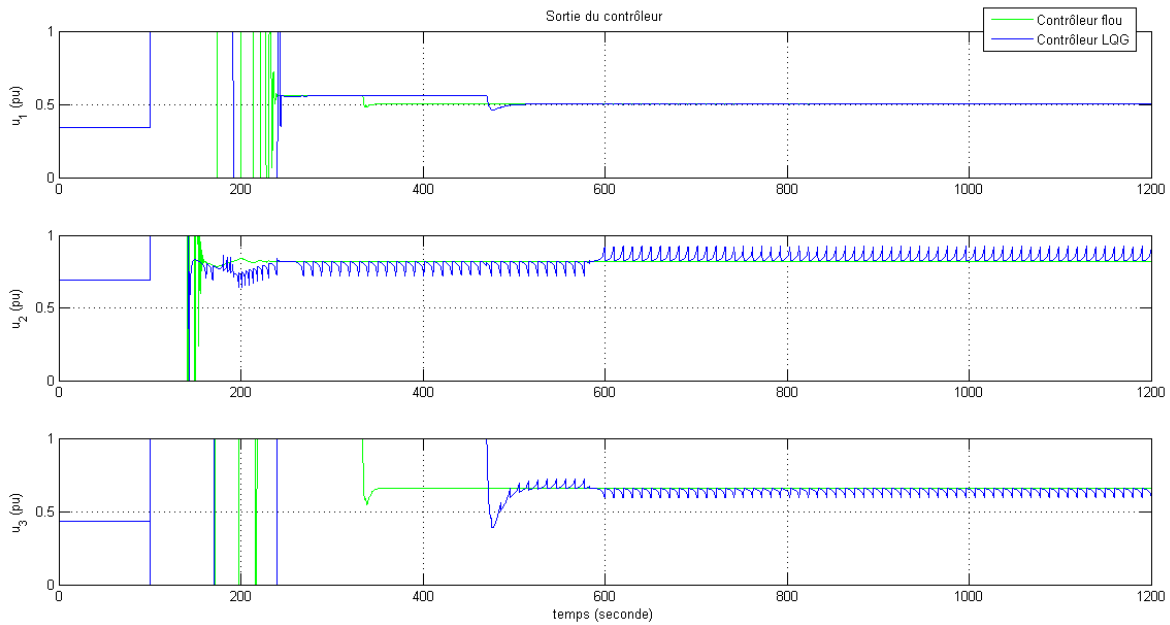


(b) Sortie du contrôleur

**Figure IV.36** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) - 10\%$ .

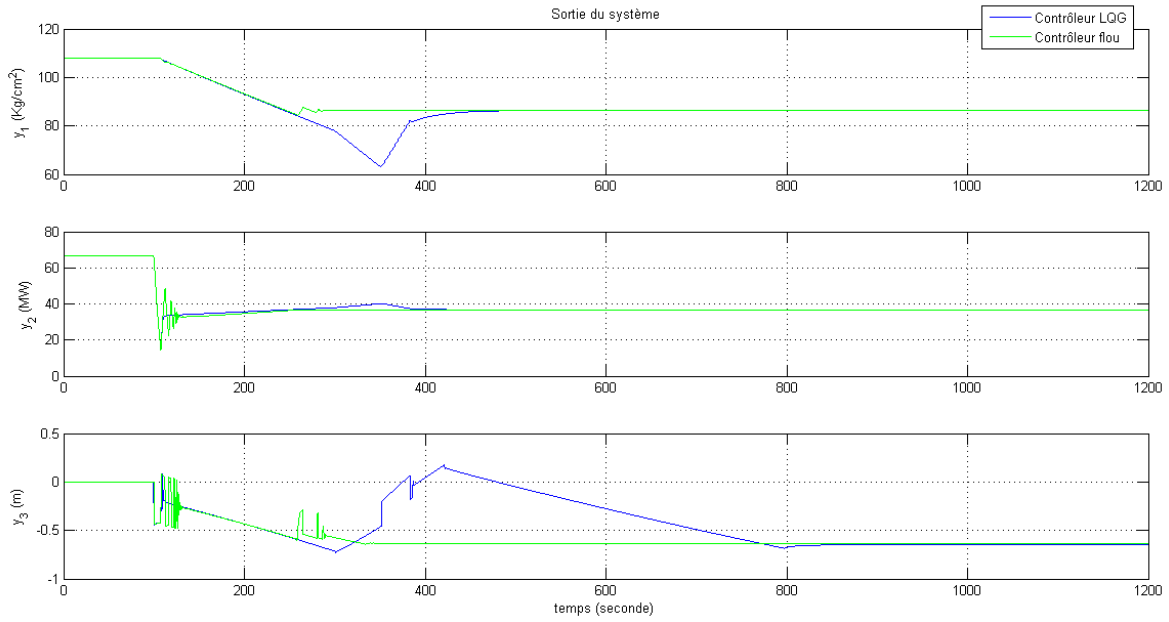


(a) Sortie du système

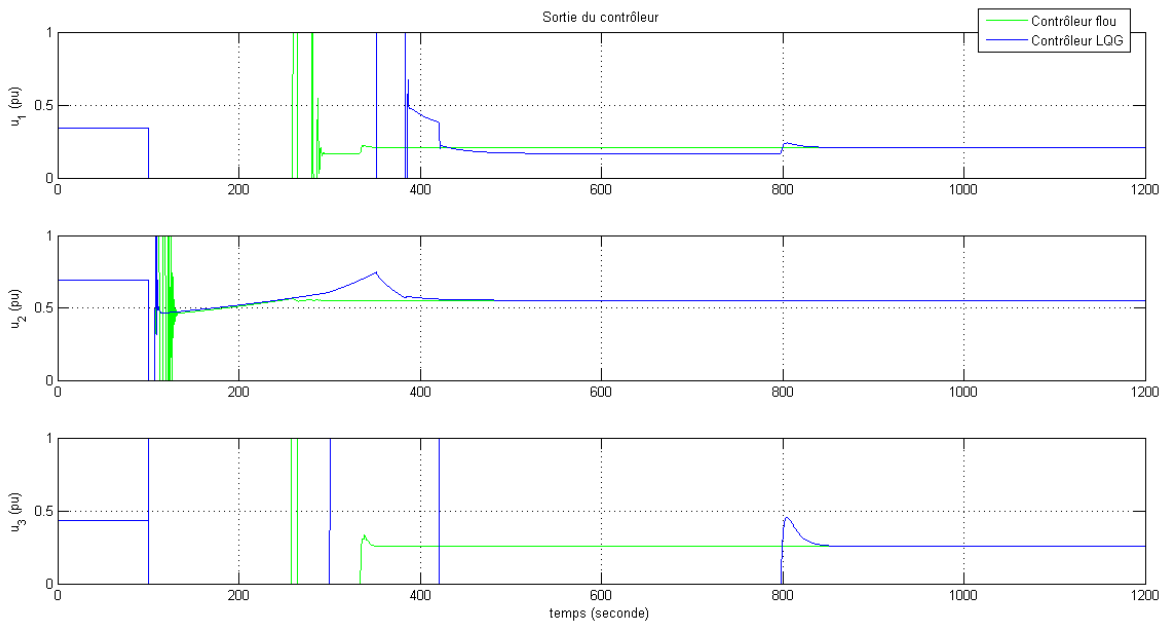


(b) Sortie du contrôleur

**Figure IV.37** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) + 20\%$ .

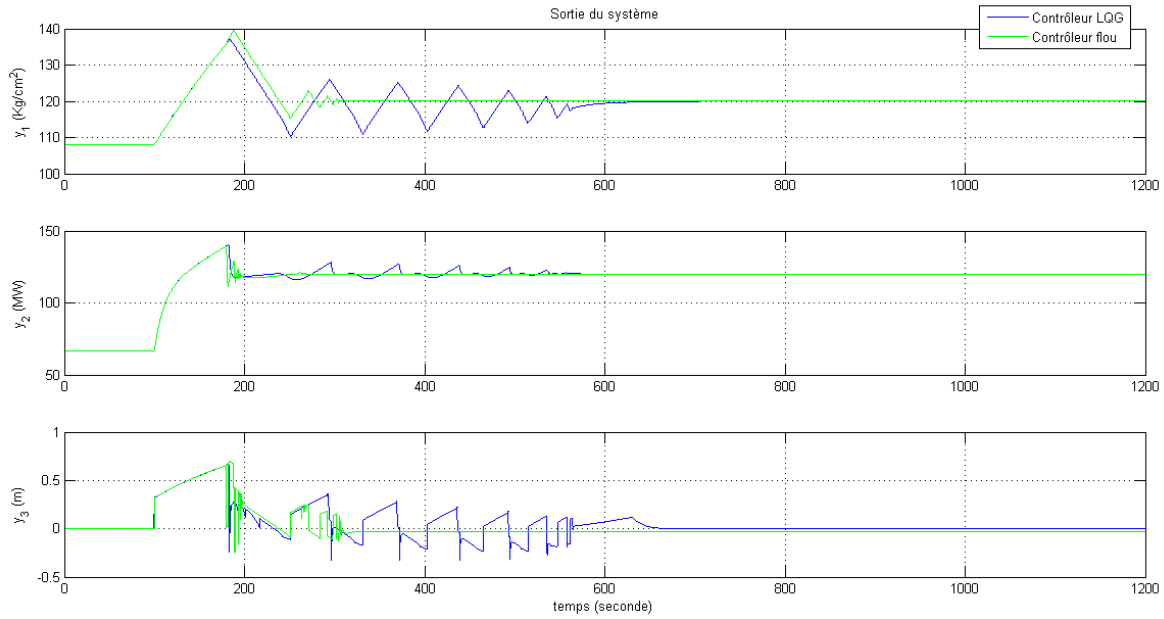


(a) Sortie du système

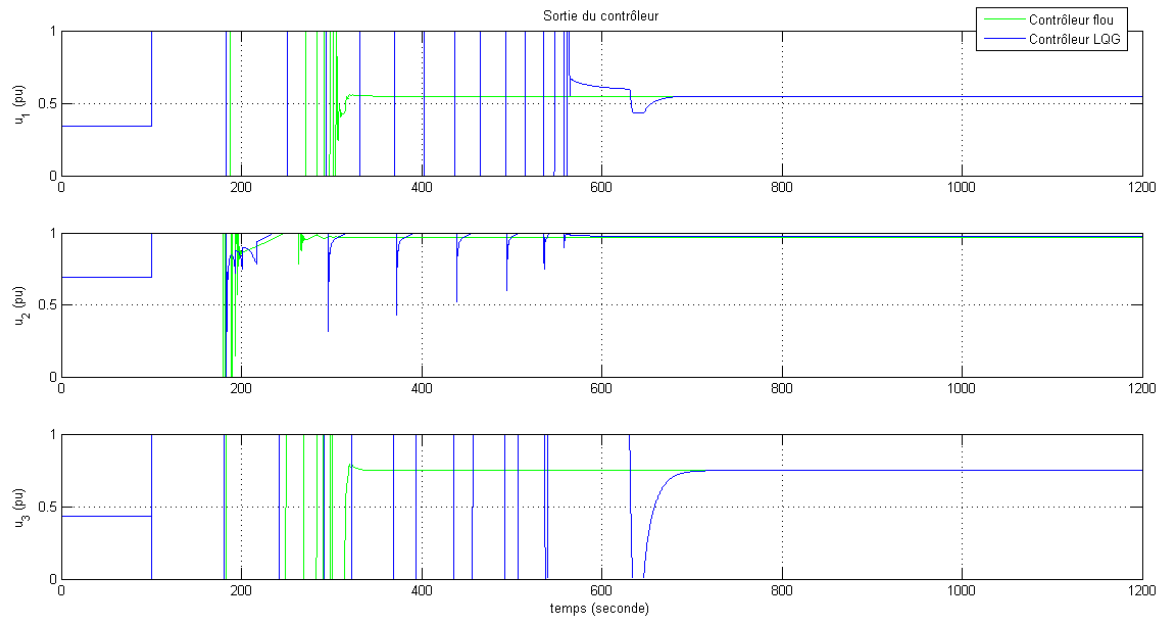


(b) Sortie du contrôleur

**Figure IV.38** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(r^0) - 20\%$ .

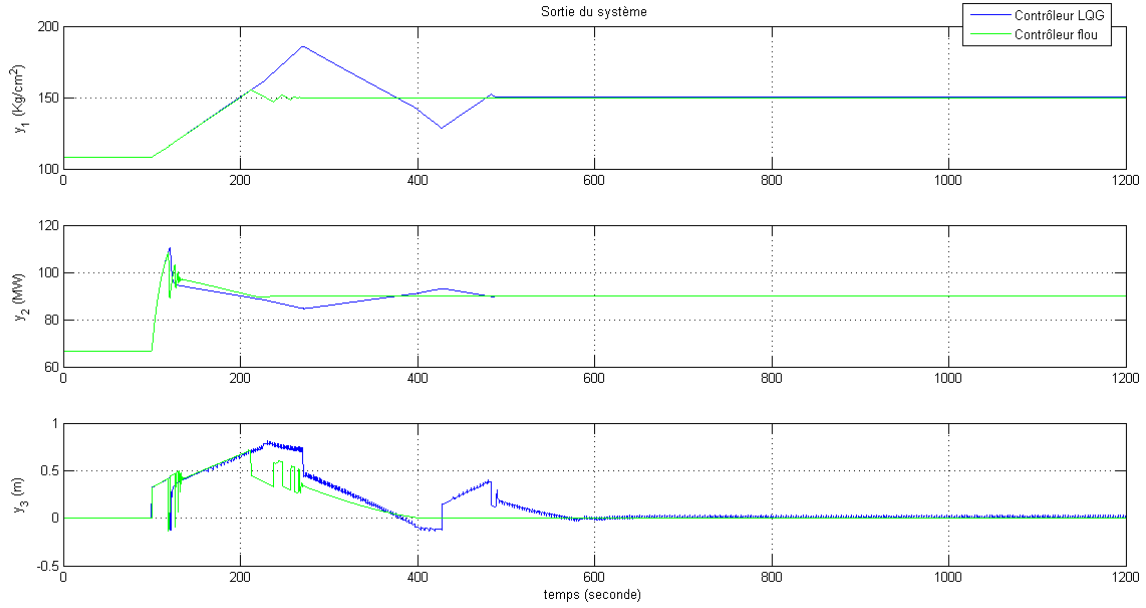


(a) Sortie du système

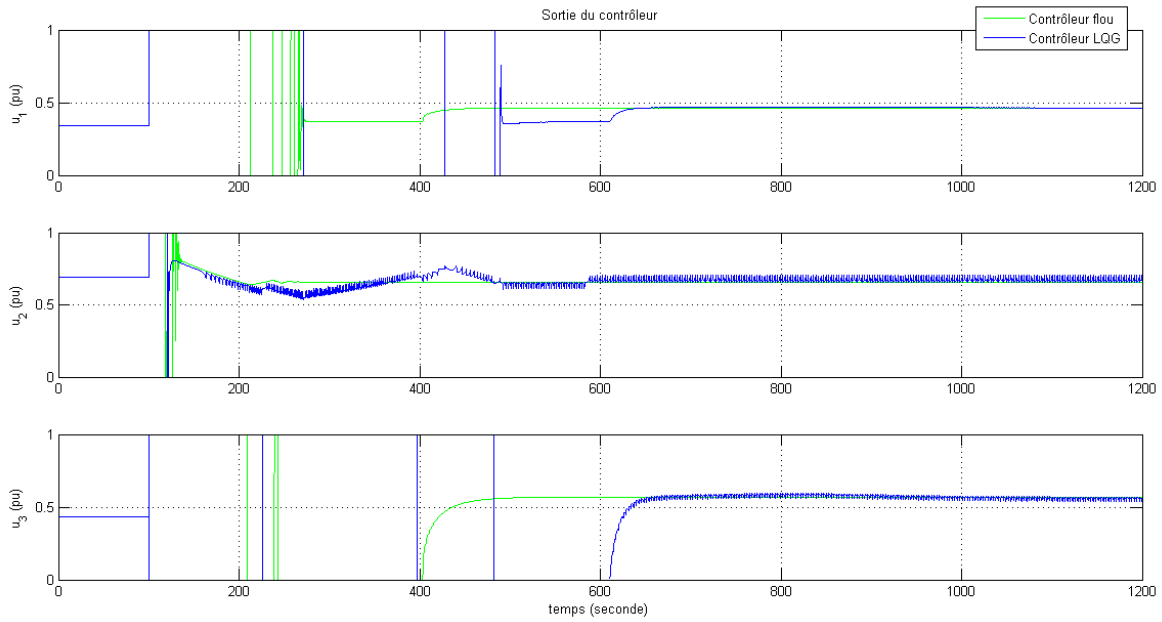


(b) Sortie du contrôleur

**Figure IV.39** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(120, 120, 0)$ .

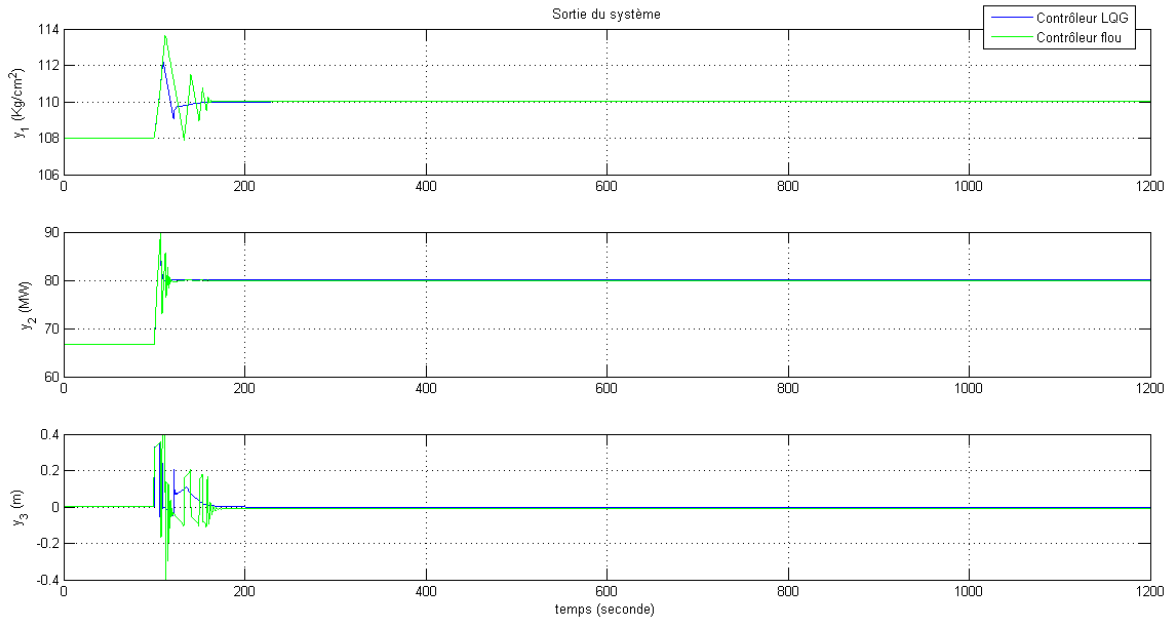


(a) Sortie du système

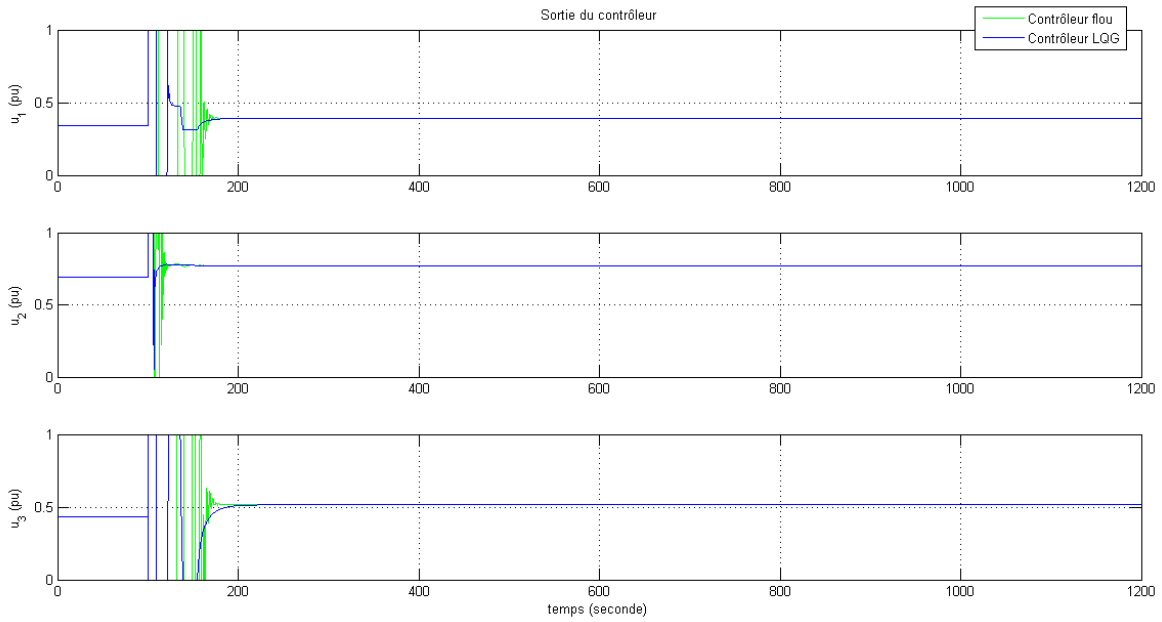


(b) Sortie du contrôleur

**Figure IV.40** Réponses du processus suite à un changement de consigne de  $\#(r^0)$  à  $\#(150, 90, 0)$ .

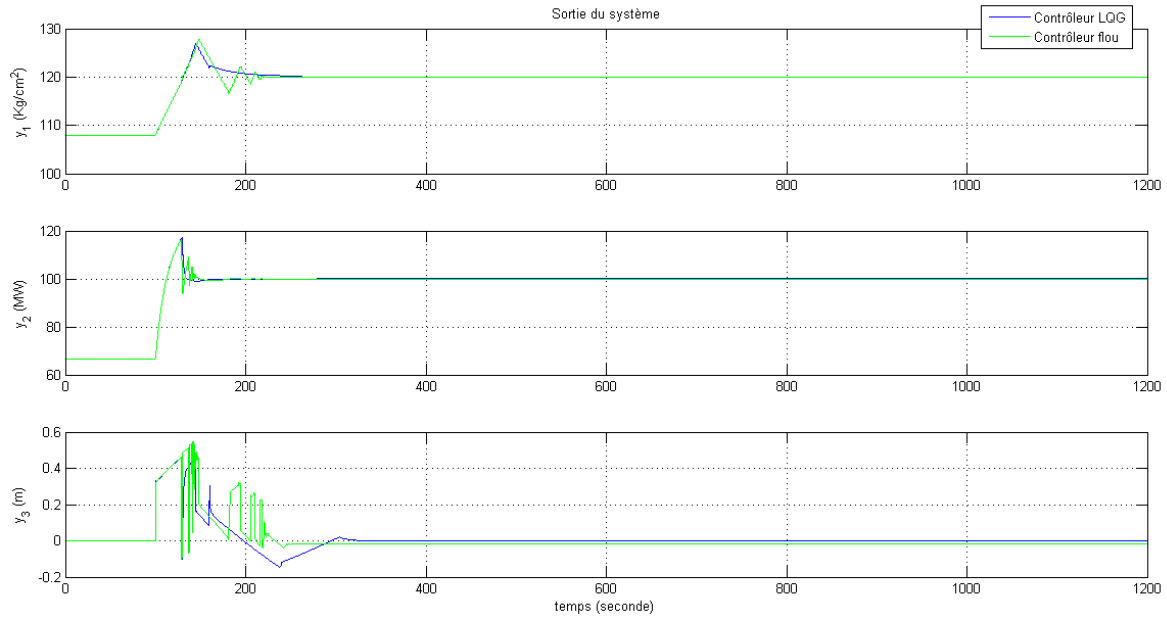


(a) Sortie du système

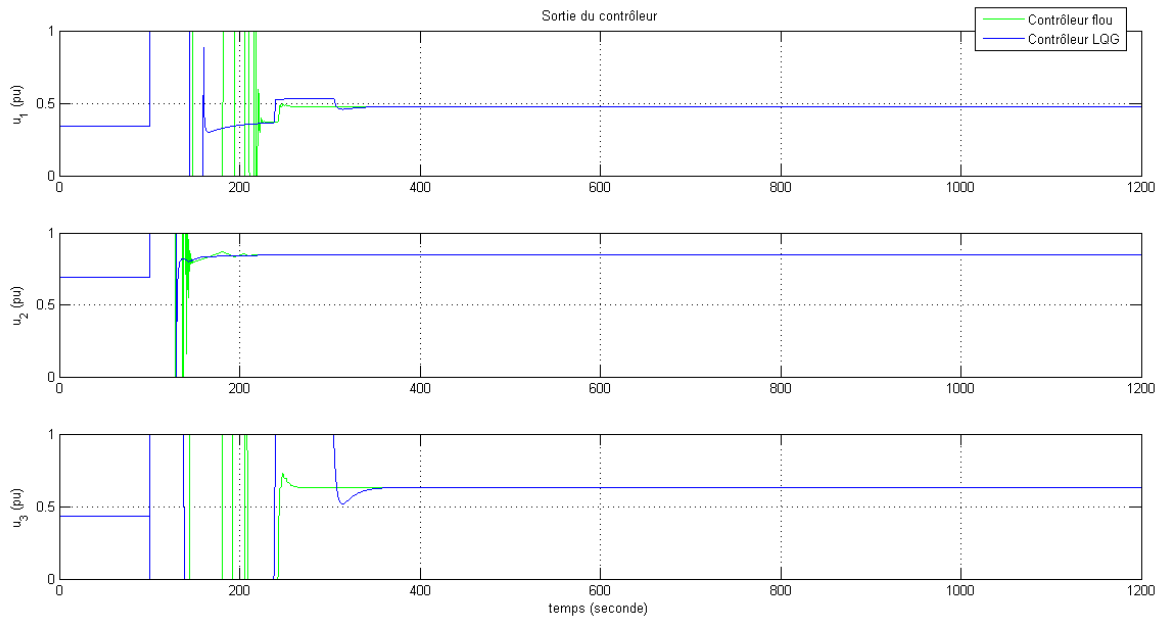


(b) Sortie du contrôleur

**Figure IV.41** Réponses du processus à la demande de puissance électrique, 1<sup>er</sup> cas.



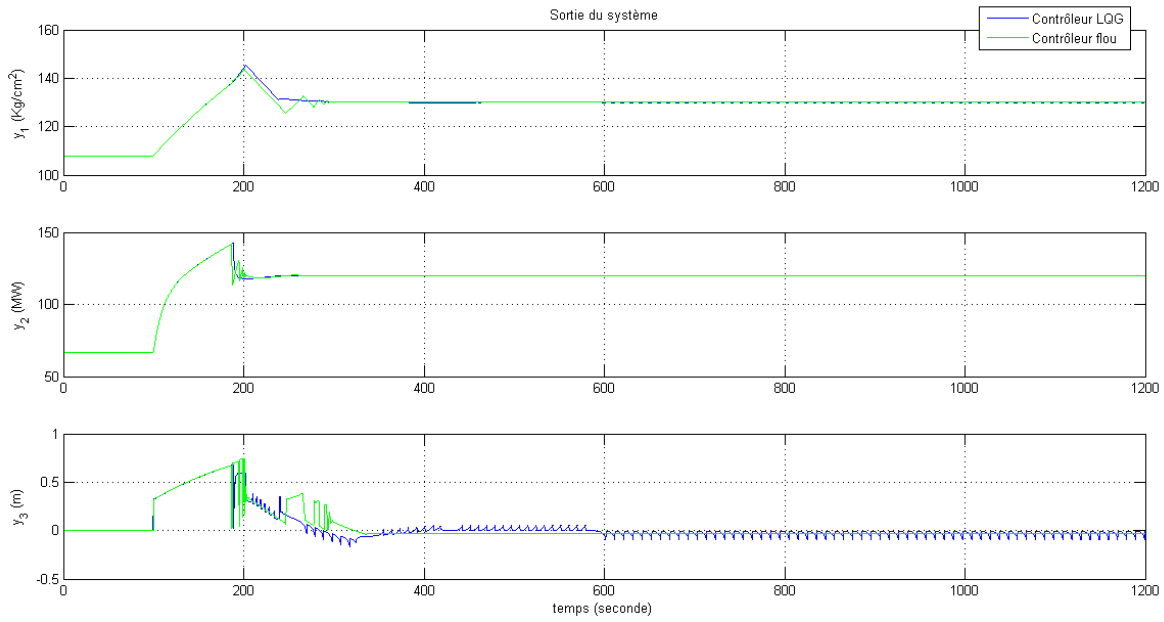
(a) Sortie du système



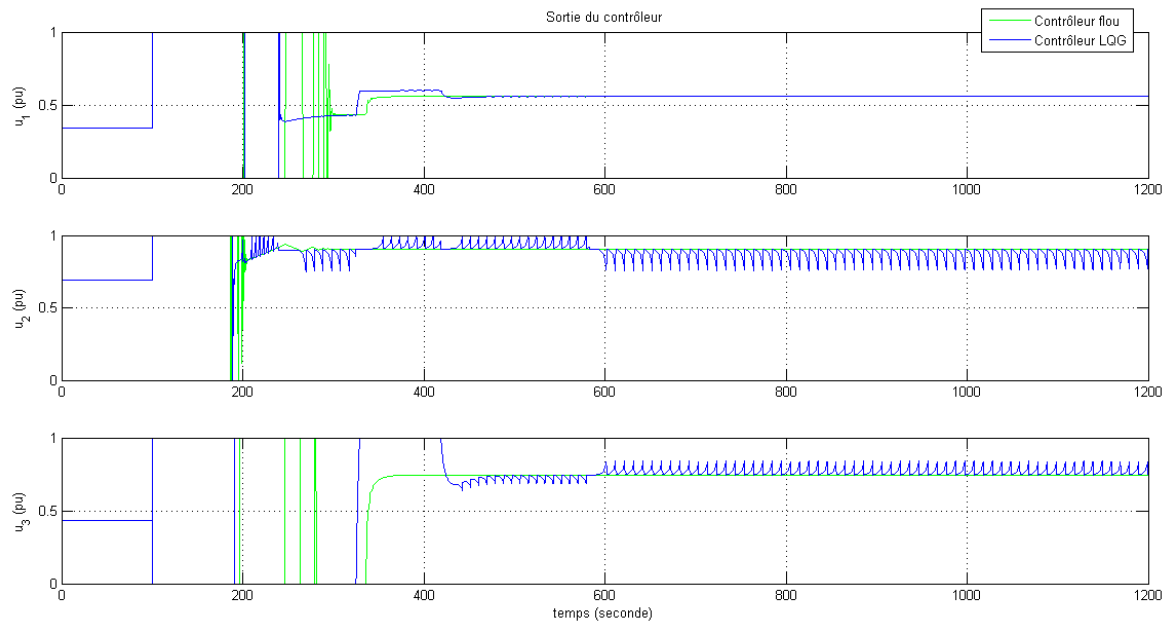
(b) Sortie du contrôleur

Figure IV.42 Réponses du processus à la demande de puissance électrique, 2<sup>e</sup> cas.



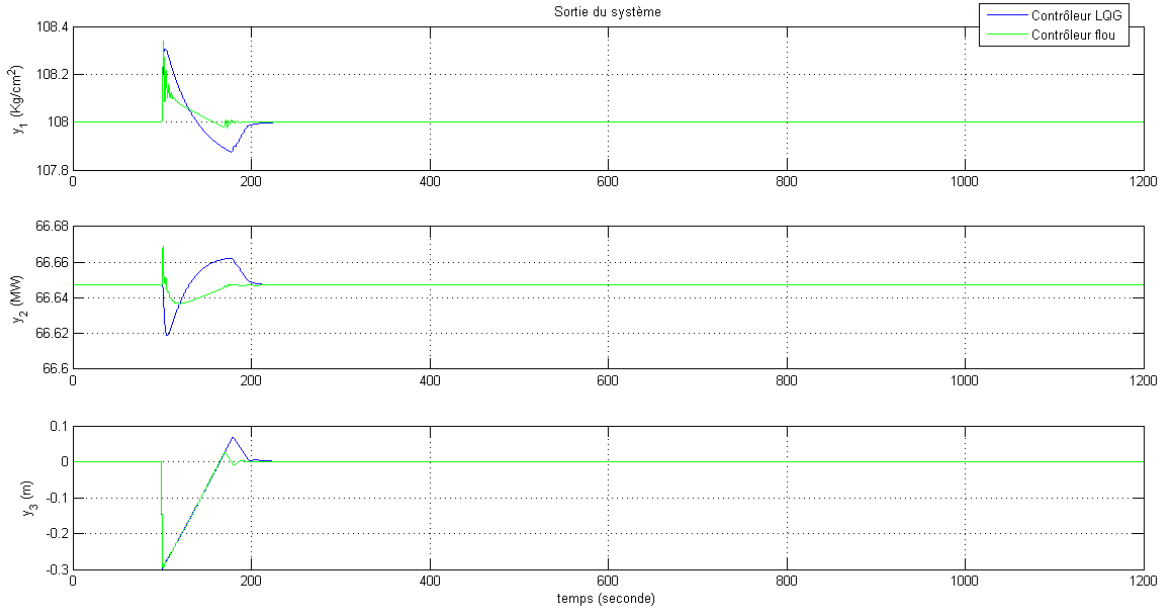


(a) Sortie du système

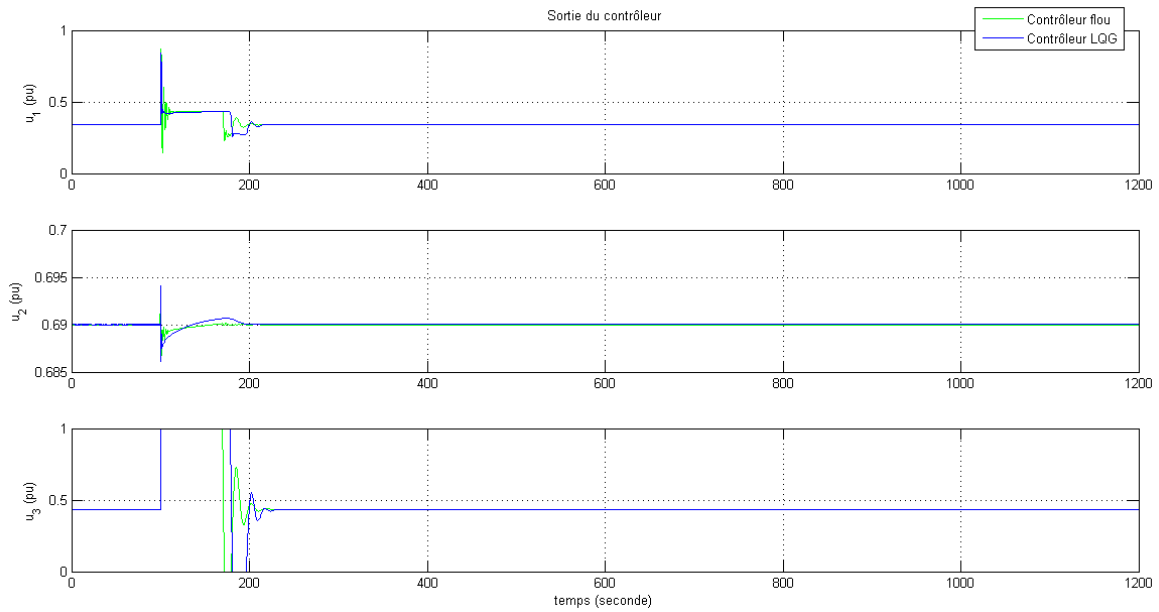


(b) Sortie du contrôleur

Figure IV.43 Réponses du processus à la demande de puissance électrique, 3<sup>e</sup> cas.

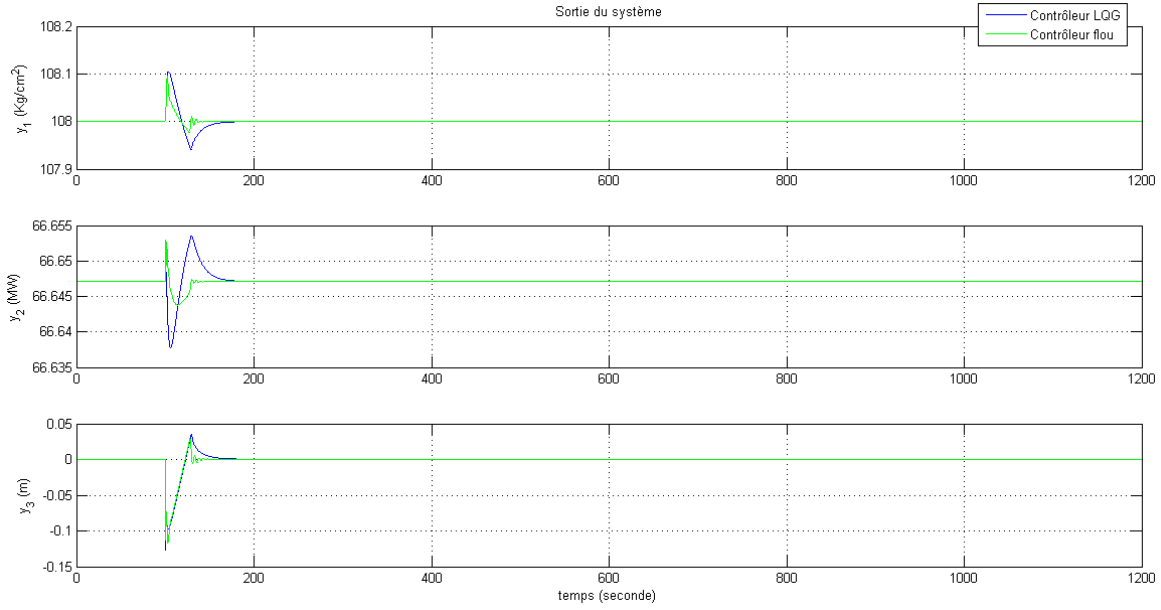


(a) Sortie du système

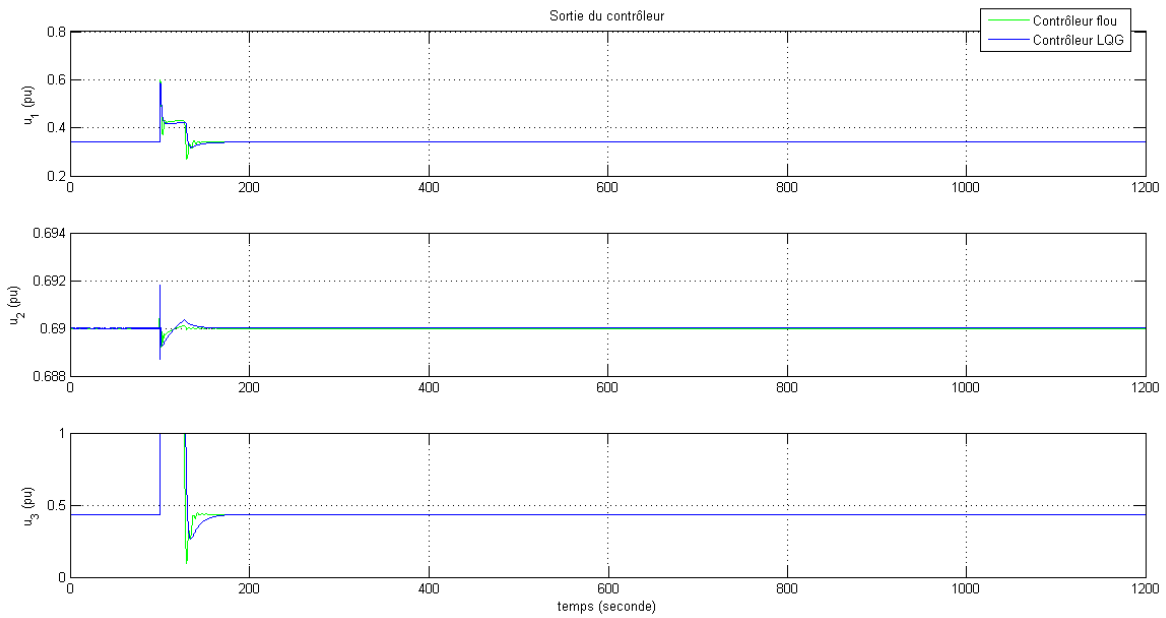


(b) Sortie du contrôleur

Figure IV.44 Réponses du processus au test de variations paramétriques, 1<sup>er</sup> cas.



(a) Sortie du système



(b) Sortie du contrôleur

Figure IV.45 Réponses du processus au test de variations paramétriques, 2<sup>er</sup> cas.

Dans les tests de changement du point de fonctionnement (Figures IV.35 à IV.38), on a pu constater une nette optimisation dans la conduite du processus sous le contrôle flou, en comparaison avec le contrôleur LQG original. Une approximation est également réalisée dans ce cas précis aussi un temps de réponse proche de celui du contrôleur original, mais aussi, on remarque quelque fluctuation au niveau des sorties du contrôleur, on les interprète par le manque existant dans la dynamique du contrôleur flou.

Les simulations de changement de consignes représentent plus qu'une bonne approximation, où dans la plupart des cas Figures IV.39 et IV.40, on peut observer un comportement optimisé de l'émulateur flou par rapport au contrôleur LQG original.

Les mêmes constatations ont été soulevées dans les autres tests de demande supplémentaire en puissance électrique (Figures IV.41 à IV.33) et de variations paramétriques (Figures IV.45 et IV.44), où l'on peut remarquer que l'émulateur flou du contrôleur LQG réalise dans certains cas un bon contrôle du niveau d'eau par rapport au contrôleur original, ce qui lui confère une bonne performance en stabilité.

## IV.5 Conclusion

Le but de ce chapitre est de tester et présenter la méthode d'émulation détaillée dans le Paragraphe III.3 dans le but de résoudre le problème d'implémentation des contrôleurs d'ordre élevés en utilisant des systèmes d'inférence floue. L'étude est menée sur un problème de commande multivariable du système non-linéaire GV-turbine pour lequel quatre stratégies de contrôle ont été conçues séparément, en l'occurrence la commande  $\mathcal{H}_\infty$ , une approximation avec un PI multivariable, la commande LQG, et un émulateur flou pour chacun des contrôleurs  $\mathcal{H}_\infty$  et LQG. Les contrôleurs flous sont générés automatiquement sur la base de données d'apprentissage acquises des modèles des contrôleurs  $\mathcal{H}_\infty$  et LQG. Ils ont montré des capacités d'approximation élevées et des performances accrues dans la plupart des cas de simulation. La structure simplifiée et la procédure de conception automatique en font une solution alternative intéressante pour traiter les problèmes pratiques pour l'implémentation des contrôleurs multivariables d'ordre élevé.

# CONCLUSION GÉNÉRALE

L'objectif principal de ce mémoire résidait dans le développement d'une méthodologie d'implémentation des contrôleurs multivariables d'ordre élevé par l'utilisation des systèmes d'inférence floue. Dans ce contexte, deux axes principaux sous-tendent les recherches entreprises.

Il s'agit, d'une part, d'étudier et d'analyser les méthodes d'approximation appliquées aux contrôleurs d'ordre élevé, synthétisés sur la base des théories de commande optimale et robuste, en particulier la commande linéaire quadratique et la synthèse  $\mathcal{H}_\infty$ .

D'autre part, la recherche d'autres approches pouvant être appliquées pour l'approximation du comportement dynamique d'un contrôleur multivariable donné, en tenant compte de l'application appréhendée. Dans ce sens, une approche systématique de synthèse est présentée dans ce mémoire. Elle se fonde sur le principe d'émulation des lois de commande multivariable sur la base de systèmes d'inférence floue. L'émulateur flou d'une commande multivariable est donc décrit par un modèle à base de règles floues que l'on propose d'extraire à partir de données numériques par l'utilisation de techniques de coalescence floue.

Il est important de noter que réussir une bonne approximation par émulation floue du système de commande multivariable est conditionné par la réalisation d'un *bon apprentissage* qui doit être mené en tenant compte de l'ensemble des considérations expliquées dans le présent document. Aussi, l'étape d'apprentissage doit être nécessairement suivie d'une phase de validation où seront manipulés des signaux d'excitation judicieusement choisis.

À l'issue des recherches effectuées, de nombreuses perspectives d'approfondissements et d'orientations nouvelles apparaissent comme prometteuses.

Tout d'abord, il faut prendre en considération que les émulateurs flous construits ne sont pas optimaux, le but de ce mémoire est de présenter une structure d'émulation avec un comportement acceptable. De plus amples améliorations peuvent être obtenues par la considération d'une analyse approfondie des signaux d'excitation à appliquer et des structures de modèles à utiliser pour la caractérisation du comportement dynamique des contrôleurs multivariables.

Par ailleurs, il existe d'autres recherches qui semblent logiques et attendent d'être testées. On peut citer, à titre d'exemple, la configuration multi-modèle floue pour tenir compte de manière explicite de la variation du point de fonctionnement du système, ou encore l'émulation floue dans le domaine fréquentiel.

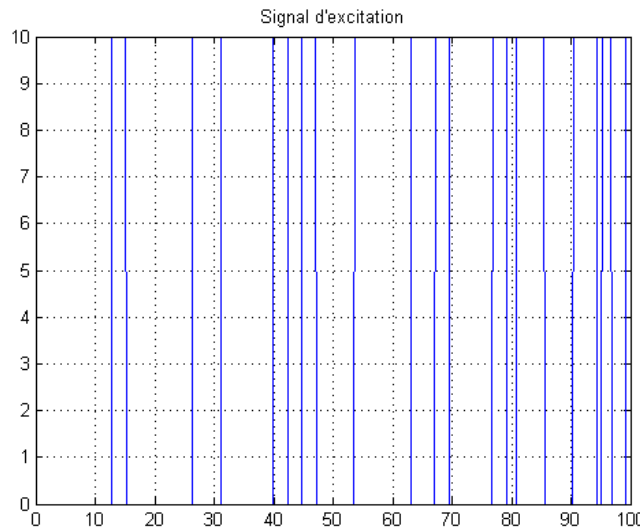
# ANNEXE I

## Signal d'excitation

Une séquence pseudo-aléatoire est une suite d'évènements créés de façon déterministe mais qui apparaît aléatoire à tout observateur non averti (deux évènements successifs ne sont pratiquement pas corrélés). Les séquences pseudo aléatoires présentent l'avantage d'être centrées, et de posséder un spectre riche. Les plus utilisées sont les séquences binaires pseudo-aléatoires (SBPA), où un signal formé d'impulsions rectangulaires modulées aléatoirement en longueur, qui approxime un bruit blanc discret.

Un signal binaire pseudo-aléatoire (SBPA) est caractérisé par deux paramètres, le nombre de registres à décalage  $n_{sr}$  et le temps de commutation  $T_{sw}$ . Dans cette étude, le signal SBPA utilisé pour chaque entrée a une amplitude  $\tilde{e}$ .

- Cas du contrôleur  $\mathcal{H}_\infty$  : La figure A I.1 représente le signal utilisé dans l'excitation du contrôleur  $\mathcal{H}_\infty$ , où ses paramètres sont comme suit,  $n_{sr} = 20$  et  $T_{sw} = 0.8$ , avec une amplitude unitaire  $\tilde{e} = 10$ .



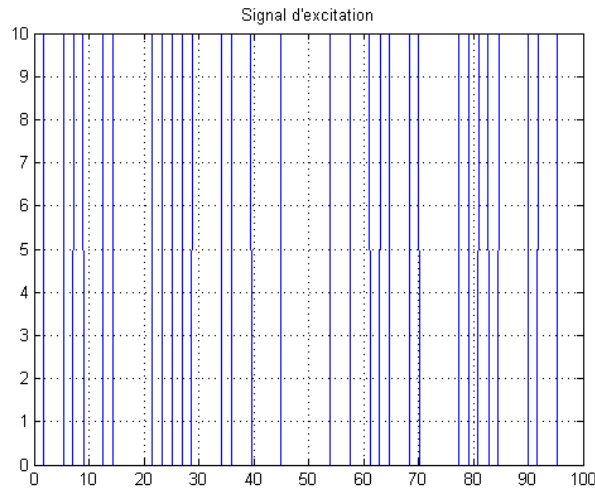
**Figure A I.1** Signal d'excitation utilisé pour le contrôleur  $\mathcal{H}_\infty$ .

Pour l'excitation de l'ensemble des entrées du contrôleur  $\mathcal{H}_\infty$ , on a utilisé plusieurs combinaisons, avec différentes amplitudes, tel que montré dans le Tableau A I.1.

$\tilde{e}_1$	$\tilde{e}_2$	$\tilde{e}_3$
10	0	0
0	10	0
0	0	10
10	10	0
10	10	0
0	10	10
0	-10	10
10	0	10
-10	0	10
10	10	10
-10	10	10
10	-10	10
10	10	-10

**Tableau A I.1** Combinaisons des signaux d'excitation utilisées pour le contrôleur  $\mathcal{H}_\infty$ .

- Cas du contrôleur LQG : Dans la figure A I.2, on a représenté le signal utilisé pour l'excitation du contrôleur LQG, où ses paramètres sont comme suit,  $n_{sr} = 05$  et  $T_{sw} = 1.8$ , avec une amplitude égale à 10.



**Figure A I.2** Signal d'excitation utilisé pour le contrôleur LQG.

Nous avons excité l'ensemble des entrées du contrôleur LQG en utilisant plusieurs combinaisons, avec des différentes amplitudes. Le Tableau A I.2 résume toutes les combinaisons utilisées.

$\tilde{e}_1$	$\tilde{e}_2$	$\tilde{e}_3$
10	0	0
0	10	0
0	0	10

**Tableau A I.2** Combinaisons des signaux d'excitation utilisées pour le contrôleur LQG.



# BIBLIOGRAPHIE

- [1] Janos Abonyi and Balazs Feil. *Cluster analysis for data mining and system identification*. Springer, 2007.
- [2] Brian DO Anderson and John B Moore. *Linear optimal control*, volume 197. Prentice-Hall Englewood Cliffs, NJ, 1971.
- [3] Karl Johan Åström. Simplified models of boiler-turbine units. (52), 1987.
- [4] Karl Johan Åström and Karl Eklund. A simplified non-linear model of a drum boiler-turbine unit. *International Journal of Control*, 16(1) :145–169, 1972.
- [5] KJ Åström and RD Bell. A simple drum level model. *Report LUTFD2*, 1976.
- [6] KJ Åström and Karl Eklund. A simple non-linear drum boiler model. *International Journal of Control*, 22(5) :739–740, 1975.
- [7] Robert Babuska. Fuzzy and neural control. disc course lecture notes. *Delft University of Technology. Delft, the Netherlands*, 2001.
- [8] RD Bell and Karl Johan Åström. *Dynamic models for boiler-turbine-alternator units : data logs and parameter estimation for a 160 MW unit*. Lund Institute of Technology, Department of Automatic Control, 1987.
- [9] Shankar P Bhattacharyya, Aniruddha Datta, and Lee H Keel. *Linear control theory : structure, robustness, and optimization*. CRC press, 2010.
- [10] Dan Chen and Dale E Seborg. Relative gain array analysis for uncertain process models. *AIChE journal*, 48(2) :302–310, 2002.
- [11] Stephen Chiu. Method and software for extracting fuzzy classification rules by subtractive clustering. In *Conference of the North American Fuzzy Information Processing Society (NAFIPS), Biennial*, pages 461–465. IEEE, 1996.
- [12] Stephen Chiu. Extracting fuzzy rules from data for function approximation and pattern classification. *Fuzzy Information Engineering : A Guided Tour of Applications*. John Wiley&Sons, 1997.
- [13] Stephen L Chiu. Selecting input variables for fuzzy models. *Journal of Intelligent and Fuzzy Systems*, 4(4) :243–256, 1996.
- [14] Brian L Cooley and Jay H Lee. Control-relevant experiment design for multivariable systems described by expansions in orthonormal bases. *Automatica*, 37(2) :273–281, 2001.
- [15] Peter Dorato. A historical review of robust control. *Control Systems Magazine, IEEE*, 7(2) :44–47, 1987.

- [16] Geir E Dullerud and Fernando Paganini. *A course in robust control theory*, volume 6. Springer New York, 2000.
- [17] Gang Feng. *Analysis and Synthesis of Fuzzy Control Systems : a model-based approach*. CRC Press, 2010.
- [18] ME Flynn and MJ O'Malley. A drum boiler model for long term power system dynamic simulation. *IEEE Transactions on Power Systems*, 14(1) :209–217, 1999.
- [19] B.A. Francis. *A course in  $H_\infty$  control theory*. Lecture Notes in Control and Information Sciences Series.
- [20] Caifen Fu, Jizhen Liu, and Wen Tan. Robust PI design for a benchmark nonlinear boiler. In *5<sup>th</sup> Asian Control Conference*, volume 1, pages 304–308. IEEE, 2004.
- [21] Douglas L Girvin and Wayne John Book. Analysis of poles and zeros for tapered link designs. *Series in intelligent control and intelligent automation. Advanced Studies of Flexible Robotic Manipulators : Modeling, Design, Control, and Applications*, World Scientific Press, Fei-Yue Wang, ed-in-chief, 2003.
- [22] Graham Clifford Goodwin, Stefan F Graebe, and Mario E Salgado. *Control system design*, volume 240. Prentice Hall New Jersey, 2001.
- [23] Mohinder S Grewal and Angus P Andrews. *Kalman filtering : theory and practice using MATLAB*. John Wiley & Sons, 2011.
- [24] H Habbi, M Kidouche, and M Zelmat. A combined LQG/LTR and fuzzy logic strategy for the improved control of a drum-type boiler-turbine plant. *Control & Intelligent Systems*, 37(2), 2009.
- [25] H. Habbi and M. Zelmat. Intelligent coordinated control scheme for improving steam power plant operations during load rejections. In *International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce*, volume 2, pages 576–580. IEEE, 2005.
- [26] Zhiqi Hu, Hanbai Fan, and Yujun He. Robust controller design for boiler system. In *International Conference on Intelligent Computation Technology and Automation (ICICTA)*, volume 1, pages 445–449. IEEE, 2008.
- [27] Fawnizu Azmadi Hussin and NW Rees. Overcoming the shrink-and-swell effect in water level control strategy on industrial boiler-drum. *Universiti Teknologi PETRONAS (UTP)*, page 10.
- [28] Petros A Ioannou and Andreas Pitsillides. *Modeling and control of complex systems*. CRC Press, 2010.
- [29] Arun Khosla, Shakti Kumar, and KK Aggarwal. Identification of fuzzy controller for rapid nickel-cadmium batteries charger through fuzzy c-means clustering algorithm. In *22<sup>nd</sup> International Conference of the North American Fuzzy Information Processing Society (NAFIPS)*, pages 536–539. IEEE, 2003.
- [30] Frank Klawonn and Rudolf Kruse. Constructing a fuzzy controller from data. *Fuzzy sets and systems*, 85(2) :177–193, 1997.

- [31] Ioan Doré Landau, Alireza Karimi, and Aurelian Constantinescu. Direct controller order reduction by identification in closed loop. *Automatica*, 37(11) :1689–1702, 2001.
- [32] Hong-Xing Li, CL Philip Chen, and Han-Pang Huang. *Fuzzy neural intelligent systems : mathematical foundation and the applications in engineering*. CRC Press, 2001.
- [33] C Maffezzoni. Boiler-turbine dynamics in power-plant control. *Control Engineering Practice*, 5(3) :301–312, 1997.
- [34] Ebrahim H Mamdani. Application of fuzzy algorithms for control of simple dynamic plant. *Proceedings of the Institution of Electrical Engineers*, 121(12) :1585–1588, 1974.
- [35] AJ Morton and PH Price. The controllability of steam output, pressure and water level in drum boilers. *Steam at work : industrial and marine steam plant*, pages 75–84, 1977.
- [36] Oliver Nelles. *Nonlinear system identification : from classical approaches to neural networks and fuzzy models*. Springer, 2001.
- [37] MEJ Newman. Complex systems : A survey. *American Journal of Physics*, 79(10) :800–810, 2011.
- [38] Andrew Packard, Kameshwar Poolla, and Roberto Horowitz. Dynamic systems and feedback.
- [39] Elie Sanchez, Lotfi A Zadeh, and Takanori Shibata. *Genetic algorithms and fuzzy logic systems*. World Scientific, 1997.
- [40] Dragoslav D Siljak. *Decentralized control of complex systems*. Courier Dover Publications, 2011.
- [41] Sigurd Skogestad and Ian Postlethwaite. *Multivariable feedback control : analysis and design*, volume 2. Wiley New York, 2007.
- [42] Tomohiro Takagi and Michio Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics*, (1) :116–132, 1985.
- [43] Wen Tan, Yuguang Niu, and Jizhen Liu.  $H_\infty$  control for a boiler-turbine unit. In *Proceedings of the 1999 IEEE International Conference on Control Applications*, volume 2, pages 910–914. IEEE, 1999.
- [44] Harry L Trentelman, Anton A Stoorvogel, and Malo Hautus. *Control theory for linear systems*. Springer Verlag, 2001.
- [45] Hao Ying. *Fuzzy control and modeling : analytical foundations and applications*. Wiley-IEEE Press, 2000.
- [46] Kemin Zhou and John Comstock Doyle. *Essentials of robust control*, volume 104. Prentice hall Upper Saddle River, NJ, 1998.
- [47] Kemin Zhou, John Comstock Doyle, Keith Glover, et al. *Robust and optimal control*, volume 40. Prentice Hall New Jersey, 1996.